Human–computer interaction

Mrs. A.G.Andurkar , Assistant Professor, Department of Electronics & Telecommunication, Government College of Engineering, Jalgaon, Maharashtra , India

Miss. Ritupriya G.Andurkar, Student , Department of Computer Engineering, Government College of Engineering, Jalgaon, Maharashtra , India

---

Abstract

Human–computer interaction (HCI) researches the design and use of computer technology, focusing particularly on the interfaces between people (users) and computers. Researchers in the field of HCI both observe the ways in which humans interact with computers and design technologies that let humans interact with computers in novel ways. As a field of research, Human-Computer Interaction is situated at the intersection of computer science, behavioral sciences, design, media studies, and several other fields of study. The term was popularized by Stuart K. Card and Allen Newell of Carnegie Mellon University and Thomas P. Moran of IBM Research in their seminal 1983 book, The Psychology of Human-Computer Interaction, although the authors first used the term in 1980. HCI has expanded rapidly and steadily for three decades, attracting professionals from many other disciplines and incorporating diverse concepts and approaches. To a considerable extent, HCI now aggregates a collection of semi-autonomous fields of research and practice in human-centered informatics. However, the continuing synthesis of disparate conceptions and approaches to science and practice in HCI has produced a dramatic example of how different epistemologies and paradigms can be reconciled and integrated in a vibrant and productive intellectual project. The user interacts directly with hardware for the human input and output such as displays ,e.g. through a graphical user interface. The user interacts with the computer over this software interface using the given input and output (I/O)hardware. Software and hardware must be matched, so that the processing of the user input is fast enough, the latency of the computer output is not disruptive to the workflow.

When evaluating a current user interface, or designing a new user interface, it is important to keep in mind the following experimental design principles:

- ***Early focus on user(s) and task(s):** Establish how many users are needed to perform the task(s) and determine who the appropriate users should be; someone who has never used the interface, and will not use the interface in the future, is most likely not a valid user. In addition, define the task(s) the users will be performing and how often the task(s) need to be performed.*
- ***Empirical measurement:** Test the interface early on with real users who come in contact with the interface on a daily basis. Keep in mind that results may vary with the performance level of the user and may not be an accurate depiction of the typical human-computer interaction. Establish quantitative usability specifics such as: the number of users performing the task(s), the time to complete the task(s), and the number of errors made during the task(s).*
- ***Iterative design:** After determining the users, tasks, and empirical measurements to include, perform the following iterative design steps:*

1. ***Design the user interface***
2. ***Test***
3. ***Analyze results***
4. ***Repeat***

Keywords: human-computer interaction, cognitive engineering, graphical user interfaces (GUI), Ergonomics and HCI Ergonomics.

I .Introduction

Until the late 1970s, the only humans who interacted with computers were information technology professionals and dedicated hobbyists. This changed disruptively with the emergence of personal computing in the later 1970s. Personal computing, including both personal software (productivity applications, such as text editors and spreadsheets, and interactive computer games) and personal computer platforms (operating systems, programming languages, and hardware), made everyone in the world a potential computer user, and vividly highlighted the deficiencies of computers with respect to *usability* for those who wanted to use computers as tools.

The challenge of personal computing became manifest at an opportune time. The broad project of cognitive science, which incorporated cognitive psychology, artificial intelligence, linguistics, cognitive anthropology, and the philosophy of mind, had formed at the end of the 1970s. Part of the program f cognitive science was to articulate systematic and scientifically informed applications to be known as "cognitive engineering". Thus, at just the point when personal computing presented the practical need for HCI, cognitive science presented people, concepts, skills, and a vision for addressing such needs through an ambitious synthesis of science and engineering. HCI was one of the first examples of cognitive engineering.

This was facilitated by analogous developments in engineering and design areas adjacent to HCI, and in fact often overlapping HCI, notably human factors engineering and documentation development. Human factors had developed empirical and task-analytic techniques for evaluating human-system interactions in domains such as aviation and manufacturing, and was moving to address interactive system contexts in which human operators regularly exerted greater problem-solving discretion. Documentation development was moving beyond its traditional role of producing systematic technical descriptions toward a cognitive approach incorporating theories of writing, reading, and media, with empirical user testing. Documents and other information needed to be usable also.

Humans interact with computers in many ways; and the interface between humans and the computers they use is crucial to facilitating this interaction. Desktop applications, internet browsers, handheld computers, and computer kiosks make use of the prevalent graphical user interfaces (GUI) of today. Voice user interfaces (VUI) are used for speech recognition and synthesizing systems, and the emerging multi-modal and gestalt User Interfaces (GUI) allow humans to engage with embodied character agents in a way that cannot be achieved with other interface paradigms.

The Association for Computing Machinery defines human-computer interaction as "a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them". An important facet of HCI is the securing of user satisfaction (or simply End User Computing Satisfaction). "Because human–computer interaction studies a human and a machine in communication, it draws from supporting knowledge on both the machine and the human side. On the machine side, techniques in computer graphics ,operating systems, programming languages, and development environments are relevant. On the human side, communication theory, graphic and industrial design disciplines, linguistics, social sciences, cognitive psychology, social psychology, and human factors such as computer user satisfaction are relevant. And, of course, engineering and design methods are relevant. Due to the multidisciplinary nature of HCI, people with different backgrounds contribute to its success. HCI is also sometimes referred to as **human–machine interaction** (**HMI**), **man–machine interaction** (**MMI**) or **computer–human interaction** (**CHI**).

Poorly designed human-machine interfaces can lead to many unexpected problems. A classic example of this is the Three Mile Island accident, a nuclear meltdown accident, where investigations concluded that the design of the human–machine interface was at least partially responsible for the disaster. Similarly, accidents in aviation have resulted from manufacturers' decisions to use non-standard flight instrument or throttle quadrant layouts: even though the new designs were proposed to be superior in regards to basic human–machine interaction, pilots had already ingrained the "standard" layout and thus the conceptually good idea actually had undesirable results.

Human-Computer Interaction studies the ways in which humans make, or make not, use of computational artifacts, systems and infrastructures. In doing so, much of the research in the field seek to `improve' human-computer interaction by improving the `usability' of computer interfaces. How `usability' is to be precisely understood, how it relates to other social and cultural values and when it is, and when it may not be a desirable property of computer interfaces is increasingly debated.

