Foundation for Distributed European Electronic Resource (DEER)

A Report on User Interface Design: Key Issues

Lily Díaz-Kommonen, Kari-Hans Kommonen, Markku Reunanen, Jyrki Rissanen
University of Art and Design Helsinki UIAH • Media Lab

ISBN 951-558-123-0
<table>
<thead>
<tr>
<th><strong>Project Number</strong></th>
<th>IST-2001-37491</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Title</strong></td>
<td>European Network of Centres of Excellence for Research and Education in Digital Culture (E-Culture Net)</td>
</tr>
<tr>
<td><strong>Document Type</strong></td>
<td>PU – Public usage of the result</td>
</tr>
<tr>
<td><strong>Document Number</strong></td>
<td>DEER deliverable 11c</td>
</tr>
<tr>
<td><strong>Contractual date of delivery</strong></td>
<td>06-17-2002</td>
</tr>
<tr>
<td><strong>Actual date of delivery</strong></td>
<td>06-17-2003</td>
</tr>
<tr>
<td><strong>Title of Document</strong></td>
<td>Key Issues in Interface Design</td>
</tr>
<tr>
<td><strong>Contributing work package</strong></td>
<td>WP6</td>
</tr>
<tr>
<td><strong>Nature of Document</strong></td>
<td>Report</td>
</tr>
<tr>
<td><strong>Contact person for E-Culture Net project</strong></td>
<td>Dr. Kim H. Veltman, Maastritch Mcluhan Institute Universiteit Maastricht Grote Gracht 82 6211 SZ Maastricht The Netherlands Phone: +31 43 3882699 Fax: +31 43 3252930 Mail to: <a href="mailto:k.veltman@mmi.unimaas.nl">k.veltman@mmi.unimaas.nl</a> <a href="http://www.eculturenet.org">http://www.eculturenet.org</a></td>
</tr>
</tbody>
</table>
INTRODUCTION ............................................................................................................... 4

BACKGROUND .............................................................................................................. 4
E-CULTURE.NET .............................................................................................................. 4

PART I: KEY ISSUES IN USER INTERFACE DESIGN ......................................................... 4

WHAT IS USER INTERFACE DESIGN? ........................................................................... 5
 The Activity ......................................................................................................................... 5
 The Object ......................................................................................................................... 6
 Command-line, Graphical, and Multimodal interfaces ..................................................... 8
 Community: User interface from the Software Developers’ Point of View ......................... 10
 The interface design process .............................................................................................. 11
 Community: User Interfaces and Participatory Design .................................................... 12

EVOlUTION OF USER INTERFACE DESIGN: A TIMELINE ........................................... 15

USER INTERFACE DESIGN: VISIONS ........................................................................... 25
 Augmentation ...................................................................................................................... 25
 Graphical User Interfaces .................................................................................................. 26
 Hyper-trails and Associative Linking .................................................................................. 27
 Cyberspace .......................................................................................................................... 27
 Gesture-based Multi-user Interaction ............................................................................... 28
 User’s Rights ....................................................................................................................... 28

FUTURE USER INTERFACES .......................................................................................... 30
 The Present State of User Interfaces is Stagnated ............................................................ 30
 The main problems in interaction are not at the interface ............................................... 30
 Personalized interfaces ...................................................................................................... 32
 Shared ontologies ............................................................................................................... 33
 Shared interaction languages ............................................................................................. 33
 Future digital cultural heritage objects .............................................................................. 33

GLOSSARY OF TERMS AND CONCEPTS ...................................................................... 35

PART II: NORDIC NETWORK IN DIGITAL CULTURE HERITAGE ..................................... 38

WHO ARE WE? .................................................................................................................. 38

ACTIVITIES IN E-CULTURE.NET .................................................................................. 39
 Logogram and Web Design ................................................................................................. 39
 Publications in the Area of Digital Cultural Heritage ......................................................... 39
 One-day Expert’s Meeting in Helsinki ................................................................................ 39
 Customizable web site as tool for collaborative work ......................................................... 40
 Expert’s Presentations ......................................................................................................... 40
 Scenarios ............................................................................................................................. 40
 Scenario 1: Distributed 3D in Cultural Heritage Seminar 2005 .......................................... 42
 Scenario 2: Wand and Footprints ....................................................................................... 44
Introduction

Background

This report is the final deliverable related to the work done on interface design for the E-Culture Net Thematic Network, a project funded by the IST program of the European Commission.

In order to establish the context for this report, we summarize in a few paragraphs the overall objectives of E-Culture Net.

In Part I, we provide a written overview of the current state of the field of Human Computer Interface Design, with specific reference to projects and milestones in the cultural heritage sector. The objectives are to highlight the trans-disciplinary nature of user interface design, to provide information that might facilitate the future projects involving user interface design, as well as to stress the relevance that the cultural heritage sector has had in the development of computer applications as a whole.

In Part II, we provide description of the activities performed by the University of Art and Design Helsinki, Media Lab, in the area of interface design for the E-Culture Net Thematic network. These activities included graphic design activities, production and documentation of research about design, as well as the organization and production of a one-day Expert’s Meeting in Helsinki. Among the results of work done during and as part of this meeting are: A tool for collaborative work in the form of a website, two fully documented scenarios outlining potential use of Distributed European Electronic Resource (DEER). (See: http://sysrep.uiah.fi/e_culture)

E-Culture Net

The project is a thematic network funded through the IST Fifth Framework. The objective of the network is to “identify/define research topics in digital culture; how teaching can focus those topics; how broadband pilots can generate new research topics among partners and lay a foundation for a Distributed European Electronic Resource (DEER) for research and higher education.”

In the context of the network, the current report focuses on issues pertaining Human Computer Interface design and development as it applies to DEER, and to further development of broadband technology.

Part I: Key Issues in User Interface Design
What is User Interface Design?

The Activity

Human activity can be seen as “conducted through actions, which take place in a unity of time and space with specific intention.” Activities themselves can be seen as bound to a goal or to an object: While a work of art can be made with the intention of earning a living, it can also be created for the purpose of making the work of art.

Computer applications are tools that can be used to enhance, and even to replace, operations in human activities. This is a fact that provides affordances as well as places constraints on the design of computer applications. For example, the 3D reconstruction of Trollberg site in Denmark created with computer graphics and available to the students through their Cultural Heritage Seminar, can only be seen indirectly through the representations projected in studio panoramas. In many cases, the object does not exist itself as something separate from the computer artifact. This is the case with the dynamically created objects such as the 3D gallery that is created by the visitor in the Raisio Archaeology Archive.

These conditions impact on interface design insofar as they reveal how the nature of the task of design of user interface is not about the creation of gadgets but rather, about the crafting of an experience. Experience design, has indeed been recognized as a new area of inquiry: “Experience Design as a discipline is also so new that its very definition is in flux. Many see it only as a field for digital media, while others view it in broad-brush terms that encompass traditional, established, and other such diverse disciplines as theater, graphic design, storytelling, exhibit design, theme-park design, online design, game design, interior design, architecture, and so forth.” See Experience Design by Nathan Shedroff at (http://www.nathan.com/ed/index.html)

Computer applications are complex tools and their design requires the cooperation of specialists. Interface design is an inherently trans-disciplinary activity. What this means is that successful development and implementation of interfaces in computing applications requires the concerted efforts of professionals from diverse fields of knowledge. This might be due to the fact that, as has been noted, human computer interfaces include a “mixture of function and form,” [and]“are the physical display of information, aesthetic, and persuasive content affording the means for interacting with that content.”

Among the disciplines that usually partake of the task of interface design we will find anthropology and the social sciences, cognitive science, design, and software engineering. Overall, this collaboration among disciplines has had a beneficial influence in the field of user interface design and in the development of computer application design in general. One of the main consequences of this involvement has been a shift and increased attention to foreground of the importance of human factors and experience.
The work of Alan Kay and Donald Norman that is cited later in this report has had substantial impact in this respect.

The Object

The term ‘interface’ is used more generally to denote a defined boundary between different layers or modules of a software system. Systems are usually divided into modules that are designed to be fairly independent of each other, so that their internal details can be hidden from other parts of the system, so that the various parts can evolve without always requiring changes in all the modules that interact with them. The design of the interface between modules is therefore very important, because the interface is what other parts of the system will know about each module, and it is the way how its features become usable to the other parts: a component can interact with another component through messages or calls that adhere to the interface definition of a component and use the interaction language understood by it.

The user interface is a very analogous concept, deriving directly from the idea of the interface in the computer system design discourse, with both its useful as well as problematic connotations. Used in this context, the term indicates the set of operating system commands, graphical display formats, and other features designed for use on a computer or a program. The user interface has also been defined as “the means by which computers and users communicate.” Through the user interface, a computer-based artifact supports the intentional and operational aspects of a computer application. In summary, in a computer system, the user interface includes the logical organization and behavior of information, the visual presentation of the data and functions, as well as the user’s interactions with a computer program.

User interfaces are created using conceptual as well as tangible components. The physical components of user interface, for example, include the myriad of input and output devices that support user interaction with the system.

Among the conceptual components of human computer interfaces are: Metaphors, mental models, navigation schemas, appearance, and interaction.

