

the sad loss of visual abstraction in graphical user interfaces

*“Is the era of abstraction
over since the introduction
of hyper-reality in a
liquified world, or has
the final hour for the icon
finally arrived?”*



Piet Zwart Institute
MA Media Design

by Jorrit Sybesma

Foreword

In short, the essay is about the sad loss of visual abstraction in graphical user interfaces, mainly researched through the question whether the era of abstraction is over since the introduction of hyper-reality in a liquified world.

From this main statement, the following research questions derived:

- What is the sad loss of abstraction?
- What is the influence of visual abstraction in a GUI on the many aspects of usability?
- What are the possible causes of the loss of visual abstraction?
- What if we would let the matter of metaphorical madness rest?
- Why are we still trapped in the paper paradigm and the endless office-metaphors?
- What if we could find the anti-metaphor solution in further GUI abstraction?

Jorrit Sybesma

Rotterdam, December 2006

-- -

* Probably superfluous to mention, but in the essay I use often abbreviations:

GUI - Graphical User Interface.

OS - Operating System (OSs, Operating Systems).

Windows - referring to the most recent version of it: Vista.

MS - referring to Microsoft.

MacOS / OS X - referring to the Mac OS, version X (10.x), unless mentioned otherwise.

Table of contents

Those research-questions are separated into the next structure of chapters and paragraphs:

Chapter #1: Reduce to the max	1
<i>From abstraction to distraction.</i>	
§ 1.1 Visuality and predictability	1
§ 1.2 Dealing with increasing complexity	5
§ 1.3 Bureaucracy of standards	7
Chapter #2: Visual Machismo	9
<i>In a place without abstraction, visual anarchy will prevail.</i>	
§ 2.1 Popularising subcultures	9
§ 2.2 Because they can	10
§ 2.3 Future of the features	11
Conclusion: “Where do we want to go tomorrow?”	13
<i>Escaping the paper paradigm.</i>	
List of references	15
Literature	19
Enclosed images	20

Chapter #1: Reduce to the max From abstraction to distraction.

Except for the fact that it is a sad loss of a specific art form - the craftsmanship of “reducing to the max”(1) - losing visual abstraction within GUIs also has negative influences on the usability and the user experience. In this essay I will focus on various aspects of the graphical representations of functionality within GUIs - and the underlying organisational structure - with an emphasising on the role of the icon (the object that reflects the concept of the desktop-metaphor best) within the OS (using Windows and MacOS as primary examples).

§ 1.1 Visuality and predictability

Abstracting means losing redundant information and reduce - in case of creating an icon - the image to its “maximum”(2). With the purpose to gain visual and functional strength. In the case of designing those tiny parts of the OS, it is not so much about trying to reflect everything that is represented by the icon, the designer should be engaged to find a shape or form that is distinctively different from others, reflecting the main task of the represented object. The difficulty is not showing as much as possible, the craftsmanship lays in visualising that what is necessary, reducing the object to its functional essence, perfectly carried out by Macintosh icon-designer Susan Kare(3).

According to its definition an icon represents an object and has some of its visual qualities. In semiotic-terms it does not mean that the object it refers to has to be copied as an one-on-one representation. In that case it would not be an icon anymore, but a photograph or picture. On the other hand, how can an icon visually represent something that is so abstract as an application, a string of 01010100. In that perspective the icon should only represent the task of the application or identity of a file. But an icon is a file, but an icon also refers to a file. How to classify what, when an alias of an original file is represented by an icon, in a file-system where every object is a file? Semiotic-wise, icons are hard to classify correctly, since an icon is a pictogram, which is a pictograph which represents a concept, idea, activity, place, event, all by illustration. Not by photographs.

Derived from an original Russian Orthodox term, the icon should be more than just an illustration. According to David Smith(4) - the guy who 'imported' the term icon - it should "embody properties of what it represents"(5). In comparison with its Russian origin, the Russian icon of a saint "is holy and is to be venerated"(6). In the case of computers, the icon should "contain all the properties of the programs and data represented, and therefore could be linked or acted on as if they were the real thing"(7).

The "as if" in the last sentence seems to be part of the "paper paradigm"-issue(8). Every object in the GUI is a metaphor, and for users it is hard to distinguish the real object from the referring one. John Siracusa(9) considers the knowledge about "relationships, representations of or interfaces to" as something that is "constituted unnecessary mental baggage", for the normal user. An icon is an icon and a file is a file, a file has an icon and an icon is not the file and the file is not the icon. It is quite abstract, but as long as clicking the icon fulfils the expectations of the user - whether that implies working with the concept of shortcuts, aliases, or files - the metaphor is successful.

The definition of the icon is quite abstract. That abstract term has to deal in itself also with abstraction when it comes to its actual representation. Although I see the art of icon designing as an art of abstraction, others see it as making miniature portrait painting(10) - somehow not entirely strange since the Russian Orthodox icons were mainly portraits of saints - practised mostly during the 16th Century in Europe. Perhaps the disappearance of the craftsmanship of both is similar. Icon-design and miniature portrait painting are both repressed due to the growing popularity of photography. Portraits were no longer necessary and icons can be lifelike pictures nowadays. In both cases with trying to represent the pictured original as resembling as possible. That means boldly: using the wrong scale, while trying to add too much details to it. The design of the icon should primarily not reflect the craftsmanship of the designer itself, or the beauty of the object nor be copy of the original it is representing, overall it should be useful and serve as "visual mnemonics"(11).

Nowadays icons seem to represent an expression of feelings. Without being articulated about what actually will happen after (double-)clicking the icon. There is no clear association between object and task, action and consequence. The icon becomes a victim of the contemporary approach of interface designers, that

everything is an experience and every task has to be spiced up, metalized, liquified. By adding another colour or flavour to its appearance. The visual elements within the GUI should reduce the complexity by increasing the predictability. It is a straight forward statement, nevertheless true and it supports my argument to have abstraction to a certain level. If you make objects that should increase predictability more complex - what seems to be the tendency - evidently the complexity of the whole increases and the predictability is reduced.