Much of the research in the field of Human-Computer Interaction takes an interest in:

- methods for designing novel computer interfaces, thereby optimizing a design for a desired property such as, e.g., efficiency of use. An example of a design method that has been continuously developed by HCI researchers is Participatory Design.
- methods for implementing interfaces, e.g., by means of software tool kits and libraries
- methods for evaluating and comparing interfaces with respect to their usability or other desirable properties
- methods for studying human computer use and its socio cultural implications more broadly
- models and theories of human computer use as well as conceptual frameworks for the design of computer interfaces, such as, e.g., cognitivist user models, Activity Theory or ethno-methodological accounts of human computer use
- perspectives that critically reflect upon the values that underlie computational design, computer use and HCI research practice

Visions of what researchers in the field seek to achieve vary. When pursuing cognitivist perspective, researchers of HCI may seek to align computer interfaces with the mental model that humans have of their activities. When pursuing a post-cognitivist perspective, researchers of HCI may, e.g., seek to align computer interfaces with existing social practices or existing socio cultural values.

Professional practitioners in HCI are usually designers concerned with the practical application of design methodologies to problems in the world. Their work often revolves around designing graphical user interfaces and web interfaces.

Researchers in HCI are interested in developing new design methodologies, experimenting with new devices, prototyping new software systems, exploring new interaction paradigms, and developing models and theories of interaction.

Following figures 1 to 4 are related with human computer interaction.

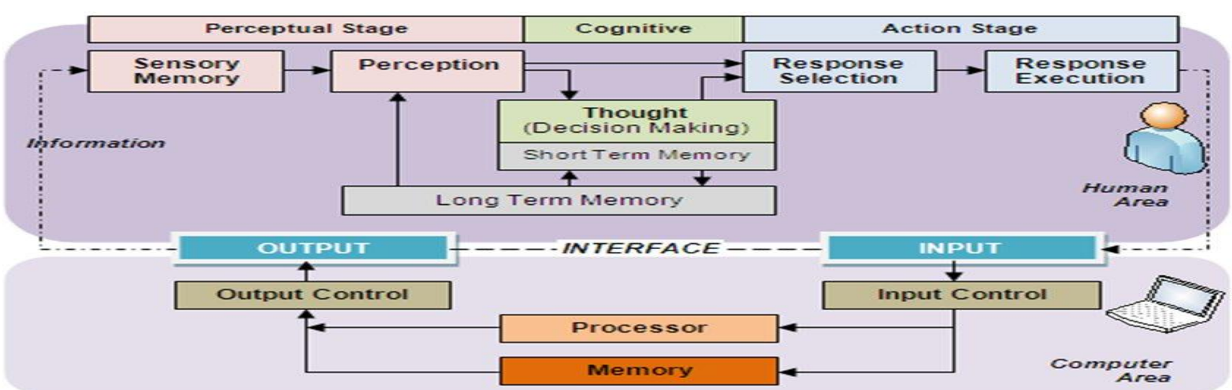

Figure1 **Block diagram of a human computer interaction system.**

Figure2 **human computer interaction example.**

figure 3 **different types of displays in human computer interaction.**

http://www.webdesignerdepot.com/2009/05/the-evolution-of-cell-phone-design-between-1983-2009/

Figure 4 . **different telephone and mobile phones for human computer interaction purpose.**

Other historically fortuitous developments contributed to the establishment of HCI. Software engineering, mired in unmanageable software complexity in the 1970s (the "software crisis"), was starting to focus on nonfunctional requirements, including usability and maintainability, and on empirical software development processes that relied heavily on iterative prototyping and empirical testing. Computer graphics and information retrieval had emerged in the 1970s, and rapidly came to recognize that interactive systems were the key to progressing beyond early achievements. All these threads of development in computer science pointed to the same conclusion: The way forward for computing entailed understanding and better empowering users. These diverse forces of need and opportunity converged around 1980, focusing a huge burst of human energy, and creating a highly visible interdisciplinary project.

1.1 Differences with related fields

HCI differs from human factors and ergonomics as HCI focuses more on users working specifically with computers, rather than other kinds of machines or designed artifacts. There is also a focus in HCI on how to implement the computer software and hardware mechanisms to support human–computer interaction. Thus, *human factors* is a broader term; HCI could be described as the human factors of computers – although some experts try to differentiate these areas.

HCI also differs from human factors in that there is less of a focus on repetitive work-oriented tasks and procedures, and much less emphasis on physical stress and the physical form or industrial design of the user interface, such as keyboards and mouse devices.

Three areas of study have substantial overlap with HCI even as the focus of inquiry shifts. In the study of personal information management (PIM), human interactions with the computer are placed in a larger informational context – people may work with many forms of information, some computer-based, many not (e.g., whiteboards, notebooks, sticky notes, refrigerator magnets) in order to understand and effect desired changes in their world. In computer-supported cooperative work (CSCW), emphasis is placed on the use of computing systems in support of the collaborative work of a group of people. The principles of human interaction management (HIM) extend the scope of CSCW to an organizational level and can be implemented without use of computers.

1.2 Design Principles:

The user interacts directly with hardware for the human *input* and *output* such as displays, e.g. through a graphical user interface. The user interacts with the computer over this software interface using the given input and output (*I/O*) hardware. Software and hardware must be matched, so that the processing of the user input is fast enough, the latency of the computer output is not disruptive to the workflow.

When evaluating a current user interface, or designing a new user interface, it is important to keep in mind the following experimental design principles:

- Early focus on user(s) and task(s): Establish how many users are needed to perform the task(s) and determine who the appropriate users should be; someone who has never used the interface, and will not use the interface in the future, is most likely not a valid user. In addition, define the task(s) the users will be performing and how often the task(s) need to be performed.
- Empirical measurement: Test the interface early on with real users who come in contact with the interface on a daily basis. Keep in mind that results may vary with the performance level of the user and may not be an accurate depiction of the typical human-computer interaction. Establish quantitative usability specifics such as: the number of users performing the task(s), the time to complete the task(s), and the number of errors made during the task(s).
- Iterative design: After determining the users, tasks, and empirical measurements to include, perform the following iterative design steps:
    1. Design the user interface
    2. Test
    3. Analyze results
    4. Repeat

Repeat the iterative design process until a sensible, user-friendly interface is created.

1.3 Methodologies

A number of diverse methodologies outlining techniques for human–computer interaction design have emerged since the rise of the field in the 1980s. Most design methodologies stem from a model for how users, designers, and technical systems interact. Early methodologies, for example, treated users' cognitive processes as predictable and quantifiable and encouraged design practitioners to look to cognitive science results in areas such as memory and attention when designing user interfaces. Modern models tend to focus on a constant feedback and conversation between users, designers, and engineers and push for technical systems to be wrapped around the types of experiences users want to have, rather than wrapping user experience around a completed system.