**Metaphors** figure prominently in user interface design. In this area of study, metaphors refer to essential similarities that can be communicated visually through words and images, or through acoustic and tactile means. Designers conveniently use metaphors to represent or denote people, objects, structures, and processes. An example of this is the dropping of a file in the trash bin as indicative of deleting a file and its contents. Metaphors can also be used to describe structures and processes, such as is the case when using an outline to describe a file hierarchy.

In user interfaces, metaphors can mimic the real world, or create a whole set of opportunities. **Real metaphors** are those that are directly based on objects of the real world. For example, the use of a library shelf to represent a set of multimedia education resources to be accessed for self-training by employees in given company.
Abstract metaphors, can use reality as a starting point, but do not try to look exactly like the real thing. An example of an abstract metaphor is that one utilized by Char Davies for the interface of *Ephemere*, one of the first large-scale virtual reality artworks featuring immersion. In this work, Davies used the metaphor of diving coupled with a hardware to reproduce the experience of diving into an information landscape: “Participants immersed in Char Davies’s all-enveloping virtual universe discover that their interaction with the computerized system is conditioned by their breathing and balance, so that they actually move through into the rhythm of their breath.”

Another example of an abstract interface is can be seen in Jeffrey Shaw’s *The Legible City*, an interactive installation in which the visitor “is able to ride a stationary bicycle through a simulated representation of a city that is constituted by computer-generated three-dimensional letters that form words and sentences along the sides of the streets. … The handlebar and pedals of the interface bicycle give the viewer interactive control over direction and speed of travel. The physical effort of cycling in the real world is gratuitously transposed into the virtual environment, affirming a conjunction of the active body in the virtual domain. A video projector is used to project the computer-generated image onto a large screen. Another small monitor screen in front of the bicycle shows a simple ground plan of each city, with an indicator showing the momentary position of the cyclist.” (http://www.jeffrey-shaw.net/html_main/frameset-works.php3)

Generative metaphors, create “new perceptions, explanations and inventions.” They provide opportunities to become aware, or to see things in a different and new way. Consequently, they can be a source for ideas on how to design.

Other examples of metaphors used in user interfaces are: Graphical User Interfaces (GUIs) that make use of the desktop to organize representations of peripheral equipment as icons, pen-based interfaces that make use of physical gestures, such as the circling of an item to indicate selection, virtual reality user interfaces that make use of marks and gestures associated with flying, such as yaw, pitch, and roll, web-based interfaces that make use of concepts such as chatting to indicate real-time communications between diverse users.

Some advantages in using metaphors in user interface design are: reduction of time needed to learn how to a the computer application, decrease in tension and stress due to the presence of familiar concepts in an unfamiliar environment, increase in learning, memorization, and use. Disadvantages of metaphors are that metaphors vary across cultures and that they change through time. Also, metaphors can get in the way of using the full potential of the virtual, immaterial and dynamic software environment by tying the system to the construct of the source of the metaphor.

A mental model refers to the organization of data, functions, tasks, and roles built into the system. The mental models of a user interface should be anchored in the activities, actors, and context in which the system use unfolds. Don Norman has noted how there is a gap that exists between the model in the designer’s mind and a final product that can be successfully employed by people. This gap results from the fact that designers do not really communicate directly with the user but rather, through the system image. That is, there is a difference between the system that is the outcome of
the designer’s effort and the system that emerges as a result of the user’s interaction with an artifact.\textsuperscript{14}

**Navigation schema** refers to the movement through a mental model or a system afforded by windows, menus, dialogue areas, control panels, etc. **Appearance** refers to the verbal, visual, acoustic, and tactile characteristics displayed by the interface. **Interaction** indicates the means by which users input changes to the system and the feedback supplied by the system.\textsuperscript{15}

### Command-line, Graphical, and Multimodal interfaces

As opposed to a command-line user interface in which the user enters input and receives feedback via the use of command-line statements, a graphical user interface (GUI) provides the user with a picture-oriented user-friendly way to see what is on a computer system.

It is also useful to note the term Application Programming Interface (API), which usually refers to the interface the system offers to the outside world as the way to programmatically use its services. The idea of a programmatic interface is important to take into account in the discussion of user interfaces, because the user can also be represented by a program, and in that case the user’s abilities of interacting with the system are defined by the API the system offers.

Aside from the already well deployed command-line and graphical user interfaces, there are multimodal user interfaces, a term that is now used to refer to those interfaces that take direct input from one of the users sensory modalities, such as speech.

There are many different definitions of multimodality that are mostly overlapping but also somewhat conflicting. Multimodality refers to the ability for end user to choose their preferred interaction mode. Interaction can be carried out for instance by using voice and visual communication, haptic interfaces and perhaps some day even olfactory sensors and smell generators... With multimodal applications users can choose whether they are most comfortable with listening to information, viewing text or pictures on screen, responding via touch-tone, voice-control or key-click navigation. The most important property of multimodal systems is that they allow users to work with multiple modes of interaction at once. For example, a user can point a map and use spoken language to ask from the system how can he travel to the place he has pointed.

Multimodal systems are capable of expanding the accessibility of computing for diverse and non-specialist users. As interaction will be more and more human-to-human, the need for actually learning how to use services will get lower and usage will be more intuitive. Multimodal systems can also provide new methods for working with computer for all kinds of designers and artists as interaction with computer don’t have to be limited to traditional input devices like keyboard and...
mouse which are very unsuitable or at least unnatural for many tasks. Multimodal systems can also provide very useful aids for physically-challenged people for their everyday life.

The development of multimodal interfaces is the subject of intense research. Psychologists, for instance, have been studying for a long time how people interact with each other and how do we mix different modalities of interaction. Still our knowledge is very limited. Also the actual implementation of multimodal systems is often much more challenging than what one might imagine. The reason for this is only partly technical but one major issue to notice is the fact that people tend to interact with computer quite differently compared to human-human communication.
Community: User interface from the Software Developers’ Point of View

Different motives support the integration of usability evaluation and interface design to the software development process: good usability sells software, helps to avoid costly errors, increases user satisfaction and reduces the cost of user education. The need for good usability and interfaces was not widely recognized until the 1990’s. Many of the relevant theories including cognitive psychology and user interface evaluation (for example the GOMS model) are older than that but the awakening of large scale commercial and scientific interest in user interface design was what really started the usability revolution. This was a movement supported by the rise of Internet, mobile phones and related technologies: a large number of new users was going to use the new services thus creating a need for better usability.

It is often convenient to think about interface design as having a threefold division: the user, the interface and the system. This division, introduced by Herbert Simon, helps to explain and visualize the concepts related to the use of a system. However, in an ideal situation this separation between the system and the interface is much less evident. What's more, in the real applications the interface might not be at all separable from the content -- in fact it might be the only purpose and content of the system. The separation of the three concepts can be used as a valuable tool when illustrating system components but not as the guideline of interface design.

A traditional approach to the creation of interfaces is to build the software first and only in the final stages add the (graphical) user interface. While this approach might be good from the software developers' point of view it is unlikely to result in an optimal user experience. It's much like the Simon's model taken to extremes without real understanding of its meaning. The modern user interface paradigms such as The Delta Method (See: http://www.deltamethod.net/) and Contextual design oppose this approach and suggest that the end user should be taken in account from the very beginning of the development in order to produce the most effective interfaces. A good user interface is not something that is glued on top of a system but more like a framework that holds the whole system together.

In the application programmer’s level the introduction of graphical user interfaces (GUIs) has changed a lot. The nature of command line and menu-based software is mostly linear and the user's actions can be well predicted. In the age of graphical user interfaces the complexity has grown significantly, thus increasing the demands and problems that the programmer will face. The applications are no longer linear but event-driven instead: the user’s actions such as mouse clicks and key presses create events. The application must interpret these unpredictable events and react to them in a consistent manner.

The design of such interaction and a usable graphical interface is a challenging task and requires understanding of both user interface design matters and programming. The traditional approach of adding the user interface to the program at the final phases makes the process one-directional. The user feedback and evaluation come in too late and the cost of revising the system can be high at this stage. To be effective the design
should be a two-way process where the content shapes the interface and the user requirements affect the underlying system. This kind of ideal process would require that either the programmer is aware of the user interface issues and has the possibility to get user feedback or works in conjunction with special usability experts. This ideal is supported by the aforementioned modern user interface design paradigms as well.

Apart from the high-level usability paradigms there are also more practical guidelines for developers. These specifications are called style guides. A great benefit of style guides is that applications following the same design rules have a consistent look and feel. A well-written style guide serves as learning material as well and helps to avoid design errors. Probably the most influential and the first large-scale interface style guide was Apple's “Human Interface Guidelines” for Macintosh developers.

The interface design process

The process described here is a coarse outline of an ideal design process. In practice it's possible to go through such a process only in relatively large projects, since funding and time are likely to pose severe limitations for small-scale projects. Neither should the process be thought as a linear one, since the different phases always overlap more or less. Also the unpredictable nature of software projects can cause backtracking or skipping some phases—for example customer requirements, timetables and application platforms are all prone to change.

