DESKTOP ICONS: A COMPARATIVE STUDY OF TEXTUAL AND ICONIC HUMAN USER INTERFACE ELEMENTS IN VISUAL COMMUNICATION

A thesis submitted to the department of Graphic Design
and the Institute of Fine Arts of Bilkent University
in partial fulfillment of the requirements for
the degree of Master of Fine Arts

By
Çağlar Önder
June, 2003

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality as a thesis for the degree of Master of Fine Arts.

_____

Assoc. Prof. Emre Becer (Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality as a thesis for the degree of Master of Fine Arts.

_____

Assist. Prof. Andreas Treske (Co-advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality as a thesis for the degree of Master of Fine Arts.

_____

Assist. Prof. Marek Brozozowski

Approved by the Institute of Fine Arts

_____

Prof. Dr. Bülent Özgüç.

Director of the Institute of Fine Arts

ABSTRACT

DESKTOP ICONS: A COMPARATIVE STUDY OF TEXTUAL AND ICONIC HUMAN USER INTERFACE ELEMENTS IN VISUAL COMMUNICATION

Çağlar Önder

M.F.A. in Graphic Design

Supervisor: Assoc. Prof. Emre Becer

June, 2003

This research examines “Desktop Icons” in correlation to contemporary developments, changes and expanding possibilities in the usage and integration of multimedia into applications and interfaces; outlining how textual and iconic elements function, work and reflect changes in the study of Human-Computer Interaction and Graphic Design as a medium of digital technologies.

Keywords: Icons, Desktop, Windows, Menus, Human-Computer Interaction, Graphical User Interface, Textual Interface, Visual Communication.

ÖZET

MASAÜSTÜ İKONLARI: GÖRSEL İLETİŞİMDE METNE DAYALI VE İKONİK KULLANICI ARAYÜZÜ ELEMANLARININ KARŞILAŞTIRMALI BİR ANALİZİ

Çağlar Önder

Grafik Tasarım Bölümü
Yüksek Lisans

Tez Yöneticisi: Doç. Dr. Emre Becer

Haziran, 2003

Bu çalışma, "Masaüstü İkonları"nı multimedyanın gelişen, değişen ve artan imkanlarla yazılımlar ve arayüzler içinde kullanımı ve bunlarla bütünleşmesi çerçevesinde incelemekte; metne dayalı ve ikonik elemanların işlevselliğini, çalışma prensiplerini ve İnsan-Bilgisayar Etkileşimi ve sayısal teknolojilerin aracı olan Grafik Tasarımdaki gelişmeleri nasıl yansıttığını ele almaktadır.

Anahtar kelimeler: İkonlar, Masaüstü, Pencereler, Menüler, İnsan-Bilgisayar Etkileşimi, Görsel Kullanıcı Arayüzü, Metne Dayalı Arayüz, Görsel İletişim.

ACKNOWLEDGEMENTS

I would like to foremost thank my advisors and former instructors Assoc. Prof. Emre Becer and Assist. Prof. Andreas Treske for their support, guidance and everlasting patience, especially at times when I delayed the deadlines. I furthermore thank them for all the knowledge and experience they have passed on to me both for my career and for my life.

I thank all my friends, Okan in particular who kept me company most of the time, Güzden, Emre and Nur who were companions at hard times and best understood and shared the difficulties of writing a thesis since they were concurrently working on their own. My special thanks go to Ahu whose support was essentially important, and who continually motivated me and showed the utmost support against all negative circumstances that would arise.

Last but not least, I thank my family – my parents Halil and Nursel, and my sister Sara Pınar who all gave me the greatest assurance and motivation, and whose support enabled this work to be completed.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

As computer systems, applications and user interfaces develop with extensive integration of various complex media types; and as their distribution and exposure rapidly continue to expand via the world-wide web, workstations and other network systems, visual communication of digital domains continues to change considerably. This research examines the "Desktop Icons" in correlation to contemporary developments, changes and expanding possibilities in the usage and integration of graphical user interfaces and outlines how the sophistication of icons reflects changes in the study of human computer interaction and graphic design as a medium of digital technologies.

Although the G.U.I. (Graphical User Interface) is the most common and widespread interface used today within the field of Human Computer Interaction, it is arguably not the most successful, or even so, is only one[1] of the many existing and imaginable ways that communication between the human user and the computer can be achieved. Many other possibilities have been researched, experimented, and are being used and

[1] See for example, Hadley, Daniel A. "Alternative User Interfaces." 6 Dec. 1999. New Jersey Ins. of Tech. 12 June 2003 <http://eies.njit.edu/~turoff/coursenotes/CIS732/samplepro/dan.html>.

developed for a great many number of purposes throughout Human Computer Interaction. My research does not intend to argue in favour of claiming the G.U.I. as a superior means of interaction or an ideal interface. Rather, this research intends to analyze the evolution of icons[2] to highlight and discuss how graphic communication within computer interaction has been so favourable among users of all kind, and how the G.U.I. has shaped over recent time. This study will not neglect the issues mentioned just precedent, and will include notes on past and ongoing issues on the development of interfaces and critical factors that compromise Human Computer Interaction, but will try to limit its boundaries to visual and graphical entities concentrating mainly on features that regard iconic property.

One problem regarding the build-up of this paper was to properly formulate and place the subject of icons into a meaningful array of information situated in one logical subject area. Since the field of Human Computer Interaction is highly interdisciplinary, related to material from many different and contrasting areas of academic and professional vocations, it was difficult to place the research of icons into a single perspective and attempting to neglect or undermine the least amount of information available at hand. Computer Interaction is equally related to a vast number of areas such as Computer Science, Visual Communication and Graphic Design, Cognitive sciences, Semiotics, and among many

[2] Derived from the idea that the icon is considerably the most characteristic element of the Graphical User Interface, allowing most for transformation, customization and diversity.

others. Even broadening the subject to User Interface design and Software design that relates more adequately to this study, the same diverse number of interdisciplinary fields are concerned.

> "The question "Who is the designer?" often means "Who understands the whole problem, and who has all the skills needed to solve it?" For software design, the answer is that no one person can be "the designer," because software design is too complex, requires too many skills, and crosses too many boundaries between disciplines. As a result, several disciplines must be brought to bear. The difficulty in this is that each brings its own methodology and point of view, and the confluence of these can blur the vision needed to archive good design...Putting the user into the process helps put things back into focus. User-centered design methodologies may be innately interdisciplinary because no single repertoire of skills can adequetaley adress the complexity of users and their tasks." (Alben, 20)

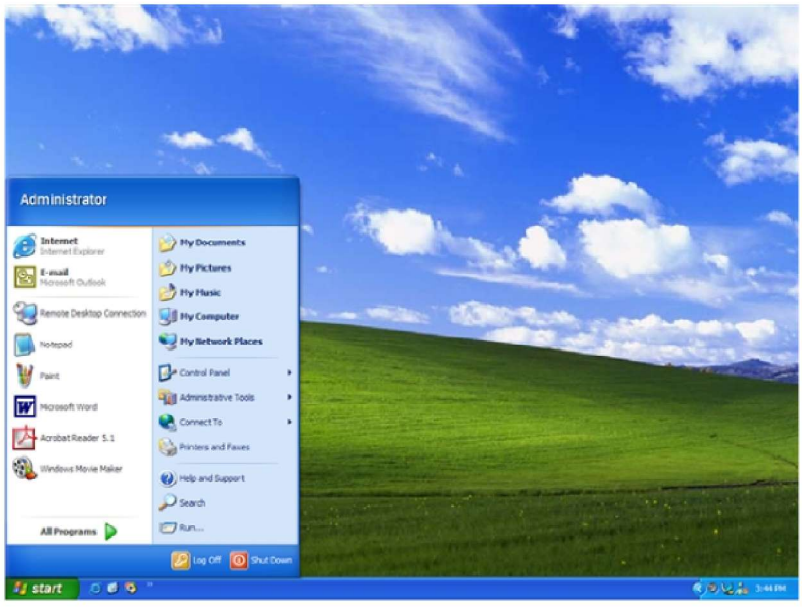
While conducting this research another problem that I faced was that I found great difficulty in finding appropriate information that was relevant to the framework that I had opted to take, and also because most sources, examples and especially visual materials related to the subject consisted of less formal, more commercial and on a large-scale non-academic material. The critical subject of a computer icon could have well been taken as part of a string of literature examining a "re-defining" of the medium of multi-media, computers, technology and user experience; whereby the basis of the problematic would consist more of about how necessary the use of icons within a H.U.I. (Human User Interface) is or not, and their degree to how much they would suffice to fulfill their intended

purposes. However, my intentions are to examine a more ‘popular,’ more widespread and a significantly more commercially affected and controlled set of icons, primarily due to the fact that they are both statistically and profoundly exposed to a greater many number of users and continue to rapidly expand bringing with this the expectancy to have yet reached their maximum exposure.

Consequently, the examples that I refer will be primarily those belonging to more widespread consumer operating systems [Microsoft Windows (XP) and the Apple MacOS (X) (Figure 1) in particular] and applications. My reasons in depicting such distinct samples neither come from the fact that they convey the most successful, aesthetic or functional set of models, nor because they are the most problematic examples to be covered in this perspective. While there may be notions that these examples are not merely successful examples and are not on the contrary it is perhaps the very thought that they are the least and the most problematic examples.

The first sections of this study briefly outlines some of the critical issues of Human Computer Interaction, the significance of the Graphical User Interface amongst this domain, and some of the key elements that form the desktop environment of operating systems and applications. The idea is to give an informative portrait of the underlying foundation that desktop icons reside amongst and a summary of the main elements that work cooperatively with icons for user interaction.

a.



b.

Figure 1: The latest widespread commercial Graphical User Interfaces:
a. Apple Mac OSX.
b. Microsoft Windows XP.
Source: Anonymous, Microsoft XP Design Guidelines.

After comprehension of the foundation, this study presents an analysis of desktop icons to define both an introductory historical background of the term, a look at the visual qualities and structural attributes to portray an anatomy of icons. Finally, the argumentative part of the study included in the latter parts deals with a comparison of textual and iconic communication that tries to formulize the issues regarding the functionality, perception, structure and necessity of iconic and textual elements in the framework of Human Computer Interaction and visual communication theory.

2. OVERVIEW OF THE HUMAN USER INTERFACE

After the invention of computers, the first interfaces consisted of keyboard inputs and text based commands as means of employing commands, communicating and task performing on machines. These developments later on gave way to the creation of menu based interfaces that used pointing devices like the mouse, and eventually developed into being fully visual Graphical User Interfaces, common in most applications and platforms today, and composed of hybrid usage of both text and graphical elements. There have been numerous attempts at developing ideal interfaces, in the most Utopian sense; interfaces that would appeal to and satisfy all the needs and desires of every user and able to form ground to accomplish every task necessary. Most of these attempts have however failed, some due to their failure to neglect certain usability features, some because of their delayed delivery and a few because of their over-complex features to pertain to average user's needs. Also coming from the fact that consumerism is at large the largest exposure of Graphical User Interfaces is achieved by large commercial enterprises that are apparently more involved in products that are sold rather than making the most functional ones. However the fundamental structure of most

interfaces have evolved from the first examples, and certain aspects remain unchanged or modified only very little since then.

The historical development and the commencement of the Graphical User Interface took place during the late 1970's at the Xerox Palo Alto Research Center (PARC), where many developers collaborated to produce research in information and physical sciences. The first example of the usage of a pointing device and a visually oriented approach to control the computer was prototyped at the Palo Alto Research Center along with other innovations regarding studies related to Human Computer Interaction. In 1982 the first personal computer (Figure 2a) was in working condition and in the years following the first WYSIWYG (What You See Is What You Get) editor, a commercial mouse for input, a graphical user interface (G.U.I.), and bit-mapped display, menus and icons and a link to a local area network (Figure 3). The use of multiple windows and the aid of a pointing device controlled cursor were first evident in the "ONLine System (NLS)" as part of the "SRI (Stanford Research Institute)" Project led by Douglas Engelbart in the 1960s. Graphical display systems for computers were developed in the "SAGE (Semi-Automatic Ground Environment)" Project and Ivan Sutherland's "Sketchpad" (Graphical User Interface, Foldoc). Alan Kay is also noted as being the founder of the graphical user interface by implementation of iconic and graphical representations of computing functions, or the icons, folders, menus and overlapping windows conventionally used in the desktop metaphor. (Multimedia, 122)

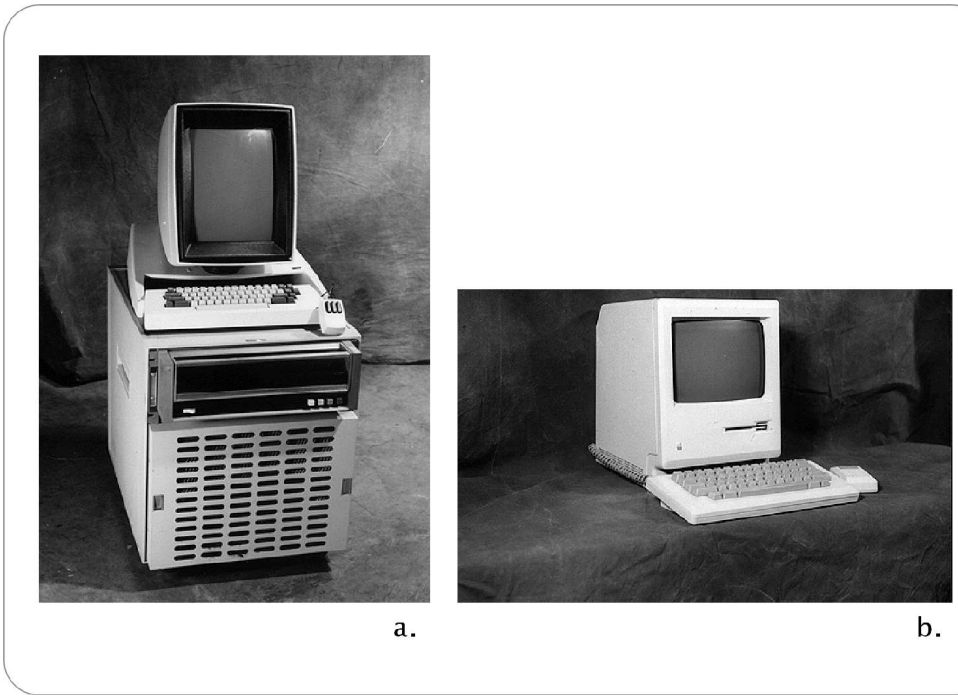


Figure 2: The initial desktop computers:
a. Xerox Parc Alto Computer
b. Apple Macintosh (1984)

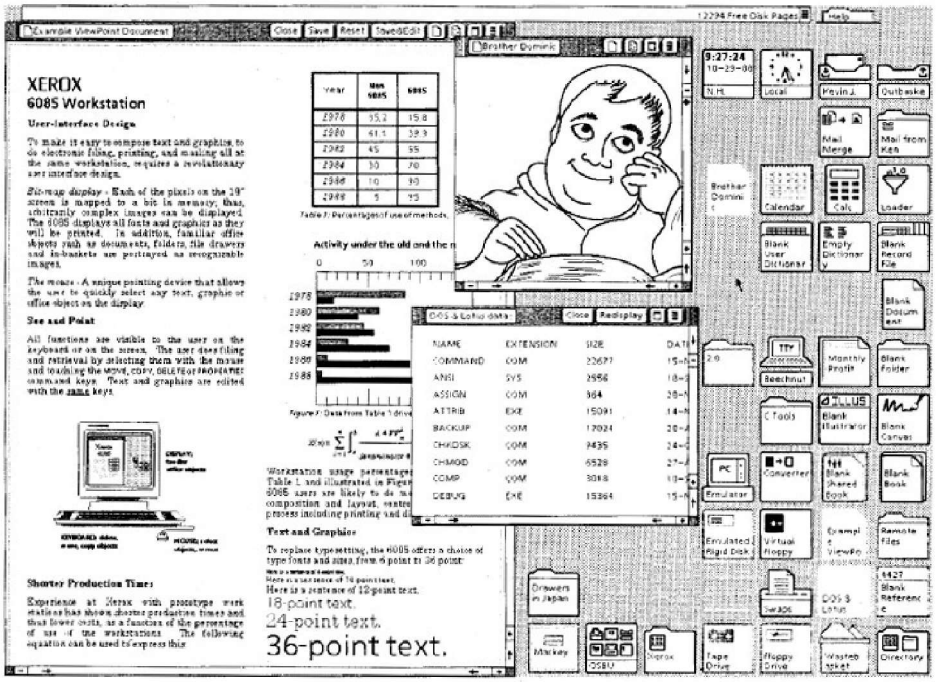


Figure 3: The Xerox Graphical User Interface

After the developments at Parc, Jef Raskin and a group of former researchers of Parc joined the Apple Macintosh Corporation and continued their established ideas on Graphical User Interfaces as a commercial product, which in effect lead to the first widespread usage of the desktop computer. In 1984 the first Apple Computer (Figure 2b) was released and is considered to be the first successful implementation of the graphical user interface as a commercial product. During the same years, Microsoft Corporation implemented this idea to their own operating system and the desktop computer with the G.U.I. which at first did not supersede the usage of the Macintosh operating system (Figure 4) but after the release of Microsoft Windows 3.0 in 1990, eventually to Windows 95, 98 and XP (The latest Microsoft Operating system that has been officially released at present) diffused to become as widespread as it is today. Figure 5 shows the first Microsoft Operating systems. (There are many other operating systems both text oriented and graphically interfaced, some window-based examples including X-Windows, Acorn RISC OS, Nextstep, Linux, Solaris, etc.

There are a number of different terms and definitions when it comes to commenting on computer interfaces. The following three are the fundamental terms that are used when dealing with issues belonging to computer interfaces; these are H.C.I.[3] (Human Computer Interaction), the H.U.I. (Human User Interface) and the G.U.I. (Graphical User Interface). H.C.I. (Human Computer Interaction) is the general name for the study of

[3] The term H.C.I. is also used on some occasions for abbreviating Human Computer Interface, synonymous to Human Interface.

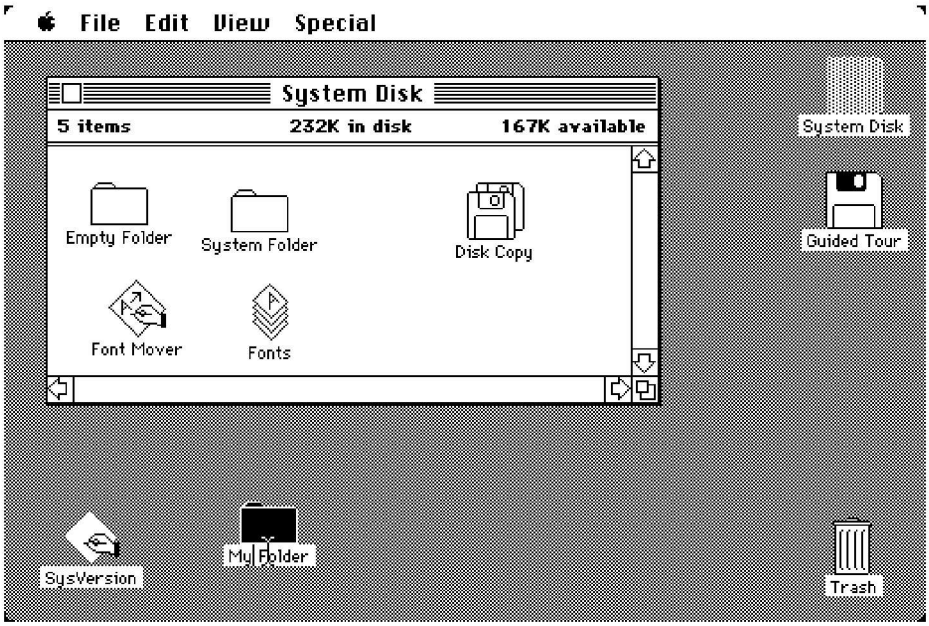


Figure 4: The MacOS 1 Operating System

how humans interact with computers and how efficient these interactions are to what extent. In this respect the H.U.I. and the G.U.I. can be considered part of the study of H.C.I. This field is a relatively new area in research, most certainly interdisciplinary and is usually defined by making use of a certain model of thought or perspective to comment on it.[4]

Considering the H.U.I. (Human User Interface) amongst the study of Human-Computer Interaction, it is specifically concentrated on the idea of an interface, and deals primarily with how communication between the user and the computer take place. The G.U.I. on the other hand is a more

[4] The ACM SIGCHI Curricula for Human-Computer Interaction makes clear notice of this point in its definition of H.C.I.. "Definition of H.C.I.." *ACM SIGCHI Curricula for Human-Computer Interaction : 2. Definition and Overview of Human-Computer Interaction.* 20 April 2003. <http://sigchi.org/cdg/cdg2.html>

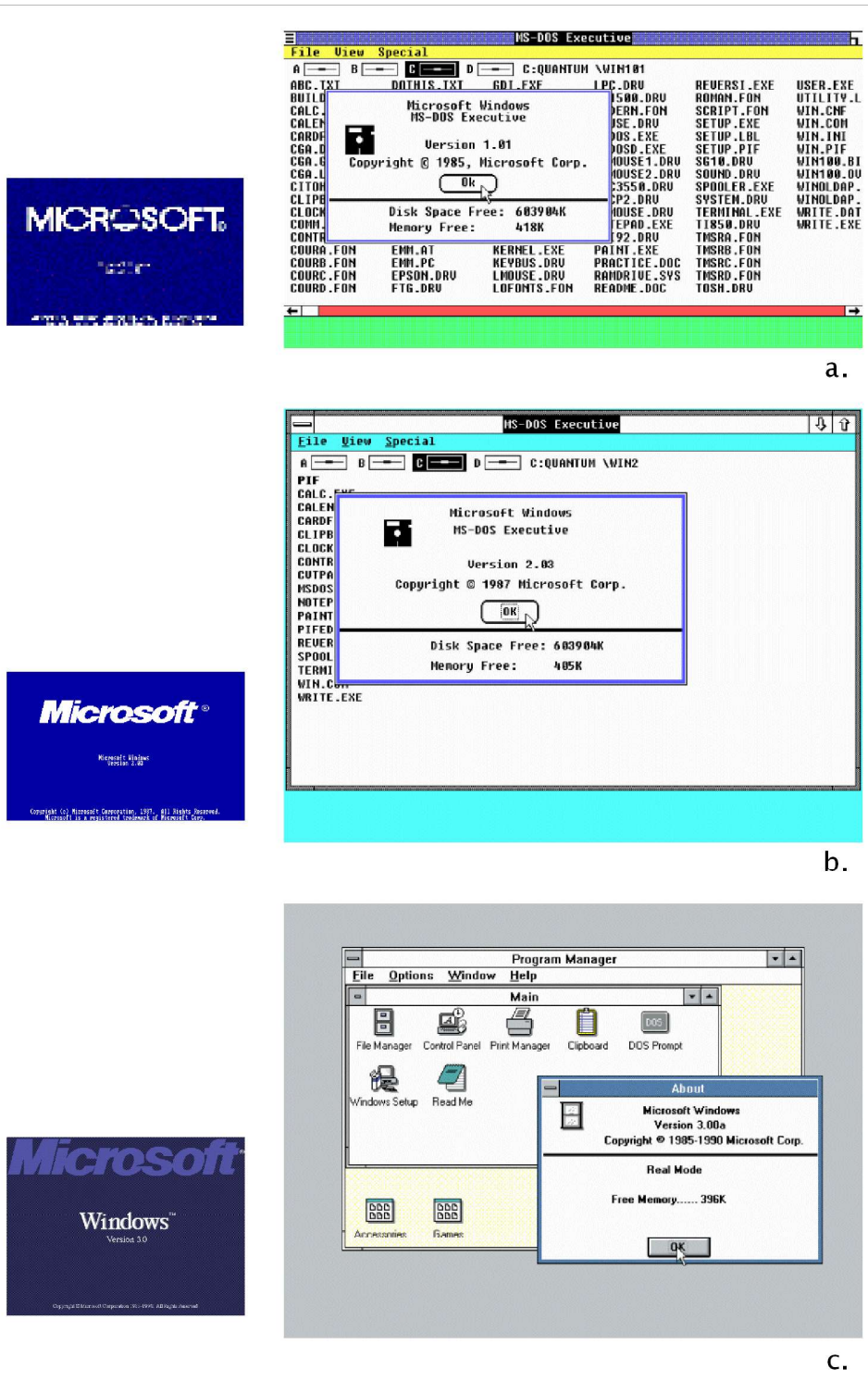


Figure 5: The first Microsoft Operating Systems:
a. Microsoft Windows 1.
b. Microsoft Windows 2.
c. Microsoft Windows 3.
Source: The GUI Gallery

specific term, an entity among H.C.I. dealing specifically with visual and graphical elements where computer interaction is concerned. As the H.U.I. stands for a more generalized concept (although in some cases denoted for G.U.I.), it does not have to consist strictly of a visual representation system as the G.U.I. does.