- Activity theory: used in HCI to define and study the context in which human interactions with computers take place. Activity theory provides a framework to reason about actions in these contexts, analytical tools with the format of checklists of items that researchers should consider, and informs design of interactions from an activity-centric perspective.
- User-centered design: user-centered design (UCD) is a modern, widely practiced design philosophy rooted in the idea that users must take center-stage in the design of any computer system. Users, designers and technical practitioners work together to articulate the wants, needs and limitations of the user and create a system that addresses these elements. Often, user-centered design projects are informed by ethnographic studies of the environments in which users will be interacting with the system. This practice is similar but not identical to participatory design, which emphasizes the possibility for end-users to contribute actively through shared design sessions and workshops.
- Principles of user interface design: these are seven principles of user interface design that may be considered at any time during the design of a user interface in any order: tolerance, simplicity, visibility, affordance, consistency, structure and feedback.
- Value sensitive design: Value Sensitive Design (VSD) is a method for building technology that account for the values of the people who use the technology directly, as well as those who the technology affects, either directly or indirectly. VSD uses an iterative design process that involves three types of investigations: conceptual, empirical and technical. Conceptual investigations aim at understanding and articulating the various stakeholders of the technology, as well as their values and any values conflicts that might arise for these stakeholders through the use of the technology. Empirical investigations are qualitative or quantitative design research studies used to inform the designers' understanding of the users' values, needs, and practices. Technical investigations can involve either analysis of how people use related technologies, or the design of systems to support values identified in the conceptual and empirical investigations.

1.4 Display Designs:

Displays are human-made artifacts designed to support the perception of relevant system variables and to facilitate further processing of that information. Before a display is designed, the task that the display is intended to support must be defined (e.g. navigating, controlling, decision making, learning, entertaining, etc.). A user or operator must be able to process whatever information that a system generates and displays; therefore, the information must be displayed according to principles in a manner that will support perception, situation awareness, and understanding.

Thirteen principles of display design

Christopher Wickens et al. defined 13 principles of display design in their book "*An Introduction to Human Factors Engineering*."

These principles of human perception and information processing can be utilized to create an effective display design. A reduction in errors, a reduction in required training time, an increase in efficiency, and an increase in user satisfaction are a few of the many potential benefits that can be achieved through utilization of these principles.

Certain principles may not be applicable to different displays or situations. Some principles may seem to be conflicting, and there is no simple solution to say that one principle is more important than another. The principles may be tailored to a specific design or situation. Striking a functional balance among the principles is critical for an effective design.

Perceptual principles

1. Make displays legible (or audible). A display's legibility is critical and necessary for designing a usable display. If the characters or objects being displayed cannot be discernible, then the operator cannot effectively make use of them.

2. Avoid absolute judgment limits. Do not ask the user to determine the level of a variable on the basis of a single sensory variable (e.g. color, size, loudness). These sensory variables can contain many possible levels.

3. Top-down processing. Signals are likely perceived and interpreted in accordance with what is expected based on a user's experience. If a signal is presented contrary to the user's expectation, more physical evidence of that signal may need to be presented to assure that it is understood correctly.

4. Redundancy gain. If a signal is presented more than once, it is more likely that it will be understood correctly. This can be done by presenting the signal in alternative physical forms (e.g. color and shape, voice and print, etc.), as redundancy does not imply repetition. A traffic light is a good example of redundancy, as color and position are redundant.

5. Similarity causes confusion*: Use discriminable elements*. Signals that appear to be similar will likely be confused. The ratio of similar features to different features causes signals to be similar. For example, A423B9 is more similar to A423B8 than 92 is to 93. Unnecessary similar features should be removed and dissimilar features should be highlighted.

Mental model principles

6. Principle of pictorial realism. A display should look like the variable that it represents (e.g. high temperature on a thermometer shown as a higher vertical level). If there are multiple elements, they can be configured in a manner that looks like it would in the represented environment.

7. Principle of the moving part. Moving elements should move in a pattern and direction compatible with the user's mental model of how it actually moves in the system. For example, the moving element on an altimeter should move upward with increasing altitude.

Principles based on attention

8. Minimizing information access cost. When the user's attention is diverted from one location to another to access necessary information, there is an associated cost in time or effort. A display design should minimize this cost by allowing for frequently accessed sources to be located at the nearest possible position. However, adequate legibility should not be sacrificed to reduce this cost.

9. Proximity compatibility principle. Divided attention between two information sources may be necessary for the completion of one task. These sources must be mentally integrated and are defined to have close mental proximity. Information access costs should be low, which can be achieved in many ways (e.g. proximity, linkage by common colors, patterns, shapes, etc.). However, close display proximity can be harmful by causing too much clutter.

10. Principle of multiple resources. A user can more easily process information across different resources. For example, visual and auditory information can be presented simultaneously rather than presenting all visual or all auditory information.

Memory principles

11. Replace memory with visual information*: knowledge in the world*. A user should not need to retain important information solely in working memory or retrieve it from long-term memory. A menu, checklist, or another display can aid the user by easing the use of their memory. However, the use of memory may sometimes benefit the user by eliminating the need to reference some type of knowledge in the world (e.g. an expert computer operator would rather use direct commands from memory than refer to a manual). The use of knowledge in a user's head and knowledge in the world must be balanced for an effective design.

12. Principle of predictive aiding. Proactive actions are usually more effective than reactive actions. A display should attempt to eliminate resource-demanding cognitive tasks and replace them with simpler perceptual tasks to reduce the use of the user's mental resources. This will allow the user to not only focus on current conditions, but also think about possible future conditions. An example of a predictive aid is a road sign displaying the distance to a certain destination.

13. Principle of consistency. Old habits from other displays will easily transfer to support processing of new displays if they are designed consistently. A user's long-term memory will trigger actions that are expected to be appropriate. A design must accept this fact and utilize consistency among different displays.