The first question that needs to be answered is: who are the actual users of the interface? Fortunately there are effective methods for approaching this problem. One approach is to do background research based on available statistics and through interviews. Also, when making commercial products, it is also beneficial to study a competitors' products and their use. Research on culture-specific practices is a key concern if the interface is designed for international use.

The user is not just an object that has specific attributes. The user has a goal or a task she wants to complete. It is crucial to discover and analyze these tasks so that the system produced is useful as well. A good interface is not enough if it doesn't serve a real purpose. A practical way of discovering the real needs of the users is to study their behavior and work habits in the actual environment. Another common method is the use of scenarios (meaning imaginary use cases) to check whether the planned features and the interface really meet the needs.

Paper prototypes are a quick way of testing interface concepts and application logic before any software is actually built. Many severe design errors can already be caught at this phase. The later the problems are found the higher the cost of fixing them. In an ideal design process the users are involved throughout the entire project, which means regular testing. Nothing can replace real usability tests, but an evaluation done by HCI experts in some phase can iron out serious mistakes in a fast and cost-efficient manner.
The actual work of building an interface is closely tied to the underlying application logic. This means that interface designers mostly work in conjunction with a group of programmers. This relation works best if both sides share the same terminology: the programmer should know a little about design terminology and vice versa. On a larger scale this same problem is reflected on the whole process. User interface design involves people from many disciplines, which may easily lead to communication problems. Also, the users also have their own area of expertise that has a language of its own.

Once the interface is completed and tested, the last phase begins. However good the testing, some problems always slip through and they are found only in long-term use. Collecting user feedback is essential in an ideal process for several reasons. First of all, the remaining problems can be discovered and fixed to achieve a better user experience. The good and bad decisions done in the design process become evident, which give valuable experience to the persons involved. Finally, the users like to get their voice heard instead.

**Community: User Interfaces and Participatory Design**

The convergence of diverse technologies into broadband makes for a complex design environment. For one, designers will have to deal with multiple and quite different technology formats that should be brought together into a coherent information landscape, that is simultaneously adaptive, while at the same time allowing and even supporting diversity.

Then there is the social and economical context—the struggle between the different stakeholders—in which the development of broadband is situated and which has been cited as one of the main factors for the inability of this vision to coalesce. A recent article published in The Washington Post on “Who is Holding Back Broadband? cites the efforts of major institutions who hold the copyright for contents as being responsible for the atmosphere of inertia that seems to pervade the broadband initiative. (http://www.washingtonpost.com/wp-dyn/articles/A11361-2002Jan7.html)

Considering the complexity of the topic, it may be necessary examine the different facets of broadband but, more important, might be to try to obtain a meta-view of broadband deployment in different environments and among diverse communities and to try to understand their knowledge production and consumption habits and practices:

“Broadband is a means to multiple, diverse ends encompassing family, work, and society in generally. In addition to enabling entertainment and e-commerce applications, broadband can enrich the Internet exploitation as a public space, making electronic government, education, and health care applications richer and more compelling and useful, and it can provide new modalities of communication, notably within communities or families. Broadband commands attention because it enables dramatically different patterns of use that offer the potential for significant changes in lifestyle and business.” 18
Any interface design approach will have to deal with the multidimensional nature of broadband, and depending on the context, broadband can signify different things:

- Diverse communication modalities
- Diverse technology formats
- Diverse potential user communities.

Participatory design approaches that bring designers, developers and communities of interest (CoI) together, from the beginning, in a co-design process can be a way in which to approach the broadband issue. In a community of interest, individuals can share knowledge, discuss issues and solve problems.

![Participatory design approach](https://example.com/participatory-design.png)

**Figure 1:** A representation of participatory design approach with communities of interest.

Indeed, communities of interest in which groups of stakeholders come together as a result of a shared interests in a topic or an activity is a concept that is already being implemented in research and development projects of computer applications. In the CIPHER project, for example, one of the objectives is to create Culture Heritage Forums “that empower communities of interest to explore, research and build content.” Ultimately, the objective is one of enabling self-sustainability of the cultural heritage resources created through research and development.

Smart communities is yet another type of community that might be of help in facilitating the process of design for broadband:

Smart communities are where leaders and stakeholders have formed alliances and partnerships to develop innovative ways to extract new economic and social value
from electronic networks and the public Internet. See the World Foundation for Smart communities:
http://www.smartcommunities.org/

The notion of Smart Community is already being implemented in countries such as Canada in the effort to bring cultural heritage and information technology together. See Canada’s Smart Communities program: (http://smartcommunities.ic.gc.ca/)
Evolution of User Interface Design: A Timeline

In the following pages we present a timeline of noteworthy ideas and inventions that have contributed to the development of user interface design.

1939: Speech synthesis

The first speech synthesis device was built by Homer W. Dudley and was called “Voder”. The device was built out of electronic components and it was possible to type in the words to be spoken with a special interface. In 1940 Dudley introduced his other speech related project, “The Vocoder” which was the first step in electronic speech recognition.

http://americanhistory.si.edu/scienceservice/037001.htm

1945: “As We May Think”

“As We May Think,” a scenario written by Vannevar Bush. The scenario fully describes the interaction between a human and a computer system. Interface components, in the manner of input and output devices, such as a screen and pointing device, are described. Clear references are made to further development of Vocoder as speech synthesis device. An online version of the article published in the Atlantic Monthly can be found at:


1950: First graphic images

The first graphic images generated by an electronic machine were created by mathematician/artist Ben Lapowsky.

1951: The Sentinel

The movie “2001: A Space Odyssey” (1968) directed by Stanley Kubrick was based on this short story by Arthur C. Clarke. Clarke also wrote a book with the same name. The computer and interaction seen in 2001: A Space Odyssey shaped the research of artificial intelligence and speech interfaces a lot during the following years.

1954: Light Pen

The light pen as input device to electronic machine was produced as part of the SAGE (Semi-Automatic Ground Environment) radar system. It was used to select a specific blip on the radar screen for tracking. See also online version of Tools for
1956: Sensorama

Sensorama, created by Morton Heilig, was a simulator ahead of its time. It featured a simulation of a motorcycle ride through Manhattan. What made Sensorama revolutionary was the fact that it had a 3D visual, stereo sound, wind effects, vibration and even smells. Despite its unique realism Sensorama was never a commercial success.24

1961: Head-mounted display

The concept of a head-mounted display (HMD) was already invented in 1956 but the first ones to construct one in 1961 were Comeau and Bryan, employees of Philco Corporation. Headsight had one single CRT element and a magnetic tracking system to determine the direction of the user's head. Headsight was used for telepresence in dangerous situations but not yet with computers.25

1962: Augmentation

“Augmenting Human Intellect: A Conceptual Framework,” a paper written by Douglas Englebart that set the agenda for the next 30 years. “Engelbart sought to define and implement the functionality necessary for computers to augment human abilities. The functions he thought necessary included links between texts, electronic mail, document libraries, as well as separate private space on computers for users’ personal files, computer screens with multiple windows, and the facilitation of work done in collaboration by more than one person.”26

http://sloan.stanford.edu/mousesite/EngelbartPapers/B5_F18_ConceptFrameworkInd.htm
http://sloan.stanford.edu/mousesite/Archive/ResearchCenter1968/ResearchCenter1968.html

1963: Sketchpad

Sketchpad, the first truly interactive computer graphics system is invented by Ivan Sutherland at MIT. “The user of Sketchpad, the precursor of all modern interactive computer graphics systems was able to draw with a light pen on the computer screen and see the results almost immediately.”27 Used click-move-click method of interaction, thus linking the hand and the eye of the user with the display screen. See an exchange of letters by pioneers Jeff Raskin and Bruce Horn about the influence of Xerox PARC ALTO system on Lisa and Macintosh: (http://www.mackido.com/Interface/ui_raskin.html). See also: http://dir.yahoo.com/Science/Computer_Science/User_Interface/
1964: The tablet

The tablet as input device is developed by a group working, with ARPA funding, at the Rand Corporation.  

1965: The mouse

The mouse as input device is invented by Douglas Engelbart. The first reference appears in a report titled “Computer Aided Display Control”, Contract NAS-I-3988, report NTIS N66-30204). “For Engelbart, the mouse, like the chord keyset, was part of an effort to optimize basic human capabilities in synergy with ergonomically and cognitively more efficiently designed artifacts.” See the MouseSite website for documentation and videoclips of the first public demonstration of the mouse: (http://sloan.stanford.edu/mousesite/1968Demo.html)

1965: Hypertext

Theodore Holm Nelson invented the term hypertext and presented it to the world. Hypertext has become a frequently used interaction metaphor in our computer environment, specially in the World Wide Web.

1965: Ultimate display

The Ultimate display was a big leap in the development of head-mounted displays. Ivan Sutherland, a scientist and researcher who has been influential in the development of computer graphics and virtual reality, created the device. The Ultimate display was already stereoscopic, meaning than you had a different image for each eye which enabled true 3D graphics. The first version had a mechanical tracking device attached to it.

1967: Haptic feedback

The first electronic devices featuring haptic feedback (also known as force feedback) were the Grope series. Starting in 1967 Fred Brooks and his research group built several different models, including a probe and a table giving haptic feedback.