The G.U.I. as its title suggests, is strictly composed of graphical components and functions on the basis of visual entities. The importance of a visually oriented interface is that it enables the user to omit learning computer languages and commands to use the computer, but rather use images as a negotiator between man and machine. This aspect is perhaps what is most relevant and parallel to the subject of this study; the fact that icons are signifiers and tools to start and manipulate applications and commands, and the tasks employed among them becoming more and more diverse as computers develop. The Graphical User Interface is a dominant feature of most applications and operating systems today, and has been developing for the past decades, changing considerably.

When the preliminary Graphical User Interface based operating system was developed, it was often referred to as a WIMP G.U.I., where the abbreviation stands for the primary elements that compose the Graphical User Interface, i.e. Windows, Icons, Menus and Pointing device. While summarizing and defining the Human User Interface I will be faithful to this approach (The main elements of the latest Graphical User Interfaces remain to have these elements as fundamental entities).

2.1. Windows

The idea of having a window as an interface is most apparent in the issue of Graphical User Interfaces, since a window defines a frame, distinct boundaries and a strict implication for a platform or application, creating the base for graphical objects that represent code, text and other non-visible features that computers possess. The window is the foundation component of the Graphical User Interface (The Microsoft Windows operating system has adopted this element as its sole unique name due to this occurrence), all other graphical objects are nested amongst it (scroll bars are located to the sides to control its dimensions, menu bars usually at its top to create a navigation pane, and icons belonging to coherent windows specify aliases and commands that are related to the contents and category of their possessive window) and it operates fully as the main interface element.

Most operating systems and applications both past and present have had similar typical features, which have been modified little over time. In this respect, as seen in Figure 6, a typical window consists of a main body space for the items that it bears and a top and bottom space for additional information and tasks, similar to that of a header and footer in text based documents. In the case of Microsoft Windows, these sections are the Title and Menu bar and the Status bar respectively.

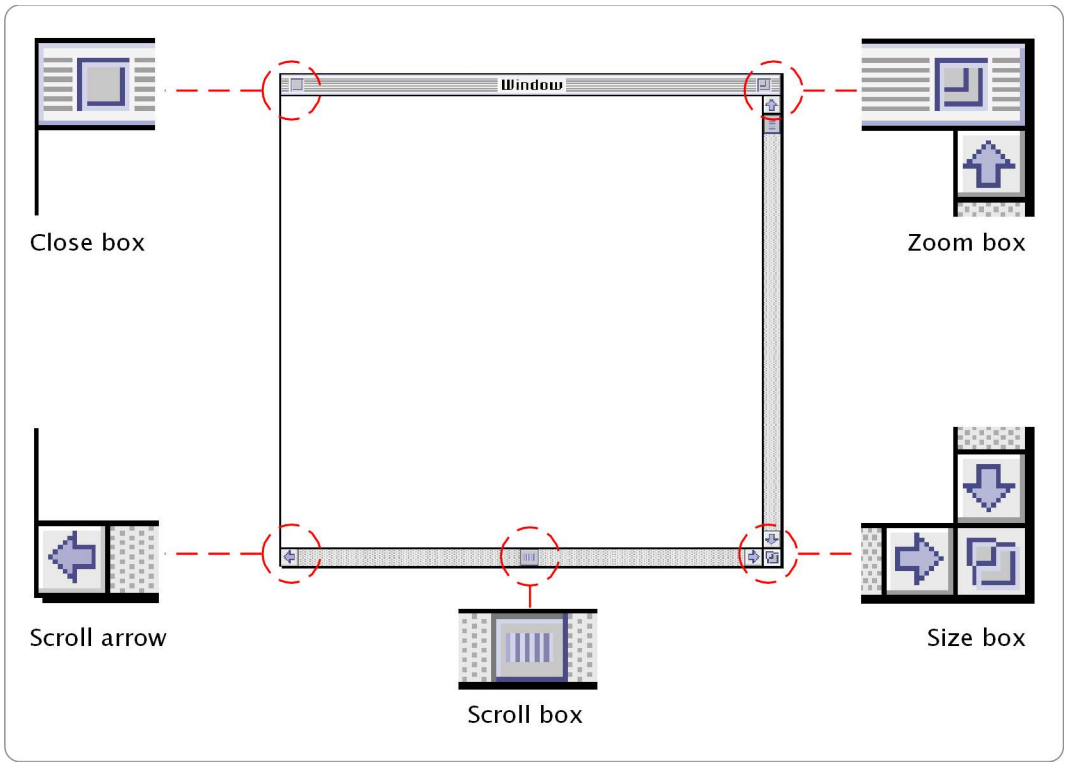


Figure 6: A classic window and it's components.
Source. Apple MacOS8 Human Interface Guidelines.

The Title bar contains both the names of the documents or applications that are open in the window along with a small icon and name for identification. The Menu bar, usually located directly below the Title bar is on the other hand where menus are classified into hierarchical categories to perform all main commands and functions. The Status bar at the bottom conveys information related to the contents of the window such as how many files are present in a directory represented by the window, the name of the file selected, the file size, etc. As I mentioned earlier, these elements have had little change overall since the beginning of their appearance such as using separators, including different sized icons and their look (shading, dimensionality, colour, and so on) (Figure 7)

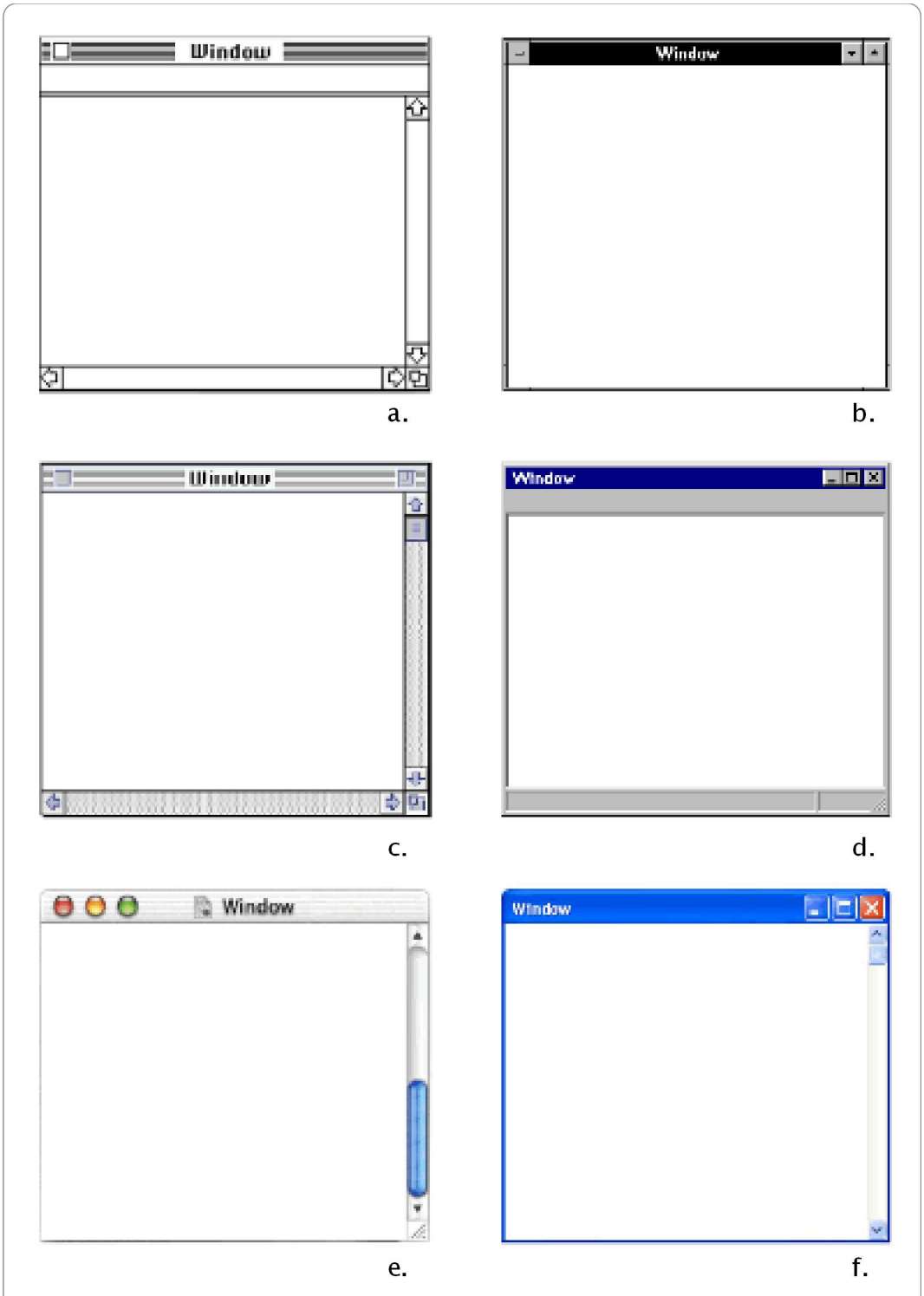
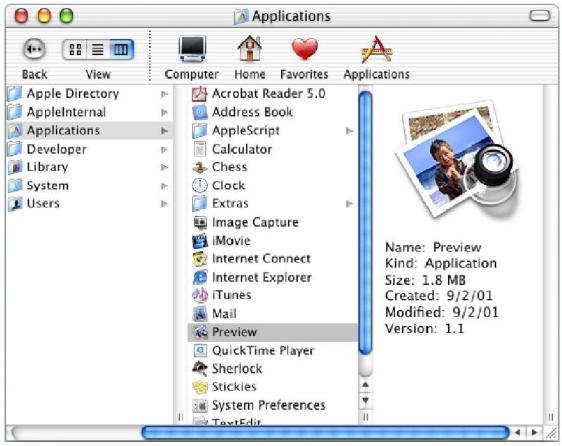


Figure 7: Various window examples over time
a. Apple Mac OS 1.0
b. Microsoft Windows 3.0
c. Apple Mac OS 7
d. Microsoft Windows 95
e. Apple Mac OS X
f. Microsoft Windows XP

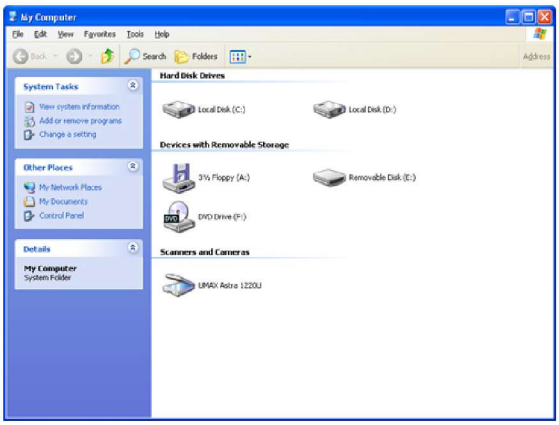
One of the oldest features of a window is its ability to resize. Usually located at the top right corner there are the maximize, minimize and close buttons to hide or shrink the whole window into the operating systems general menu bar (Taskbar as it is called in Windows, or Dock as a newer term for the MacOSX) and ranging from platform to platform various other preset sizing and locating features. Again, usually the bottom right includes a “gripper” or a “size box” that lets the user manually adjust the size of the window into whatever dimensions are needed. Both the appearances and exact functions of these elements do vary between different operating systems and applications but function in very similar properties for the same purposes, hence I find it unnecessary to explain each and every one separately.

As mentioned in the beginning of this section, the window can be considered the primary element of the Graphical User Interface. Speculating on the characteristics of windows will reveal that it renders a virtual visual space and a platform for visual entities to represent and operate commands, instructions and applications hence forming a basis for visual communication and interaction driven by graphical elements. In terms of innovative development, it has remained pretty much unchanged over time except for visual enhancements and functional properties such as colours, textures and sizes. The fact that other elements residing amongst windows have evolved also contributed to overall window development. (Figure 8)

a.



b.



c.

Figure 8: Specialized Windows
a. A textured application window in Mac OS X.
b. Applications window for browsing files in Mac OS X.
c. An explorer window in Microsoft Windows XP.
The 3 figures depict examples of features and elements newer in present operating systems and applications.

2.2. Icons

Although the subject of this thesis primarily concentrates on the issue of iconic evolution and visual communication related to the icon, this subcategory describes the fundamentals of the icon just to give a brief definition and description of this element as a context of its belonging to the Graphical User Interface and as a part of Human Computer Interaction. The latter chapter will be a more detailed analysis describing in depth all features of the icon, and thorough explanations will be given therein. This section of the thesis is placed here in order to follow the preliminary structure of user interfaces, i.e. the WIMP GUI model that was initially developed in order to form a hierarchical definition of the Graphical User Interface.

Icons are pictorial and graphic representations of files, folders and applications used to employ commands. The use of icons were employed to substitute conventional command prompting by textual input devices such as the keyboard to graphical point and command systems aided by pointing devices. Icons were derived from providing the user with advantages and features of using graphical entities, to use a different approach to comprehensibility, usability and task performing. Figure 9 shows a screenshot of a folder icon in an icon editing program. The various sizes (left in figure), colour palettes (right in figure) and the enlarged icon in the grid emphasize characteristics of the icon.

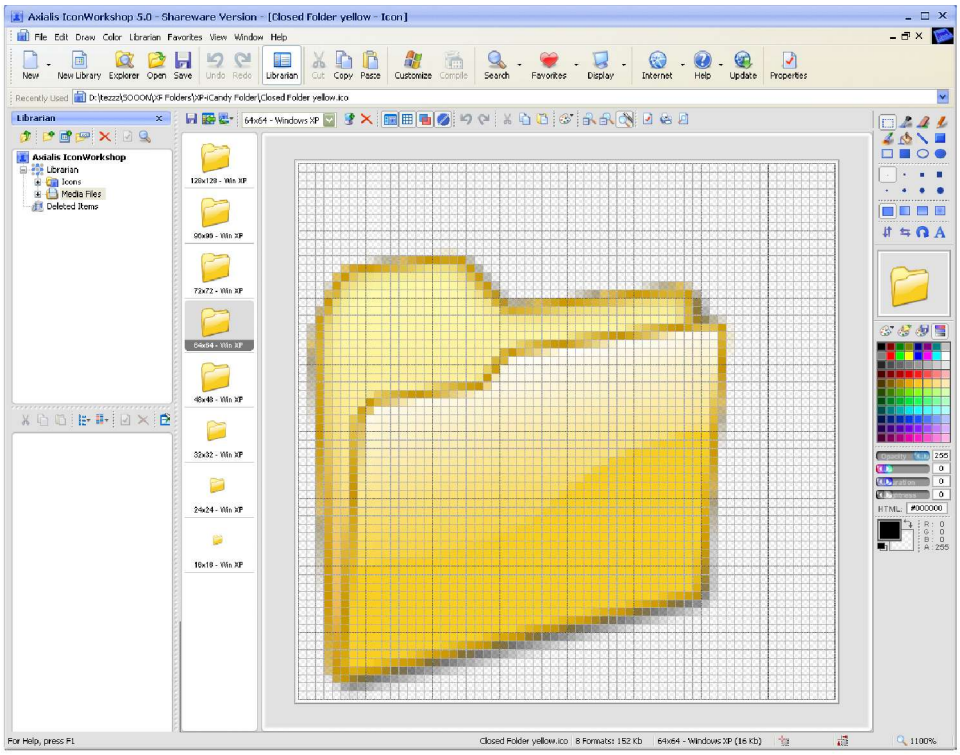


Figure 9: Screenshot of a folder icon opened in Axialis Icon Workshop.

Apart from regular system and application icons, customization of icons made by graphics designers, software designers, etc. can also be noted to be present. Users can customize their working environments with these icons. (Example Figure 10) Many icons are distributed, shared and sold as or with software packages and through the Internet contributing to icon design. Figure 11 Shows examples of customized icons.

Icons differ from one another both with respect to (a) the features of the corresponding commands that they represent; and (b) how the command-features are represented. Icons directly or indirectly represent either the operations a command carries out (called "command-operations"), or the objects operated on (called "command objects"), or both; and the objects/operations are commonly represented by conventional or abstract symbols, or by depictions of relevant objects. (Hemenway, 21)

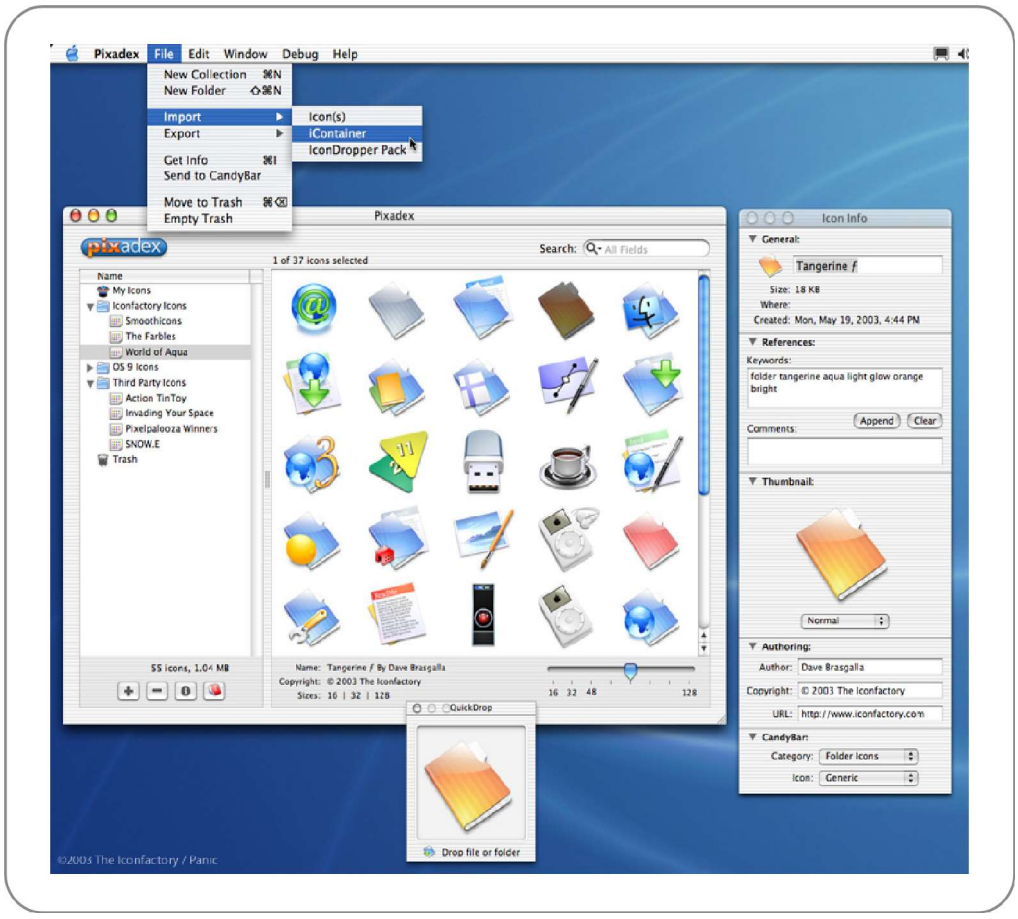


Figure 10: Icons inside an icon customization application window in MacOSX GUI. Source: Iconfactory

Figure 11: Various custom-designed alternative system and application icons. Source: Iconfactory

2.3. Menus

Menus (Figure 12) are one of the oldest methods of achieving tasks in applications. A menu can be defined as a list from which the user can select operations to be performed. This is achieved by the use of a pointing device such as a mouse as stated earlier, and this method of computer use was the transition to Graphical User Interfaces. Menus are often part of the various bars located inside windows (also described in the previous sections) and are usually nested to the top, bottom or the sides. Similar to the nature of windows, menus have also remained similar throughout the development of the Graphical User Interface. Apart from visual changes, colour and stylistic conventions however, one main evolution is apparent which is the change in the hierarchical structure of

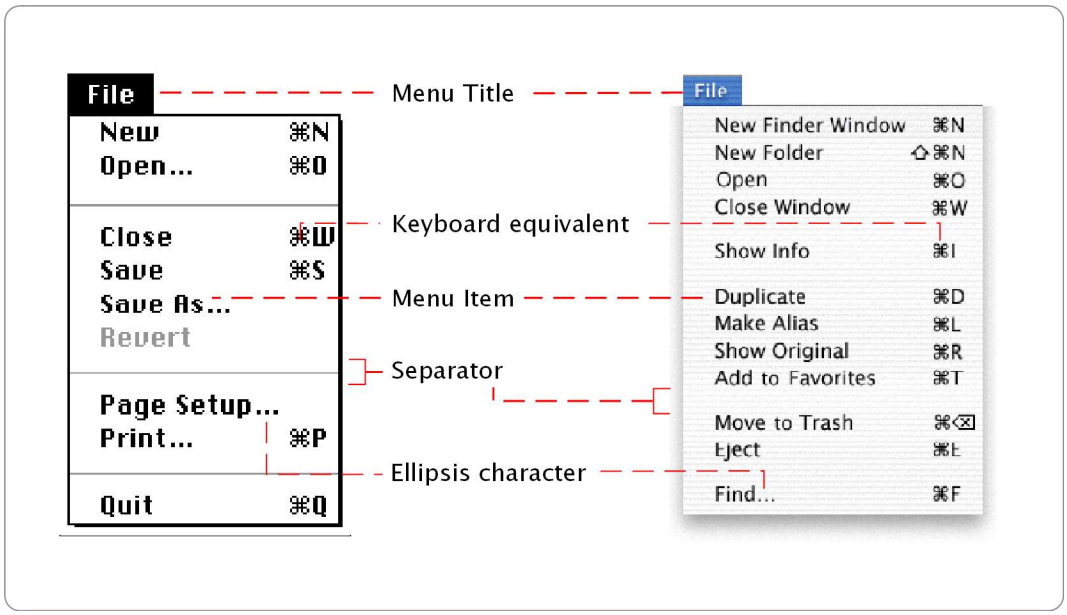


Figure 12: A typical pull-down menu and it's components.
Source: Apple Aqua Human Interface Guidelines

the menu. In older systems and applications, menus would consist of simple single level lists that would either open a corresponding window or operate a command. This functionality has expanded to include subsets of sub-menus and commands, expanding to deeper levels and providing access to functions that are of lower hierarchical level.