Given the contemporary shape of HCI, it is important to remember that its origins are personal productivity interactions bound to the desktop, such as word processing and spreadsheets. Indeed, one of biggest design ideas of the early 1980s was the so-called messy desk metaphor, popularized by the Apple Macintosh: Files and folders were displayed as icons that could be, and were scattered around the display surface. The messy desktop was a perfect incubator for the developing paradigm of graphical user interfaces. Perhaps it wasn't quite as easy to learn and easy to use as claimed, but people everywhere were soon double clicking, dragging windows and icons around their displays, and losing track of things on their desktop interfaces just as they did on their physical desktops. It was surely a stark contrast to the immediately prior teletype metaphor of Unix, in which all interactions were accomplished by typing commands.

even though it can definitely be argued that the desktop metaphor was superficial, or perhaps under-exploited as a design paradigm, it captured imaginations of designers and the public. These were new possibilities for many people in 1980, pundits speculated about how they might change office work. Indeed, the tsunami of desktop designs challenged, sometimes threatened the expertise and work practices of office workers. Today they are in the cultural background. Children learn these concepts and skills routinely.

As HCI developed, it moved beyond the desktop in three distinct senses. First, the desktop metaphor proved to be more limited than it first seemed. It's fine to directly represent a couple dozen digital objects as icons, but this approach quickly leads to clutter, and is not very useful for people with thousands of personal files and folders. Through the mid-1990s, HCI professionals and everyone else realized that search is a more fundamental paradigm than browsing for finding things in a user interface. Ironically though, when early World Wide Web pages emerged in the mid-1990s, they not only dropped the messy desktop metaphor, but for the most part dropped graphical interactions entirely. And still they were seen as a breakthrough in usability (of course, the direct contrast was to Unix-style tools like ftp and telnet). The design approach of displaying and directly interacting with data objects as icons has not disappeared, but it is no longer a hegemonic design concept.

The second sense in which HCI moved beyond the desktop was through the growing influence of the Internet on computing and on society. Starting in the mid-1980s, email emerged as one of the most important HCI applications, but ironically, email made computers and networks into communication channels; people were not interacting *with* computers, they were interacting with other people *through* computers. Tools and applications to support collaborative activity now include instant messaging, wikis, blogs, online forums, social networking, social bookmarking and tagging services, media spaces and other collaborative workspaces, recommender and collaborative filtering systems, and a wide variety of online groups and communities. New paradigms and mechanisms for collective activity have emerged including online auctions, reputation systems, soft sensors, and crowd sourcing. This area of HCI, now often called social computing, is one of the most rapidly developing.

The third way that HCI moved beyond the desktop was through the continual, and occasionally explosive diversification in the ecology of computing devices. Before desktop applications were consolidated, new kinds of device contexts emerged, notably laptops, which began to appear in the early 1980s, and handhelds, which began to appear in the mid-1980s. One frontier today is ubiquitous computing: The pervasive incorporation of computing into human habitats — cars, home appliances, furniture, clothing, and so forth. Desktop computing is still very important, though the desktop habitat has been transformed by the wide use of laptops. To a considerable extent, the desktop itself has moved off the desktop.

The focus of HCI has moved beyond the desktop, and its focus will continue to move. HCI is a technology area, and it is ineluctably driven to frontiers of technology and application possibility. The special value and contribution of HCI is that it will investigate, develop, and harness those new areas of possibility not merely as technologies or designs, but as means for enhancing human activity and experience.

The task-artifact cycle

The movement of HCI off the desktop is a large-scale example of a pattern of technology development that is replicated throughout HCI at many levels of analysis. HCI addresses the dynamic co-evolution of the activities people engage in and experience, and the artifacts — such as interactive tools and environments — that mediate those activities. HCI is about understanding and critically evaluating the interactive technologies people use and experience. But it is also about how those interactions evolve as people appropriate technologies, as their expectations, concepts and skills develop, and as they articulate new needs, new interests, and new visions and agendas for interactive technology.

Reciprocally, HCI is about understanding contemporary human practices and aspirations, including how those activities are embodied, elaborated, but also perhaps limited by current infrastructures and tools. HCI is about understanding practices and activity specifically as requirements and design possibilities envisioning and bringing into being new technology, new tools and environments. It is about exploring design spaces, and realizing new systems and devices through the co-evolution of activity and artifacts, the task-artifact cycle.

Understanding HCI as inscribed in a co-evolution of activity and technological artifacts is useful. Most simply, it reminds us what HCI is like, that all of the infrastructure of HCI, including its concepts, methods, focal problems, and stirring successes will always be in flux. Moreover, because the co-evolution of activity and artifacts is shaped by a cascade of contingent initiatives across a diverse collection of actors, there is no reason to expect HCI to be convergent, or predictable. This is not to say progress in HCI is random or arbitrary, just that it is more like world history than it is like physics. One could see this quite optimistically: Individual and collective initiative shapes what HCI is, but not the laws of physics.

A second implication of the task-artifact cycle is that continual exploration of new applications and application domains, new designs and design paradigms, new experiences, and new activities should remain highly prized in HCI. We may have the sense that we know where we are going today, but given the apparent rate of co-evolution in activity and artifacts, our effective look-ahead is probably less than we think. Moreover, since we are in effect constructing a future trajectory, and not just finding it, the cost of missteps is high. The co-evolution of activity and artifacts evidences strong hysteresis, that is to say, effects of past co-evolutionary adjustments persist far into the future. For example, many people struggle every day with operating systems and core productivity applications whose designs were evolutionary reactions to misanalyses from two or more decades ago. Of course, it is impossible to always be right with respect to values and criteria that will emerge and coalesce in the future, but we should at least be mindful that very consequential missteps are possible.

The remedy is to consider many alternatives at every point in the progression. It is vitally important to have lots of work exploring possible experiences and activities, for example, on design and experience probes and prototypes. If we focus too strongly on the affordances of currently embodied technology we are too easily and uncritically accepting constraints that will limit contemporary HCI as well as all future trajectories.

1.5 Humanizing Technology:

The march of progress in computing is a climb. Each big step forward is also a step up, so that communication is further away from the machine, more on human terms.

And each time, the number of people who can use computing increases dramatically. At first, programming languages were the medium of communication between man and machine. Fortran, the breakthrough computer language, was designed to resemble the algebraic formulas familiar to scientists and engineers — reasonably enough, since they were the only people anyone could imagine using the relative handful of giant calculating machines back then.

Today, billions of people roam the Internet from computer phones they hold in their hands. Dramatic advances in hardware, of course, are a big part of the explanation, notably the flywheel of technological dynamism known as Moore's Law, celebrating the chip industry's ability to double computing power every couple of years (there's a debate about whether the pace is tailing off, but that's another story).