1968: Hypertext Editing Systems

Hypertext Editing System jointly designed Andrew Van Dam, Theodore Nelson, and two students at Brown University with funding from IBM. The system was one of the first generation hypertext systems.
1968: The Demo

The Demo, presented in 1968, was a demonstration of the work done by Douglas Engelbart and his research group in Stanford Research Institute. The group had been working on the concepts since 1962. During The Demo several new interaction methods and devices were introduced for the first time. The most important ones were mouse, windows (not yet overlapping), hierarchical hypertext and multi-user interaction. Engelbart's work set the direction for user interface development and his influence can still be recognized on the paradigms and systems of today.


1969: Unix

The creation of Unix (originally called Unics) by Ken Thompson and Dennis M. Ritchie at the Bell Labs set the standard for text-based user interfaces for years to come. The operating system was first given free to universities and later various commercial versions started to emerge. Originally Unix was used through dumb character-only terminals, which shaped the user interface significantly. The modern Unices and clones like Mac OS X and Linux do have a graphical user interface, but the traditional text-based metaphor can still be accessed.

http://www.opengroup.org/

1970: Software Information Technology: Its New Meaning for Art

The first public demonstration of a hypertext system, Labyrinth, by Nelson Goodman and Theodore H. Nelson, took place at this exhibition held at the Jewish Art Museum in New York. The exhibition served as a publicity vehicle that helped to introduce the public to a host of personalities from the emerging art and technology scene. Alan Kaprow, Nicholas Negroponte,

1971: Touchpad

Dr. Sam Hurst invented the first touchpad, “Elograph,” while working at the University of Kentucky. Nowadays the touchpad has become the standard mouse replacement in almost all the portable computers. Three years later, in 1974, the concept of touchpad was developed further and the first touch screen was created.

http://www.elotouch.com/company/history.asp

1972: GUI

The Learning Research Group was formed at the newly founded Xerox PARC (Palo Alto Research Center). Led by Alan Kay, the group worked on crucial aspects of human-computer interactivity, such as the graphical user interface (GUI). “Kay introduced the idea of iconic, graphical representations of computing functions – the
folders, menus, and overlapping windows found on the desktop – based on his research into the intuitive processes of learning and creativity. Kay came to understand—as he put it—that doing with images, makes symbols. This was the premise behind the GUI, which enabled viewers to formulate ideas in real-time by manipulating icons on the computer screen.”

For more information about Xerox PARC and its interdisciplinary research projects see: [http://www.parc.com/]

1973: ALTO

ALTO system created at Xerox PARC. It had overlapping windows, two menus per window (one for cut-copy-paste), a multi-button mouse (with different buttons for doing different things), and a chorded keyboard for doing entry. “This was the first manufactured machine to use a mouse, and the Ethernet protocol was established to network the machines within Xerox. The main processing unit, the size of a small refrigerator, was designed to sit under a desk or table, while the display, mouse, and keyboard sat on the tabletop. The display was shaped like a piece of letter-sized paper and in conjunction with the mouse employed a graphical user interface (GUI). This was perhaps the most farsighted feature of the machine, as it took another 20 years for GUI's to become standard on desktop computers.”

[http://www.cedmagic.com/history/xerox-alto.html]

1977: Dataglove

The Sayre glove was invented by artist Dan Sandin, Richard Sayre and Thomas Defanti at the University of Illinois. The glove used a bend-sensing technique unlike modern datagloves that are mostly based on optical fibers. The first optical glove by Thomas Zimmerman followed in 1982. So far datagloves have remained expensive and unavailable to ordinary users apart from one exception: Mattel Powerglove from 1989, which was a low-cost device used with Nintendo entertainment system and sold to the consumer market.

1974: Altair

Altair, the first really successful home-kit computer is shipped. This machine did NOT have any input/output devices such as keyboard or monitor - just switches and LED's. It marks, however, the beginning of personal computers.

1979: Interactive multimedia

The Aspen Movie Map project is realized at the Center for Advanced Visual Studies at MIT. The Aspen Movie, in which the viewer is literally transported to another place, is one of the first examples of interactive multimedia. For a live demonstration...
of the Aspen Movie and more information about Michael Naimark see: (http://www.artmuseum.net/w2vr/timeline/Naimark.html#NaimarkText)

1979: Affordance theory

Perception theorist J. J. Gibson published his affordance theory in 1979. The radical ideology of affordance theory is that we directly perceive affordances (eg. possibilities for action) from our environment. This point of view has influenced and advanced user interface design a lot during the following years, even though the theory itself is not widely accepted as a whole.33

1979: Visicalc
The first spreadsheet application. The representation on the screen used a simple columnar spreadsheet (as had been done for years on paper) and featured embedded formulas that could be manipulated by user.

1983: Videoplace

Myron Krueger, a visionary and a researcher, created a virtual environment called Videoplace at the University of Connecticut. The main concept of Videoplace was the digital groupwork of two users who stood in front of screens and performed tasks together. The users' gestures were recognized from video input and let them do some basic interaction. The rise of video conferencing and web camera based interaction prove the relevance of Krueger's work even today.34

1983: GOMS

An early and influential user interface design model introduced by Card, Moran and Newell in 1983. GOMS stands for Goals, Operators, Methods and Selection rules. Originally the model was used for text editing tasks, but later was found usable in several other human computer interaction areas as well.35

http://tip.psychology.org/card.html

1983-84: Macintosh and Lisa

In 1983, the Lisa computer was introduced. The Lisa already featured many of the innovations that later made the Macintosh famous. Among the user interface inventions that belong to Mac and that we now take for granted are: Finder, drag and drop file manipulation, direct manipulation and editing of document, types and creators for files, disk and application names, clipboard, multiple views of the file system, desk accessories and control panels, pull down menus, one button mouse and click and drag methods, and self-repairing windows. One person that has done significant work in the development of graphical user interfaces is Susan Kare, the designer of the original Macintosh icons. Her goal was to help people to operate computers that were seen as cold and intimidating. Later on she also designed icons
for such systems as Windows 3.0 and IBM OS/2 Warp. The well-known Windows Solitaire graphics are designed by her too.

1984: Neuromancer

*Neuromancer*, a novel written by William Gibson, has greatly influenced the development and terminology of virtual reality. The book is a sci-fi story about the fight between governments, companies and individuals in the world of virtual reality. Words like “cyberspace” and “matrix” used by Gibson have established their role in modern computer terminology. *Neuromancer* is the first part of a trilogy and the following novels were *Mona Lisa Overdrive* and *Count Zero.*

1985: VIEW

VIEW, Virtual Environment Workstation Project headed by Scott Fisher at NASA’s Ames Research Center. The objective of the project was to develop a multisensory ‘virtual environment’ workstation for use in Space Station tele-operation, tele-presence and automation activities. The VIEW Project pioneered the development of many key VR technologies including head-coupled displays, data-gloves, and 3-D audio technology. For more information as well as a video demonstration see ArtMuseum Net website sponsored by Intel Corporation: (http://www.artmuseum.net/w2vr/archives/Fisher/04a_VIEW.html#VIEWText)

1985: Windows 1.0

The first version of Microsoft Windows was released in 1985 as a graphical interface for MS-DOS operating system. The interface concepts are heavily borrowed from the user interface of Macintosh. The first versions are not commercially successful but Windows 3.0 finally brings the graphical user interface to the PC using big audience in 1990.

http://members.fortunecity.com/pcmuseum/windows.htm

1985: UTOPIA, Scandinavian Collective Systems Design, participatory design

“The UTOPIA project was a Nordic cooperation that specifically concerned newspaper production and new page-make up and image processing based on the emerging workstation display technology. The technology was developed in close cooperation between graphic workers, their trade unions and systems designers (computer scientists as well as product designers.) … To achieve these goals a number of participative methods and techniques like extensive use of mock-ups and early prototypes in combination with design games were applied in novel participatory ways forming the Scandinavian collective systems design approach”
“So the impact of UTOPIA is continuing to expand, and the idea that workers and their unions have an important role in the design of new technology is reaching a wider and wider audience. Today, Scandinavia, tomorrow perhaps the rest of the world.”

1986: X Window System

Started with the name “Project Athena” at MIT. In 1986 the name was changed to X Window System. It is the standard graphical interface for Unix workstations. A very fundamental feature of X is the network transparency, which lets the users run programs over a network from other computers, regardless of the hardware used. X itself does not offer user interface elements such as buttons or menus, but they are built on top of it as a separate layer instead. http://www.x.org/

1987: Hypercard

Hypercard is introduced on the Macintosh. Based on the concept of hypertext, hypercard is the first high-level, visual programming tool to allow non-programmers to build applications. “The software's power lies in its ease of use: Information is stored in a series of cards arranged into stacks. Cards can be linked to each other, just like hypertext links on the Web. A built-in, plain-English programming language, HyperTalk, executes commands. The result is both simple and powerful. Fifth-graders get to grips with HyperCard in minutes, building databases of their Pokemon cards. Cyan, the game publisher, used it to create fiendishly complex games like Myst and Riven. It allowed ordinary people, such as Jacqueline Landman Gay—who had hardly touched a computer before—to start a successful software company.” (http://www.wired.com/news/mac/0,2125,54365,00.html)

1987: Perseus Project

Named for the Greek hero Perseus, who learned by traveling, the Perseus Project presents students with a world of ancient Greek culture and literature. Initially the project was realized using hypercard. Currently the project resides on-line and is accessible through the World Wide Web.