Types of menus include pull-down menus and context-sensitive menus as most common. Pull down menus (Figure 13) consist of content that reveal when the user activates it with the use of a pointing device and the menu items appear below the title. After clicking either on the item or somewhere else, the menu automatically closes itself. The items in the menu can be selected by dragging the mouse from the menu title to the item and releasing the mouse or by clicking the title and then the item. Context-sensitive menus (Figure 14) on the other hand are operated by secondary buttons of the pointing device (such as the right button of a typical mouse) and include quick access to fundamental functions that are usually from the menu bar. However, there is no rule to the operations and contents of any of the menu types and again they vary from platform to platform. For example a typical Microsoft Windows application would call a context-sensitive menu by right-clicking, whilst a Macintosh application would bring up the menu by pressing the mouse button (The majority of mice used for Macintosh systems are composed of only one button) for a longer duration of time, as with other applications, no context-sensitive menu may be present.

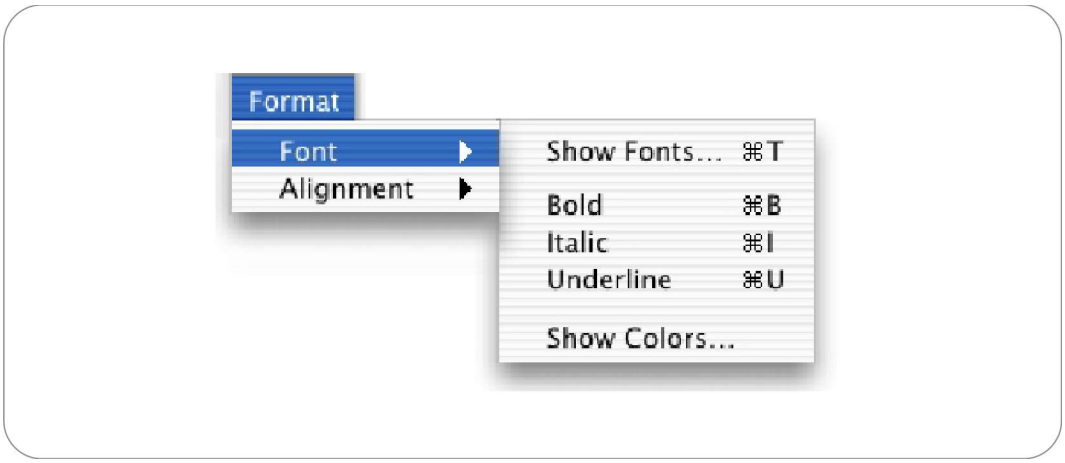


Figure 13: A pull-down menu with a sub-menu
Source: Apple Aqua Human Interface Guidelines

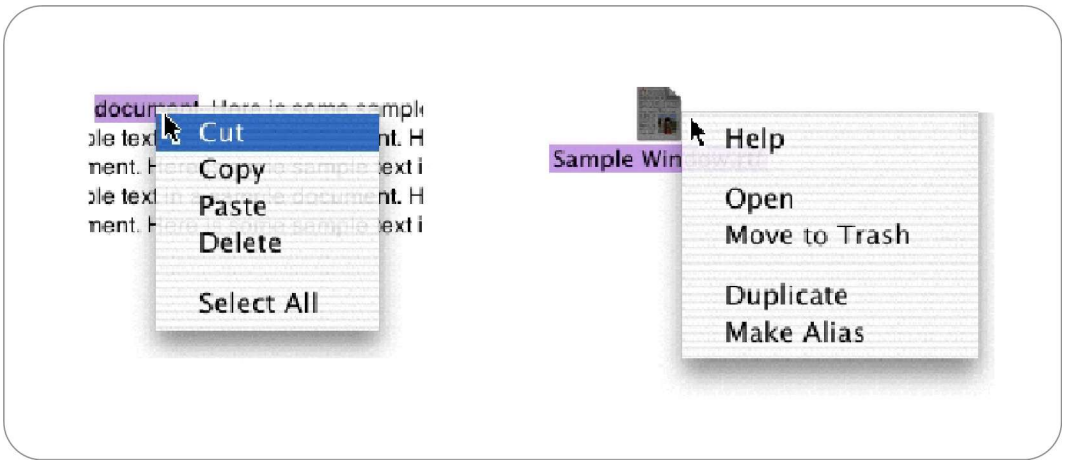


Figure 14: Examples of context sensitive menus
Source: Apple Aqua Human Interface Guidelines

2.4. Pointer (Pointing Device)

A pointing device is a hardware component that allows a user to input spatial data to a computer. In the case of Graphical User Interfaces the pointing device allows for physical gestures to control and provide data to the computer. The current types of pointing devices include the Mouse, Tracker balls, Trackpads, Lightpens, Digitising tablets and Data gloves. (Figure 15) Among these devices, the mouse is the most commonly used (Developed in 1968 by Douglas Engelbart). The gestures that control pointing devices consist of a variety of movements, for example in the case of the mouse, the device allows the user to point, click and drag. The physical movement of the mouse on the desktop surface is represented on-screen in a meaningful relation and the activation of buttons and switches on the apparatus cause other functions to operate.

A roller-ball mouse (the most widespread pointing device that is used) includes a ball that lies in an opening at its bottom which rotates synchronous to the mouse's movements. A formation inside the mouse including shafts and sensors that detect and measure how much the ball, hence the shafts, have rotated send this information to the computer through a wire or an infrared device enabling it to render the mouse pointer on the screen mimicking the mouse's movements, in other words following the hand gestures of the user. This action is operated on the surface of the desk, usually on a mouse mat. The mouse usually includes a number of buttons from 1-3 which are used to click and operate

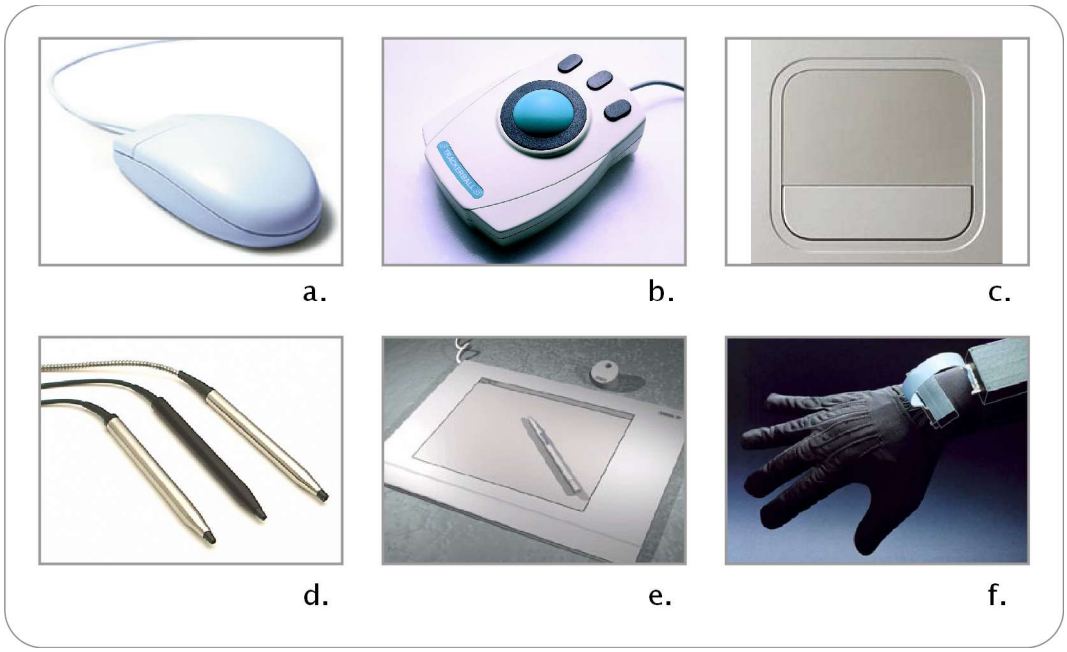


Figure 15: Examples of pointing devices:
a. Mouse
b. Rollerball
c. Trackpad
d. Light pen
e. Digitizing tablet
f. Data glove

applications, open files, etc. The primary mouse button (usually the left one on a two button mouse) is operated to activate represented objects on the screen, whilst the other button(s) usually operates secondary functions (differing from system to system and depending on the graphical user interface type) such as opening context-sensitive menus, modifying selections, pasting texts and commands of a similar nature. Mouse with fewer buttons may require a longer duration of depressing or a combination of mouse and keyboard functions to perform the same

tasks. A Mouse may also include wheels for scrolling, and software to setup the mouse to perform special actions and changing the functions of its buttons.

There are five main gestures that can be performed with a typical mouse. These are point, click, double-click, right-click, and drag. These fundamental gestures allow the user to perform most primary operations on any application of system task under a Graphical User Interface. Apart from specially programmed or scripted applications, the sole movement of a mouse pointer has no function until positioned or gestures over a menu or an icon or other object.

Pointing devices are usually standard equipment that are used with most Graphical User Interfaces. Portable computers such as Laptops and Notebooks generally include other pointing devices such as a Trackball, Touchpad or Trackpoint which have the same usage as a mouse but are more space-efficient. Other types of pointing devices that are derived from the conventional mouse are the infrared mouse, the foot-controlled mouse, the optical mouse and similar devices.

2.5. Metaphors

The previous sections explain the basic elements of a Graphical User Interface, elements that have constructed the basis of most Operating systems and applications today. The WIMP GUI model consists of Windows, Menus, Icons and the Pointer in terms of physical elements to compose the skeleton of an Interface. However there is also a conceptual property of the Graphical User Interface, that is the use of metaphors to convey information and compose a representation system to employ means of conveying information and tasks amongst the user Interface, also a very common attribute in today's systems and applications.

If we consider the computer as a product or tool that is used for storing, manipulating, and modifying digital data, the existence of merely graphical elements is not enough to achieve an interface for interaction between the computer and the user. A hierarchical structure and a model for arranging, organizing and rules are needed to perform tasks and maintain computer use as intended. The foundation of the Graphical User Interface depends on the desktop metaphor.

Before the usage of the computers for work and personal use, for publishing, data manipulation, and so forth, all tasks were done tangibly in a more physical manner, i.e. writing on paper, organizing on files, typing on typewriters, figuration and drawing on various materials, and a

number of methods in all fields that had no digital substitute. The concept of metaphors transported these tasks onto a simulated virtual environment onto the screen or other viewing device.

The name 'Interface' stands for an in-between mechanism of translation, a tool for controlling a mechanism and a communication device translating actions of one source to another. The Graphical User Interface was developed in this order to mimic the activities of a “desktop” and all functions were developed in correlation to elements of the desktop and their functioning conditions. Perhaps the most cliché example to this use is the text-editors which are based on the typewriter metaphor, and try to adapt old features of the typewriter along with new features that are not possible in physical conditions to employ a tool which is more advantageous to older technologies and machines. Although alternative approaches to the desktop metaphor may seem like an inevitable advancement, this does bring with it problems that are regarded in the following parts of this study.

After around 30 years of Human Computer Interaction development and the rise of the Graphical User Interface, the world is now at a point where nearly all tasks are accomplished by computer use, where communication, work and leisure takes place on the computer screen and User Interfaces are a compulsory part of many people’s lives. Even though developments can be easily seen in the technological sense and significant change can be traced back to very recent past, recent criticisms tend to regard the

progression of Human Computer Interaction as being more negative than positive. Jakob Nielsen, and Don Gentners essay entitled the “Antimac”, starts with the following words,

> “At recent user interface conferences, several speakers have lamented that the human interface is stuck. We seem to have settled on the WIMP (windows, icons, menus, pointer) model, and there is very little real innovation in interface design anymore.”

In this article, Nielsen and Gentner give reference to the Macintosh guidelines, and criticize that the metaphors based on real instances from daily life can often lead to disadvantages and damage navigation and interactivity, limiting and slowing down user experience. They depict the examples that use 3D virtual spaces in opposition to the common desktop metaphor, but does not speak in its favour. In this respect their criticism regards the idea of using metaphors of any kind, and if overstated, perhaps to the idea of whether using a Graphical User Interface is profound or not.

There have been attempts to use different metaphors, even by Microsoft (Figure 16), owner of the most widespread operating system, but problems of usability never enabled it to succeed as a commercial product. To brief the problems that they show about metaphors, the following can be summarized; firstly the functions that are not possible in real life circumstances (the reference for the metaphor) are often easily

Figure 16: Microsoft BOB, a product conveying an alternative approach rather than the desktop environment, but without success to stay on the market.
Source: The GUI Gallery

unnoticeable when represented in the user interface because of the very fact that they are referenced to something that cannot be done. Nielsen and Gentner give the examples of the following:

> The three classic problems with metaphors are:
>
> - The target domain has features not in the source domain" (e.g., telling the user that "a word processor is like a typewriter would not lead the user to look for the replace command)."
>
> Some other problems that they relate to can be quoted as follows:

- The source domain has features not in the target domain (a typewriter has the ability to mark up any form you receive in the mail, but a user trying to do that on current computer systems will normally fail).

- Some features exist in both domains but work very differently (the treatment of white space representing space characters, tabs, and line feeds is very different on typewriters and in word processors). Therefore, users may have trouble seeing beyond the metaphor to use the system in the ways it was intended. It is possible to design interfaces to map very closely to metaphors (e.g., Bob Mack's NOSE-no-surprise editor-really did act like a typewriter in all respects), but those interfaces will often be very low-powered and suited mainly for walk-up-and-use situations. (Gentner, 72-3)

This article refers to many features that the desktop metaphor has forcefully tried to adapt within itself in order not to give way to any restrictions compared to the tangible environment, causing misinterpretations and clumsiness in effect. As an example Gentner and Nielsen also draw attention to the working principles of the trash can icon (Figure 17) , a metaphor of the wastebasket, where when all items disrespectful to where they physically are located end up in the same trash can folder before complete deletion, and although they seem to be located in the same volume, are actually in different trash can folders respectful to their initial physical location, hence when are deleted the operation takes place in all physical drives. This results in unnecessary operations and confusion on the user's behalf. “The desktop metaphor has enforced a limitation on the interface that does not serve the user's real needs.”

Figure 17: Common Trashcan icons (Empty and full)

One other important aspect of metaphors that they emphasize is the restrictions that metaphors cause for people both using and designing the interface. Since metaphors are directly related to analogy and their main purpose is to represent in some conceivable form, actions or objects related to real life situations, any situations bring limitation compared to alternative means that could result in operations that are much more efficient and meaningful. The traditional desktop may have been the tangible user interface of the 1980s and could be argued to be the best way of doing things, however today the metaphor is “tired” in their words and tasks, communication methods and items of the desktop need replacing in order to present the most up-to-date and efficient objects and tasks in using interfaces.

Many more ideas can be developed from these propositions rendering the desktop metaphor as exhausted; The generation that started using the first examples of graphical user interfaces and the concept of a desktop metaphor were adapting their real life skill into the digital platform. In

other words they were using the typewriter, using physical file cabinets and folders and were used to all the applications, objects, appliances and chores that the desktop environment simulate. However the new generation of present time, children especially have become directly acquainted to the older revisions and higher capable devices that have built up on the premature graphical user interface and desktop metaphor. Files and folders, typewriters, and all other common objects that are descendants of the desktop metaphor are no longer used anymore, are obsolete so the idea of representing exhausted, replaced technologies and analogies that do not refer to the “real world” any longer is deeply problematic in trying to improve, expand or evolve a user interface, hence to develop and progress in what is actually the aim and function of the user interface.

> The desktop metaphor assumes we save training time by taking advantage of the time that users have already invested in learning to operate the traditional office with its paper documents and filing cabinets. But the next generation of users will make their learning investments with computers, and it is counterproductive to give them interfaces based on awkward imitations of obsolete technologies. Instead, we need to develop new interface paradigms based on the structure of computer systems and the tasks users really have to perform, rather than paradigms that enshrine outmoded technology. The way to advance the interface is not to develop ever-more-faithful imitations of the desktop, but instead to escape the limitations of the desktop especially as computers themselves become ubiquitous [21] and are used away from the desk. (Gentner)

Although metaphors, the Graphical User Interface and hence the use of iconic communication to interact with the computer may have its problems, it is still the most widespread example of human computer interaction today, and when compared to alternatives is leading by far. Many users are overly familiar with visual interfaces and iconic communication to the extent that passing to textual or language based interfacing or other means of gestural interaction may be too awkward to be faced with. Although much can be imaginable as alternatives to the graphical user interface, it would take an extremely difficult path to be realized and just as pure textual interfaces have their advantages in certain cases over graphical and iconic interfaces and are far from diminishing, newer alternatives that are of highest promise will most likely fall into the same situation.

The graphical user interface has its limitations and has indeed become exhausted to a degree that is not to be disregarded. As can be expected without surprise, newer ideas on interfaces will most certainly replace certain criteria amongst the conventional graphical user interface and therefore also iconic communication, but just as the mutual relationship between images and text is concerned, these new innovations are most likely to adapt and implement themselves within the roots of current Human User interfaces.

3. DESKTOP ICONS

After the development of the first Graphical User Interfaces, the first set of icons were produced as pictorial representations of commands, files and folders, software and hardware, and were used for the first time in accordance with the other new visual features that were discussed in the first chapter of the study. (Parc) The concept of the Graphical User Interface was entirely new along with the use of computers regarding people that were not engaged distinctively in areas of computer research and development and had just superseded the conventional command prompt that was more familiar to software developers and computer programmers (Some users still prefer text only programming platforms due to faster commandment and excess of unnecessary functions). The concept of the desktop metaphor was a new one and users at that time were trying to get used to both using the computer and the interface, and getting used to a new way of accomplishing tangible tasks that would conventionally be done outside the presence of computers. The first set of icons along with the other Graphical User Interface elements, whether directly accredited to or not, were the first examples that started the expansion and current computing practices used today.

The early application of icons consisted of simple but effective 1-bit black and white illustrations occupying a very limited amount of space and performed simple straightforward tasks having both similar and different functionalities as compared to ones that are designed today. Present system and applications icons however, have the capacity of complex high resolution photo-realistic or photo-illustrative imagery; being capable performing multiple tasks and conveying different categories of information with a wide band of flexibility. Such an evolution is correspondent to the fact that user interaction, navigation and possibility of multiplicity in applications have changed substantially. Together with technical advancements, higher memory allocation, faster processing units and a build-up of software developments, iconic function has somewhat paralleled the capacities and functional purposes of multimedia applications and developing operating system capabilities.

The following segments of this chapter will include an in-depth definition of all aspects regarding the significance of icons in the Graphical User Interface and will define and interpret the characteristics and features that icons possess. First a brief definition will be given to explain what the fundamental functions and roles of the icon are and where the concept has been derived from, followed by the technical features and classification methods that are conducted for defining icons.

3.1. The Definition of the Icon

The term ‘Icon’ originates back to a considerable past and its roots come from the Greek word for ‘Image’. In its most traditional context, it is a term referred to a single image that represents a religious subject or person usually of divine importance, and exists most commonly in the form of paintings of holy people, places, subjects or ideas. Traditional “Icons” (Figure 18a) were especially used in the practices of the Christian Church and although their functionality and visual impact has changed compared to its past, it is still used and practiced today in this context. Analogical to traditional iconic imagery signifying importance, divinity and holy subjects, computer icons intend to perform the task of representing the most fundamental and important groups of applications, functions and commands on the desktop environment. Since the whole idea of the desktop metaphor and the essentials of the Graphical User Interface rely on, or more modestly put; have progressed upon representation and metaphors, the concept of the icon can also be regarded as a metaphor to its traditional meaning.

In its most general sense, the icon is a term used “to mean any image used to represent a person, place, thing or idea.” (McCloud, 27) McCloud talks about symbols, language and science related icons and pictorial icons in depicting examples in his definition of the icon. Pierce's definition as a semiotic model is also commonly used where he separates the icon, index and symbol in the framework of signs. (Commens, icon) In the case

a.

b.

Figure 18: The first signs of iconic representation
a. An example of a "traditional icon," as a portrait of Jesus.
b. Photograph of a cave painting from Lascaux Caves.

of the computer icon, many definitions apply and can be useful for analysis and speculation. The computer icon uses many mixed semiotic and visual communication attributes to signify meaning and perform functional tasks. Although Pierce separates the icon from the symbol for example, a computer icon can be in either or both forms. In this sense the definition of the icon will be related with representation in a more general sense, and in relation to this study, will also be closely related to writing.