Yet there is another force in the striking democratization of computing beyond hardware, one that is more subtle but still crucial. That is the steady stream of improvements in the design of computer products, mainly software, which have opened the door to new users by making computers easier to use. The term most used now is "user-interface design." But that suggests a narrower, product focus than the field that stretches back several decades, called human-computer interaction, which embraces psychology, anthropology and other disciplines.

"I think human-computer interaction designs have had as much impact as Moore's Law in bringing the web and mobile devices to the world," said Ben Shneiderman, a professor at the University of Maryland, College Park.

To try to raise the profile of prominent people in the field, Mr. Shneiderman last week published a web site for what he calls "The Human-Computer Interaction Pioneers Project." It is a personal project for Mr. Shneiderman, who founded Maryland's Human-Computer Interaction Laboratory in 1983, and is an avid photographer. The web page for each of his pioneers includes a brief text description and photographs, often several, that Mr. Shneiderman has taken of the person, at professional conferences and elsewhere over the years.

He is starting with 45 subjects, and plans to add more. Some of the people featured are familiar names in computer history like Douglas Engelbart (computer mouse), Alan Kay (laptop) and Ted Nelson (hypertext). But many others are not. And going through the list, and talking to Mr. Shneiderman, it becomes clear how much the field has evolved over the years and its broad influence in business these days.

His list includes scientists like Sara Bly, who was early to move beyond lab settings to study how people actually use technology at work and at home, and Jennifer Preece, who has researched online communities and how design choices affect social behavior.

The best people in the field, Mr. Shneiderman said, possess "a deep knowledge of technology and a real sensitivity to human needs."

Getting out in the field to see how people really use products and seeking to empathize with customers rather than merely exploit are becoming priorities in the tech industry and beyond. They are key ingredients in two of the most popular management trends today: lean start-up techniques and design thinking.

The latter is being embraced by Charles Phillips at Infor, a large business software company, and by Virginia M. Rometty at IBM. Design thinking is the theme of the September issue of The Harvard Business Review, with a collection of articles on the subject. One of Mr. Shneiderman's human-computer interaction pioneers, Donald Norman, is also a prominent design-thinking consultant.

The A/B testing that is a common practice at major Internet companies like Facebook, Amazon and Google is a form of human-computer research. In A/B tests, different features or web-page layouts are presented to different groups of users — often samplings of millions at a time — and observing how they react.

Programming remains central to the profession of human-computer interaction. Mr. Shneiderman's lifetime achievement award from the Association for Computing Machinery, the largest computing professional society, is for his work on "direct manipulation" software that helped enable hyperlinks and touch screen keyboards. (He included himself among the 45 human-computer interaction pioneers.)

But the field, Mr. Shneiderman said, is once again headed further up, away from the machine. The goal of tomorrow's software, he suggests, will be less on features on a web site or smart phone and more on design that encourages better outcomes for individuals and society in, say, education or health care.

"The future lies in these social designs," he said. "Design to encourage trust, empathy and responsibility, while protecting privacy. That's the next big thing."

Human–computer interface

The human–computer interface can be described as the point of communication between the human user and the computer. The flow of information between the human and computer is defined as the *loop of interaction*. The loop of interaction has several aspects to it, including:

- *Task environment*: The conditions and goals set upon the user.
- *Machine environment*: The environment that the computer is connected to, e.g. a laptop in a college student's dorm room.
- *Areas of the interface*: Non-overlapping areas involve processes of the human and computer not pertaining to their interaction. Meanwhile, the overlapping areas only concern themselves with the processes pertaining to their interaction.
- *Input flow*: The flow of information that begins in the task environment, when the user has some task that requires using their computer.
- *Output*: The flow of information that originates in the machine environment.
- *Feedback*: Loops through the interface that evaluate, moderate, and confirm processes as they pass from the human through the interface to the computer and back.
- *Fit*: This is the match between the computer design, the user and the task to optimize the human resources needed to accomplish the task.

User customization

End-user development studies how ordinary users could routinely tailor applications to their own needs and use this power to invent new applications based on their understanding of their own domains. With their deeper knowledge of their own knowledge domains, users could increasingly be important sources of new applications at the expense of generic systems programmers (with systems expertise but low domain expertise).

Embedded computation

Computation is passing beyond computers into every object for which uses can be found. Embedded systems make the environment alive with little computations and automated processes, from computerized cooking appliances to lighting and plumbing fixtures to window blinds to automobile braking systems to greeting cards. To some extent, this development is already taking place. The expected difference in the future is the addition of networked communications that will allow many of these embedded computations to coordinate with each other and with the user. Human interfaces to these embedded devices will in many cases be very different from those appropriate to workstations.

Augmented reality

A common staple of science fiction, augmented reality refers to the notion of layering relevant information into our vision of the world. Existing projects show real-time statistics to users performing difficult tasks, such as manufacturing. Future work might include augmenting our

social interactions by providing additional information about those we converse with.

Social computing

In recent years, there has been an explosion of social science research focusing on interactions as the unit of analysis. Much of this research draws from psychology, social psychology, and sociology. For example, one study found out that people expected a computer with a man's name to cost more than a machine with a woman's name.[1] Other research finds that individuals perceive their interactions with computers more positively than humans, despite behaving the same way towards these machines.

Factors of change

Human–computer interaction is affected by the forces shaping the nature of future computing. These forces include:

- Decreasing hardware costs leading to larger memory and faster systems
- Miniaturization of hardware leading to portability
- Reduction in power requirements leading to portability
- New display technologies leading to the packaging of computational devices in new forms
- Specialized hardware leading to new functions
- Increased development of network communication and distributed computing
- Increasingly widespread use of computers, especially by people who are outside of the computing profession
- Increasing innovation in input techniques (e.g., voice, gesture, pen), combined with lowering cost, leading to rapid computerization by people previously left out of the "computer revolution."
- Wider social concerns leading to improved access to computers by currently disadvantaged groups

Traditionally, as explained in a journal article discussing user modeling and user-adapted interaction, computer usage was modeled as a human-computer dyad in which the two were connected by a narrow explicit communication channel, such as text-based terminals. Much work has been done to make the interaction between a computing system and a human. However, as stated in the introduction, there is much room for mishaps and failure. Because of this, human-computer interaction shifted focus beyond the inter-face (to respond to observations as articulated by D. Engelbart: "If ease of use was the only valid criterion, people would stick to tricycles and never try bicycles."