Besides increasing predictability, by reducing complexity, abstraction can play a role on a larger scale within the GUI. When navigating through the system, clicking somewhere can have unexpected results (unpredictability), due to strange combinations of (inter)active and inactive parts of the GUI. In my search for finding proof to support my arguments that visual abstraction (“reduce complexity”) within GUIs should be considered as something precious, I bumped into Fitts’ Law(12). An almost forgotten mathematical experiment done in the 1950’s, when mathematician Paul Fitts did research to whether and how the size and the distance of a target - which had to be reached by the use of a pressure-sensitive pencil - would influence the approach speed towards the object. The outcomes were not really resounding: the speed of the pencil increased when the size of the target is larger and the speed of the pencil increased when the distance to the target is shorter.

Some may consider the GUI of Windows more efficient, since every application has its own environment(13), in practice - according to Fitts’ Law - the principle of the MacOS is more efficient. Let me clarify that - because it seems a bit contradictory - since in theory the distance between mouse and target is smaller in Windows, that would implicate that Windows would work more efficient and faster. Here comes the power of abstraction. The MacOS menu-bar is visually very simplified, fixed on the top of the screen, physically cut of by the border of the screen. But virtually it has an immense vertical height, which is - because of obvious reasons - invisible. Efficiency of abstracted objects, in this case, has major influences on the effectiveness of working. When using MacOS, you simply can ‘throw’ the mouse cursor to the top of the screen and it will always end up on one of the abstracted menus. Working with Windows implies that users have to navigate carefully towards the menus, since these are not on the side of the screen and since the ‘clickable’ area is pretty vague, due to over-designed and detailed surroundings. Conclusively one could say: “less is more”(14) - a form of abstraction - indeed. Now scientifically proved.

Icons are nothing more than “visual mnemonics”(15). The icon as an object has to be easily associated with a specific task, source, action or goal. Referring again to the concept of implementing an universal language across the OSs, every object is recognisable for its purpose, regardless the system one uses. On the other hand, one could oppose to that by addressing that the personal and emotional association by users with specific objects should be incorporated when it comes to providing options for customising the GUI or icons as a part of that.

Icons become more realistic, photo-realistic - “as if you are dragging around portraits of saints”(16) - and larger in size, designed for 128x128 pixels (Xerox Star-icons were 2.5 cm in size). When those giants are downsized, almost nothing is left of what should be the association between the visual and the functional, except for a little “smear”(17). Objects become more realistic without becoming more meaningful(18). An interesting icon to focus on is the hard-drive icon, on MacOS a photograph of an actual hard-drive. Experts associate that icon with the hard-drive inside their machines, after a while non-experts will do as well. The point is that there is no good reason for using the whole picture, without downscaling the details. For someone who is not aware of the fact that a real hard-drive looks similar as the icon does, the extra visual information is not meaningful. Extra visual information does not add extra meaning to the icon. It is better to leave abstract terms abstract in their representation, no photographer or designer can explain the world in an icon. Anyhow, associative learning (connecting icon with purpose) benefits from the relation between objects in the physical world and their metaphorical use within a virtual environment. A perfect example is the wastebasket on everyone’s desktop since 1981. An abstraction from the real object fulfils our expectations, without being photo-realistic.

Visual strong represented functions (not over-detailed), have not only influence on our behaviour, or memory in a virtual environment. Even outside the box, those elements influence our language and thinking (thinking ‘undo’ after dropping a glass of water). A student-research project “Digital Terminology”(19), showed that the strength of the visual representation of specific interface-elements, determined whether these elements become part of a ‘new-speak’. The association of object with function seems not to be constrained to a specific medium, a virtual object with a visual and functional equal and resembling physical one can act and is encountered with the same expectations as the ‘real’ one in a virtual environment. On the other hand, virtual functionality of an object can easily be projected

by accident on a physical similar object, with a different function(20). Association of the object with its function seems to be stronger, when abstraction is done to a certain level.

I freely adopted a research-method performed by researcher Bill Buxton(21) to support my argument that visual abstraction influences memorising functionality positively. I asked ten people to draw the object that is used to 'delete' files and folders on their computer.

Although the object does not represent its functionality correct(22), every questioned could reproduce the icon. Not the 3D-objects were drawn - those contain too much visual information that is not necessary to remember what the function of the object is that is represented by the icon - but the most abstracted wastebaskets(23) were remembered best. That pleads unmistakable in favour of using abstraction when it comes to "shaping the end user's mental model of the system"(24). Achieved by reducing visual objects to the essence. It also implies that the argument of the Microsoft development-team that customisation is the best answer to better visual mnemonics is perceptible invalid(25). Uniformity and simplicity are the keywords according to my opinion, supported by the outcome of the drawing exercise(26). It proves that the effort that has been put into the development of realistic icons (photographs), is wasted, it has influence on the user-experience on a superficial level.

§ 1.2 Dealing with increased complexity

The tasks of the machines are getting more versatile, first the computer was therefore to work on. Now it is part of our daily lives, even when relaxing at home. The computer seems to get more complex, and the tendency is that the interface becomes more complex equally. Even the tiniest elements of the interface reflect the struggle - trying to incorporate even the experience the user has using the application as a part of the icon - of the designers with the pc as an obsolete concept in the form of a fixed, boxed machine. Applications have to be more versatile as well, dealing with more information streams than ever, being interconnected and intertwined with local and remote destinations as a part of a multi-user environment.

Instead of simplifying that complexity by adding a literally crystal clear, easy to use, minimised interfaces, the designers of the contemporary GUIs seem to feel the urge

to 'show it all', by trying to translate the technological power into a visual answer by adding eye-candy, features, widgets, gadgets, animated pictograms, dancing windows, real lifelike drop shadows, exposé's, dashboards, front rows, et cetera. We have to work in a world of 'experiences', listening to music becomes an experience, surfing on the Internet becomes an experience, watching a movie becomes an experience. Every task seems to be transformed into an interconnected system-wide 'Disneyfication'(27) of its original. These kinds of experiences are accompanied by advertisements in applications, flashy images and unnecessary waiting time before one can actually start. Icons become emoticons. For the interface designers it should not be their task to dictate our feelings and expressions, or tell us what to think, but it should be their task to break up the complexity and versatility of the computer into pieces. Perhaps literally, physically.