Regarding the originations of iconic imagery, it becomes primarily important to establish the differences and similarities of some definitions between images and writing. Haramundanis compiles an account of the of icons' relationships with the origins of writing by referring to

documentation from Gelb and Diringer where she refers to as sources having “exhaustively documented” the history of writing (4) From these sources she mentions that Pictographic and Logographic scripts were developed before the presence of alphabetic scripts, hence concluding from her research, what may be called the initial forms of writing were closer to an imagery based system rather than a text based system, linking writing and text further to each other. (Example, Figure 19)

Figure 19: Examples of Chinese Ideographs, a writing system that has evolved from imagery.

Goppold's more concentrated account on the typology and history of writing states that the oldest forms of symbolic representation dated back to over 400,000 years ago appear in the Altamira and Lascaux paintings (Figure 18b) Although it is hard to depict what exactly these symbolic

figures represent, these examples of a combination of pictorial and abstract symbols are noted to be the initial forms of symbolic representation. In terms of the first forms of writing, the Mesopotamian, early Egyptian and Chineese are exemplified in Goppold's article. Haramundanis also gives the primary examples to these initial logographic scripts in her article as Sumerian, Babylonian cuneiform and Chineese logographs. (Figure 20, Figure 21)

When tracing back iconic representation to writing, it then becomes necessary to categorize the different typologies of writing systems in order to differentiate how images are related to writing systems and how they have evolved. In Goppold's article (quoted to be derived from Microsoft Encharta) these are first taken as “Limited Writing Systems” and “Full Writing Systems.” Limited writing systems intensely include the various pictographic and ideographic symbols capable of limited expression, whilst full systems include the categorizations of words (logographic), syllabic or alphabetic; able to fully express all language formulation. Although these systems are taken separately, they are related to one another essentially as once again highlighting how iconic representation is interlinked with writing. The Mesopotamian web site of the British Museum illustrates an instance of how Pictographs eventually evolved into cuneiform scripts in the early Mesopotamian culture.[5]

[5] See <http://www.mesopotamia.co.uk/writing/story/sto_set.html> to see the examples.

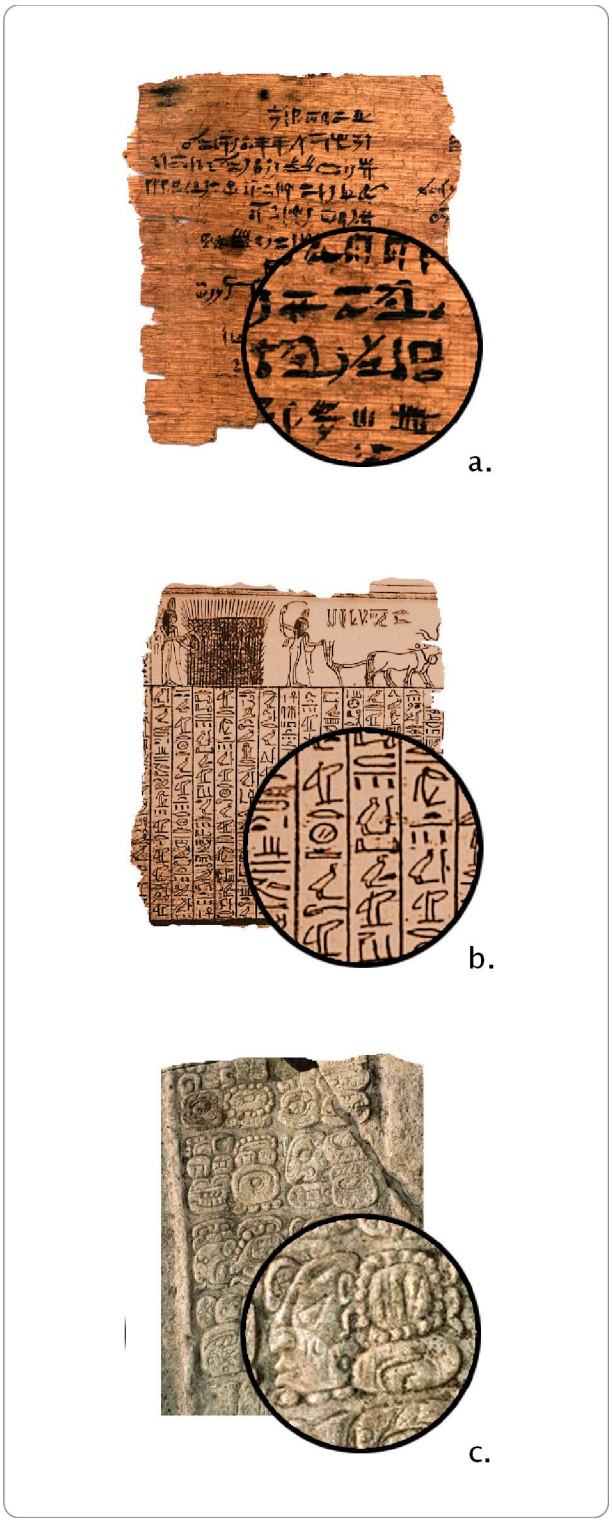


Figure 20: Examples from the first writing systems:
a. Early Chinese Script.
b. Egyptian Hieroglyphics.
c. Mayan Stone Carving.

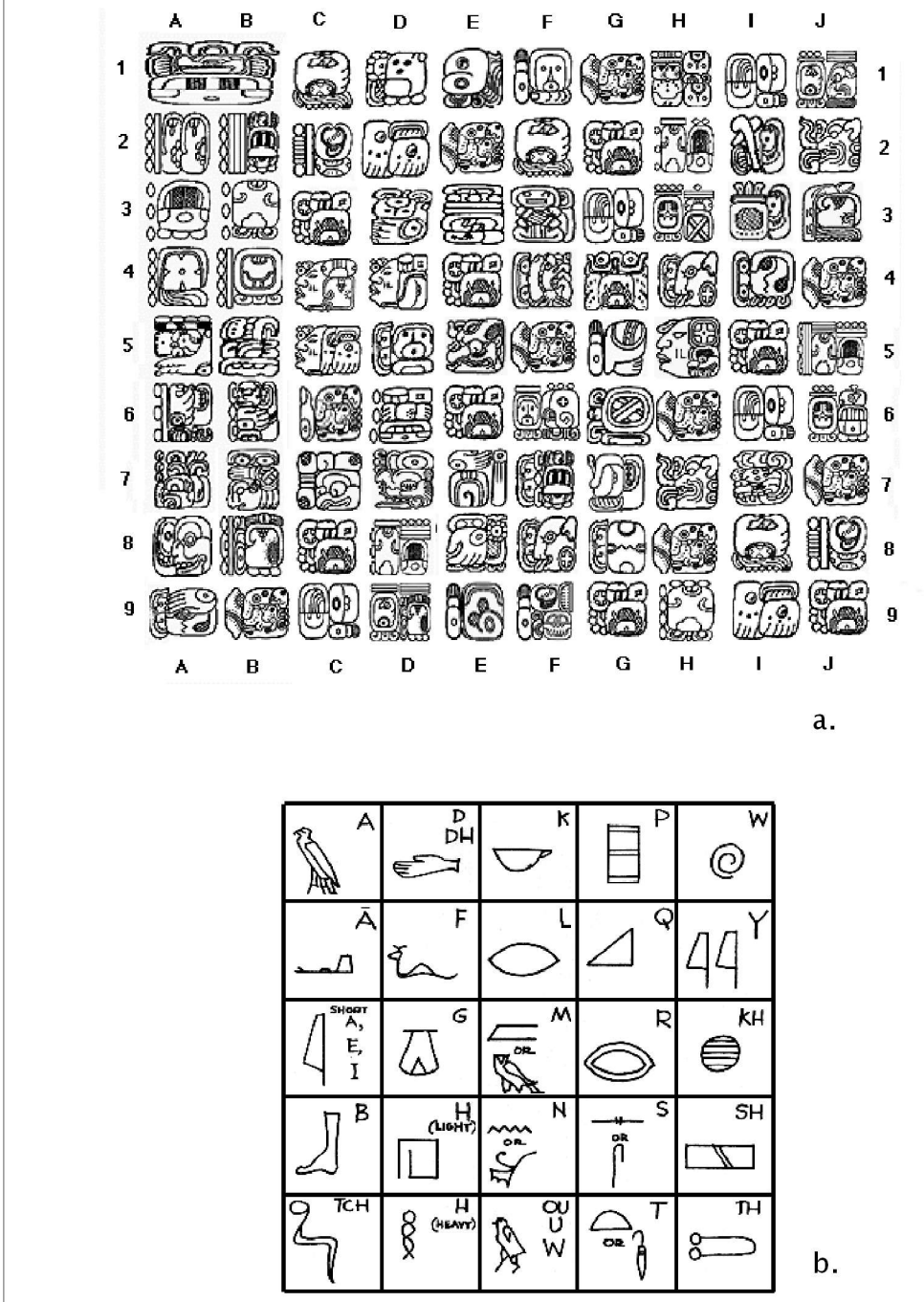


Figure 21: Mayan alphabet (a) and Egyptian Hieroglyphics (b) examples

The fundamental difference here is that pictograms stood for whole words and concepts whilst scripts used characters representing sounds to form words. Hence it is possible to write the same language by both forms of symbolic and writing systems and in the same sense it is possible to write several different languages by using the same cuneiform script. (Figure 22)

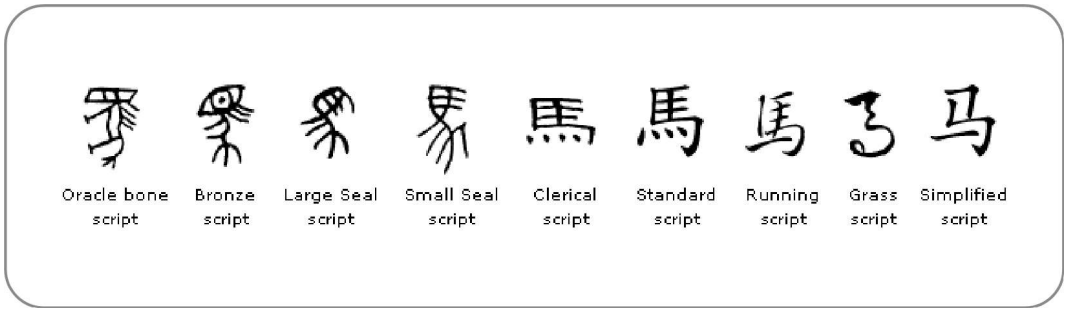


Figure 22: The same letter in various different Chinese Script examples.
Source: The British Museum's Mesopotamia Page.

Writing systems are evident in the major forms consisting of Logographic, Pictographic, Ideographic, Phonographic, Syllabic and Alphabetic[6] (Goppold, 16.2) According to this categorization, Abstract Logographic symbols are composed of forms that are often common and universal, such as mathematical, scientific, notational and technical symbols. McCloud refers to these types as icons belonging to the "practical realm." (27) Pictographic symbols on the other hand are those that represent a word or phrase.

[6] The following section includes similar terms under the section "Forms of Icons," although these terms are related to the visual attributes of icons in particular, and regarded through a perspective somewhat different to the history and topology of writing and origins of iconic imagery.

Haramundanis's main argument, evident in the title of her article, “Why Icons Cannot Stand Alone,” emphasizes that the origins of icons and writing come from similar roots and that there is no clear distinction between text-based scripts and iconic imagery. (Haramundanis, 1) Therefore her ideas on this subject could be summarized as icons and text being regarded together without isolation, that icons must be guided or founded upon some form of written explanation or label. A more detailed argumentation about this idea will be presented in the following chapter that compares textual and iconic communication. Regarding the definition of the icon, Rubens comments on how they can be regarded, and places them into another visual category that has a wider and clearer terms of classification:

> The difficulty of placing icons into discrete domains presents a major impediment to the development of useful iconic vocabularies. One way of considering the impact of the presence of various domains of icons in one context can be found in the anthropological studies of the development of glyphs. Glyphs are arbitrary visual signs with no associated vocal form (much like mad signs in which a right arrow represents the shape of the road but not the word “curve”): in addition they are not part of a system designed to account for a complete “real” language or vocabulary. Glyphs generally fall into five categories: pure, bound, set, system, and proper. Pure glyphs cannot be analyzed into component parts by applying systematic grammatic rules. Bound glyphs act as signs for components of a message in a particular language; they normally represent nouns or functors. Set glyphs are members of a system associated with a set of combinatorial rules independent of a spoken grammar, heraldry, chemical formulae, musical notations, and many international symbols for specific

> professions are examples of these glyphs. A glyph system contains glyphs that have a systematic relationship. Particular lighting systems, such as traffic signals, and directional signs for various purposes fall into this category. Finally, a proper glyph stands for an individual institution; corporate logograms, trademarks, and certain cultural icons -- crosses, yin and yang, etc.-- ail belong to this category. (Rubens, 27)

The context of the metaphor regarding cultural and historic significance is important as Brami shortly mentions. Change in time, cultural traits and other factors substantially change the comprehensibility and the meaningfulness of an idea, concept or understanding and hence the whole metaphorical concept that depends on it. These traits however, can also work in favour of empowering the comprehensibility of the icon.

> “Throughout human history, the typical course of development in a culture’s creation of iconographic imagery is to first produce archetypes that are frontal and rigid, in “poses” that reflect a particular action, attitude, or meaning. As the culture develops the imagery, it becomes more specific in appearance and more loaded with nuances.” (Brami, 13)

What this idea suggests is that through continued depictions of a certain concept that underlies as the foundation of the representative icon, certain ideal figures, poses or imagery with stylistic similarities start to evolve and become typical examples for that particular concept. When applied to desktop icons, we can encounter a similar tendency. To exemplify this idea, certain icons can be considered; the 'text-document' , file icons, “folder” icons, as those which are application related, and the 'zoom' icon, the 'selection' icon as command icons. When we observe the

instances of these icons in a number of different operating systems and applications, we encounter very similar, in some cases near-identical representations and metaphoric depictions of these functions and concepts. For example the zoom icon that is used in a number of graphic and document editing programs, browsers and other similar applications is commonly and widely represented by a magnifying glass, and usually inherits a '+' or a '-' sign to relate whether it is zooming in or out. Even though the names of the icons may be different from one application to another, such as magnify, zoom, enlarge, the representative figure used to convey this metaphor is substantially the same.

Just as there are sets of ideographs forming the unified array of languages, computers also include many sets of icons to enable widespread use of a diverse number of actions and commands. Some icons are directly referenced to older civilizations' pictorial and symbolic cases, and communication methods operate in a very similar nature to these instances.

> "The practice of grouping collections of attributes together is becoming increasingly common. For example, the PHIGS graphics standard [2] uses "bundles" of drawing attributes, while Microsoft Word [5] allows the user to group parameters controlling the appearance of text, such as font, size, and spacing, into a single text "style." In our work we apply the notion of grouped attributes to a graphical interaction method,. (Salesin, 103)

In order to perform tasks as intended and function properly icons have to be properly designed and formed. As Brami defines it, the iconic image must be “a recognizable image with a power to transcend cultures and be immediately understood by the viewer” (16) A user has to quickly distinguish one icon from another, recognize it and understand what the icon signifies or what command the icon deploys. Whether figurative, illustrative or photographic, black, white or coloured, the main purpose of icons are to represent the needed task. An icon is basically a pictorial representation of a concept or idea and therefore its technical features can be somewhat parallel to that of an image, only to have certain limitations as to the area it occupies, the number of colours it uses and its symbolic properties. Rubens states the following about designing sign systems,

> ”If one attempts to define the salient features of a conventional sign system, one could suggest that it will have three aspects: leveling, sharpening, and assimilation. Leveling simply means that extraneous detail and objects have been deleted. Sharpening involves making the remaining detail stand-out from the background. Finally, assimilation means that exaggeration and other deformation techniques are used to interpolate from mimetic, or real, to imaginative, or metaphoric, detail. Techniques of this variety allow developers of icons to represent fairly complex environments with relatively simple graphics.” (Rubens, 25)

3.2. Visual Qualities of Icons

Icons are the most intricate, customized and varying elements of the Graphical User Interface. Other elements such as windows, scrollbars and menus include differences in terms of their categorizations and content but in visual form are regularly consistent throughout. Changes in themes, tones, colours, textures and features such as menus, displaying options and bars do vary within these elements coresponding to their functional properties but unlike icons that rely on unique image representations, they are not intended to differ among themselves.

> Icons and windows are the most important visual objects as they correspond to real objects of the applications. Buttons and scrollbars are used in the decoration of windows: buttons trigger operations on the window (for instance open/close), and scrollbars are used to control the visible part of a window's content. (Beaudouin-Lafon, 145)

Many sources aim to define or at the least interested in establishing points regarding the classification and categorization of icons to produce a comprehensible portrait and understanding of the general factors of the icon. When a number of these are examined, two different perspectives seem to be of most importance. The first of these approaches is the icon as taken in the framework of semiotics; the study or science of signs, and the other approach analyzing the icon in terms of visual quality as part of graphic design/ visual communication. Although these two perspectives

rely on and are related to one another, it is important to look at both separately to underly different factors. Bryne's account of visual factors is one that effectively summarizes the different qualities of icons:

> Size. Generally, all of the icons on a display are the same size. So while size may be a useful way to discriminate between icons, it is typically not employed in current displays.
>
> Color. This is an interesting dimension, because many GUI displays do not support color. Thus, even if using color as a discriminating feature is effective, reliance on color differences may not be appropriate for icon designers.
>
> Form. This is the primary dimension on which icons vary and includes a number of sub-dimensions, such as the level of detail in the form and the meaningfulness of the form.
>
> Spatial organization. Some interfaces impose a grid-like organization on icons, others a"staggered grid" organization, and still others no organization at all-icons can be anywhere. This may or may not have an effect on search.
>
> Number of objects. Both the size of the search set and the number of icons with pictures matching the target can vary widely from trial to trial. Probably the best-studied factor in visual search is that of set size, while the effect of multiple visual matches has been relatively ignored. (Bryne, 447)

3.2.1.Form

There appear to be three major categories for classifying the different forms of icons. Although sources give different names, and some include more than three, the underlying descriptions suggest that they can be grouped into Pictographic, Representative and Abstract.

> "Manning posits three modes that serve as models for comparative visual analysis: order, graphic, and literal. The first ranges from high to low order the second from high graphic to low graphic; and the third from high literal to low literal. In each instance, he makes some suggestions about visual representations that characterize each of these modes." (Rubens, 26)

Horton states different terms, "We can draw icons in five different degrees of detail and realism:Photograph, Drawing, Caricature, Outline, and Silhouette" (Horton, 372) The instances of the three types of forms included in this study are relevant to what is being represented and how the icon is to function. In some cases all methods may be suited, whilst in other situations, neither three will be able to properly represent the intended function. Rubens explains this situation:

> One difficulty presented by icons in computing, already discussed, is that they span the boundary of several iconic types. The representational or mimetic sign appears as surrogates for corporate logos; the pictogram is employed for specific elements or tasks (such as file folders and disks); diagrams (such as arrows) are used to indicate direction: prosodic (in the form of punctuation marks) and mathematical signs (as notation for mathematical and boolean operations):

> and verbal support to provide redundancy occur. Thus, viewers, as users of computer equipment, must be facile enough at decoding iconic signs from various domains to be able to move successfully among those domains with some degree of comprehension. (Rubens, 29)

3.2.1.1. Pictographic Icons

As older operating systems and applications permitted limited use of graphical entities due to technological limits, earlier examples of icons fell primarily to this category of visualization. Figuration or illustration was used for pictographic depictions of representing an idea, concept or subject by using fewer lines, summarizing details and conveying only the most important features of the object. This type of representation proved very effective, functional and suited for earlier operating systems and applications primarily to the fact of technical restrictions, and was an economical solution for allowing metaphoric communication methods. Icons were pure and simple in this sense and were remembered, easier distinguished and perceived at shorter times. The overall use of figurative and illustrative icons proved to be well suited for fast and easy task performing and apprehension. A comparison of this method can be directly related to logos, emblems and symbols used in industries, companies, institutes and other foundations which employ the use of a simple image to represent their institution in the most comprehensive and effective way. This information is apparent in the interface as many

software developers use their company logotype as icons for their products and applications, or in other words the logo of a company can be directly substituted to function as an icon.

> "For the most part, these icons would be termed symbolic signs. Symbolic signs can be defined as a pictogram; that is, certain analogies connect the icon to its referent, but the primary referent is deformed to allow the icon to refer, by analogy, to a secondary referent. For instance, in our later discussion of a telephone icon for a telecommunication program, we will maintain that this icon is appropriate even though the telephone is associated with voice rather than data transmission."(Rubens, 28)

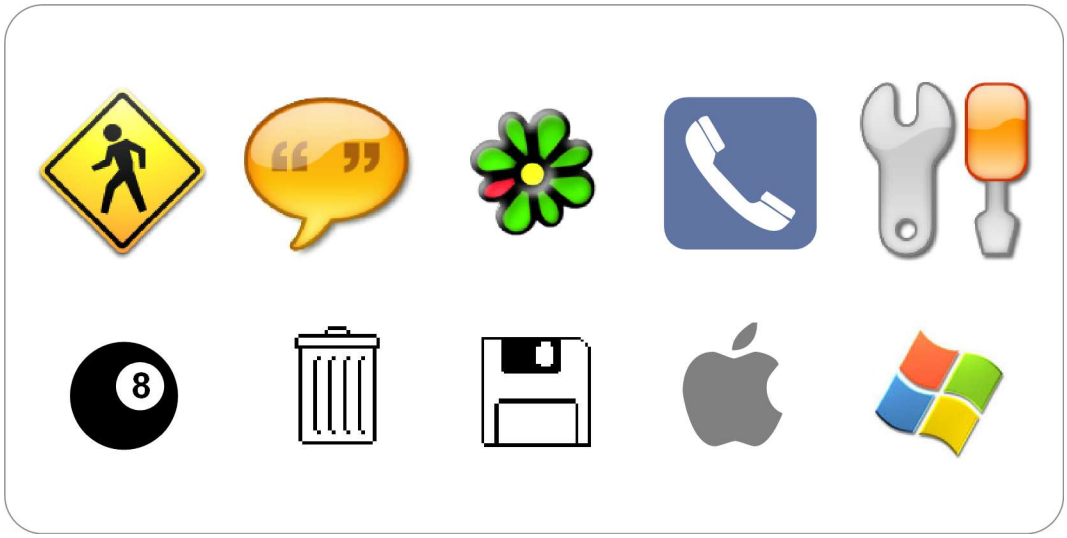


Figure 23: Examples of Pictographic Icons
Source: Iconfactory, anonymous.