The means by which humans interact with computers continues to evolve ra

The future for HCI, based on current promising research, is expected include the following characteristics:

- *Ubiquitous communication*. Computers are expected to communicate through high speed local networks, nationally over wide-area networks, and portably via infrared, ultrasonic, cellular, and other technologies. Data and computational services will be portably accessible from many if not most locations to which a user travels.
- *High-functionality systems*. Systems can have large numbers of functions associated with them. There are so many systems that most users, technical or non-technical, do not have time to learn them in the traditional way (e.g., through thick manuals).
- *Mass availability of computer graphics*. Computer graphics capabilities such as image processing, graphics transformations, rendering, and interactive animation are becoming widespread as inexpensive chips become available for inclusion in general workstations and mobile devices.
- *Mixed media*. Commercial systems can handle images, voice, sounds, video, text, formatted data. These are exchangeable over communication links among users. The separate worlds of consumer

electronics (e.g., stereo sets, VCRs, televisions) and computers are partially merging. Computer and print worlds are expected to cross-assimilate each other.

- *High-bandwidth interaction*. The rate at which humans and machines interact is expected to increase substantially due to the changes in speed, computer graphics, new media, and new input/output devices. This can lead to some qualitatively different interfaces, such as virtual reality or computational video.
- *Large and thin displays*. New display technologies are finally maturing, enabling very large displays and displays that are thin, lightweight, and low in power consumption. This is having large effects on portability and will likely enable the development of paper-like, pen-based computer interaction systems very different in feel from desktop workstations of the present.
- *Information utilities*. Public information utilities (such as home banking and shopping) and specialized industry services (e.g., weather for pilots) are expected to proliferate. The rate of proliferation can accelerate with the introduction of high-bandwidth interaction and the improvement in quality of interfaces.

2. System Development

2.1 The human

Introduction

In 1983, Card, Moran and Newell described the Model Human Processor: a simplified view of the human processing involved in interacting with computer systems.

MHP comprises 3 subsystems: the perceptual system, the motor system and the cognitive system. Each of them has a processor and memory. MHP also includes a number of Principles of operation which dictate the behavior of the system under certain conditions.

Input-output channels :In interaction with a computer, the human input is the data output by the computer vice versa. Input in humans occurs mainly through the senses and output through the motor controls of the effectors. Vision, hearing and touch are the most important senses in HCI. The fingers, voice, eyes, head and body position are the primary effectors.

Vision :Visual perception can be divided in 2 stages: the physical reception of the stimulus from the outside world, and the processing and interpretation of that stimulus. The eye is a mechanism for receiving light and transforming it into electrical energy. Light is reflected from objects in the visual field and their image is focused on the back of the eye, where it is transformed into an electrical signal and passed to the brain. The most important components are the cornea and lens and the retina with the blind spot and photoreceptors: rods and cones, located on the fovea. Rods are highly sensitive to light and usable under low illumination, but do not distinguish fine details. The cones are less sensible to light and can distinguish color. The eye can perceive size and depth using the visual angle. If two objects are at the same distance from the eye, the larger one will have a larger visual angle. Similarly, if two objects of the same size are at different distances from the eye, the furthest one will have the smaller visual angle. The visual angle is given in degrees or minutes of arc (1 degree = 60 minutes of arc)

Visual acuity is the ability of a person to perceive small details. If the visual angle is too small, the details will not be perceived. The minimum visual angle is approximately .5 seconds of arc. However, according to the law of size constancy, our perception of size relies on more factors than the visual angle, for example, the perception of depth. Depth can be perceived through various cues, e.g. indications in the visual context about an object's distance and familiarity with the size of the object. Perception of size and depth are highly intertwined. Perception of brightness is a subjective reaction to levels of light emitted by an object: luminance Contrast is related to luminance, since it is the function of the luminance of the object and the background. Visual acuity increases with increased luminance. However, on screen, the flicker also increases with the luminance. The eye perceives color because the cones are sensitive to light of different wavelengths. It should be reminded that 3-4% of the fovea is sensitive to blue, making blue acuity lower. The context in which an object appears allows our expectations to clearly disambiguate the interpretation of the object. . First, the visual pattern of the word is perceived. Second, it is decoded with reference to an internal representation of language. Finally, the word is processed as part of the sentence or phrase using syntactic and semantic analysis. During the first two stages, the eyes make saccades (jerky movements), followed by fixations. The eye moves both forwards and backwards over the text, called regressions, which is increased when the text is more complex.

Hearing :The ear receives vibrations on the air and transmits them through various stages to the auditory nerves. The ear compromises 3 sections, the outer ear , middle ear (tympanic membrane and cochlea) and inner ear (with cilia). The inner ear is filled with cochlean liquid. The sound waves are transmitted to the liquid. The vibrations, now in the liquid, bend the cilia which releases a chemical transmitter. The transmitter causes impulses in the auditory nerves. The human ear can hear frequencies from 20 Hz to 15 kHz. The sound we perceive is (selectively) filtered, which is illustrated by the cocktail party effect: we can notice our name spoken out in a noisy room. Sound (vibrations) have a number of characteristics. The pitch is the frequency of the sound. The higher the frequency, the higher the sound. The loudness corresponds to the amplitude of the sound. Timbre relates to the type of the sound, independent of frequency and amplitude.

Touch The apparatus of touch (haptic perception) is not localized. Stimuli are received through the skin, which contains various types of sensory receptors. Mechanoreceptors, responding to pressure, are important in HCI. There are 2 kinds of MR's: Rapidly adapting mechanoreceptors, responding to immediate pressure as the skin is intended. They stop responding as continuous pressure is applied, to which slowly adapting mechanoreceptors respond. Some areas of the body have greater sensitivity/acuity than others. This can be measured using the two-point threshold test. A second aspect of haptic perception is kinesthesis: awareness of the position of the body and limbs, due to receptors in the joints. There are 3 types: rapidly adapting (respond when moving limb in direction), slowly adapting (respond to movement and static position) and positional receptors (only responding to static positions).

Movement When making movements, a stimulus is received through the sensory receptors and transmitted to the brain. After processing, the brain tells the appropriate muscle to respond. The movement time is dependent on the physical characteristics of the subjects. The reaction time varies according to the sensory channel through which the stimulus is received. Accuracy is a second measure of motor skill. A fast respond does not always mean a less accurate response. The time taken to hit a target is a function of the size of the target and the distance that has to be moved. This is formalized in Fitts'law, which is commonly written as: Movement time = a + b log2 (distance=size + 1) where a and b are empirically constants.

Human memory:We can distinguish 3 types of memory: sensory buffers, short-term memory (or working memory) and long-term memory.