It seems that the developers are buying time, or we as consumers are actually paying for that. The interface needs the time to grow with the increasing complexity of the machine and somehow I consider the superficial direction of nowadays developments, not as the right one. These dirty old men try to seduce us with candy, to keep the user interested in buying new software and therefore new hardware and therefore new software and therefore new hardware. Till someone comes up with a new concept for either the theoretically ideal user interface - will it be Jef Raskin's mode-based Archy(28) - or the pc of the future.

In the late-nineties there were some attempts to change the GUIs radically, referring to the Microsoft's Bob-metaphor(29) and Navigator by Packard Bell(30). Absurd examples of metaphoric madness where the desktop metaphor was extended to a the size of living room, represented inside the computer. The public was not really interested in both products. Operating a computer through a clumsy virtual reality, was not the visual and conceptual Walhalla, a three-dimensional environment makes operating a system even more complex. Our contemporary computers are not well suited for that job(31). Due to the lack of 3D-controlling devices (mouse is a 2D-navigation device) and technically most pc's can not handle the overload of visual data. It is all "putting your interface in the way of the goal"(32).

§ 1.3 Bureaucracy of standards

A logical response to the vast growing complexity of the OS, the GUI and the computer as a whole, would be visual standardisation. In order to get more grip on the graphical qualities of each function within the GUI, narrowing down visual inconsistencies. Something that is probably only reachable through developing guidelines across multiple platforms, which could fix various aspects that are specifically related to the representation of system-functions within the GUI.

Something that has been done before, when the term 'icon' was incorporated and applied to the field of interface design as what we nowadays now as iconography, by David Canfield Smith, in 1975. Developers at that time (1980's) seem to be convinced by the power of the use of images, instead of "squares"(33), or other too abstracted forms that represented a function. Through the years, the original definition has been expanded and abused to its maximum. What inevitably should result in a re-definition of the original term, or there has to be done some expanding work in the vocabulary of modern iconography, by adding new terms and definitions. One could consider thinking about a term that would incorporate dynamic qualities - "icons changing appearance depending on the properties of object they represent"(34) - of what is now called an icon.

For every visual element that is or will be made for the MacOS-platform, an extensive set of guidelines(35) is issued by Apple. Unfortunately these guidelines are quite general and optional to apply when it comes to the development of icons and other interface related items and they focus mainly on experience aspects regarding size, colour, shadow and transparency. None of the guidelines deal with real usability issues. Apple suggests however to hire an icon designer, Susan Kare perhaps? The consequence anyhow is an inconsistent interface(-skin) where system-wide functions are referred to by different icons.

However, a solution to actual increase predictability and reduce complexity would be considering an universal visual language, preferably used across multiple OSs. Similar to the 'pictoral'-utopia Neurath and Arntz(36) promised us, with their attempt in the 1930's to create an universal pictographic language(37). Maybe a better example is the attempt by Charles Bliss, to create a pictographic language(38) to overcome linguistic problems, which is now still in use. In general all efforts in creating an universal pictographic language failed, often because even icons are

subject of discussion and can be explained in multiple ways. In the context of user interfaces, it would be a horrible formalisation. In the field of designing communicating complex instructions or tasks, it is often part of the process to get 'certified'. During my time as an intern at a design firm(39) specialised in wayfinding, certifying an icon or a pictogram was a regular part of the process. Certified icons were added to an international library(40), containing thousands of pictorials, which are shared amongst the design firms that work in the same field. It is quite bureaucratic and it would limit the artistic freedom within the design field of iconography.

Chapter #2: Visual Machismo In a place without abstraction, visual anarchy will prevail.

Standardising does not seem to be the solution to achieve uniformity in the visual language within interfaces. Every attempt to achieve the implementation of an 'universal' language - regardless its scale - failed, whether it was physical (pictorial). or virtual (GUI-inconsistencies). In this next chapter a few underlying issues are treated that possibly had effect on the interfaces, as we know them by now.

§ 2.1 Popularising subcultures

Interesting to see is that software-applications which are part of the popular mp3-software culture, seem to earn most of the attention by so called interface designers. Some examples show that the interfaces are not generic in use anymore, but are very specific for controlling music only. The term "MP3Faces"(41) is born, and it were those applications that even before the 'Aqua-life' of MacOS and Vista showed derogatory interfaces. As if an extraterrestrial force attacks my local machine by software with poor 'game'-quality (inter)faces(42). It is a strange way of trying to customise something that is very generic. This customisation is similar in the by MTV popularised car-pimping bling-bling subculture(43) shows. To pick up the garage and car metaphors of Neal Stephenson(44) and Bill Puxton(45). Very common cars are 'adjusted' to the needs of an individual. Often the customisation is only on an aesthetic level, technically these cars (or media-players) are not capable of doing things differently, after these space-attacks. Hilarious enough often the software is slower, and the pimped cars as well due to adding heavy-weight non-functional visual elements. What You See Is What You Get?

The interface should almost be invisible, since the user is not really interested in the interface, but in the actual work that has to be done through the interface. In the case of the mp3-software it is quite clear that not so much the functionality of the interface itself is interfering in the workflow, but the visuality of the interface is the noise in the communication between user and computer. It would be superfluous to mention that visual abstraction of specific functions of and within the GUI would be in favour of the usability here.

Those “MP3Faces” seemed to get so articulated and visual separated from the rest of the software - living autonomously inside the machine - that they give birth to an actual physical device. The perfect example of such a trajectory is the transformation of iTunes into the iPod. The iPod was already there since the introduction of iTunes in 2001, only virtual. The consequence of being too explicit and getting out of control is that the interface ‘searches’ a new medium to evolve further in, escaping out of the box and starting a new life outside the environment of the paper paradigm. Finally the whole computer will be disintegrate into small pieces. If it is the result of lack of abstraction, than it would be a positive consequence.