Caricatures, comics and cartoons also fit into this category. Pictographic icons can be said to be the most common and widespread types of icons used, presumably because the tradition of older systems still enforces pictography to be popular, and because the use of highly representative icons have only possible in recent time, available in only the newest operating systems and applications. Although some icon designers aim to experiment and deploy the latest technical possibilities and trends, may stick to more solid standards and design for the widest audience expected to be using older systems and applications. Although icons are capable of resizing into a number of standard sizes, highly realistic icons cannot be effectively displayed with the lower dimensions that most systems are limited of displaying, so most executions of icon design are aimed directly to these limited sizes to preserve design and style consistency.

> Glyph research provides a glimpse into the difficulty of creating icons that fit into specific categories. An additional difficulty in creating icons, and perhaps a more far-reaching one, is the difficulty in designing a finite icon set. International road signs are a good case in point. For the most part, road signs try to represent their meaning independent of the sound of a spoken language. Pictures are used to represent objects rather than the sound of the object's name. Highway pictographs, for example, depict an upcoming bend with a curved arrow. The arrow represents the shape of the road rather than the sound of the word "bend". Pictographs, therefore, are very useful when people speak different languages. (Rubens, 28)

3.2.1.2. Representative Icons

This is perhaps the most apparent form of evolution that has occurred considering the visual features of the computer icon. As discussed previously, this evolution is mostly regarded to technical advancements and the developing of platforms which allow for more complex dimensions, details and digital expansion. With the coming of technical possibilites, newer user interfaces such as the Apple Mac OSX and Microsoft Windows XP can now equipped with near photo-realistic and photo illustrative icons. Departing from this new opportunity, the number of representative and realistic icons have grown. The new technical possibilities allow icons to possess direct realistic depictions of objects or subjects that refer to a concept, similar to a photograph or realistic drawing. For metaphors that represent tangible objects and material subjects, this method provides an effective and applicable technique.

However parallel with symbolic properties and necessities for successful implementation of metaphors and analogies, and also because of space consistency, this type of icon form also includes restrictions, in most apparent terms, the represented object must be clearly depicted, usually in full size against a solid colour background to be perceptible. It must be simple enough to communicate its message clearly and represent a key concept without trouble. At the moment within the latest technical possibilities, iconic communication can only benefit from using highly representative techniques within certain limits.

Figure 24: Examples of Representative Icons
Source: Apple Aqua Human Interface Guidelines, Iconfactory

3.2.1.3. Abstract Icons

Abstract icons are very common in applications and operating systems since it is not possible to represent concepts, ideas or commands that do not possess physical material quality. Icons of this type either assign symbolic and expressive figures to convey its message or take advantage of conventions to display ideas. Usage of national flags to convey language settings can be an example to the use of conventional symbolism. (Flag icons also fit into the other two categories depending on its stylistic properties and resemblance to the flag as a physical object.

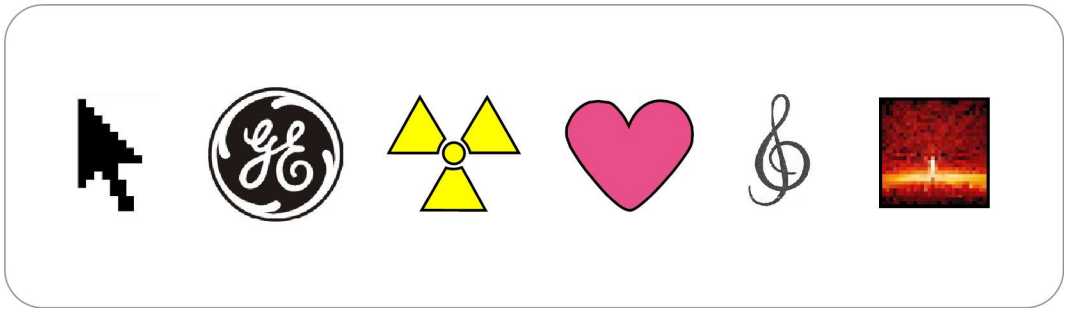


Figure 25: Examples of Abstract Icons
Source. Anonymous

Although abstract symbols and icons may make successful expressions of representing less clear concepts and commands, and can prove to be highly recognizable due to their different complex visual language, they can and are commonly problematic. Without prior knowledge, explanatory text or some form of explanation, there is no way of comprehending the icon initially. This means that there is no way to figure out what the represented command or object is, or is related to, except for triggering the icon to find out. Also memorization is required to perform the task for the next time and abstraction makes remembrance harder.

3.2.2. Colour and Resolution

The conventional computer screen colours are measured in a number of bit-depths that display screens are capable of showing. Bit depth also refers to resolution and therefore effects both the colour and how clear an image is as related to this study. Since colour contributes to the clarity of an image and display, both are taken together in one section that is described in the following parts.

Until quite recently, in earlier versions of operating systems, the maximum number of bit-depth that computers could display was 256 colours, previous versions were even less at displaying monochrome 1-bit images consisting of only black or white. Newer systems and platforms however have the possibility of including alpha channels for visual components in their environments and can mask icons to be placed on a variety of backgrounds, letting the icon show through. High colour resolution images also allows for a wide spectrum of colour enabling representations of transparent objects, textured surfaces and lightened environments. This is one of the most important aspects of colour regarding the Graphical User Interface and Desktop icons. As systems increase colour capabilities and spectrums, user selected schemes, use of variety and alternative depiction of elements become possible. A common example of this is desktop schemes, that are apparent in almost all recent operating systems. (Figure 26)

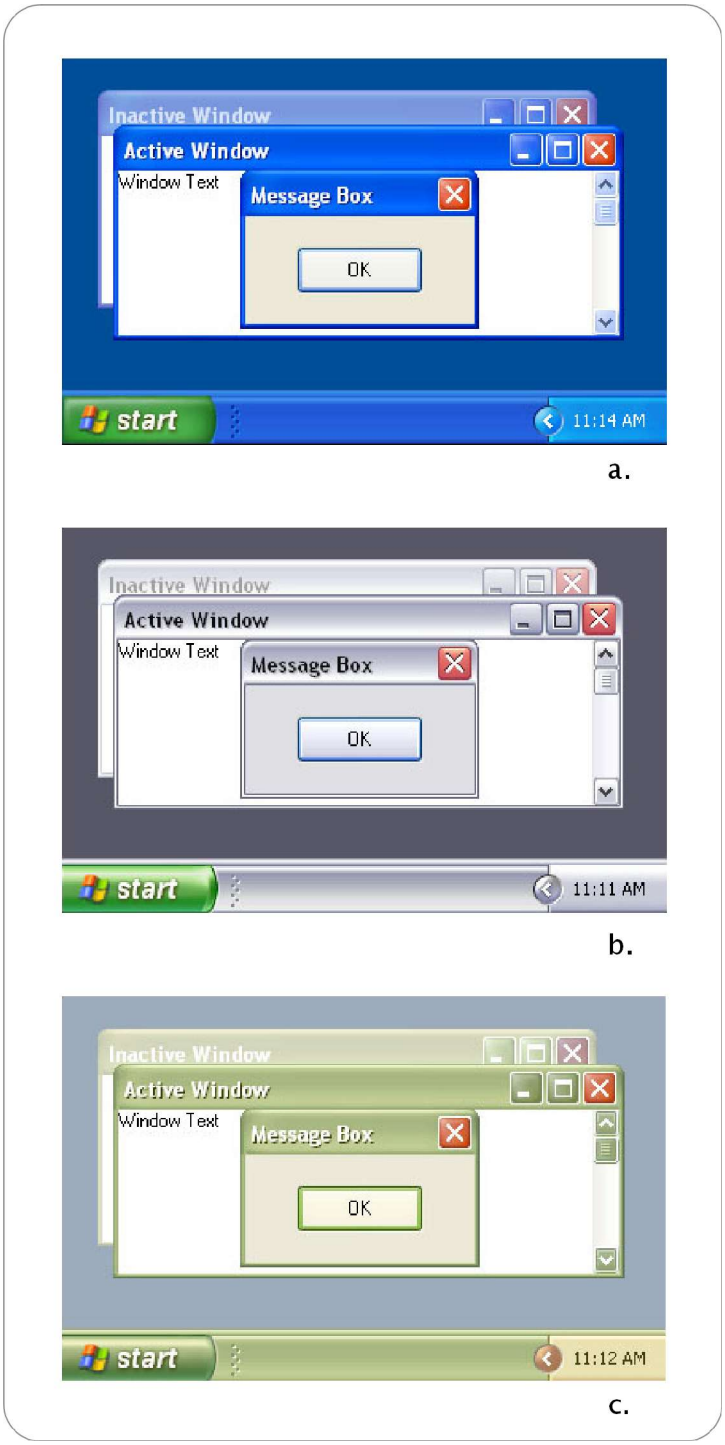


Figure 26: The different desktop colour schemes of Windows XP:
a. Classic
b. Silver
c. Olive Green
Very little has changed regarding the general structure of typical windows. The most fundamental differences of change include changing of colours and styles.

The qualities that are regarded to transparency and opacity are also a case dependent on technological development. Where it is difficult to express space with perspective in small areas consisting of an area of 16 pixels, and only primary geometric shapes can be defined in such an area, a larger area such as 128 pixel square, space can easily be manipulated into different viewpoints and geometrical perspective drawings. Shadows and colour further contribute to emphasize volume and depth.

Colour is one of the most important factors of visual communication and applies to icons with the same importance. Alben states that the main principles of visual communication is colour and colour theory (18 – 19) and stresses that it is vitally important for perception. Alben mentions his use of colour as part of a work including the design of an interface, teaching the principles of another interface (Making it Macintosh) stating that colour functioned as separation between the different parts of the interface while also providing a thematic relationship throughout. Colour applies to icons as does to basic design principles. Ergogero.com includes a very comprehensive summary of all points of colour and interface design regarding colour:

> 1. Color consists of three properties: hue (red, green, etc), brightness/lightness/value (light-dark) and saturation/chroma (vivid, pale, etc.)
>
> 2. There are two contexts for color: aperture color and surface color. They follow somewhat different rules.
>
> 3. There are 4 perceptual primary colors: red, green, blue and yellow. Brown is sometimes also seen as primary.

4. Background has a strong influence on color appearance.

5. Ambient light can affect color appearance.

6. All humans divide hue into eleven basic categories: black, white, red, green, yellow, blue, orange, pink, gray, brown and purple.

7. Color similarity is the best way to convey that two things are the same type. Color differences are the best way to convey that two things are different type.

8. The meanings that people attach to color changes with culture. But it also changes with context in the same culture. I. E., blue can sometimes mean power and at other times sadness.

9. People in all cultures, organize color the same.

10. For people to be able to read words, there must be a lot of brightness (light-dark) contrast. Hue contrast is not much help. The biggest single mistake that designers make is insufficient brightness contrast.

11. When there are more than about 6 colors, ability to pick out individual elements declines.

12. Color is too important to consider only aesthetics. It greatly affects effectiveness, visibility and conspicuity.

Computer monitors, (CRT[7] displays as the most common type) represent pictures in small dots (pixels where computer terminology is concerned) similar to printed material in this respect, which are small enough to compose a unified single piece of imagery when viewed from a suitable

[7] Cathode Ray Tube

distance. Typical monitors are composed of 72 or 96[8] ppi (pixels per inch) which means that for every inch there are 72 or 96 pixels to form the display respectively, and have a total resolution of 800 by 600 pixels meaning the display has a grid of 800 pixels horizontally and 600 pixels vertically. Most contemporary monitors have options to change the resolution and provide the user with alternatives for substantial lower and higher resolution. The more pixels there are occupying the smallest area of the display, and the smaller the pixels, the clearer the image becomes and the sharper the details will be. This principle also applies to printed material or photography to exemplify.

A pixel also consists of colour-depth or a bit depth as it is referred to. (Figure 27) Monitors use additive colour, forming all possible colours by juxtaposition of light using the three primary colours consisting of RGB (Red, Green and Blue). The absolute presence of these three primary colours at maximum intensity result in white (can be associated with hard light) and the absence of the three result n black (since there is no light whatsoever coming through), all other colours present in the palette are achieved by using alternating combinations varying the intensity for each of the coloured lights. A monitor's or document's bit-depth is defined by the number of colours it is able to exhibit and formulated by 2^x where x represents the bit-depth and the resulting figure represents the total number of colours used.

[8] This difference is related to the different standards between Macintosh displays that are usually 72 ppi and PC displays that are 96 ppi.

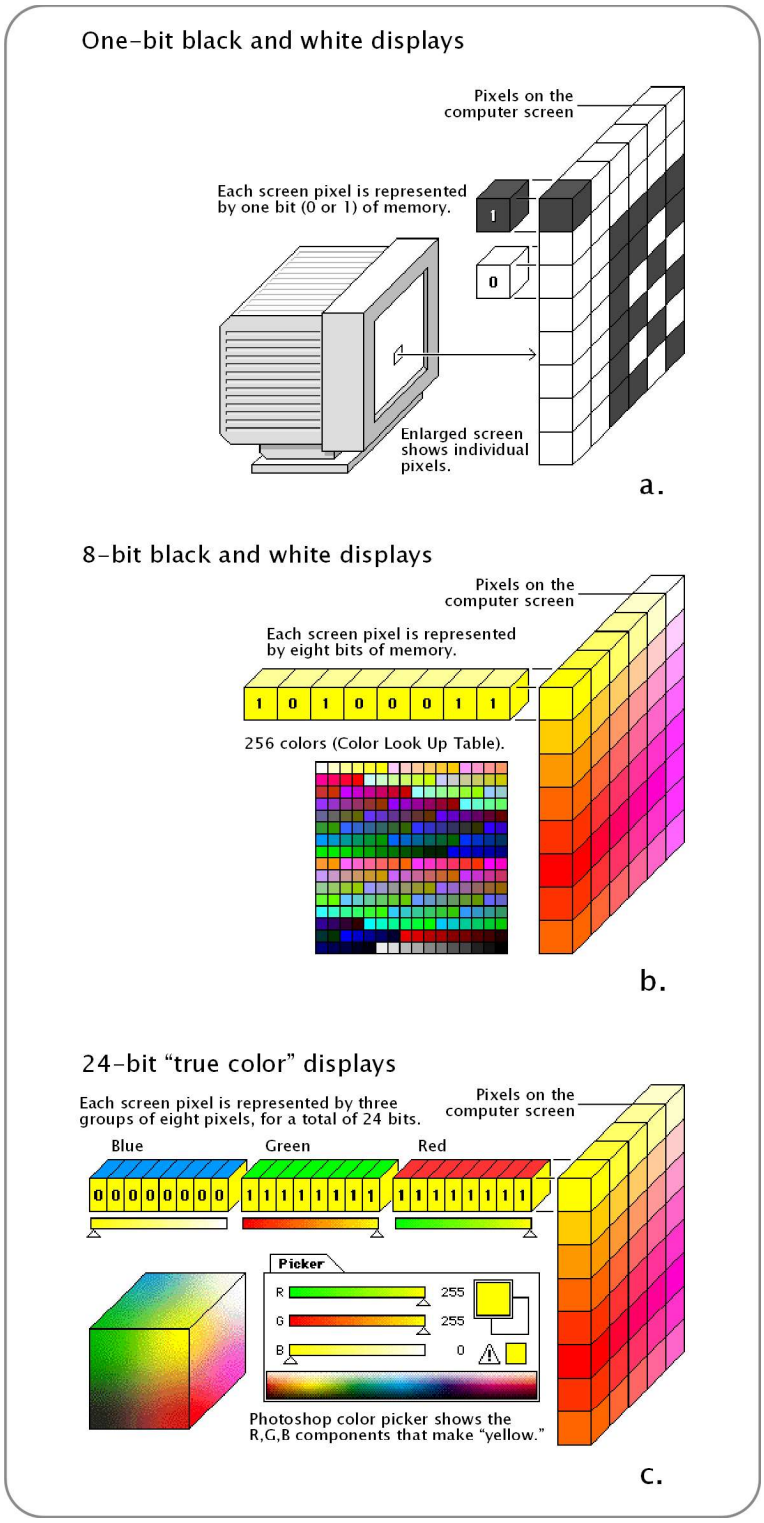


Figure 27: Illustrations showing 1-bit, 8-bit and 24-bit display resolutions.
Source: Web-Style Guide. (www.webstyleguide.com)

Earlier monochrome operating systems consisted of 1-bit colour depth consisting of pixels with either a value of black or with white, with no gray tones, then upto 8-bit (256 colours) displays and eventually to 24-bit and 32-bit images. 24-bit and 32-bit images allow for photo-realistic representations and values with and consist of millions of colours. As stated before, older icons were very limited in terms of colour depth while it is presently possible to produce near-photographic representations in icons. Commonly, operating systems and applications used to include only a number of colours and bit-depth, some of which were not capable of displaying higher number of colours and in other cases, to preserve system resources and memory. (Figure 28) The browser safe palette was also designed with the intention of producing the same results for all users and computers. Newer systems include a variety of different colour palettes, the colour schemes of desktops mentioned earlier being an example to this situation.

Initial icons of the first desktop operating systems such as the Mac OS 1 consisted of very simple depictions consisting of 1-bit images of an area about 16 x16 or 32x32 pixels. This meant that a very simple but straightforward pictorial symbolic picture was composed to represent a key concept. In newer applications and operating systems, exemplifying the Mac OS X, the maximum size of an icon has increased to 128x128 pixels, 16 times the area of earlier icons. Another aspect of newer icons is that they consist of a number of various different sizes, including 16x16, 24x24, 32x32, 64x64 in and includes complexer 32-bit (millions of

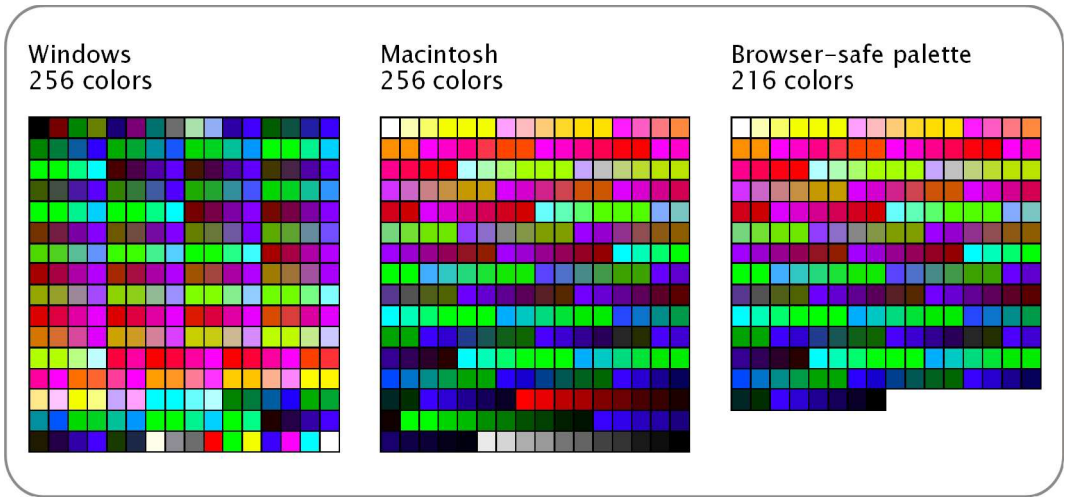


Figure 28: System and web-safe colour palletes.
Source: Web-Style Guide. (www.webstyleguide.com)

colours) images with alpha channels, the simplest 1-bit (black and white) capable versions of the same images and th 8-bit, 256 colour images (Figure 29) as middle values, a combination enough to present various features that allow for complex shading and dimension on both older and newer systems. Figure 30 shows examples of system icons with varying sizes and bit-depths. Newer systems displace 1-bit depth icons, and include newer sizes such as 48x48 on Windows XP and upto 128x128 on MacOSX.