1987: Palenque

One of the first pilot applications that made use of motion video and interactive technology to create an impression of immersion in a cultural heritage site. The pilot was co-developed by the DVI technology group at David Sarnoff Research Center, and Bank Street College in New York City.

1989: Rediscovering Pompeii
“Between 1987 and 1989, more than one hundred scientists, archaeologists, architects, art historians, anthropologists, and specialists from IBM Italy and FIAT Engineering collaborated to develop useful new applications for computer technology in the field of archaeology… Rediscovering Pompeii was a museum exhibition designed to encourage the visitor to explore everyday life in 1st century Pompeii, and presented a comprehensive picture of the life, death and re-discovery of the ancient city. The exhibition presented a combination of the world of archaeology and advanced information technology.”

1991: CAVE

CAVE stands for Cave Advanced Virtual Environment and is one of the most advanced virtual reality systems in use today. The first CAVE was built at the University of Illinois. A CAVE can have up to six walls that are shaped like a cube around the user. Behind each wall there is a video projector which displays a computer-generated image on the wall. The user, being inside the cube, feels like being present in the virtual environment. The CAVEs usually also feature shutter glasses for true 3D graphics and a magnetic tracking system in order to create a correct view for the user.

http://cave.ncsa.uiuc.edu/about.html

1991: WWW and HTML

The World Wide Web (WWW) and the related Hypertext Markup Language (HTML) and Hypertext Transport Protocol (HTTP) technologies were first launched in 1991 at CERN. The main architects of the system were Tim Berners-Lee and Robert Caillau. The original purpose of the web was to share device-independent documents among the researchers at CERN. After the release of a graphical browser, NCSA Mosaic (1993), the web began to its well-known rapid expansion. NCSA Mosaic source code was later used in widely popular Netscape Navigator and Internet Explored browsers. In addition to hypertext publishing the web has also become an important application platform and has its own document-based user interface paradigm which differs from the traditional user interfaces.

http://www.w3.org/

Though it has not been updated since 1997, the MOSAIC website is still kept online at: (http://archive.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/help-about.html)

1992: OpenGL

Silicon Graphics Inc. introduced the OpenGL 3D application programming interface (API) in 1992 for its workstations. OpenGL was the successor of
the earlier IRIS GL. During the following years OpenGL spread from
the Silicon Graphics workstations to other platforms and became the standard
programming interface in professional 3D graphics. Currently OpenGL features
an established, well defined and portable set of functions for graphics
programming.

http://www.opengl.org/

1993: Contextual design

An important trend in user interface design, originally developed by
K. Holtzblatt and S. Jones. The principle of contextual design is to research
and model the user in her actual workplace in the first stages of the
software development cycle.

Holtzblatt, K. & Jones, S. Contextual Inquiry: A Participatory
Technique for System Design. Participatory Design: Principles and
Practice. Aki Namioka and Doug Schuler (Eds.), Lawrence


1995: VRML 1.0

Virtual Reality Modeling Language was designed to be the virtual reality equivalent
for HTML. The consortium was founded in 1994 and the first official specification
was published in 1995. Later on followed VRML97 (1997) and the current version,
X3D. The acceptance of the standard has been hindered by incompatible
implementations and the absence of authoring tools. Currently the amount of 3D web
content is still small.

http://www.web3d.org/
User Interface Design: Visions

Augmentation

To augment means to increase or to enlarge, especially in size. Augmentation as a paradigm is related to the idea of the infinite extension of the human sensory capabilities. In information technology, augmentation refers to the potential of technology to enhance human agency. Technology is seen as possessing the capacity to increase the physical limits of the organism. This is realized in three ways: Sensory enhancement through tools that extend the scope and reach of the human perceptual experience, mental enhancement through the tools that supplement human cognition, and through the co-evolution of technology and human practice so as to produce new forms of human/machine interaction.

In information technology, the paradigm of augmentation first appears in the work of Vannevar Bush: “The impulses which flow in the arm nerves of a typist convey to her fingers the transplanted information which reaches her eye or ear, in order that the fingers may be caused to strike the proper keys. Might not these currents be intercepted, either in the original form in which the information is conveyed to brain, or in the marvelously metamorphosed form in which they then proceed to the hand?” … “With a couple of electrodes on the skull the encephalograph now produces pen-and-ink traces which bear some relation to the electrical phenomena going on in the brain itself. True the record is unintelligible, except as it points out certain gross malfunctioning of the cerebral mechanism; but who would now place bounds on where such a thing may lead?”

The vision of augmentation is continued through the work of Douglas Engelbart, who is the inventor of many of the physical artifacts and concepts in contemporary information technology, including the basic concept of computer interface:

“In the early 1960s, Engelbart began the Augmentation Research Centre (ARC), a development environment at the Stanford Research Institute. Here, he and his colleagues (William K. English and John F. Rulifson) created the On-Line System (NLS), the world's first implementation of what was to be called hypertext. Yet this was only a small part of what ARC was about. As he states in Working Together, Engelbart was particularity concerned with ‘asynchronous collaboration among teams distributed geographically.’ This endeavor is part of the study of Computer Supported Co-operative Work (CSCW); software which supports this goal is often called groupware.” (http://www.iath.virginia.edu/elab/hfl0035.html)

Engelbart introduced the concept of Bootstrapping which means “to promote or develop by initiative and effort with little or not assistance.” The augmentation project in Engelbart’s work involves: “a reflexive application of the cybernetics principles as embodied in the notion of bootstrapping to help people to learn how to become better at realizing their abilities to deal with the challenges that confronted
them.” His approach is distinct in that it requires that the human learn to adapt to the computer. In Engelbart’s vision exchange between the computer and the human takes part in a larger framework H-LAMT/T system (Human using Language, Artifact, Methodology in which he is trained.) The point of contact between human and computer intelligence, the boundary that separates them and joins them exists as a metaphor. As a language machine, for Engelbart the computer can serve as a genuine boundary-spanning machine.44

Among the items invented by Engelbart are: the computer mouse, multiple windows, shared-screen teleconferencing, computer-aided meetings, context-sensitive help, distributed client-server architecture, and hypermedia.

Bootstrap Alliance
http://www.bootstrap.org/engelbart/index.jsp

Graphical User Interfaces

The ideas and interface methods seen in Douglas Engelbart’s The Demo in 1968 were the foundation for development of graphical user interface (GUI). See: (http://sloan.stanford.edu/mousesite/1968Demo.html) The Demo already featured multi user interaction, windows and mouse. The modern interfaces found in today’s computers are indirectly successors of Engelbart's work. After “The Demo” the next big milestone in graphical user interface was the work done in Xerox PARC, which soon lead to creation of Apple Lisa and Macintosh computers (1983/1984) that brought the new interface paradigm to personal computing.

The Learning Research group of PARC was lead by Alan Kay, who was interested in helping people, especially children, to learn how to use a computer. His visions of learning are reflected in the user interface that was realized in the ALTO computer in 1973. The interface seen in Alto develops further the concepts of Engelbart and already features most of the elements found in a modern GUI, such as folders, menus, icons and overlapping windows. An interesting fact about the development is that the interface was built simple and graphical because of Kay’s observations about the learning process of children. The management of Xerox was skeptical about the radical ideas found in Alto and thus not willing to commit large funding for the project. The young and unbiased people of Apple computer visiting PARC in 1979 were not and replicated the paradigm in their upcoming Lisa computer. See: (http://ei.cs.vt.edu/~history/GASCH.KAY.HTML)

The other visions of Alan Kay have also influenced the way we use and think of computers today. Kay was one of the first people to research a programming paradigm called object-oriented programming (OOP), which is very different from the traditional procedural programming. The basic idea of object-oriented programming is to raise the abstraction level of programming to include objects: entities that incorporate data and functions. These objects often correspond to real-world entities thus making the programming more task-oriented and understandable by humans. Smalltalk (1980), a programming
language developed by Alan Kay is regarded as the father of object-oriented languages. Kay is credited as the inventor of portable computers, but unfortunately the technology to build such a device was not available yet.

**Hyper-trails and Associative Linking**

Hypertext has its origins in Vannevar Bush’s concept of the Memex, a micro-fiche based machine that allowed the user to forge his/her path through trails of information:

“In hypertext, the basic unit is a node. Nodes are connected to each other by electronic cross-references called links. To access more information about a specific topic, the hypertext reader simply points to a link anchor… the computer screen immediately changes to reveal the contents of the node to which the link refers.”