§ 2.2 Because they can

Another aspect is technical and legitimised by the non-argument: “because they can”. As an example I would like to take a look at the liquid interface by Apple as well as Microsoft. For Apple to argument to use the Aqua-interface was actually quite plain and simple. Jonathan Ive designed the original iMac in 1998, which had translucent and transparent elements, combined with shiny colours. On that iMac the boring and grey MacOS 8.5 was installed, which did not reflect the ‘Think different’(46) approach by Apple. During the same time OS X was in development, first in the same grey colour-scheme, very late in the process of development, a “liquid, click it, you can lick it”(47)-look was introduced. Probably it were the visual similarities between the GUI and the physical hardware that would form the actual argument of implementing the Aqua-interface. Now there is the brushed-metal-interface, which not only seems to represent the visual similarities between the hardware and the OS, but also represents the changed - closed and defensive - kernel-policy of Apple(48).

The technical progression of hardware elements and specific graphic-software was the actual reason for the introduction of the liquid-world. If it was not technically possible, it would not have been done. It is a show-off, designers try to exceed each other, by constantly raise the level of reality. Fuelled by the gaming industry(49) and its needs for sophisticated hardware and OpenGL-software. Heavy visual hardware-dependent GUIs seem to become a selling point. The OS requires new hardware and the new hardware requires new software again. A perfect self-fulfilling commercial machine, very well illustrated by the strategy of the introduction of Microsoft’s Windows Vista.

After seeing Apple using liquid-graphics, Microsoft could not wait to “start their copiers”(50) in Redmond. They incorporated even more “visual features” and “aesthetically pleasing solutions”, to make Vista “more radical”. The business-strategy to sell ‘visuality’ is more interesting. Illegal users of Vista do not have all these visual elements and hardware-heavy graphics and therefore this group of users have a faster pc. Consumers that are willing to pay for Vista, are confronted with the strategy that for each level of visual complexity another few dollars have to be paid(51). The cheapest system does not contain “transparency or translucency, window animation, Windows Flip 3D” - what a pity - but is anyhow the fastest. The more expensive the OS becomes, the slower it gets.

Strangely the need to simulate reality is not only visible in the GUIs. It is a fashion item, one of the features of for instance the Web 2.0-promise is also the power of liquid aesthetics(52). Perhaps we are in the era of liquid aesthetics, it seems not to be limited to interfaces, icons and websites only. These effects are also adopted as functions in various software applications. Reflections, liquid shapes and colours appear in commercials, advertisements, brand identities and publications. This tendency shows some similarities with the visibility of new Photoshop-filters in each and every design, during the early days of the desktop publishing revolution. When new software functionality was quite often adopted by designers without any criticism.

§ 2.3 Future of the features

Nothing has been fundamentally improved or changed since the introduction of the paper paradigm in 1981. Since that date, all the so called ‘innovations’ of the OS are mostly visual, instead of dealing conceptually with the constraints of the original invention of the desktop paradigm. The metaphors are part of the visual and conceptual translation of a desktop into a computer environment, and although that is a subject worth a pile of essays, it is one of the key-reasons for the stalled innovation and the visual cacophony. The developers at Apple and Microsoft seem to have the idea that they found the ultimate metaphor already.

The lack of real innovations in OSs has been covered up by eye-candy, to mislead the user and in order to invent yet another commercial machine, transforming normal users into hardware-addicts. Developers seem to be distracted, only developing fun-features. Avoid thinking thoroughly about the future of the OSs, which seem to

grow towards each other, the differences lay only in the number of unnecessary gadgets - for a normal consumer who does not care about the underlying technological and conceptual differences between Windows and MacOS. Modern day computing is a war of aesthetics.

At some point in the nearby future, this visual rat-race will also stall, due to a technical halt. Causing panic at the feature-developers of Apple and Microsoft. Eventually both OSs will merge - they can emulate each other already - and according to an article on that subject - Windows will be the OS on every Mac(53).

The idea that we enter the era of the feature, is also confirmed in the image of the future GUI, sketched in Hollywood-productions. Where the impossible is raised to be the standard, virtual reality and three dimensions form the bases of those idiocy. Complex and unpredictable, useless visual spatial systems challenge the hero, who is - of course - able to understand the working principles of every GUI immediately(54). Relevant to refer to this 'future' is not so much the visual nonsense or to support my assumptions that "3D is there for demos, and 2D for work"(55). More relevant are the implications of those GUIs on the expectations that consumers and investors have of the technology and usability. These science fictional ideas do have their influence on research-funding. When specific 'effects' are incorporated in existing GUIs, those become evidently more complex, in this context the term 'Disneyfied' seems to be justified.

On the other hand if a hero can control any GUI, as a spectator one identifies him or herself with the hero and expects to understand every GUI as well. Automatically blaming themselves for not being able to work with encountered GUIs. "We need people to start demanding easier design and blaming the technology when it is too hard to use"(56). Anyhow, it should be so that the designs are "human-centred" instead of "technology-centred"(57).

I am afraid though that indeed concepts mentioned above like three dimensional GUIs will be reality within the nearby future. Apple already has patents for spatial piling on the desktop(58) and much more visual nonsense. That extra layer of spatial complexity is already added to the new OS of Apple(59), and will be part of Vista as well. Other concepts which are a little more radical are probably waiting to be commercialised within the next five years...

Conclusion: “Where do we want to go tomorrow?” Escaping the paper paradigm.

The sad loss of abstraction is a much larger issue than I originally anticipated, extending to a wide area underneath the surface, from metaphors to technical issues, all with a great influence on the interface itself. The development and the focus of attention on the GUI nowadays are symptoms of a far more serious problem than just the sad loss of abstraction. It is the lack of fundamental innovation in the area of interaction between human and computer. Developers are convinced of their desktop paradigm and all the changes that do happen these days, are superficial and indicates we are in a period of relative peace.