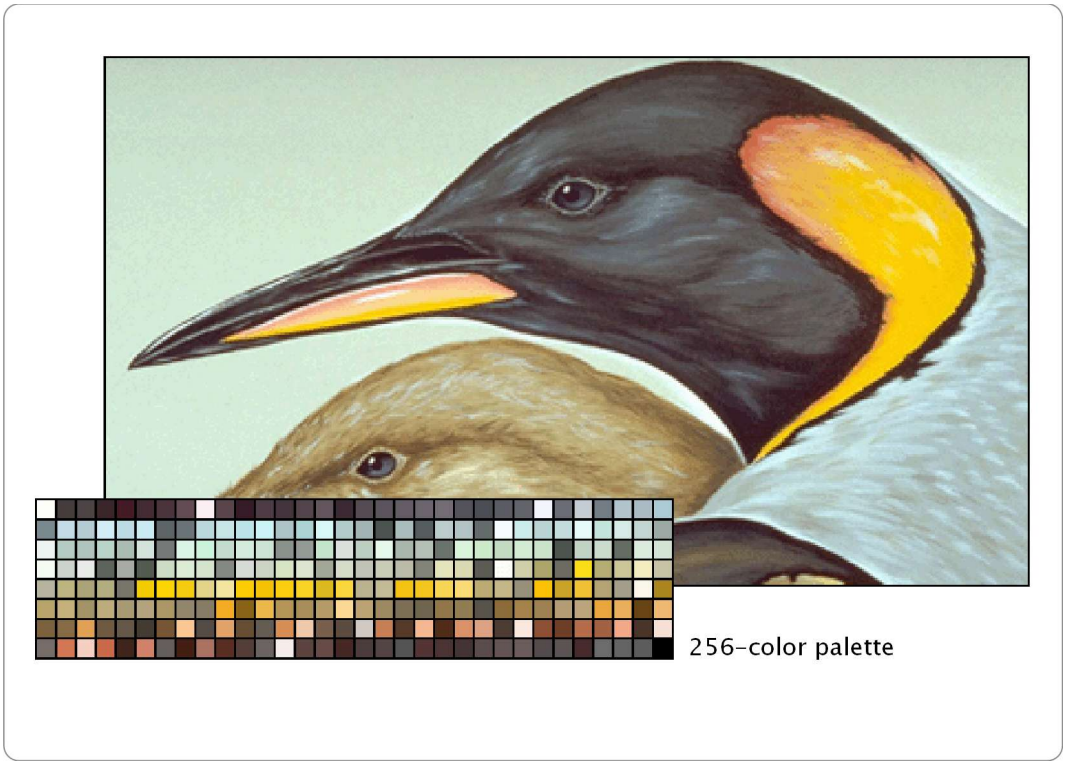


Figure 29: Example of an image in 256 colour and its custom colour pallete.
Source: Web-Style Guide. (www.webstyleguide.com)

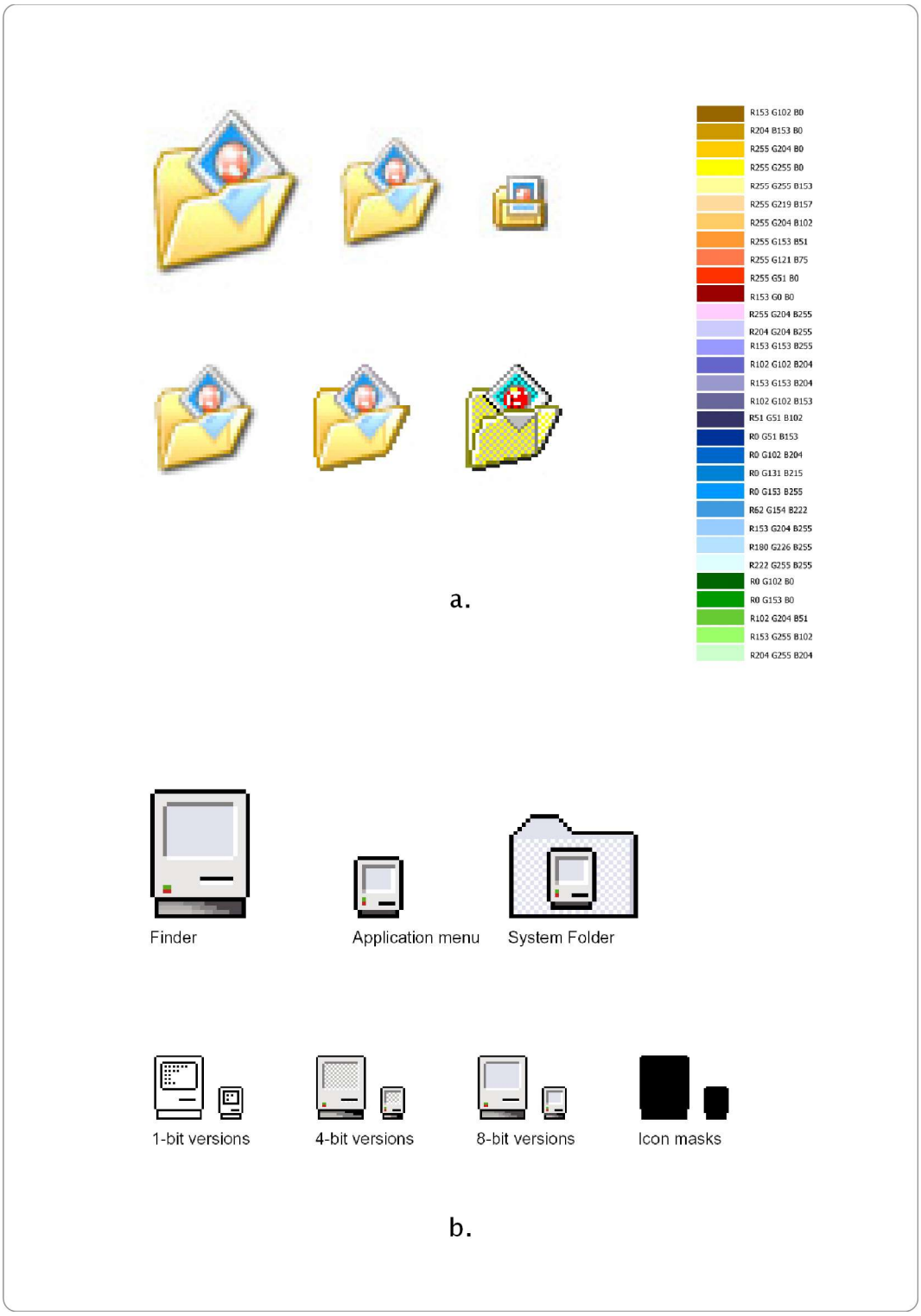


Figure 30: Examples of icons with varying size and bit-depth:
a. Windows XP icons in different sizes and resolution with icon colour palette.
b. MacOS8 icons in different sizes and bit-depths.
Note that since MacOS8 is an older system, it also includes a 1-bit version whilst Windows XP does not.

3.2.3.Motion

> "Animated icons can bring to life symbols representing complete applications or functions within an application, thereby clarifying their meaning, demonstrating their capabilities, and even explaining their method of use." (Baecker, *Bringing* 1)

Along with the properties of multimedia development, a major enhancement in icons is that they do not necessarily have to be static within new applications and operating systems. As with being in different sets of sizes and colour depths, icons have also started to become more dynamic, at the least including basic animation and effects. (Figure 31 shows an example of an animated icon.) A newer example of this occurrence is the upward-downward movement of icons in the "dock" bar in the latest Mac OS X. Baecker's study is defined as an "Article proposing the use of dynamic visual representations to augment traditional static text as documentation... suggest using live demonstrations or moving pictures to show people how to do things, and not just using written or spoken words to tell them what to do" (Baecker, Showing 10)

Baecker asserts this idea in one of his other articles and concludes that, "The results showed significant benefit from the animations in clarifying the purpose and functionality of the icons." (Baecker, Bringing 1) and later, "demonstrates that in every case the users understood the purpose

of each icon after viewing the animations." (Baecker, Bringing 4) He defines animation as an "effective means of portraying complex processes evolving over time" (Baecker, Bringing 1).

Animation and movement are evidently important aspects that liven imagery and extend capabilities of functionality. Apart from enhancing narrativity and providing guidance to perform tasks and navigate through application and workspaces, animation also includes psychological aspects. Newer operating systems and software such as those on the Mac OSX include moving icons to show processing, background task performing and speed. As well as informing the user about what the computer is doing, these features also entertain to occupy the user, motivating and giving the notion that the system is performing faster and better.

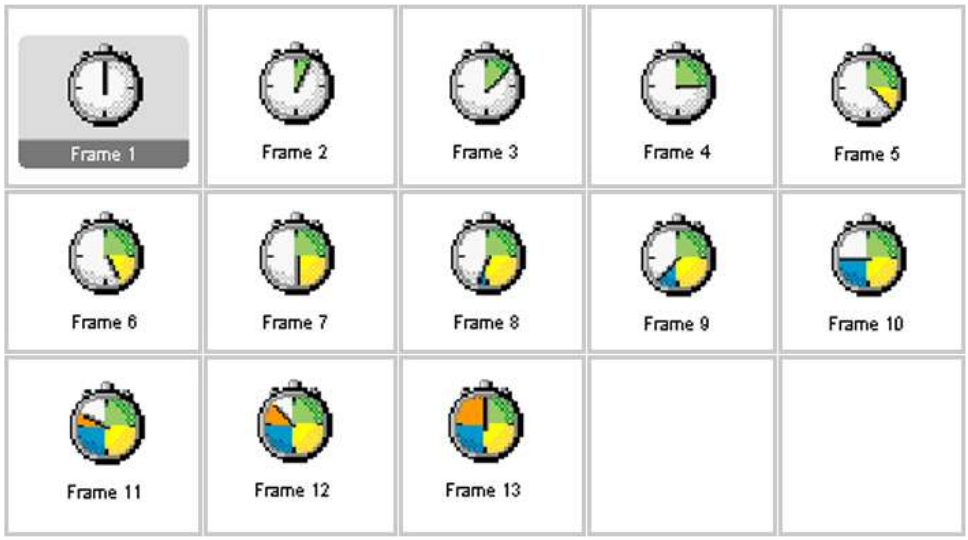


Figure 31: Example of frames from an animated icon.

3.3. Classification of Icons in the Graphical User Interface

The Aqua Interface Guidelines for the latest Macintosh operating system classifies icons into genres such as the user application icons and the utility icons. This helps utilizing distinguished categories while using the computer. The main difference of these genres comes from the stylistic conventions and expressive qualifications of the images used for representing an idea. Similar categorizations and attempts to organize the different types of icons are evident in most Graphical User Interfaces although the names although the names and categories differ slightly. However the general logic is that icons are in its most apparent forms used to represent applications and commands.

The three main categories that are included in the Aqua Guidelines are the application icons, non-application icons and toolbar icons. Application icons (Figure 32) are used mainly in regard to the represented application for finding, opening and closing programs. Subcategories in application icons include User application icons, Viewer, player and Accessory Icons and Utility Icons.

Non-application icons (Figure 33) include document icons, icons for preferences and plug-ins and icons for hardware and removable media. Most typical examples of non-application icons are those used to relate documents that were composed in a certain application to the group of icons that represent that application.

Figure 32: A selection of Mac OS X Application Icons.
Source: Apple Aqua Human Interface Guidelines.

Figure 33: A selection of Mac OS X Utility Icons.
Source: Apple Aqua Human Interface Guidelines.

The last category consists of toolbar icons (Figure 34) wherein a bar where most common used functions can be easily accessed with a variety of the different icon types are nested to provide quick and easy commandment.

Figure 34: Mac OS X Toolbar and Icons.
Source: Apple Aqua Human Interface Guidelines.

One salient difference between categories of icons that is likely to affect both initial comprehension and subsequent retention involves the "directness" of the link between what's depicted and the command. Whereas some icons depict objects and operations directly involved in the commands, others depict objects or symbols related to the command by analogy. In the former cases, discovering the "link" between a depiction and the corresponding command is quite straightforward. Assuming that the depiction is clear and unambiguous, and that the depicted command-objects and implied transformations are characteristic of the command, initially discovering the link and subsequently using it to retrieve the command should be simple processes. In contrast, both icons that are conventional symbols and icons depicting common tools (that represent the commands by analogy)are linked less directly to the corresponding commands: they represent the commands by drawing parallels between the features of the commands and features of familiar entities or meanings of familiar symbols. For these icons, the quality of the analogy is critical: whether the link is easy or difficult to discover initially, and whether it is subsequently retained in memory (facilitating retrieval of the command) , undoubtedly depends on how good the analogy is. More generally, the impact of analogies on users' retention of icon-command correspondences, and on users' understanding of a system, are presently open questions (see [7]), and good topics for empirical research. (Hemenway, 22)

4. THE COMPARISON OF TEXTUAL AND ICONIC COMMUNICATION

Summarizing what has been included in this study until now, the main points show that, from the first invention of the computer working on principles of switches and dials, information stored digitally in a form of 1s and 0s, the need for efficient communication was needed to fulfill expanding needs and to perform more complex actions that computers were becoming capable of doing. As the primary means of communication amongst ourselves 'language' was apparently taken as basis for task performing on machines, and eventually the command prompt or text-based user interface was developed. Since the computer functioned on a basis of mathematical calculation, the computer programs or languages developed for communication consisted strictly of figures, numbers and words. While this interface provided a considerably suitable environment for scientists and computing professionals dealing intensely with computer technology, programming and related fields, it was apparently not suitable for the average or novice user that did not have authoring capabilities or the necessity to learn the infrastructures of programming and computing. Soon after the employment of the command prompt, a model for a visually dominant graphical user interface, also a direct

manipulation interface as related to the issue, whereby features and functions of the former model was integrated with the new visual features. This model has been the general working principle of both past and contemporary operating systems and applications, and has evolved to take the form of the desktop computer used and needed by a vast number of users throughout the globe.

This chapter will distinguish the similarities and differences between the two different models of user interface and also look at examples that use a combination of the two. Apart from the oldest textual user interfaces and those which have chosen to preserve the conventions of command prompting for specific reasons, it would actually be fairly correct to say that no interface exists solely on the property of a single user interface model, or to clarify this statement, there are almost no interfaces that contain merely iconic or only text-based features. There are rather, examples that have a tendency towards the general characteristics to one of these two models. Therefore the categorizations depicted below, apart from the typical command prompt textual user interfaces should be regarded as those with mostly textual, or in opposition iconic, tendency.

4.1. Textual Communication

Textual communication was briefly discussed earlier in the overview of the human user interface and as mentioned, from the initial development of computers the most fundamental method of communication was developed based on linguistic property. However, the main problem related to such an approach was that the computer could not understand and interpret language as did humans, so as part of the interface that allowed communication between the user and the computer, external languages were necessary in order to effectively employ commands and perform tasks. The conventional form of textual communication related to the command prompt, typically the DOS Operating System (Figure 35) or similar (On which nearly all graphical user interfaces use as foundation to operate). This kind of operating system would, after the computer was switched on and initialized, display a prompt at which the user would type commands in the appropriate language that the platform was capable of understanding. The computer would then perform the requested action or task and reply back in a series of responses related to the request.

Although text based interfaces have advantages, especially for tasks that require text such as programming, networking, controlling external devices, etc. other tasks that are done on Graphical User Interfaces can also be performed on textual interfaces. The fundamental reason for choosing text based interfaces is that they do not waste computer capabilities and resources for visualization, as Graphical User Interfaces

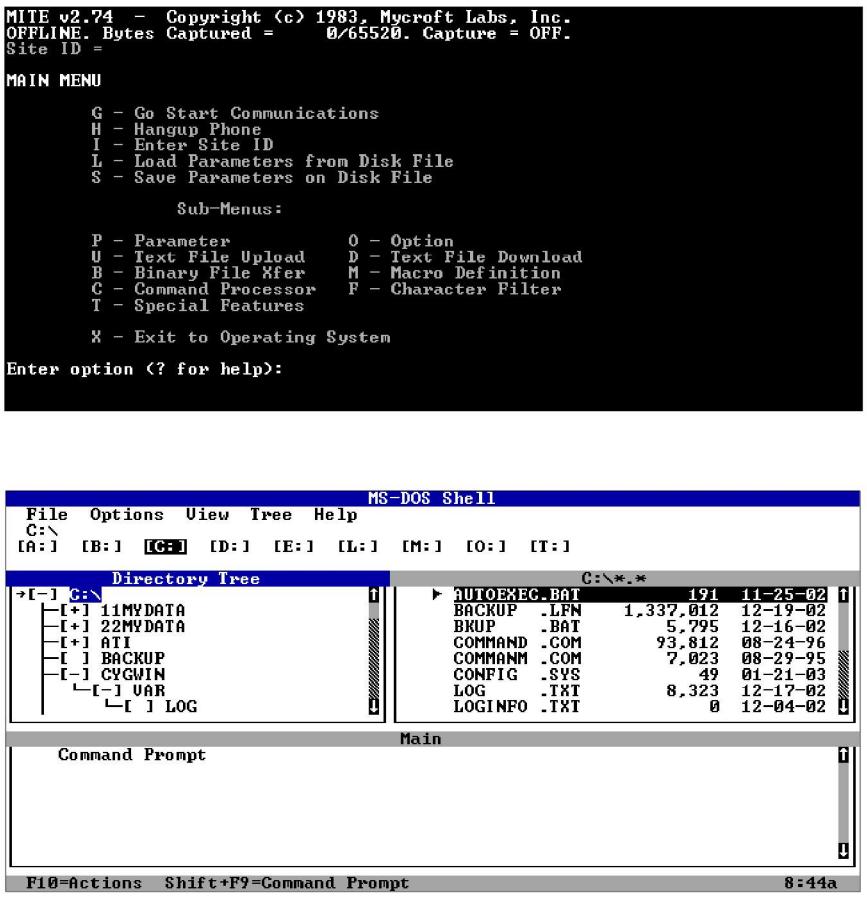


Figure 35: Examples of command prompt operating environments (MS DOS)
Source: The GUI Gallery

do. Although imagery, graphics and other forms of intricate multimedia have features that have substantial advantage, implementation of these features is highly resource dependent. Time, speed, storage space are often abused by Graphical User Interfaces and Visual applications. Users of text based interfaces can and do manipulate tasks composed of these media types, but their manipulations are performed only when needed

without extra visualization that Graphical User Interfaces are bound to perform even for the most simplest tasks.

On the other hand, Graphical User Interfaces do not stand as an opposition to textual properties. The initial development of the Graphical User Interface adapted most of the features of conventional task prompting into its system, and regarding the evolution of its operating conditions have not given up on these traits. “Representing command-set structure in icon-set structure may affect the user's ability to determine what the icons depict, to link what's depicted with the corresponding commands, and to understand the commands and the system.” (Hemenway, 24) Menus, explained in the Overview of the Human User Interface are typical and commonly used textual entities that have evolved form text based systems and interfaces. Graphical User Interfaces suggest many reasons for their uprising, including examples of typical situations that people have the tendency to not read textual property and documentation. Even including documents intended purposely for guidance and, ”despite all the skill and care applied to the creation of these texts, many users don’t read manuals and often find “online help” unhelpful.” (Baecker, Showing 11)

4.2. Iconic (Graphic) Communication

As the opportunities and advantages of visual communication had been understood by many people in a number of fields and purposes, it could be said that iconic and graphic communication inevitably served as an alternative to textual communication based primarily on linguistics and words. Especially in terms of learning and conveying messages, a more universal and simplistic way of interpretation was found to be the leading characteristics of employing iconic communication. As far as the roots of the first graphical user interfaces are concerned, this type of human computer interaction provided ground for an environment much richer in terms of deploying gestures, interaction and providing alternative means of communication.

Although articles such as Nielsen and Gartner's “Antimac” offer alternative solutions to the Graphical User Interface and even though these are not strictly against the concept of using image and icon based interfaces, they do not disregard the importance of iconic communication, especially in regard to earlier text or command based interfaces. Leaving the issue of alternative interfaces aside, the significance of the reasons for developing an icon based interface can be accounted for in may sources: Some examples include, Chu's study of icon size where he describes the reasons for his research referring to Gittins's and Lodding's accounts of iconic interfacing as: “This study addresses the issue of icons because icons are effective, useful and succinct in depicting specific functions or

operations of a window based application" (Chu, 314) Baecker also asserting a similar notion while describing his study, proposes that visual communication should be preferred to linguistic communication when computer interaction is concerned: "This paper asserts that answers can and should often be delivered visually, in other words, users should be shown how instead of being told what to do." (Baecker, Showing 11)

Bryne also draws attention to the diversity of iconic communication, suggesting that it has been undermined, having greater potential, and "Despite the prevalence of icons in graphical user interfaces (GUIs), the basic processes underlying the interaction of humans and icons are not well understood. (Bryne, 446) and continues by saying that "Icons are used in a wide variety of ways in GUIs, one of the more common of which is the representation of objects recognized by the operating system, such as files and directories... ...the use of icons in the GUI is that icons seem to work better than having to recall and type file names, as is typical with a command-line interface."

Derived from the knowledge of imagery's power, even stronger statements are apparent for the promotion of iconic communication, one example being:

> “Curiously an obvious human capability has so far (rarely) been exploited: the human brain is more easily capable of processing picture than text. It seems that about 80% of human knowledge is based on visual perception. An image Is more explicit and representative of a domain than several hundred words describing this domain.” (Frasson, 146)

When formalized, the general traits of iconic communication are easily noticeable and referred to by many: Icons “help users work smarter,” “represent visual and spatial concepts,” “save space,” “speed search,” provide “immediate recognition” and “better recall,” makes possible that “users do not have to read,” “help interfaces go global” “reduce translation,” “simplify learning,” “improve intelligibility of text,” and “give products an international look,” (Horton, 371-72) Haramundanis adds,

> ”In his book, Horton defines icons as "the small pictorial symbols used on computer menus, windows, and screens," and quite rightly limits his discussion of glyphs, symbols, signs, and signals to keep his work practical. Horton is right on many counts:
>
> • icons are useful as reminders (cues).
>
> • distinguishable icons can aid recognition.
>
> • icons can save space on screen real estate.
>
> • icons assist users whose native language is not English.
>
> Icons assist recall of a function, but they are not a substitute for training in the language of the user.” (Haramundanis, 2)

Concluding from all the comments that are concerned with the advantages and reasons regarding the use of iconic communication, the main argument is that iconic communication is empowered by the characteristics of graphic imagery and symbiolic communication.

> "When we communicate, whether we are communicating facts, numbers, or pure emotions, we do so through symbols. They may be pictures, words, musical notes, a tone of voice, or a facial expression. Often they are subtle and other times, direct." (Horton, 371)

By using symbolic properties, metaphors and iconic elements, software visualization is achieved and a graphical representation system is formed to produce a more effective means of Human Computer Interaction:

> "Software visualization has been defined [Price, Baecker, and Small, 1993] as "the use of the crafts of typography, graphic design, animation, and cinematography with modern human-computer interaction and computer graphics technology to facilitate both the human understanding and effective use of computer software." (Baecker, Showing 11)

4.3. The comparison of Textual and Iconic Communication

As mentioned in the opening of this section, while distinguishing between the two methods of interface communication composed of textual and iconic entity, it is most apparent that the two models are necessarily used cooperatively and have both advantageous and disadvantageous features over one another. This inevitably means that the isolation of one of the models, or the sole use of one or the other is rather inefficient and difficult. As a catalyst of the Graphical User Interface, the command prompt does evolve around textual entity, but this makes performing certain tasks almost impossible to achieve and fulfills only a limited number of purposes, satisfying a minority of users. Since this is clearly evident, the comparison of textual and iconic communication can be regarded as an analysis of the tendencies towards one or both of the two approaches and highlighting instances where each provides substantial advantage over one another. (Figure 36) The following parts aim to formalize these factors and discuss examples of certain cases.