The concept was further developed by Ted Nelson, with the Xanadu project. Though the name was chosen in 1967, Xanadu was started in the early 1960’s: “Project Xanadu was the explicit inspiration for the World Wide Web (see Tim Berners Lee’s original proposal for the World Wide Web), for Lotus Notes (as freely acknowledged by its creator, Ray Ozzie) and for HyperCard (acknowledged by its developer, Bill Atkinson)” See Xanadu’s home page: (http://xanadu.com/xuhistory.html)

Hypertext is an elegant interaction approach that makes use of the human brain’s natural tendency to think associatively. It also builds on the strength of both computer and the human: the computer holds the data and the human chooses the path of navigation by pointing and clicking at each juncture.

**Cyberspace**

The term “cyberspace” was used by writer William Gibson to describe the networked virtual worlds in his books. Long before the Internet age Gibson already sketched a worldwide data network with a virtual reality interface. The terminology used by him includes words like “cyberspace” and “matrix” that are still used in everyday talk. Computer viruses, system security, Internet (including related technologies such as VRML) and 3D user interfaces all have their counterparts in Gibson’s fantasy world. The visions found in Gibson’s vision has on the other hand shaped the things we see today and on the other hand also been a surprisingly accurate prediction of the direction of the development.

By far the most famous of Gibson's books is *Neuromancer* (1984), the first novel in the *Neuromancer* trilogy. The original book was followed by *Mona Lisa Overdrive* and *Count Zero* that take place in the same milieu. In *Neuromancer* Gibson presents the reader a large-scale vision of a global interconnected network where companies, governments and independent cyberjockeys have a fight going on over money, power and information. This pessimistic vision of the future is known as
“cyberpunk” and has been replicated in numerous books, movies and computer games ever since.46

**Gesture-based Multi-user Interaction**

Myron W. Krueger was the first visionary to research the field of gesture-based human computer interaction in a large scale. The work of Krueger has influenced several different areas of virtual reality and user interfaces such as video conferencing, multi-user interaction and gesture-based interaction involving the full body. One fundamental difference between Krueger's work and the other research of the time is the use of the full body of the user as a natural input device. Most of the other systems were based on special input devices and displays such as head-mounted displays. Krueger regarded those devices as physically uncomfortable and tried to find more natural alternatives. Krueger himself called his concepts as “artificial reality.”47

Videoplace (1983) is the most important of his works. Krueger also built a smaller version of the system called the Videodesk. The central goals in the use of Videoplace were to explore the possibilities of gestures as an interaction method and to share the interaction between multiple users.

Videoplace is a system where the user stands in front of a large screen that displays computer generated images. There is a video camera that records the user's gestures and sends the image to a computer that interprets the gestures and movements. Two users can be connected via a video link and they can perform actions at the same time, resulting in multiuser collaborative interaction. Things like 3D object management, document sharing and Teleconferencing were among the subjects that were researched using the Videoplace system.48

During the years the themes and questions risen from Myron Krueger's work have not lost their importance. The introduction of affordable web cameras and digital video cameras has led to wide interest and use of gestures as an alternative interaction method. One interesting example of this development is “Quiqui's Giant Bounce” ([http://www.kukakumma.net](http://www.kukakumma.net)), an interactive web camera based game for children.49 The teleconferencing tools of today owe a lot to Krueger, yet the implementations have not come even close to the goals set by him.

**User’s Rights**

For more than two decades, Don Norman has been an advocate of user’s right. According to Norman, for a technology to be truly successful, it has to disappear. This does not mean that the Internet or computers will go away but rather that their presence will fade, as their ubiquity increases. According to Norman: “You don’t think about the process of how a light switch works when you turn it on. It just works. That’s what needs to happen to computers in order for people to get the most out of them.”
Norman often illustrates his concepts with everyday examples such as doors or faucets that are hard to use because of the flawed design. If even simple things such as doors are illogical then how much more difficult is it to design complex computer applications that would be easy to use? One fundamental principle of Norman is that there are no stupid users, only bad design.


Don Norman’s publications online: [http://www.jnd.org/dn.pubs.html](http://www.jnd.org/dn.pubs.html)
The Present State of User Interfaces is Stagnated

The popular user interfaces of today have not seen dramatic developments in many years. The biggest change took place when the graphical user interface became popular in the late 1980’s.

The World Wide Web has had a significant standardizing effect on user interfaces, not so much of because of its technical benefits, but more because of its social acceptance as a global information technology infrastructure that provides a uniform interaction language to the widest possible community across operating system boundaries.

One of the new popular developments has been the evolution of high performance three-dimensional game environments and a variety of new interaction styles within them. Small devices like mobile phones and digital cameras have also begun to have their own computer style interfaces that borrow some features from the more powerful platforms but generally have very idiosyncratic solutions for all of their features.

In systems specially designed for particular purposes, such as scientific, industrial or business systems, or museum or art exhibits, several different kinds of more experimental or exotic interfaces have been tested and utilized.

However, and in spite of significant investments into research and development in the area, the mainstream user interaction ideas and patterns have not evolved very much during recent years. Why is this?

The main problems in interaction are not at the interface

One of the reasons for slow development in the interface development is probably that the research and development has not addressed the right problems. It is usually assumed that by making the interface better, significant benefits can be achieved. However, when all attention is paid to the interface, and when it happens in the context of specialized applications, two important dimensions are usually left without proper attention: the depth and the breadth.

The depth dimension means the layers and structures behind the interface, that provide the content and functionality that users should have access to. The breadth dimension means the diverse combinations of activities and systems that form the digital ecosystem of the user, and within which the system in question should seamlessly and usefully be integrated.

These kinds of problems can’t be solved with interface design – they are more architectural, more fundamental concerns, that would require more concerted and collaborative efforts than current research and development systems tend to favour.
In fact, the GUI became popular because of very deliberate efforts by first Apple and then Microsoft to change the whole computer use experience in large, orchestrated revolutionary developments.

Apple not only produced a commercially viable version of a GUI, but also developed a coherent set of user interaction standards – the Human Interface Guidelines – and designed the whole operating system so that it would facilitate these and would gently guide other developers to adopt their proposed interaction models. Terry Winograd offers this as an example of a design language.  

It was probably because of this design language, its well thought guidelines that were consistently and efficiently encouraged and supported by the operating system features, and the deliberately and thoughtfully developed supporting culture with its evangelists, that the WIMP (Windows, Icons, Mouse, Pointing) GUI became so successful. The computer industry had never seen such a uniform way of doing similar things across programs. This uniformity made it easy for non-expert computer users to adopt the tools, and facilitated the entry of personal computers into the everyday life routine of people.

The advance from the idiosyncratic character oriented interfaces to GUIs was a big social step for computing, and it introduced a new culture where users acquired an understanding of how to accomplish the basic interaction patterns, such as moving around documents in windows, typing and editing text, changing fonts, printing, and so on. The GUI has since then not been replaced, because those kinds of things it still manages quite well. The problems of the interaction are now on a deeper level. When computers were new in the everyday life, it was understandable that you had to teach them everything, and that programs were simple and fairly stupid.

However, as we are now surrounded by computers, networks and information, it becomes more and more tedious to repeat the same actions with various non-communicating pieces of software over and over again. Usability problems have nowadays less to do with the interface than the underlying logic and knowledge of the world and user’s life and preferences that the software exhibits.

For example, people who work daily with personal laptop computers have often gigabytes of information of things that relate to their life and activities in the computer. This information may consist of documents they have written or have received, emails, information systems that track their business transactions, calendars, project plans, photographs, digital voice mail records and so on. However, these pieces of information are not organized by their meaning by the operating system, and there are no links between the documents that relate to the same customer, for example. It is up to the user to invent ways and strategies for managing the complex collections of digital data, and for this, the GUI or other interface innovations offer little help.

The underlying software, in the depth dimension, is too simple and stupid to cater for the growingly sophisticated needs of users, and the interface technologies cannot
soothe or hide that reality. Therefore it is likely that significant advances in the user interaction are hard to achieve until the software can ‘behave’ in ‘smarter’ fashion, for example by utilizing the information it possesses about user’s circumstances and previous activities in determining how it reacts to user’s current actions.

Smarter behavior, in turn, is usually interpreted as something that requires Artificial Intelligence (AI). AI has for decades tried to realize an intelligent machine, starting from the premise that intelligence can be designed based on the capability of the computer to perform logical operations. This has not succeeded, mainly because it seems that human intelligence is a product of some other kind of information processing than was expected. Therefore, AI has sought to find other, less grandiose applications for its technologies. Some of these will be useful for achieving ‘smarter’ behavior, but it is probably important to stress that the ‘smartness’ that counts here is not some internal intelligent capability of the computer as much as it is the perceived anticipation, adaptation and support that the system gives the user in different circumstances.

Therefore, simple and effective customization features that let the user make sure herself that the system behaves in expected and beneficial ways in certain circumstances will deliver the kind of ‘smartness’ that makes a difference, regardless of whether any AI is employed in realizing those features.

In the breadth dimension, even if software systems have similar GUIs that all share similar text manipulation and copy/paste functionalities, it is still very challenging to integrate them into a working whole. The management of the personal software ecosystem is the users’ responsibility, and different systems do very little to aid the users in this. When a component in the system changes, for example the mobile phone, it may render the user inoperable for a few days as she spends her time learning the idiosyncratic functions and interface conventions of the new tool and moves all relevant information from the old to the new one. Most likely also the integration glue that the user has developed, for example routines for keeping a calendar on two devices in sync, will become useless and a new solution needs to be custom developed.