In this conservative time-span, developers reach back to the values and concepts and all that their predecessors already have achieved. Now that it is technically possible the developers are in pursuit of simulation reality in GUIs to the level of perfection, instead of finding a new purpose for using that technological progression. Nothing has really been improved since 1981, technical progression has not led to conceptual change, we are still stuck with the paper metaphor of the ‘desktop’ with icons that refer to real objects. In 1981 that was radical and the visual abstraction was necessary due to technical constrains, those constrains forced us to think about reducing to the max. Now everyone can uncritically give expression to each and every form or function, a craftsmanship is lost: everyone is a designer. On the other hand, essential changes in social structures and society, question the use of the desktop metaphor, since we no longer constrained to our ‘office’.

Create a new visual language, all attempts to find a solution in virtual reality failed, do not search the solution in simulating reality. The icon is not “here to stay”(60), it is an in-between solution while waiting for a truly new environment, where we lose metaphors, get rid of paper paradigms, feel not constrained by physical values of reality, act like grown-ups and develop through further abstraction a real visual virtual vocabulary, distinguish function from form again.

Too many arguments (linguistic, semiotic, historic, artistic, mnemonic, didactic, metaphoric, mathematic) plead for functional abstraction, and the arguments to work in a candy store, are just not valid, that is at least what my intentions were to prove.

I do not provide a solution in this essay, it was my intention to find legible arguments to support my assumptions. Conclusively the loss of abstraction is indeed sad because it did not led to anything, but interface-project developments by for instance Jef Raskin and his Archy seem to prove that there will be an opportunity to escape from the paper paradigm and its constraining desktop metaphor, somewhere in the nearby future...

Jorrit Sybesma

Rotterdam, December 2006

Notes

(1) “Reduce to the max” is a term used by design critic Max Bruinsma in the article ‘Proportional deconstruction’ on Swiss graphic design (2000), about deconstructing design to its essence (source: <http://www.xs4all.nl/~maxb/benzin/>).

(2) In the sense of being ‘visually communicative’.

(3) Susan Kare designed the original Macintosh icons in 1983, but was also involved in the development of interface aspects for NeXT and Windows 3.0 from Microsoft (source: <http://library.stanford.edu/mac/primary/interviews/kare/index.html>).

(4) After David Smith was working on his Ph.D. thesis in 1973 when he joined the Xerox Star project and implemented the icon, after receiving advice from Alan Kay. Based on information from the article ‘Of mice and menus: Designing the user-friendly interface’, 1989 by Tekla Perry and John Voelcker.

(5) “embody properties of what it represents”, from the article ‘Of mice and menus: Designing the user-friendly interface’, 1989 by Tekla Perry and John Voelcker.

(6) “is holy and is to be venerated”, from the article ‘Of mice and menus: Designing the user-friendly interface’, 1989 by Tekla Perry and John Voelcker.

(7) “contain all the properties of the programs and data represented, and therefore could be linked or acted on as if they were the real thing”, from the article ‘Of mice and menus: Designing the user-friendly interface’, 1989 by Tekla Perry and John Voelcker.

(8) “Paper paradigm” is “The paper paradigm consists of, usually, black text on a white background, files within folders, and a desktop” (source: http://en.wikipedia.org/wiki/Paper_paradigm)

(9) John Siracusa is a computer engineer and author of various articles amongst others for Ars Technica (source: <http://arstechnica.com/articles/paedia/finder.ars/>).

(10) Description of miniature paintings is mentioned in an article by Joshua Lee, ‘We could just start carrying around pictures of saints or something’ and in a wikipedia article on icons in operating systems (http://en.wikipedia.org/wiki/Icon_%28computing%29).

(11) “visual mnemonics”, are devices that help you remember associating the visual and the functional (source: http://en.wikipedia.org/wiki/Mnemonic#Visual_mnemonics)

(12) Fitts’ Law was applied in the factories of the Ford Motor Company in the late fifties, where production speed could be increased (source:).

The law was put in a drawer and was resurrected again in the computer-mouse-cursor age, during the eighties and nineties. When GUI design could possibly benefit from Fitts’s findings in the fifties. His experiment could be applied to nowadays computing, during usability-research to the effectiveness of Windows in comparison with MacOS. Practical appliances of the Fitts’ Law in modern day computing are very hard to find actually, the only application that is partly based on the ideas of Paul Fitts is the radial-menu-based-application Quicksilver for Mac OS X (source: “Fitts and Spiff”, Dutch article by graphic designer Finn Stapelkamp, published in the Macintosh-community magazine MacFan #55 and http://en.wikipedia.org/wiki/Fitts'_law)

(13) Windows GUI-environment and its effectiveness: “containing a document window, inside an application window”.

(14) “less is more” referring to the well-known by Ludwig Mies van der Rohe (adopted?) statement “less is more” and his minimalist philosophy.

(15) Icons are “visual mnemonics”, according to Ciam Saywe (Microsoft Windows XP Expert Zone Producer) in his article ‘Let’s Make a Theme: Customize Windows XP’, 2003.

(16) “as if you are dragging around portraits of saints” referring to an article by Joshua Lee, ‘We could just start carrying around pictures of saints or something’ and in a wikipedia article on icons in operating systems (http://en.wikipedia.org/wiki/Icon_%28computing%29).

(17) “smear” referring to description of very small unclear icons in the article ‘One Thousand Squar Pixels of Canvas’ from by Marcin Wichary, 2003 (source: <http://sediment.semifat.net/entry/2003/07/08-175509.html>).

(18) “Objects become more realistic without becoming more meaningful”, see the diagram of ‘the process of abstraction’ by Amsterdam-based German information designer Gerlinde Schuller, who was also a visiting tutor in 2005 at the Piet Zwart Institute, media design course (source: <http://www.gschuller.net>). See images on page 24.

(19) The research-project (at the St. Joost Academy in Breda) “Digital Terminology”(0) by Hans Gremmen and Monique Gofers is published in the design journal Items, #6 in 2001, BIS Publishers.

(20) Digital terminology examples: Some of those objects are so strong, that it would even be possible to use for instance the old abstracted wastebaskets of various OS’s - which is a digital metaphor for a physical one - to re-physicalised again, without loosing their strength or meaning. Another example is for instance the popularity of the disc-burning application Toast which gave for a certain group of computer-users an extra meaning to the physical object of a toaster. Besides hands and breads, also discs could be burned. See images of the project on page 23.