Directly related to this issue, and is revealed initially in the title “Why Icons Cannot Stand Alone,” Haramundanis' article clearly addresses the reasons for the impossibility of sole iconic execution. Her main argument is that the roots of the icon has direct relevance related to the development of writing, and her introductory statement starts by articulating that without

Figure 36: The Microsoft Start Menu (XP) and Apple menu (OS 7) are examples of integrated text/iconic elements.
Source: Microsoft XP Design Guidelines, Apple Human Interface Guidelines

descriptive and supporting material in the form of written text, icons cannot function. (2) Other sources, although not distinctly proposing that iconic communication cannot be used alone, include similar observations,

> "Icons representing files typically only distinguish between different data types, such as Word documents or Excel spreadsheets. The actual designation of each file is given in accompanying text. Icons representing functions, on the other hand, are often intended to convey completely to a user the function he or she is activating and its purpose." (Baecker, Showing 12)

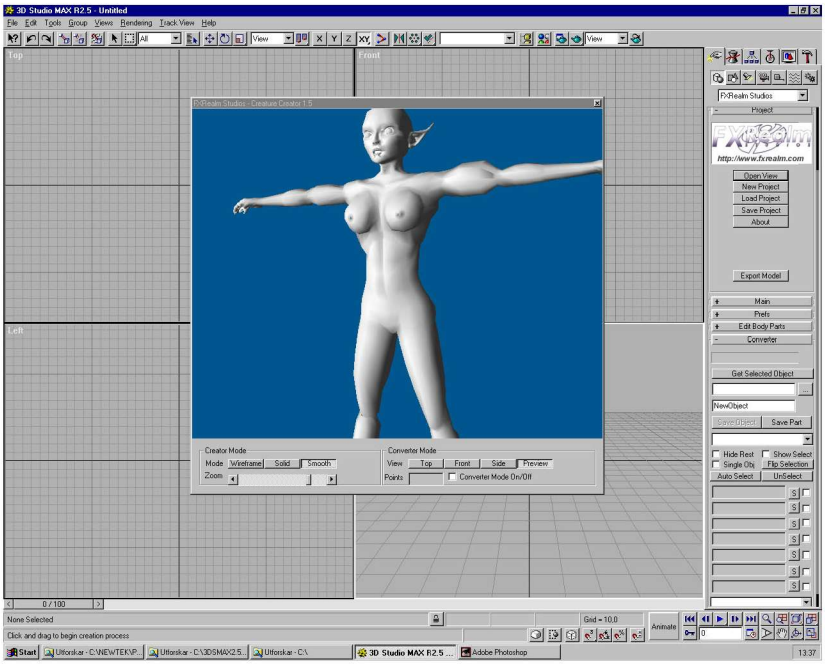
Even considered the fact that each and every object, application or command could be represented by an icon independent of text, the practicality of such an execution would extremely difficult. In present operating systems, it could be said that a harmony of both elements is sought after and a balance is necessary for ease of use, comprehension and usability. In fact, there exists a demand not only for the use of text and graphics, but of all multimedia types; “It is important not to use graphics in isolation, but to combine animation or video with text.” (Baecker, Showing 15)

One of the most perceptible examples that nears opposition to this idea and at the same time validating it, is Poser by Curious Labs. Poser is a 3D-character design and animation application for creating 3D characters, figures, scenes, movies and images. Its interface is a WYSIWYG (What You See Is What You Get) console and complimentary attributes to the Graphical User Interface can be observed in the program. It includes libraries of preset models of human and animal figures, objects and other characters that can be imported, opened, modified and composed in a 3 dimensional environment, complete with lighting and camera angles and allows for manipulation in character details such as gestures, texture, fabric and organic material like hair.

The significance of this application in this study is the general design and functionality of the interface and interface elements that are used for operating this software. Poser is composed a highly visual and graphical

elements whereby most fundamental tasks are guided through iconic communication as opposed to textual commands. Nearly all main elements are composed of symbolic elements that use the maximum capacities in the limits of graphical possibilities. Many symbolic depictions are made for performing tasks in various sizes and colours, and command elements that are regularly performed with menus are often substituted by sets of icons. What is interestingly evident in Poser is that the earlier versions were developed from a command prompt type model, contrastingly an inverse in terms of using the software. As the earlier versions of Poser and other 3D applications including new ones, deploy a more command – response type model of execution, the development of newer versions have a strict progression to a direct manipulation model where functions and results are throughly represented by graphic, symbolic and animated elements some of which have either no or minimal set of textual commands. (Figure 37) Salesin's idea describes why these types of applications strictly prefer iconic manipulation:

> Typical painting and graphic design programs present the user with a menu of operations, such as drawing, erasing, and text placement. Each of these operations, in turn, can have many attributes, such as open width, color, and fill pattern. Often, a user may wish to alternate between several configurations of attribute settings. For instance, an artist might want to use an opaque blue line, then a semitransparent red wash, and then continue with the blue line. Unfortunately, alternating between sets of attributes may require performing many tedious menu operations. Moreover, it may be difficult or impossible to duplicate a previous set of attributes exactly. (Salesin, 103)



a.

b.

Figure 37: Two different 3D application Interfaces.
a. 3D Studio Max 5
b. Poser 4

Therefore, when the required output is composed of graphical, or multimedia related elements, using an iconic or graphical system of commandment becomes better suited, allowing for easy manipulation and easier association. Compared to textual commandment, “graphic symbols are superior to words for representing variation among a set of commands on a graphic dimension (e.g., length, width, brightness) or on a dimension that is easily translated into graphic representation (e.g., representing "more" with "longer)" (Hemenway, 20)

When the principles of the previous sections are depicted, certain points can be extracted. Chu, for example marks some of the following points:

> Icons are visually more distinctive than text and they can be easily recognized.
>
> Icons' syntax are simpler than commands.
>
> Icons are succinct in information representation.
>
> Icons have less learning time than text commands.
>
> Icons are international. (Chu, 315)

To classify certain concepts in order to analyze the different features and functions of textual and visual communication, certain criteria can be established. Many sources either attempt or mention such points that can be regarded as criteria, some of which include, These are Immediacy in Recognition, space occupation, universality, identification, interpretation and modification.

4.3.1. Immediacy in recognition

One of the most fundamental advantages of using an icon based interface comes from the fact that icons “are compact and usually quickly recognizable.” (Baecker, Showing 12). As the foundation of this study is based upon the idea that icons are unique representative images of objects, applications and commands, this proposes that an icon has the ability to stand out among other icons and express individual characteristics, forming singular identity whilst at the same holding on to a family or set of other icons. (Example, Figure 38)

Figure 38: The toilet symbols are an example to iconic communication providing fast and easy recognition.

Text, on the other hand is formed of a very limited number of characters and numerals that are limited to the alphabet of the particular language that it belongs to, and as far as the majority of contemporary languages are concerned, include only tens of characters at most. This makes the discrimination of different texts less comprehensible. Another occurrence

of textual elements is that “The human brain is designed for the handling and understanding of spoken language, but the written, alphabet-based form of a language represents a quite intricate code as Sampson shows” (Vaillancourt, 9) Text commands depend on memory and require that their place be remembered (while icons have a more direct impact of comprehensibility and recognition. As visual forms, all textual elements look similar and there are only very few possibilities of variation, those including font size, emphasizing by underlying, boldness or italicizing and using different typefaces. Icons however have the diversity of alternating in all aspects of imagery, especially after technical limitations have lessened in recent computer system developments.

> “Graphic symbols can be visually more distinctive from one another than a set of words can, and consequently, it is easier to spot a graphic symbol among symbols than it is to pick out one word among words. So, both on visually busy maps and visually display screens, graphic symbols make good, distinctive targets. Also, graphic symbols can represent a lot of information in very little space, and space is at a premium on both maps and terminal display screens.”(Hemenway, 20)

However this does not mean that icons are autonomously easily recognized. Adding too many elements to a glyph or icon based system leads to the creation of a parallel written language, destroying the usefulness of iconic representation. (Vaillancourt, 9) (Figure 39) The recognition process depends on the successful design of the icon and proper association with the signified representative. In terms of design, icons require many properties that enable them to be depicted in this

Figure 39: Figure showing incomprehensibility of icons when too many reside together and no textual label are aiding.
Source: BEOS Icons

way. Another important necessity that regards recognition and perception is for icons to be properly distributed and located throughout the spatial desktop environment. If too many icons reside together then single items become hard to differentiate and hence the fast recognition process fails.

Simplicity, clarity and spatial relief is essential in both individual examples of icons and those residing between groups of icons. Bryne points out the following about the complexity of individual icons:

> "for icons to be effective aids to visual search, they must be simple and easily discriminable. Simple icons are more effective with larger set sizes, allow effective use of icon knowledge, are less affected by a lack of file name knowledge, and are especially effective when they are unique to the display. With simple icons, there is reason to accept the design assumption that icon pictures make finding files easier.
>
> Complex icons are consistently worse than even blank icons. They seem to clutter the display with information that users are unable to employ to their advantage. Thus, if the icons are complex and difficult to discriminate quickly, the assumption that icon pictures are useful in this task clearly does not hold." (Bryne, 452)

Another substantial possibility of intensifying recognition is the use of animated or dynamic icons. Although this method is not as widely used as static graphic elements, it is still apparent in a number occasions for the purpose of conveying certain messages, or helping to show particular actions or functions. This has been observed by a number of people (for example Baecker, Showing 12) and has proved to be effective, resulting in better and faster comprehension compared to static use of visual entities. Although motion applies to both textual and iconic forms, iconic imagery has a greater potential and wider possibility to use animation for

intended explanatory situations, whilst textual elements have only very limited options with the possibility to perform only basic movements or effects.

Recognition leads to another factor that can be used in the analysis of interface communication, which refers to identification and interpretation explained in the following subsection.

> “Icon's distinctiveness could be divided into physical distinctiveness, which is related to recognition of the objects the icon is comprised of, and perceptual distinctiveness, which is related to the understanding of what the objects in the icon represent. An icon is usually designed in an environment that consists of a set of icons. In this setting, the icon has to be physically and perceptually distinguishable among the other icons in the set. Icons performing similar functions, however, should bear family resemblance to increase family distinctiveness.” (Kurniawan, 159)

4.3.2. Identification and Interpretation

> "Icons are designed to represent a function. Therefore, icon has to be related its referent. An icon could be physically recognizable and easily distinguished from other icons, but if its represented function is not understood, the icon could be considered (perceptually) indistinguishable. Numerous studies pointed out that level of representation is a function of domain expertise, familiarity, meaningfulness and appropriateness." (Kurniawan, 160)

Unless icons have been well designed, there may be no way for the conventional user to understand what it represents. Dependency on prior knowledge is one of the main problems that might arise in the case of using icons. Abstraction and simplification are techniques widely used in the design of icons and have a great potential for being misunderstood, or misinterpreted by users. Even very familiar icons may make no or little sense to novice users that have had never used it before. Associations and representations must be interpreted carefully in order for an icon to function properly. In many cases it will be almost impossible to accurately represent an object or command whereby leaving the user to either learn the properties beforehand from other sources or master the function in trial and error, damaging the users experience and task needed to be performed. Complex functions and objects therefore cause problems when trying to be represented by icons. Hemenway comments on the recognition and interpretation processes of an icon:

> "Briefly, according to the model, interpreting an icon for the first time involves:

(a) discovering what the icon depicts; and

(b) "linking" what's depicted with the corresponding command. Subsequent to the initial interpretation of an icon, the ability to recall a command in response to the corresponding icon "cue" entails recognizing what the icon depicts, and retrieving from memory the "link" between what's depicted and the command. In general, it will be easy to determine what an icon depicts, both for the first time and subsequently, if the user is familiar with the object or symbol depicted, and if the depiction is clear and unambiguous." (Hemenway, 22)

Further research for the comprehension introduces the idea of repetition. Apart from initial recognition and "linking" an icon to its referent,

The repetition of graphic elements in several icons can facilitate both initial comprehension of the icons and subsequent retention of the correspondences. First, and most obviously, when graphic elements are shared between icons the user has less to learn than when they aren't, simply because when the user has comprehended one of the icons, he or she can apply that learning directly to the second one. Second, the repetition of a graphic element in several icons can make it easier for the user to identify the features of the depicted object that correspond to features of the commands. (Hemenway, 24)

Repetition further enhances comprehensibility as Hemenway adds that "When an icon for a familiar command and an icon for an unfamiliar command share an important element, the user will be able to make predictions about important characteristics of the unfamiliar command." Text in this respect is easier interpreted, inversely from the fact that its recognition process takes longer time, since text definitions are based upon grammar and language that exist for clarifying ambiguity. The

application of what text signifies applies little to the appearance or familiarity of the word but is regarded with the significance of what the text literally defines in linguistic property.

With the diverse possibilities of similarities, analogies, and a wider possible limits of distinction, although positive factors can be examined, iconic communication also becomes subject to negative aspects and vulnerable to problems. Modest use and design of icons are needed for the above factors to prove successful, and even if these conditions are met, some effects do damage icon interpretation. For example Baecker states that, "There is strong evidence that the impact of visuals can decay significantly over time. Results are also highly dependent on the quality of the visualizations.(Baecker, Showing 15) Another problem is that the diversity of functions that the icon possesses bring with it burdens of performing tasks with the same level of diversity, "The major weakness of icons is that they are sometimes non-obvious. It is also hard to scale a set of icons to handle hundreds or thousands of objects." (Baecker, Showing 12)

One idea that is related with the identification of icons is that of Moyes proposal of considering icons' location as well as the icons shape for determining recognition and identification. Coming from the expectancy that "users learn to associate commands with an icon's shape, and eventually learn to associate its position." he concludes that, "subjects

appear to select either position or shape to associate an icon with a command but not both. Which one they select appears to depend on whether the icons' shapes are easy or hard to learn." (Moyes, 284)

Figure 40 shows this idea in textual menu elements, and a more detailed account of Moyes' research is given below:

> It seems clear therefore, that subjects trained on representational icons, i.e. those whose shape was easy to guess and to learn, relied on that shape, If the shape was changed their performance was disrupted, if it was maintained then randomizing position had little effect. On the other hand subjects trained on abstract icons (i.e. ones whose meaning and association with function was hard to guess and to learn) relied on position to identify the icon rather than shape: if position was maintained then changing the shape had little effect, while if position was randomised their performance was disrupted. (Moyes, 284)

Figure 40: Examples showing the importance of position for recognition and similarity. Internet explorer uses the term "refresh" whilst Netscape Navigator and Opera use the terms "Reload" On the other hand both Explorer and Navigator's refresh and reload are located in the same "view" menu whilst Opera places "Reload" in the Navigation menu. Users become confused when such positions and terms vary throughout the range of programs and interfaces.

4.3.3. Space Occupation

Space occupation and manipulation is a very important aspect of interfaces, especially in Graphical User Interfaces where many elements of size, and dimension are confronted regularly. Spatial property can be regarded in two main points. The first is about the physical space occupation related to the areas in which elements occupy and secondly in spatial organization, that corresponds to how elements are organized and grouped within the spatial environment. When an analogy is made with graphic design principles, this issue could be regarded as economy of visual elements, and order, balance, harmony and rhythm concepts referring to spatial organization. The importance of the factors of spatial property have increased, especially as “The development of new handheld and wireless technologies has resulted in additional documentation needs and in challenging new small screen formats for documentation presentation and access.” (Baecker, Showing 10) (Figure 41)

Regarding the first principle of spacial occupancy, generally icons form a more solid and compact way of representing applications and commands, taking up less space on the whole and having the ability to summarize descriptions. Text, on the other hand occupies more space because it needs, in the least a word's length of descriptions of the functions that it represents. Going even further from this aspect, space allocation for text must be predetermined and chosen for the longest possible entry (Figure 42) that may reside if updated information is to take place, and for

Figure 41: An example of palm device icons. Small devices require space efficient sizes and colors, much like the icons of older operating systems. Designing smaller icons for wider screens makes it unnecessary to design separate icon sets for these types of devices.

international issues (mentioned in the following section). However text is used in limited sizes and positions, so the spatial occupation regards the number of entries more than it does the characteristics and individual sizes of textual items.

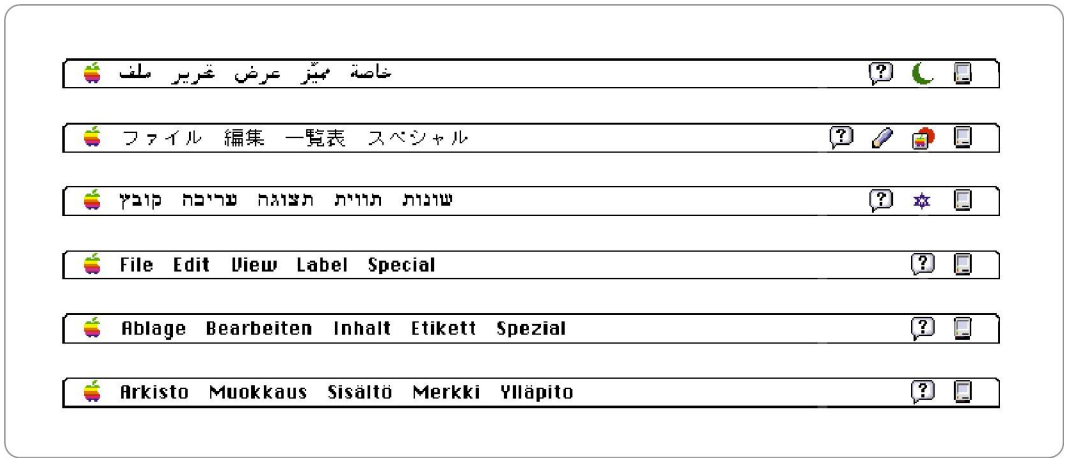


Figure 42: Example showing the importance of space occupation. The longest possible entries must be considered when defining area for textual residement, otherwise the area will not be enough for other languages.
Source: Apple MacOS8 Human Interface Guidelines.

Icons on the other hand are designed in a variety of sizes and bit depths, so require more attention. Chu's “study to define the optimal icon size for small display (wireless and mobile computing) with a psychophysical approach” (314) aims to find the appropriate size for using the same set of icons in a number of platforms (Figure 41) with differing screen sizes. This is an example that icons must be carefully designed to occupy the least space possible on larger displays, whilst be perceptible on smaller

ones. “Although it is possible to use the same icon size that appears on a17-inch screen to a 5-inch screen, the cost of such design is that of limited workspace” (Chu, 314) Chu also notes that icons are more suitable for space efficiency when regarded to those of text, “An envelope icon, for example, consumes less area than the text 'email'.” (Chu, 315)

Regarding the second point, spatial organization must also be considered when designing icons and icon sets, and textual entries and groups. Moyes research seems to underline the importance of spatial organization in terms of identification, ”The experiment reported suggests that if the icon’s shape is hard see as meaningful then subjects rely on position rather than shape in order to identify the appropriate icon.” (Moyes, 283) Vaillancourt also stresses this issue in terms of the users perspective, “Users must make a separate effort to find the icons, choose relevant ones and place them correctly. It is both an analytical and a clerical chore. It is a difficult task, since a correctly filed document on a virtual desktop must be faithful to some form of context, if it is to be retrieved later, as Barreau noted.” (qtd. in Vaillancourt, 9)

Regarding text, menus are the most apparent elements that require planning, since badly organized hierarchal command will require the user to reach deeper and deeper into the subsets of pull-down menus and strings representing the levels of folder hierarchy. If not carefully distributed and organized, it will be very difficult to finds commands and documents that are needed and form a logical visualization of command

locations."Considering that organization has a major influence on retention, the more a user explores the relationships between commands, and, consequently, the more he or she organizes the command set in his or her mind, the more the user will learn and remember about each individual command." (Hemenway, 24) However, apart from the overall organization, textual entries are not problematic as individual entities. (Figure 43, Figure 44) Iconic organization on the other hand requires that the number of icons expected to be located together should be manageable and in appropriate measures. This regards the interface design as well as the design of icon sets and families, requiring that a suitable area and its subsiding elements be properly established to allow successful iconic conveyance, recognition and usability.

> "The challenge for interactive system designers is to balance a limited workspace for information display while maintaining a consistent application interface with Graphical User Interface objects such as Windows, toolbars, icons, menus and pointers." (Chu, 314)

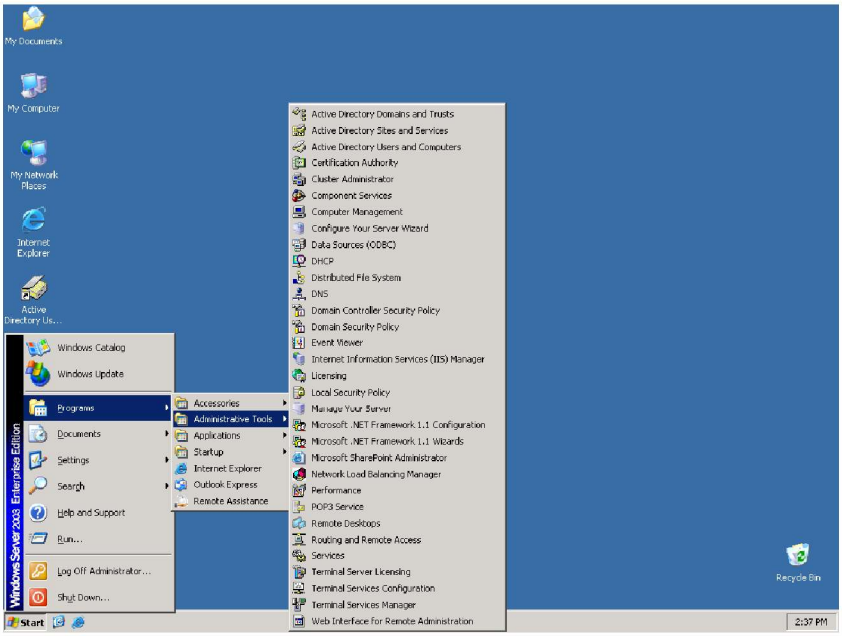


Figure 43: Example showing the levels of menu hierarchy. Textual entities must be considered as deeper levels make it harder to find a target and can also occupy space, damaging recognition.
Source: Windows Server 2000 desktop.

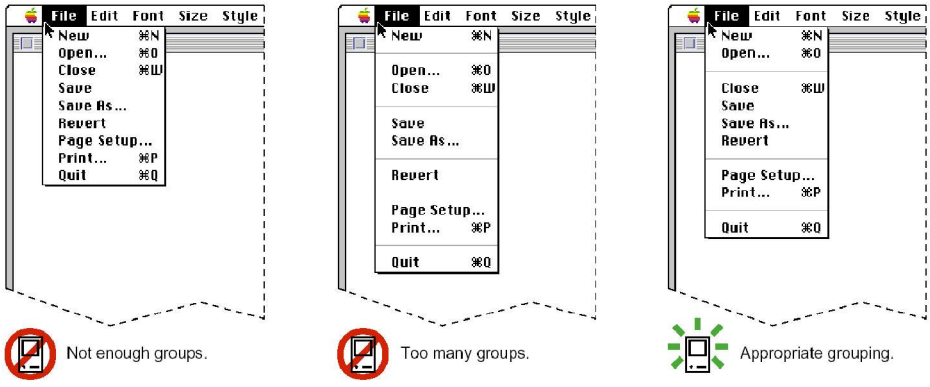


Figure 44: Menu hierarchy. Textual entities do not have as many possibilities of organization as icons. However organizations such as suitable grouping helps usability considerably.
Source: Apple MacOS8 Human Interface Guidelines.

4.3.4. Universality

Since representation and images belong into the category of visual communication, the rules and grammar that are needed to effectively transmit information and enable comprehension can be regarded universal in may respects. Visual communication can be considered apparent in all societies and cultures, and although there are differences and different sets of rules in every culture, it would not be wrong to say that the majority of people all have a basic understanding of visual literacy. "The objective is to facilitate the interaction between the users and their environment by appealing to an immediate understanding of the information represented by the icons. (Frasson, 146) Textual communication on the other hand depends directly to language, grammar and linguistics.