Again, the usability problems apparent in these kinds of examples are not in the interface, they are in the strategies of the developers, or in the lack of coherence and collaboration in their design. A new interface will not help the situation. Instead a new system layer that brings the desirable coherence seems necessary.

**Personalized interfaces**

This is where the layered idea of interfaces comes in to the discussion. In the future, it is likely that the coherence and smartness that may eventually deliver better usability and utility to users results from new interface layers between the objects of interest and the users. These would make it possible for the users to customize their
interactions with any software objects according to their personal preferences and according to a coherence strategy that they choose.

This requires that the objects of interest, for example here the cultural heritage software objects, must provide programming interfaces (PI), so that the user’s interaction style can be accommodated. The objects should not require direct access to the users, although it is a good idea to provide always a default interaction mechanism that can be used when special preferences are not given. In the same fashion, interesting special interactions could also be supported and may also be passed through the user’s own customizations.

Shared ontologies

Shared ontologies and directories are another direction of developments that will make future interactions more satisfying and interesting. These will be new layers for systems that extend both their depth and their opportunities for horizontal integration and collaboration. The existence of shared ontologies, better metadata and better directories of information in ways that would be useful across systems would make new applications and uses possible.

Shared interaction languages

In the same fashion as information content can be described, also functionality and behavior could be described. In the future, there will probably be sets of standard behaviors that digital objects can be chosen to support.

Having a standard in interaction languages has two major benefits. Firstly, the user could concentrate on her actual tasks instead of learning a new paradigm for each platform. Secondly, interconnected "building blocks" allow the user to do more than the traditional approach of having a separate application for each task.

Future digital cultural heritage objects

These capabilities are examples today of the kinds of universals that should be introduced into the world of digital systems in coherent ways, much like the GUIs from Apple and Microsoft did about 20 years ago. It seems that breakthroughs in the interfaces of the future will first and foremost require cultural developments and social innovations that introduce a new level of coherence and social intelligence into the systems.

In considering the how to design digital cultural heritage objects that should be useful for future users with their future interface technologies, it is therefore important to realize that the interface only a part of a system, and the system is a part of an ecosystem. The usefulness, utility and usability of the object is dependent not only of
its interface but of its behaviors, programming interfaces, metadata, and connections to other resources, which in turn can make it possible to develop intelligent and interesting interfaces.

Future users should not be restricted to the interface standards of today; it is therefore important to offer not only a direct user interface but also a programming interface to the users, so that they or third parties that serve them can upgrade the functionality with their own technology to suit the evolving needs.

In order to decide upon the right set of basic programming interface features for digital cultural heritage objects, it would be useful to develop fairly deep and informed scenarios of future interactions that illustrate the diversity of possible applications well enough. The field needs shared design vision of what its possibilities are. With an effort put into developing shared concepts and shared design languages, also better design outcomes may be achieved.
Glossary of Terms and Concepts

**Affordance**: A term used by J. J. Gibson to describe a possibility for action. A door handle and stairs are examples of affordances.

**Application Programming Interface (API)**: A set of functions, procedures and variables that act as an interface between a software package and an application.

**Artificial Intelligence (AI)**: A common name for the set of different techniques that aim at teaching computers how to think. Here “thinking” refers to the human way of thinking.

**Augmentation**: To extend the capabilities of a human being through technology.

**Client-Server Architecture**: A software architecture that has two components: the server that acts as a common repository of data and the clients that communicate with the server.

**Clipboard**: A temporary storage in computer memory necessary for the common copy and paste operations between applications.

**Command Line**: A user interface paradigm where the user interacts with the system purely by textual means

**Context-sensitive**: An object whose function depends on the current state of the application. A typical example is context-sensitive help, meaning that the user gets help for the specific task she's trying to complete.

**Cultural Heritage**: Cultural heritage is a broad term meant to include all forms of cultural and artistic expression inherited from the near or distant past of a given country or cultural area.

**Cyberspace**: A term used by writer William Gibson to describe a networked virtual reality.

**Data Glove**: An input device that resembles a glove. The computer measures the movement of the user’s fingers that allows for natural gesture-based interaction.

**Drag-and-Drop**: A popular interface method which is based on picking elements (with a mouse) and dropping them on other elements.

**Graphical User Interface (GUI)**: A "graphical user interface" (GUI pronounced: goo-ey) provides users with a picture-oriented user-friendly way to see what is on a computer system.

**Groupware**: A common nominator for applications that let multiple users share information and work on the same task.
**Haptic:** Related to the sense of touch and its implementation in computer applications.

**Head-mounted Display (HMD):** A display device that is attached to the user's head. Some models can display a different image for each eye that enables the creation of true 3D graphics.

**Hypermedia:** A special case of multimedia that includes hypertext navigation.

**Hypertext:** Information that is structured as a network of nodes and links. Following a link brings the user to another node. The word hypertext was first used in 1965 by Theodore Holm Nelson. World Wide Web is the most common form of hypertext in use today.

**Immersion:** The subjective feeling of “being there.” A term used for describing the credibility of a virtual environment.

**Input Device:** Any device that lets the user interact with the computer. Mouse and keyboard are the most typical input devices in personal computers.

**Multimedia:** Media that contains some or all of the following: images, audio, animation, and video.

**Multimodal interface:** An interface that involves more than one sense.

**Object-oriented Programming (OOP):** A popular programming paradigm involving high-level entities called objects. The objects can contain both data and functions.

**Output Device:** A device that displays information for the user. For example a printer, monitor or a loudspeaker.

**Pull-down Menu:** A menu that becomes visible after the user selects its title.

**Scenario:** A description of an activity, in narrative form, and a view of a system from the outside, or from the user’s perspective.

**Shutter Glasses:** A device that shuts off the view of each eye consecutively, thus enabling stereoscopic viewing because a different image can be shown for the left and the right eye.

**Simulator:** A system that is built to act as some other system. The first computer-based simulators were flight simulators used in pilot training.

**Stereo graphics, Stereoscopic:** Providing a different image for both eyes in order to make computer graphics look truly three-dimensional. The need for such a construct rises because the human depth perception relies heavily on the interpupillary distance.
Tangible: Employing material objects and devices to allow manipulation of immaterial, virtual objects.

Teleconferencing: Discussion and groupwork over a video and voice link.

Terminal: A device that handles the input and output of data over a network connection.

Touchpad: An interaction device usually found in laptop computers. A flat pad that measures the movement of fingers on it.

Tracking System: A device that enables the measurement of the user’s direction or location. Can be attached to different body parts to track for example the movement of head or hands.

Transparency: Consistency in an interface that hides the complexities and implementation specific details of a system.

Ubiquity: In the present report the term is used to refer to the pervasive presence of computing devices in all aspects of human existence. In the current trend, ubiquity is accompanied by transparency and disappearance of the computer so that it is no longer a conscious factor in the user’s experience.51

Usability: Measure of the quality of the user’s experience when interacting with a product or system — whether a Web site, a software application, mobile technology, or any user-operated device. Usability factors include: Ease of learning, efficiency of use, memorability, error frequency and severity, as well as user satisfaction. See: Usability Basics at (http://usability.gov/basics/index.html)

Virtual Reality: An artificial environment created by the means of 3D computer graphics, spatial audio and in some cases also haptic feedback.

WIMP: Windows, Icons, Mouse/Menus, Pointer. The standard graphical user interface paradigm found in today's computers.
Part II: Nordic Network in Digital Culture Heritage

Who are we?

The Nordic Network in Digital Cultural Heritage was brought together as part of the initiatives of Nordic Interactive and the University of Art and Design Helsinki, Media Lab. Already in 2001, the network held a seminar in Digital Culture Heritage as part of the NIC 2001 conference held in Copenhagen, Denmark. Dr. Kim Veltman was featured as an invited speaker in this seminar. The period of inactivity that followed this event ended with a request, from Dr. Veltman, for representation of the Nordic partners in the E-Culture Net thematic network. The reactivation of the Nordic Network in Digital Cultural Heritage, has been one of the direct consequences resulting from participation in E-Culture Net.

The following institutions have participated in the E-Culture Net thematic network: University of Oslo, Intermedia in Oslo, Norway; Interactive Institute Stockholm in Sweden, University of Lund, Department of Information Technology; Aarhus University, Department of Information Technology and Media Studies and the Centre for Advanced Visualization and Interaction (CAVI), and University of Art and Design Helsinki, Media Lab in Finland.

The Media Lab of the University of Art and Design Helsinki and Nordic Interactive has led the work effort for the network. As has been noted, the activities of the Nordic partners have focused in the area of interface design.

The Media Lab was formed in 1993 as a separate department within UIAH. The Lab became a faculty of UIAH in 1998. MA and Doctoral programs of study in new media are offered. The mission of the Media Lab is to explore, discover and comprehend the new digital technology and its impact in society; to find and exploit the possibilities it opens to communication, interaction and expression and to evaluate, understand and deal with the challenges it poses to design and creative production.