(21) Bill Buxton (William), Bill Buxton is a designer and a researcher concerned with human aspects of technology. Buxton’s research specialties include technologies, techniques and theories of input to computers, technology mediated human-human collaboration, and ubiquitous computing (source: <http://www.billbuxton.com/>).

(22) Actual function of the wastebasket: is objects are not deleted, when trashed, but recycled over time.

(23) See images added to this essay, page 20.

(24) “shaping the end user’s mental model of the system” referring to Bill Buxton in his article ‘Less is More (More ore Less)’, McGraw Hill, New York (USA) 2001.

(25) referring back to note 15.

(26) Sketch shows the old trashcan icons and the new ‘Aqua’-versions which replaces the old version. Added to the existing icons, the outcomes of the limited research exercise are placed. Most individuals mentioned that they were not able to draw their current trashcan, I asked them to draw the one they did remember.

(27) “Disneyfication” freely referring to the term used by Neal Stephenson in his text ‘In the Beginning was the Command Line’, Neal Stephenson, 1999 (source: <http://www.cryptonicon.com/beginning.html>). The term “Disneyfication” is a more or less common term to refer to specific overdone designs, part of the graphic design vocabulary.

- (28) Archy “is a proposed radically new system for interacting - on a (text)mode-basis - with many kinds of computers. Designed by human-computer interface expert Jef Raskin” (source: <http://en.wikipedia.org/wiki/Archy> and <http://www.raskincenter.org/>).
- (29) Microsoft Bob was “a failed attempt at introducing room-based GUI for beginners. Available in 1995. Developed by Microsoft” (source: <http://www.guidebookgallery.org/guis/bob>) and “Is Bob really a Kids-world”, PC Magazine 1995 by John C. Dvorak (source: <http://www.guidebookgallery.org/articles/isbobreallyakidsworld>).
- (30) Packard Bell Navigator was “a family-oriented GUI included with Packard Bell computers. Available from 1993 to 1996” (source: <http://www.guidebookgallery.org/guis/pbnav>).
- (31) forwarding to note 55.
- (32) forwarding to note 55.
- (33) “Squares” were used in ‘The Flex Machine’ by Alan Kay in 1967 to represent specific applications or functionality, from the article ‘Of mice and menus: Designing the user-friendly interface’, 1989 by Tekla Perry and John Voelcker.
- (34) “icons changing appearance depending on the properties of object they represent” referring through ‘One Thousand Squar Pixels of Canvas’ from by Marcin Wichary, 2003 (source: <http://sediment.semifat.net/entry/2003/07/08-175509.html>) to AskTog, “Apple Squandering the Advantage”, published on asktog.com of the ‘usability-brothers’ Nielsen Norman Group (source: <http://www.asktog.com/columns/035SquanAdv.html>).
- (35) Rules issued through the Human Interface Guidelines (source: <http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHIGuidelines/index.html>).
- (36) Otto Neurath (anti-philosopher) and Gerd Arntz (illustrator) developed the pictorial language Isop-type(37) to communicate educational information, based on a non-linguistic principle.
- (37) Isotype means International System of Typographic Picture Education.
- (38) The visual language was named Semantography of Blissymbolics, still used as a communication tool for people with a language-related handicap (source: “Words divide and pictograms unite” an article by Bas van Lier, published in the design journal ‘Items’ in April 2005, BIS Publishers).
- (39) Bureau Mijksenaar, Amsterdam and New York (see: <http://www.mijksenaar.com>), “wayfinding” is a non-existing English term for designing signs and pictograms.
- (40) The library is part of the UIC, the International Union of Railway Corporations, their commissions determine which icons will be part of the international visual language. More information can be found at: <http://www.uic.asso.fr/>
- (41) “MP3Faces” is a contraction of mp3-players and interfaces. Term is not widely spread or used, quoted from an article by Peter Mertens. Published in the Dutch Design journal ‘Items’, October 2000.
- (42) See images added to this essay, page 21.
- (43) Bling-bling is the term (hip hop slang) referring to a expensive lifestyle, which is “normally displayed through various forms of visual stimuli” (source: <http://en.wikipedia.org/wiki/Bling>). MTV broadcasted the

show 'Pimp my Ride', a car customization program, hosted by the technical incompetent rapper Xzibit (source: http://en.wikipedia.org/wiki/Pimp_my_Ride). See images on page 22.

(44) Neal Stephenson used a car-garage-metaphor in his text 'In the Beginning was the Command Line', see note 27.

(45) Bill Buxton used a car-metaphor in his text 'Less is More (More ore Less)', see note 24.

(46) 'Think different' has been the slogan for an international advertising campaign by Apple during the mid-nineties, to emphasize the differences between choosing for Apple or a - in their opinion - boring pc with Windows installed. Source: http://en.wikipedia.org/wiki/Think_Different

(47) "liquid, click it, you can lick it" were the words used by Steve Jobs during the introduction gala of the first edition of Mac OS X, back in 2001.

(48) Closed kernel policy: "developers no longer have the freedom to alter, rebuild, and replace the OS X kernel from source code" from: "Apple closes down OS X" article by Tom Yager of InfoWorld, published on 'Macworld.com' on May 17, 2006 (source: <http://www.macworld.co.uk/news/index.cfm?NewsID=14663&Page=1&pagePos=8>).

(49) Playing the popular Atari videogame "Breakout", was for Steve Wozniak the main motivation to develop new hardware for the Apple II (source: http://www.knuddle.be/index.php?page_name=gui&page=4).

(50) "Redmond start your copiers" is a slogan used by Apple to refer to the copycat behaviour of Microsoft, regarding the introduction of Windows Vista (and other products like Zune), which do show some striking similarities with the products Apple introduced earlier.

(51) As in good Microsoft tradition, they always seem to fail when it comes to imitating others, in this case Aero is a copy Apple's Aqua-interface. Aero is an acronym for Authentic, Energetic, Reflective and Open. (source: <http://windowsvistablog.com/blogs/windowsvista/archive/2006/11/09/the-sounds-of-windows-vista.aspx>). Vista comes in a few flavours with different graphical qualities: Aero, Standard, Basic and Classic. Known criticism: "reviewers have also noted similarities between Vista's Aero interface and that of Apple's Mac OS X operating system, particularly around the use of transition effects" (source: http://en.wikipedia.org/wiki/Windows_Vista).