Easterby (1970) stresses the advantages for international use of symbolic displays over those that are language-based, Lodding (1983) asserts that, because people find images "natural," because the human mind has powerful image memory and processing capabilities, because icons can be easily learned and recognized, and because images "can possess more universality than text," iconic interfaces can "reduce the learning curve in both time and effort, and facilitate user performance while reducing errors." (qtd. in Baecker, Bringing 2)

Using icons in comparison to textual commands hence allows for wider comprehensibility with the need for less translation and adaptation required. “An icon can be sufficiently expressive. The international system of road signs is an example of icons that everybody can easily understand. In various domains (medicine, scientific applications, mechanic, traffic control, trade,...) icons are used for communicating with groups of users.(Frasson, 146) Internationalization (Figure 45) is an important issue that software and application developers take into account. The use of icons makes this task much easier to develop and advantages are evident. As Haramundanis mentions, “once the nonnative speaker has learned the basics of the tools, it is much easier to click on a recognizable icon than to read text and click on it, or to enter a command from the keyboard to launch an application or invoke a tool.” (Haramundanis, 2)

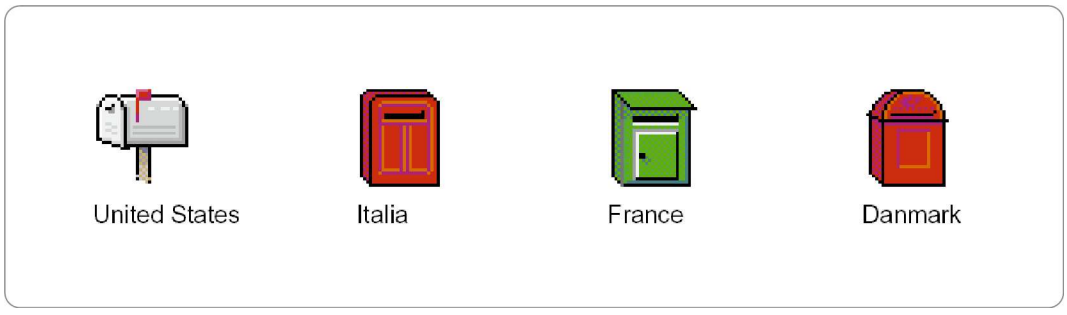


Figure 45: Different iconic representations for various countries, to establish proper signifiers.
Source: Apple MacOS8 Human Interface Guidelines.

Lack of universality, related to textual communication has other disadvantages as well. For example, since the word for a command is composed of a different number of characters in every language, the allocated space for that command in a particular menu, text list or other field would have to be composed to fit the widest possible word amongst all languages (in other words, enough space must be allocated for every text field so that the longest case amongst all languages must be able to fit) Icons however, even if some modification is present do not need alteration in size, presuming they have been considerately designed in the first place, apart from the exceptions of icons that include text, which again is related to language factors.

4.3.5. Similarity in Depictions

Similarity in depictions refers to the the appearance of different icons that belong to alternative functions, programs or commands. The similarity of one icon resembling another can both become an advantageous or a disadvantageous situation. For commands or objects that are common and widespread, depictions of similar appearances in different situations will require the user to only remember and recognize that certain appearance instead of memorizing many alternative appearances for the same or similar type of commands or objects. “when graphic elements are shared between icons the user has less to learn than when they aren't, simply because when the user has comprehended one of the icons, he or she can apply that learning directly to the second one.” (Hemenway, 24)

For example, the representation of very common elements like the folder or file icons, or those which are fundamental commands such as copy, paste, delete, magnify, etc. are all usually composed of similar figures or illustrations with similar stylistic conventions, and use the same metaphorical and conceptual elements. Therefore the user easily distinguishes, recognizes and interprets the functions of these common icons no matter which application, platform or interface s/he uses.

> Similarities between commands that otherwise might not be remembered (or even perceived) maybe perceived and remembered if they are explicitly represented in the icons. The similarities between icons may also encourage more general comparisons

> between particular commands: a user maybe more likely to compare and contrast commands that have similar icons than commands that don't. (Hemenway, 24)

However, similar depictions may also cause confusion, especially for functions and objects that include complexity as those elements that are straightforward and common. To exemplify this point, an icon with very similar figurations of a pair of scissors can be used both as the “cut” command in one program, whereas it can also be used for a program named “Scissors” that could be of an entirely different nature and hence the icon would be used to start the application. Since there are uncountable numbers of applications and commands used within the same platform, occurrences of this nature are likely to take place. Even though two programs may include the same depiction of an icon to perform a function that is almost identical in terms of what is to be achieved, the working principles may differ causing misunderstanding when executing.

To overcome the mixing-up the functions and enable proper discrimination, the generally proposed solution is to produce and use clear and distinguishable images. “In general, simple icons (those discriminable based on a few features) seem to help users, while complex icons are no better than simple rectangles.” (Bryne, 446) In cases where more complex complimentary explanations and referents are needed, the icon is proposed to be aided by other elements:

"Ideally, all attribute values should be reflected in a tool's icon to provide a visual reminder of the attribute settings and to help disambiguate multiple instances of the tool. Unfortunately, it may be difficult to present all of this information in a single, compact, visually appealing, and easily understood icon. However, additional means of identifying particular tools can also be used, such as labeling or otherwise decorating the icons, and organizing the icons on the screen and in (labeled) desk drawers." (Salesin, 106)

4.3.6. Modification

One major disadvantage of using icons compared to using text is that all features that make the advantageous sides of icons are related to the fact that they are thought of, designed, drawn and composed. The act of producing icons is very time consuming and often technically difficult. Cooperation is essential between the interface or software designer and the graphic designer, and requires that the functions of the icon are clearly depicted, and later conceptually adapted and realized to fit its purpose. Every icon requires unique characteristics, considerations and separate composition. While texts can be computed instantly rapidly, modified ad disposed, icons are much harder to compose and create. Only strictly universal and easily perceived icons are exceptions that require little or no modification. Textual entities also require consideration on some aspects however, for example in the case of using different languages, one must bear that conventions can change, (Figure 46) and hence must account for variations and prevent problems.

This is another supporting fact of hybrid usage of textual and iconic elements. Instead of producing separate icons for certain groups of objects or commands, text labels serve as the differentiating attribute, both for distinguishing and for easy manipulation and change. As text accompanies icons to allow easy manipulation to iconic elements, icons make possible the modification of arrangement of textual elements amongst the desktop environment.

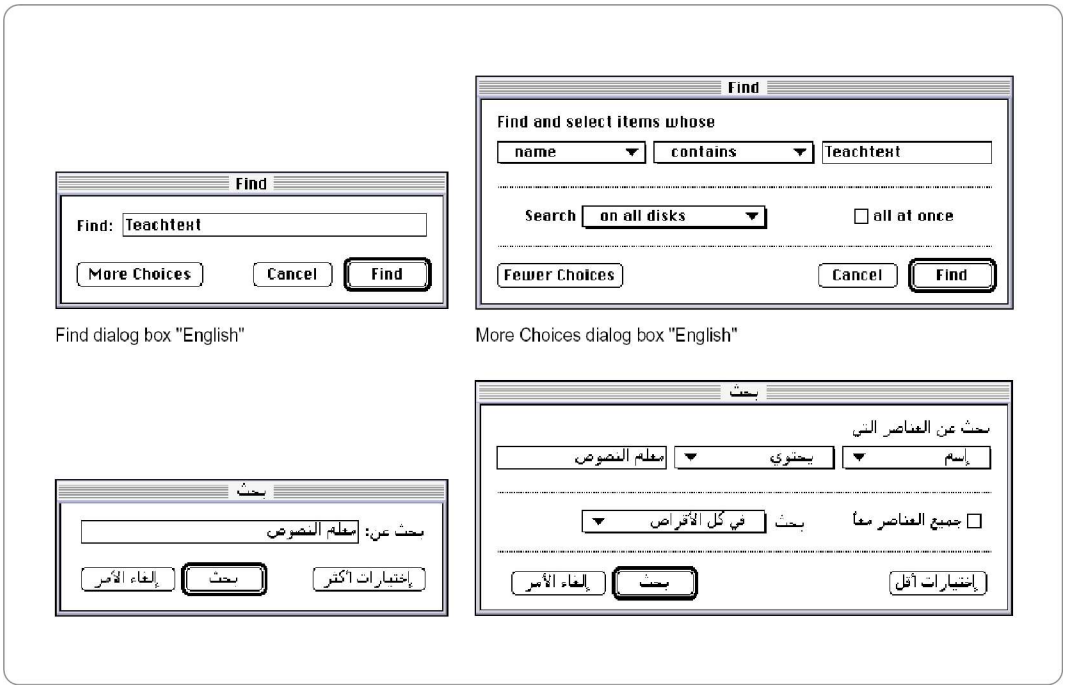


Figure 46: Example depicting the different conventions of text use.
Source: Apple Macintosh Human Interface Guidelines.

5. CONCLUSION

The subject for this study was inspired from the fact that we are constantly affronted with a vast number of tiny images in our increasing interactions with computers. Although at some point we lose interest in individual visual identities and usually take for granted our experiences with these small representations, the aim of this study was to provide a better understanding of how icons have structurally evolved, and how and why they work to order our interactions and tasks on computers. The study of Human Computer Interaction is becoming exceedingly important as computers continue to immerse into many experiences of the daily life of many people. This study could be summarised as an attempt to locate desktop icons in the framework of Human Computer Interaction, concentrated on visual communication.

There are many critical issues in both Human Computer Interaction alone, and the significance of the Graphical User Interface amongst this domain especially including the elements that form the desktop environment of operating systems and applications. After an analysis of desktop icons to define both an introductory historical background of the term and a look at the visual qualities and structural attributes to portray an anatomy, the

critical part of the study included in the latter parts regarded a comparison of textual and iconic communication that tried to distinguish, formulise and compare these two approaches within the framework of Human Computer Interaction and visual communication theory.

Upon completion of the research, the most evident factor that validated my earlier predictions regarding the subject was that the study of Human Computer Interaction is extremely complex and involves knowledge in a vast number of fields and areas. One can encounter many different approaches and perspectives regarding computers and visual communication. It was in my great interest to seek and examine sources on this matter from as many views as possible and to incorporate all findings into one study. However I found this to be difficult and hardly manageable. Coining the different styles, positions and methods of different areas is one hard task and in certain points impossible to accomplish.

One fundamental conclusion I reached while conducting the study was that the present gap of cooperation and understanding of visual communication in different perspectives has improved unimaginably. As many researchers, theorists and practitioners take note, collaboration is vitally important when multimedia applications and software visualisation is concerned. While artists and designers may find difficulty in applying their ideas and works into executable products and digital works, experts in the areas of science and engineering are often deprived of enough

aesthetic practise and graphic design principles to promote and enhance their applications. Apart from producing successful visual and practical applications and systems, usability and psychological issues on behalf of the user are also required. Every detail of a digital project requires the expertise of the regarded area and the successful implementation of all details to form the whole.

If desktop icons are considered in this understanding, they to are reliant to innumerable factors. Besides the distinction from textual properties and other sign systems, icons are in the system of software visualisation and interface design. Representing a command or a single application in a small simple image may seem an easy enough task, however such is not so as my research shows. The creation of successful icons depends on an understanding of many factors, only some of which are highlighted in the body of this work, and ideally require the interpretation of working principles of interfaces, usability, function, aesthetics, and careful consideration of the problems that may arise while producing and using them. The crucial point also evident in this research is that the design of human computer interaction and its elements are tied to past standards, tendencies, and habits and foundation for future models, developments and innovations.

On the whole my research dealt with issues regarding image based communication and textual based occurrences amongst a graphic platform. As the findings also show, the Graphical User Interface is a

descendant of textual based interfaces and stands in a place to further textual entities rather than rival them. In this sense the two different models are interlinked and cannot easily be put into distinct opposing categories. However, comparing the two different methods of Interface elements and using the textual model as a tool provided a solid way to better understand the problems and working principles of the Graphical User Interface, and iconic communication.

Technology and computers are subject to constant change and development so regarding interaction and interfaces, new subjects and research areas come in to light in parallel frequency, leaving ground for many future research areas. Regarding more specific considerations related to future research on desktop icons, a few ideas can be suggested from this study.

First of all iconic communication uses all power of imagery and traits of the objects and environment that it represents. In this sense iconic communication parallels the developments and the expanding possibilities of interfaces and applications to perform in the most efficient manner. Regarding the evolution of icons, technical possibilities limited better icon performance and since the represented objects were simpler and less capable, could not go beyond their initial capacity. However, as these circumstances are improved and expanded, so too are the methods of communication.

The tendency in newer Graphical User Interfaces clearly show vast usage of graphics, dynamic media, audio, video, virtual reality and artificial intelligence, bringing with it highly multimedia driven working environments with dynamic media and spatial depth. Dimension is a key innovation of user interfaces, one that has been imagined long ago, but has only recently been imaginable as a common desktop workspace. The latest commercial interfaces have begun implementing features of larger, clearer and more powerful 3-dimensional attributes and features regarding "time." As cultural desires and living styles also adopt to more cyber and hyper types, the tools and environments develop to render this possible.

Virtuality and virtual reality is starting to become a form of life, as people start to form identities on the Internet, work and play in fictious realms and increase non-physical contact. Interfaces have become the tool for this form of communication substituting for transport, physical structures and enhancing representation, pseudo objects and time perception. Descendant on image dependent and visual oriented societies, the Graphical User Interface has seemed to have evolved as the primary interface in such a path. The importance of Iconic communication comes from this idea that the user is in constant desire to better visualize and participate in such a representation oriented environment.

My predictions from such a study would be to say that iconic and textual communication will continue to expand. Rather than substituting one to the other, they will be further interlinked with each other, with stronger bonds and newer innovations. Moving images, cinema, video and all forms of multimedia will be further integrated into both the interfaces and hence the tools, objects and elements of the Human User Interfaces.

One main responsibility of iconic and textual elements have been to be adaptive to their environment and provide easy recognition and manipulation throughout. Older systems and environments included simpler tasks and limited functions whilst present ones have a complex and diverse features and uses. This brings with it the need for enhanced methods of signification, description and details whilst at the same time being able to clarify and simply the burdens of present chores.

Desktop icons are significantly important in the present situation and are evolving to stand up to this responsibility. Currently, the most noticeable advancement has been the increase in their sizes, colours, visual qualities and semiotic properties. Dynamic icons can also be noticed throughout newer products including simple animations, simulations and flexible size manipulation. They will continue to evolve and will develop to be highly interactive, dimensional and integrated to the complex paradigms of interfaces substituting many other features beyond their visual and graphical capabilities to further their roles in Human Computer Interaction.

REFERENCES

Alben, Lauralee, Jim Faris, and Harry Saddler. "Making It Macintosh: Designing the message when the message is design." Interactions 1.1 (1994): 11-20.

Baecker, Ronald, Richard Mander, and Ian Small. "Bringing icons to life." Proceedings of the SIGCHI conference on Human factors in computing systems, Gaithersburg, Maryland, United States (1991): 1-6.

Baecker, Ronald. "Showing instead of telling." Proceedings of the 20th annual international conference on Computer documentation New York: ACM, 2002. 10-16.

Barrett, Sonia. Graphical User Interface GUI. Nov. 1999. Curtin University Faculty of BEAD Multimedia Design Program. 14 Feb. 2003 <http://multimedia.design.curtin.edu.au/cache/g/0001/>.

Beaudouin-Lafon, Michel. "User interface support for the integration of software tools: an iconic model of interaction." Proceedings of the third ACM SIGSOFT/SIGPLAN software engineering symposium on Practical software development environments, Boston, Massachusetts, United States (1988): 143-152.

Brami, Russ. "Icons: A Unique Form of Painting." Interactions 4.5 (1997): 15-28.

Byrne, John G., David C. Ngo, and Lian S. Teo. "Evaluating Interface Esthetics." Knowledge and Information Systems 4.1 (2002): 46-79.

Byrne, Michael D. "Using Icons to find Documents: Simplicity Is Critical." Proceedings of the conference on Human factors in computing systems January, 1993. Boston: Addison-Wesley, 1993. 446-453.

Chu, Josey, Mikael Goldstein, and Mikael Anneroth. "Icon size as a function of display screen." CHI '99 extended abstracts on Human factors in computer systems (1999): 314-315.

"Definition of HCI." Ed. Baecker, et al. ACM SIGCHI Curricula for Human-Computer Interaction 6 Dec. 2002. ACM/SIGCHI 20 Apr. 2003 <http://sigchi.org/cdg/cdg2.html#2_1>.

Edwards, Alistair D.N. The rise of the Graphical User Interface. Rochester Institute of Technology. 3 Mar. 2003 <http://www.rit.edu/~easi/itd/itdv02n4/article3.html>.

Every, David K. Microsoft, Apple and Xerox: The History of the Graphical User Interface. 11 Sept. 2002. MacKiDo. 16 Feb. 2003 <http://www.mackido.com/Interface/ui_history.html>.

Francik, Ellen. Computer- & screen-based interfaces: Universal design filter. 6 June 1996. Trace Research and Development Center. 1 May 2003 <http://trace.wisc.edu/docs/comp_screen_interfaces_96/comp_screen_interfaces.htm>.

Frasson, C., and M. Er-radi. "Principles of an icons-based language." Proceedings of the 1986 ACM SIGMOD international conference on Management of data, Washington, D.C., United States (1986): 144-152.

Gentner, Don, and Jakob Nielsen. "The Anti-Mac interface." Communications of the ACM 39.8 (1996): 70-82.

"Graphical User Interface." FOLDOC - Computing Dictionary. 25 Apr. 2003 <http://foldoc.doc.ic.ac.uk/foldoc/foldoc.cgi?query=GUI>.

GUI - Graphical User Interface. 2001. Computer Knowledge. 25 Apr. 2003 <http://www.cknow.com/ckinfo/acro_g/gui_1.shtml>.

"GUI." SearchWebServices.com, presented by TechTarget. 25 Apr. 2003 <http://searchwebservices.techtarget.com/sDefinition/0,,sid26_gci213989,00.html>.

Hadley, Daniel A. Alternative User Interfaces. 6 Dec. 1999. New Jersey Ins. of Tech. 12 June 2003 <http://eies.njit.edu/~turoff/coursenotes/CIS732/samplepro/dan.html>.

Haramundanis, Katherine. "Why icons cannot stand alone." ACM SIGDOC Asterisk Journal of Computer Documentation 20.2 (1996): 1-8.

Hemenway, Kathleen. "Psychological issues in the use of icons in command menus." Proceedings of the SIGCHI conference on Human factors in computing systems, Gaithersburg, Maryland, United States (1982): 20-23.

Henry, Tyson R., and Scott E. Hudson. "Multidimensional icons." ACM Transactions on Graphics 9 (1990): 133-137.

Horton, Sarah, and Patrick J. Lynch. Web Style Guide. 28 Apr. 2003. 5 May 2003 <http://www.webstyleguide.com/index.html>.

Horton, William. "Designing icons and visual symbols." Conference companion on Human factors in computing systems. April, 1996. Vancouver. British Columbia, Canada (1996): 371-372.

Huang, Shih-Miao, Kong-King Shieh, and Chai-Fen Chi. "Factors affecting the design of computer icons." International Journal of Industrial Ergonomics 29. (2002): 211-218.

Kare, Susan. Susan Kare User Interface Graphics. 14 July 2002. Home Page. 5 Feb. 2003 <http://www.kare.com/>.

Kurniawan, Sri H. "A rule of thumb of icons' visual distinctiveness." Proceedings on the conference on universal usability, 2000 on Conference on Universal Usability, Arlington, Virginia, United States, ACM (2000): 159-160.

Lineback, Nathan. The GUI Gallery. 4 Jan. 2003. Personal Page. 3 Mar. 2003 <http://toastytech.com/guis/index.html>.

Ludi, Stephanie. "Animated icons: re-inventing visual cues for the visually impaired computer user." Proceedings on the conference on universal usability, 2000 on Conference on Universal Usability, Arlington, Virginia, United States. ACM Press (2000): 145-146.

Marcus, Aaron. "Icons, Symbols, and Signs: Visible Languages to Facilitate Communication." Interactions 10.3 (2003): 37-43.

Marinilli, Mauro. The Theory Behind User Interface Design, Part One. 2003. Developer.com. 2 June 2003 <http://www.developer.com/design/article.php/1545991>.

Microsoft Developer Network. Critical Thinking in Web and Interface Design. 2003. MSDN Home Page. 1 June 2003 <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwue/html/ch14a.asp>.

Microsoft Developer Network. Visual Design. 2003. MSDN Home Page. 1 June 2003 <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwue/html/ch14a.asp>.

Moyes, Jackie. "When users do and don't rely on icon shape." Conference companion on Human factors in computing systems. Boston, Massachusetts. ACM (1994): 283-284.

Multimedia: From Wagner to Virtual Reality. Ed. Ken Jordan, and Randall Packer. New York: W. W. Norton & Company, Inc, 2001.

Nielsen, Jakob. "Noncommand user interfaces." Communications of the ACM 36.4 (1993): 83-99.

PARC History. 15 Oct. 2002. Palo Alto Research Center (PARC). 17 May 2003 <http://www.parc.com/company/history/>.

Patten, James, and Hiroshi Ishii. "A comparison of spatial organization strategies in graphical and tangible user interfaces." Proceedings of DARE 2000 on Designing augmented reality environments (2000): 41-50.

Press, Larry. "Personal computing: Windows, DOS and the MAC ." Communications of the ACM 33 (1990): 19-26.

Rubens, Philip, and Robert Krull. "Communicating with icons as computer commands." Proceedings of the 6th annual international conference on Systems documentation, Ann Arbor, Michigan, United States (1988): 25-34.

Salesin, David, and Ronen Barzel. "Adjustable tools: an object-oriented interaction metaphor." ACM Transactions on Graphics (TOG) 12 (1993): 103-107.

Shneiderman, Ben. Designing the User Interface. 11 Dec. 2002. Addison-Wesley & Benjamin Cummings. 14 May 2003 <http://www.aw.com/DTUI/>.

The Human User Interface. 5 Feb. 2003. ScienceNet and Science Line. 24 May 2003 <http://www.sciencenet.org.uk/soundofsci/interface.html>.

User Interface Design. 23 Feb. 2003. Usernomics. 20 Apr. 2003 <http://www.usernomics.com/user-interface-design.html>.

Vaillancourt, Alain D. "Generated glyphs as memorable desktop icons for document." Proceedings of the 1998 workshop on New paradigms in information visualization and manipulation, Washington, D.C., United States (1998): 9-12.

Wagner, Annette, Patrick Curran, and Robert O'Brien. "Drag me, drop me, treat me like an object." Proceedings of the SIGCHI conference on Human factors in computing systems, Denver, Colorado, United States. ACM Press/Addison-Wesley Publishing Co (1995): 525-530.

Writing. 12 Feb. 2001. The British Museum. 10 May 2003 <http://www.mesopotamia.co.uk/writing/home_set.html>.