The information society with its converging media is a complex environment that requires an interdisciplinary approach to design. This is reflected in the co-operative education and research projects of the Media Lab. Within UIAH the Media Lab plays an active role in strategic planning and integration of matters relating to interactive media.

Nordic Interactive is an open group that focuses on initiating and stimulating research, development and education in the area of interactive digital technology in the Nordic countries – and within the Nordic region as one common entity. Nordic Interactive provides a network for Nordic research, development and education in interactive technology, and thereby promote visibility and achievability of high-level expertise for potential international partners, and for a continued Nordic growth.

Emphasis is on the already prominent level achieved in many areas of interactive technology and related areas such as pervasive computing, human computer
interaction, mobile computing, hypermedia, object technology, pattern recognition, and human centered information systems design. In addition, Nordic Interactive has included a special initiative to foster research and development in the area of applications of digital technology in the cultural heritage sector.

**Activities in E-Culture Net**

**Logogram and Web Design**

The work performed by the Media Lab as part of their work on interface design included the logo and identity programme for E-Culture-Net as well as the graphic design for the E-Culture Net website and related application (See Appendix 2.) The work done on the logogram included four logotype examples and three iterations of design. The logo and identity programme is now implemented in all electronic and printed media produced by the E-Culture Net thematic network.

**Publications in the Area of Digital Cultural Heritage**

A paper documenting original basic research into the application of Mary Douglas’ “Grid and Group” method as a way to organize communities of users in a broadband universe, was written by Lily Díaz-Kommonen and presented by Philip Dean at the December conference of the network. The paper will also be presented in EVA 2003 London in July. A copy of the paper is included in the Appendix to this report.

**One-day Expert’s Meeting in Helsinki**

EU Thematic Networks are designed to allow for the collaboration of experts, for exchange of information and expertise, and with the aim to create future research agendas and projects.

A one-day experts’ meeting took place in Helsinki, in the Marilyn theatre of the LUME Media Centre at the University of Art and Design UIAH, on Friday March 28th 2003. The meeting was organized as part of the official efforts of Media Lab UIAH within the E-Culture Net Thematic Network. The target audience was the Nordic partners in the E-Culture Net project and experts working in cultural heritage in Finland.

The objective of the meeting was to work on identifying key issues development of a broadband Distributed Electronic European Resource (DEER), concentrating specifically with issues of interface and challenges of interface design for such a system. The programme for the day included introductory lectures/presentations, as well as collaborative workshops. The keynote speaker was Dr. Manfred Thaller from the University of Cologne, one of our partners in E-Culture Net.
Customizable web site as tool for collaborative work

As part of the work to prepare for the meeting, a tool for collaboration in the form of a website was created. The website was designed so that it could be customized by registered participants. It included a manual that explains how to make use of its features. It allowed participants to submit the work that they were asked to prepare for the meeting. This task included the submission of scenarios of relevance to the question of potential uses for DEER.

It was also used to take notes and record ideas as they developed throughout the meeting. The website is located at: http://sysrep.uiah.fi/e_culture

Expert’s Presentations

The presentations given as part of the meeting included a presentation by Prof. Manfred Thaller of the work done with cultural heritage databases at the University of Cologne.

Dr. Lily Díaz-Kommonen gave an update of the work being done in Helsinki as part of the CIPHER project. CIPHER, the acronym stands for Communities of Interest to Promote the Heritage of European Regions, is a EU funded project under the IST 5th Framework. Using Self-Organizing Technologies, CIPHER aims to produce advanced resource organization and navigation tools. As part of its methodological tools, CIPHER is also creating Culture Heritage (CH) Forums. The concept of the CH Forum aims to link digital technology research and development with institutions from the physical world, with the final objective of self-sustainability.

Prof. Andrew Morrison, from University of Oslo Intermedia Center, gave a presentation on the work that is currently being done at this institution on the topic of digital interfaces for dance and performance. The abstract of the paper, to be published in Digital Creativity, is included in the Appendix.

Professionals currently working in the field of culture heritage in Finland also attended the meeting. Riikka Haapalainen, researcher at the Finnish National Gallery, and Marjo Maenpää, Director of the Professional Transfer Program at the Media Lab presented their work in the MUMMI project for the year 2002-2003. Mr. Sampsa Saarela, a researcher, gave a presentation about the Semantic Web Project at the University of Helsinki.

Scenarios

The Helsinki Expert’s Meeting produced two complete scenarios. One of these, Distributed 3D for Cultural Heritage Seminar was produced solely at Aarhus University in Denmark. At the request of the Helsinki organizers, Aarhus submitted
the scenario as part of the work in preparation for the seminar. The scenario was used during the seminar as an example for those who were unfamiliar with the use of this tool for design. The materials for this scenario in the website include the diagrammatic illustration and the narrative submitted by Aarhus.

The second scenario, Wand and Footprints, was produced by one of the two groups who participated in the scenario building task session during the Expert’s meeting. The materials for this scenario in the website include video documentation of the presentation as part of the concluding session. A diagrammatic illustration outlines the technology components, as well as the workflow, the full text of the narrative.
Scenario 1: Distributed 3D in Cultural Heritage Seminar 2005
Kim Halskov, Svend Erik Søfeldt, CAVI.dk
26.03.2003

Theme: Distributed 3D Cultural Heritage

We are in the year 2005 the European Cultural Heritage Year and students attending the European Cultural Heritage masters program are on their weekly meeting. The meeting takes place in a 3D panorama facility at their own university.

Today students throughout Europe are using broadband connection made available through DEER to hear a lesson about the Danish Viking Age. In the virtual studio in Denmark archaeologist Svend Pedersen welcomes: “Good morning every!!” which is being echoed from France, Great Britain, Greece, Finland etc.: “GOOD MORNING!!"

Svend Petersen: “I am standing here in main yard of Trellebrog a Danish Viking Age fortification.” The 3D stereo camera pans across the façade that is actually a digital reconstruction.

The 3D video connection is switched to the 3D ENG camera at Trelleborg.

Svend Petersen continues: “As you can see only little remains left of the buildings.” [walking across the yard]: “I will return to the fortress later ..... over here we have a piece of a old Viking found in the fjord of Roskilde .......”

Michelle [Sitting in the Panorama in Paris interrupts]: “Sven, what kind of axes was used during that period of time?”

Svend Petersen: “Let me show you.” [The 3D stereo camera focuses cutting mark on piece of wood from the Viking Ship]. Michelle [in the Paris Panorama]: “It looks like an iron axe to me?” [Half an hour later] The closing remark from the virtual studio in Denmark: “Looking forward to see you all in the virtual world next week when we all will be in Italy to listen to a demostration and lecture about the Roman Period.” France, Great Britain, Greece, Finland etc.: “Thanks, see you.”

To read more about 3D panorama and virtual studio technology see: http://www.CAVI.dk
Figure 2: Scenario describing technology and activities for a Distributed 3D Culture Heritage Seminar in 2005.
Scenario 2: Wand and Footprints
Riikka, Riitta, Katri, Manfred, and Kim
28.03.2003

Theme: Personalized museum collection for secondary school children

User groups: Teachers and pupils at secondary education, roughly 10 - 14 year old ones.

Communities: Museum professionals, educators, students

Scenario events:
Mrs. Maria Virtanen is a school teacher in cultural subjects at a secondary school in Finland. She is preparing her teaching concerning the life in Finland and Greece. She uses the DEER to get access to artworks about Greece in the 19th century. A few days later she shows her findings to the students in her class, explaining what they tell about the daily life of the times. Soon after that, she takes her class on a visit to the local art museum with the task to find images that illustrate the difference of the local lifestyle to the Greek one.

When they enter the museum each student is issued an “intelligent wand.” Seeing an image that is appropriate for the task, the students point the intelligent wands at a barcode below the images. A microphone at the wand allows them to enter comments. When they leave the museum, they return the wands, receiving a floor plan of the museum, with a line of footprints connecting all the items they clicked at, in temporal order.

Coming home, they receive a copy of all the images in their e-mail, in a special image format. This image format will make itself unreadable after seven days. A second copy is delivered to the school's computer on the account of the teacher. Both sets of images are accompanied by the appropriate comments.

After they have studied the images at home, they can use the copy at the school to present their “homework.” The price paid for admission decides about the number of images that can be collected and about the quality of the images delivered.

Technology components: Protected image formats

Key interface issues: Intelligent wands, map that outlines footprints of visit, admissions transaction.
Figure 3: Scenario describing technology and activities for use of DEER by secondary school students visiting a museum.
Sources:
4. Raisio Archaeology Archive: (http://mlab.uiah.fi/mulli/e_index.html)
8. Bodker, Susanne. Through the Interface...36.
15. Marcus, Aaron. Graphic Design for Electronic Documents and User Interfaces, 118.
18. Ibid., [Summary and Recommendations, Introduction].
24. Kalawsky, Roy S. The Science of Virtual Reality and Virtual Environments,


34. Kalawsky, Roy S. *The Science of Virtual Reality and Virtual Environments*.


47. Kalawsky, Roy S. *The Science of Virtual Reality and Virtual Environments*.