'Interface Windows Vista keeps out software pirates' (Aero is connected to Windows Genuine Advantage), Dutch article by Remco Mourits, published at April 14, 2006 (source: <http://www.zdnet.nl/news.cfm?id=55639>).

Windows Vista pc's getting more expensive: Information based upon various sources, amongst others this has been published in the German newspaper 'Die Welt' on December 11th 2006 (source: <http://www.webwereld.nl/articles/44129/-vista-pc-kost-zeker-20-procent-meer-dan-xp-pc.html>).

(52) According to the Web 2.0 article in 'Wikipedia, The Free Encyclopaedia', although that part has been removed in the meantime (source: <http://en.wikipedia.org/wiki/Web2.0>).

(53) "Windows will be the OS on every Mac" John Dvorak wrote that prior to the introduction of Apple's Bootcamp(0). "Will Apple adopt Windows", article by John C. Dvorak, published on PCMAG.com on February 15, 2006 (source: <http://www.pcmag.com/article2/0,1895,1923151,00.asp>).

(54) "Usability in the Movies", article by usability-guru Jakob Nielsen, published on 'useit.com' on December the 18th, 2006 (<http://www.useit.com/alertbox/film-ui-bloopers.html>).

(55) "2D is better than 3D", article by usability-guru Jakob Nielsen, published on 'useit.com' on November the 15th, 1998 (source: <http://www.useit.com/alertbox/981115.html>).

(56) referring back to note 55.

(57) 'The Invisible Computer', (Why good products can fail, the personal computer is too complex and information appliances are the solution), Donald A. Norman. The MIT Press, Cambridge (USA) 1998/1999.

(58) Referring to a passage in the article "Apple Squandering the Advantage", published on asktog.com of the 'usability-brothers' Nielsen Norman Group (source: <http://www.asktog.com/columns/035SquanAdv.html>).

(59) Time Machine is a spatial system - that will be implemented in Mac OS X Leopard - in order to travel through the history of the filesystem, it is a heavy 3D-application (source: http://en.wikipedia.org/wiki/MacOS_X).

(60) "Icons are here to stay" from the article 'One Thousand Squar Pixels of Canvas' by Marcin Wichary (convinced by the use of icons in the future OSs), 2003.

(61) "Where do you want to go today" was part of the so called progressive image of the Microsoft corporation during the nineties, as a response to this campaign many people called or sent a letter or fax to the company with their answer to that question: "Microsoft asked me where I wanted to go today, so I told them: Egypt!". Microsoft dropped the campaign in 1996 (source: <http://www.denounce.com/mswhere.html>). The slogan also had a 'reverberating' effect, where did Microsoft wanted to go to tomorrow, did they really need the customer to point out their product-strategy? (source: <http://www.microsoftmonitor.com/archives/001768.html>)

(62) Front and back cover: photograph of a 'piece' at the Statentunnel, near the central station of Rotterdam, taken in 2004.

General resources:

'In the Beginning was the Command Line', Neal Stephenson, 1999 (source: <http://www.cryptonomicon.com/beginning.html>).

'The Anti-Mac Interface', Jakob Nielsen and Don Gentner, 1996 (source: <http://acm.org/cacm/AUG96/anti-mac.htm>)

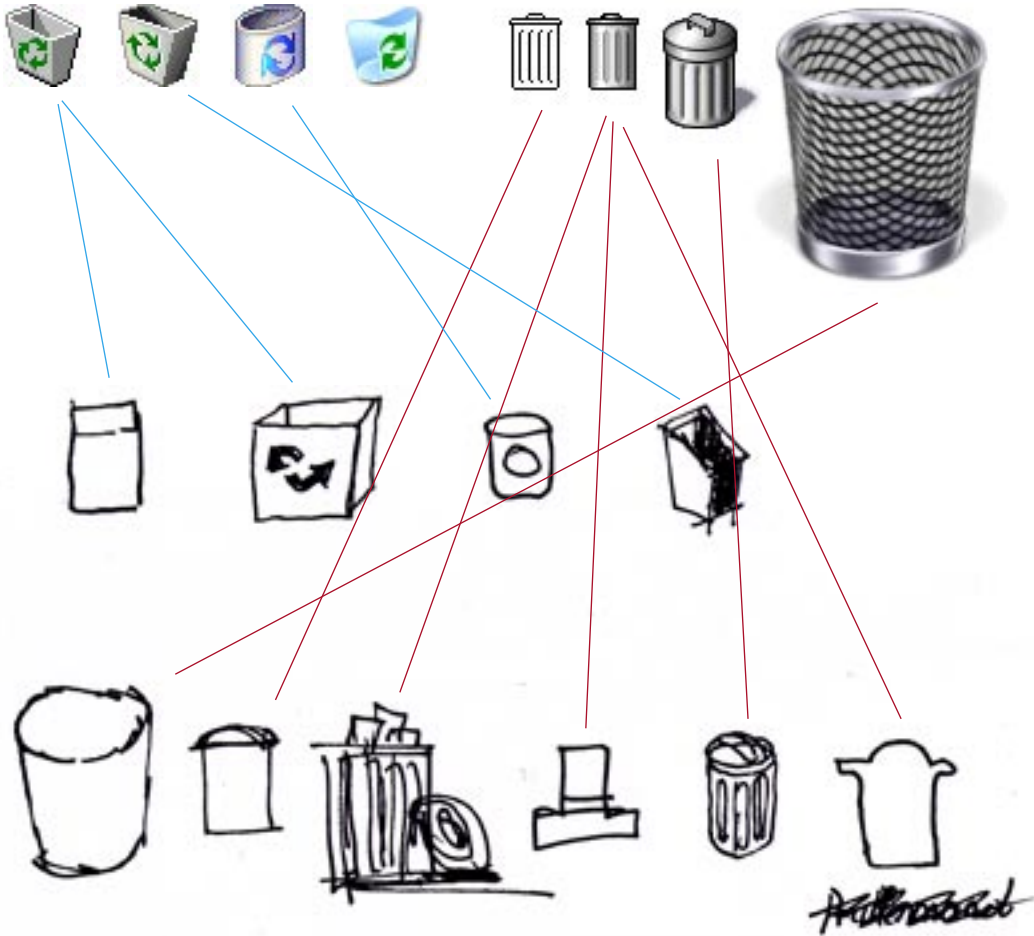
'Down with GUIs', Jef Raskin, 1993 (source: http://www.wired.com/wired/archive/1.06/1.6_guis.html)

'Less is More (More ore Less)', essay by Bill Buxton, McGraw Hill, New York (USA) 2001.