Sensory memory: The sensory memories act as buffers for stimuli received through each of the senses: iconic memory for vision, echoic memory for sounds and haptic memory for touch. These memories are constantly overwritten by new information coming in on these channels. Information is passed from the sensory memory into short-term memory by attention, filtering the stimuli to those that are at that moment of interest (arousal, or shift of attention).

Short-term memory: STM is used to store information which is only required temporarily. STM can be accessed rapidly, however, also decays rapidly. It has a limited capacity. Miller stated the 7+/-2 rule, which means that humans can store 5-9 chunks of information. Chunks can be single items or groups of items, like 2 digits of a telephone number grouped together. Patterns can be useful as aids to memory

HUMAN MEMORY

The recency effect suggests that STM recall is damaged by interferences of other information. Long-term memory is not affected.

Long-term memory: LTM differs from STM in various ways. It has an unlimited capacity, a slow access time and forgetting occurs more slowly or not at all. Information is stored here from the STM through rehearsal. There are 2 types of LTM: episodic memory and semantic memory. Episodic memory represents our memory of event and experiences in a serial form.

Semantic memory is a structured record of facts, concepts and skills that we have acquired, derived from the episodic memory. According to the semantic network model, the semantic memory is structured as a network . The more general the information is, the higher is the level on which it is stored. This allows us to generalize about specific cases. The connections in the network are made using associations. There are other models about the organization of our LTM. Structured representations like frames and scripts, for example, organize information into data structures. Frames have slots to add attribute values. A script comprises a number of elements, which, like slots, can be filled with appropriate information. Another model is the production system, which holds IF-THEN rules: if information coming into the STM matches one of the conditions in the LTM, the appropriate action is executed. There are 3 main activities related to LTM: storage of information, forgetting and information retrieval. Storage: The rehearsal of a piece of information from the STM stores it in the LTM. If the total learning time is increased, information is remembered better (total time hypothesis). However, the learning time should be well spread (distribution of practice effect). But repetition alone is not enough: information should be meaningful and familiar, so it can be related to existing structures and more easily incorporated into memory. Forgetting: There are 2 main theories of forgetting: decay and interference. Decay suggests that information held in LTM may eventually be forgotten. Jost's Law states that if 2 memory traces are equally strong at the same time, the older one will be more durable. Information, however, can also be lost through interference: if we acquire new information, it causes the loss of old information: retroactive interference. It is also possible that the older information interferes with the newly acquired information: proactive inhibition. Forgetting is affected by emotional factors too. Retrieval: There are 2 types of information retrieval: recall and recognition. In recall the information is produced from memory. It can be facilitated by providing cues, e.g. the category in which the information may be placed. In recognition, the presentation of the information provides the knowledge that the information has been seen before.

THINKING: REASONING AND PROBLEM SOLVING

Thinking: reasoning and problem solving Thinking can require different amounts of knowledge. Some thinking activities are very directed and the knowledge required is constrained. Others require vast amounts of knowledge from different domains. Thinking can be divided in reasoning and problem solving.

Reasoning

Reasoning is the process by which we use the knowledge we have to draw conclusions or infer something new about the domain of interest. There are different types of reasoning: deductive, inductive and additive. Deduction: Deductive reasoning derives the logically necessary conclusion from the given premises. The logical conclusion does not have to correspond to our notion of truth. The human deduction is weak at the points where truth and validity clash. Induction: Inductive reasoning is generalizing from cases we have seen to infer information about cases we have not seen. In practise, induction is used to Öll in missing details while reasoning. Abduction: Abduction reasons from a fact to the action or state that caused it. Abduction is used to derive explanations for the events we observe.

Problem solving Problem solving is the process of finding a solution to an unfamiliar taste, using (adapting) the knowledge we have. There are different views on problem solving: Gestalt theory: The Gestalt theory states that problem solving is both productive and reproductive; insight is needed to solve a problem.

Problem space theory: The problem space comprises problem states and problem solving involves generating these states using legal state transition operators. People use these to move from the initial state to the goal state. Heuristics (e.g. Means-end analysis) are employed to select the right operators. Use of analogy: Problems are solved by mapping knowledge relating to a similar known domain to the new problem: analogical mapping. Skill acquisition Experts often have a better encoding of knowledge: information structures are fine tuned at a deep level to enable effi cient and accurate retrieval. According to the ATC model, these skills are acquired through 3 levels: The learner uses general-purpose rules which interpret facts about a problem. (slow, memory-demanding)

EMOTION

The learner develops rules specific to the task, using proceduralization. The rules are tuned to speed up performance, using generalization.

Errors and mental models There are different types of errors: changes in context of skilled behavior can cause errors. An incorrect understanding/model of a situation can cause errors too, because humans tend to create mental models, based on experience, which may differ from the actual situation.

Emotion Emotion involves both physical and cognitive events. Our body responds biologically to an external stimulus and we interpret that in some way as a particular emotion. That biological response (affect) changes the way we deal with different situations and this has an impact on the way we interact with computer systems.

Individual differences: The principles and properties discussed apply to the majority of people, but humans are not all the same. Differences should be taken into account in the designs: divide the users in target groups, for example.

Psychology and the design of interactive systems

Guidelines General design principles and guidelines (straightforward or complex) can be and have been derived from the above discussed theories. Models to support design Psychological analysis has led to the development of analytic and predictive models of user behavior.

2.2 The Computer:

Introduction Interaction (with or without computer) is a process of information transfer. The diversity of devices reflects the fact that there are many different types of data that may be entered into and obtained from a system, as there are many different users. In the early days, batch processing was common: a large mass of information was dumped into and processed by the computer. Nowadays, computers respond within milliseconds and computer systems are integrated in many different devices.

Text entry devices

The alphanumeric keyboard :The vast majority of keyboards have a standardized layout, known by the first six letters on the top row: QWERTY. The non-alphanumeric keys are not standardized. This layout is not optimal for typing, but dates from the time of mechanical limitations of the typewriter. Today, the keys can also be arranged in alphabetic order (the alphabetic keyboard), but this does not improve typing performance. The DVORAK keyboard does, placing the keys in a different order on a similar layout as found on the QWERTY keyboards. The layout minimized the stretch of fingers and the use of weak fingers, reducing fatigue and increasing typing speed (10-15%).

Chord keyboards :On chord keyboards, only a few keys are used. Letters are produces pressing multiple keys at once. They are smaller than conventional keyboards and have a short learning time.