'The Invisible Computer', (Why good products can fail, the personal computer is too complex and information appliances are the solution), Donald A. Norman. The MIT Press, Cambridge (USA) 1998/1999.

Uses wikipedia-pages on topics:

ITunes, Bling, Pimp_my_Ride, Windows_Vista, MacOS_X, Icon, Susan_Kare, miniaturists, Symbol, Icoon_semiotiek, Icon_computing, Skin_computing, Mnemonic#Visual_mnemonics, Paper_paradigm, The_Mother_of_All_Demos, Desktop_environment, Desktop_metaphor, User_interface, Desktop_Window_Manager, Eye_candy, Web2.0, Windows_Aero, Windows_Vista, Fitts'_law.



IMPORTANT NOTE

This small research is (of course) not representative for a larger group of people. All the questioned were familiar with computers (although no experts), no artistic talents, all above 20 years of age (below 60), not forced to draw, and none of them was related to the Piet Zwart Insitute. After the drawing exercise the operating system versions were checked, all the Mac-users used Mac OS X.x (two of them still had Classic installed and worked with it). All Windows-users had a version of the latest XP edition installed.



00:00:00

VOLUME
TITLE
ILLROFT WINAMP3 VER3.10
QUALITY
K81PS KHZ

LORD OF THE RINGS

Eddie Grant - De... 00:46

Paused
Let It Be
The Beatles
96 kbps Let It B
19/31 2:12

FAZELONLINE.COM

2

Balance
SRS WOW Effects
Bass Boost
Trellis
WVW Effect

Cent D

000

VOL
-00:00
HALO 2

FM107.5
107.5
FM107.5
107.5

PLAYLIST





PHENOMENON OF ABSTRACTION

Top

Continuation of text



World
Environment
Technology
Energy
Economy
Education
Health
Culture
Politics
Science
Sports
Arts



Continuation of text



Continuation of text

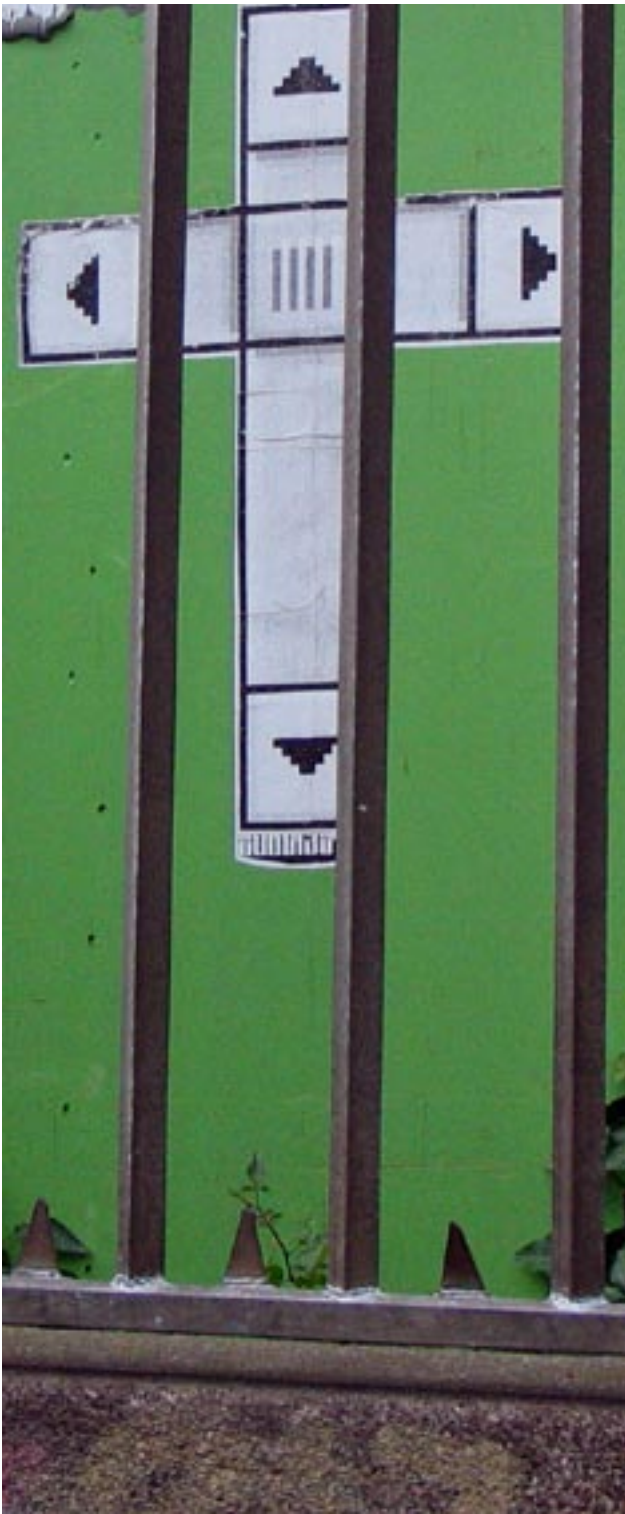


Continuation of text

Bottom

the loss of
visual language
in graphic design
interest

"Is the era of abstraction
over since the introduction
of hyper-reality in a
digital world, or has
the final hour for the icon
finally arrived?"



Piet Zwart Institute
MA Media Design

by Jorrit Sijbesma