Phone pad and T9 entry The numeric keys on a cellphone can be pressed more than once to enter letters. Most phones have 2 keypad modes: a numeric and an alphabetic mode. Most phones have additional modes for entering (initial) capitals. On modern phones you can also Önd the T9-algorithm. This uses a large dictionary to disambiguate words by typing the relevant letters keys once. 2

Handwriting recognition :Current technology is still fairly inaccurate and makes a lot of mistakes, partly due to the enormous differences between people ís handwriting. HR deals mostly worth stroke information: the way in which the letter is drawn, not the letter itself. Therefore, online recognition is most accurate. HR has the advantage of size and accuracy over small keyboards and are therefore often used in mobile computing.

Speech recognition :The performance of speech recognition is still relatively low, even for a restricted vocabulary. Adjusting the system for use with natural language gives birth to even more problems: different voices, emotions and accents etc. This means the system has to be tuned for each different user. SR can be used in 3 scenarios: as an alternative text entry device, replacing the keyboard in the current software, with new software especially designed for SR and in situations where the use of keyboards is impractical or impossible.

Positioning, pointing and drawing :

The mouse :The mouse is an indirect input device, because a transformation is required to map from the horizontal nature of the desktop to the vertical alignment of the screen. Invented in 1964 by Engelbart, his mouse used 2 wheels that slid across the desktop and transmitted x; y-coordinates to the computer. There have been experiments with foot-controlled mice.

Touchpad :Touchpads are touch-sensitive tablets, operated by sliding the finger over it and are mostly used in notebook computers. Performance can be increased using accelerators.

Trackball and thumbwheel :A trackball is an upside-down mouse: instead of moving the device itself, the ball is rolled to move the cursor. Trackballs are often used by RSI users. Thumbwheels (in 2 dimensions) offer less usability because they can only manipulate the horizontal and vertical movement of the cursor. 1-dimensional thumbwheels are often included on the normal mice the enhance the scrolling.

DISPLAY DEVICES

Joystick and keyboard nipple :There are two types of joysticks: absolute sticks, in which the position of the cursor corresponds to the position of the joystick in its base, and isometric sticks, in which the pressure on the stick (in a certain direction) controls the velocity of the cursor in that direction. Keyboard nipples are tiny joysticks that are sometimes used on notebook computers.

Touch-sensitive screens (touch screens): Touch screens detect the position of the user's finger or stylus on the screen itself and are therefore very direct. They work by having the finger/stylus interrupting a matrix of light beams, making capacitance changes on a grid overlaying the screen or by ultrasonic reflections. It is a direct device: no mapping is required. However the selection of small area's is difficult and intensive use can be tiring.

Stylus and light pen: For more accurate positioning, systems with touch-sensitive surfaces often employ a stylus. An older technology for the same purpose is the light pen, which emits radiation detected by the screen. A difficulty of this and other direct devices is that pointing obscures the display, making it more difficult to use in rapid successions.

Digitizing tablet: A device used for freehand drawing. A resistive tablet detects point contact between two separated conducting sheets. Magnetic, capacitive and electrostatic tablets use special pens. The sonic tablet requires no pad: an ultrasonic sound emitted by the pen is detected by 2 microphones.

Eyegaze: Eyegaze allows you to control the computer by looking at it, while wearing special glasses, head-mounted boxes etc. By tracking a laser beam reflection in the eye, the direction in which the eye is looking is determined. The system needs to be tuned and is very expensive, but also very accurate.

Cursor keys and discrete positioning: For 2D-navigation, cursor keys can sometimes be preferable. The same goes for remote-controls and cell phones.

Display devices

Bitmap displays, resolution and color:A bitmap-base means that the display is made of a fixed number of dots or pixels in a rectangular grid. The color or intensity at each pixel is held by the computer's video card. The more bits per pixel, the more colors/intensities are possible.

DEVICES FOR VIRTUAL REALITY AND 3D INTERACTION

Also is the resolution of the screen: the total number of pixels (in a 4:3-ratio) and the density of the pixels. Anti-aliasing: softening the edges of line segments, blurring the discontinuity and making the juggles less obvious.

Technologies In a CRT-monitor a stream of electrons is emitted from an electron gun, which is than focused and directed by magnetic fields. As the beam hits the phosphor coated screen, the phosphor is excited by the electrons and glows. Flicker can be reduced by increasing the scanning rate or by interlacing, in which odd lines are scanned first, followed by even lines. In LCD ís a thin layer of liquid crystals is sandwiched between two glass plates. External light passes through the top plate and is polarized. This passes through the crystal and is reflected back to the user's eye by the bottom plate. The polarization of each single crystal can be turned electronically.

Large displays and situated displays: There are several types of large displays. Some use gas-plasma technology and usually have a 16:9-ratio. Several smaller screens can also be places together in a video wall. Projectors are possible too, in two variants: projectors with 3 lenses (red, green and blue) can build a full-color image. LCD-projectors have a small screen, through which light is projected on a screen.

Digital paper: Thin flexible material that can be written to electronically, but keeps its contents when removed from the power supply.

Devices for virtual reality and 3D inter action : Positioning in 3D Changing from 2D to VR does not mean going to 3 degrees of freedom, but (sometimes) to 6, because except for moving in 3 dimensions, you can also roll, turn, twist etc. Humans can use a 3D-environment with a 2D-device (mouse). The human mind is therefore capable of handling multiple degrees of indirection. A 3D-input device is the 3D-mouse, which has 6 degrees of freedom: 3 for position (x,y,z), 1 for pitch, yawn and roll. However, sometimes it's better to use a data glove: a lycra glove with fibers laid around the fingers, detecting the joint angles of the fingers and thumb. The position of the head can be tracked using a VR-helmed, which can also display the 3D-world to each eye. With other devices, e.g. special clothing or a modified trampoline, the position and movement of the whole body can be tracked. PHYSICAL CONTROLS, SENSORS AND SPECIAL DEVICES

D displays 3D can be displayed on normal screens using shadows, depth etc. It is also possible to generate the natural stereoscopic images for both eye positions and have them delivered to the eyes using a VR-helmed. Finally, users can enter a VR cave, where the VR world is projected around them. If the VR-system performances too slow, and there is a delay between movement and image, disorientation and sickness may occur.

Physical controls, sensors and special devices

Special displays :Except for CRT and LCD, there are numerous other display devices, e.g. LEDs, gang's, dials and head-up displays.

Sound output: We do not yet know how to utilize sound in a sensible way to achieve maximum effects and information transference in HCI. However by having sounds confirm a right action, we can speed up interaction.

Touch, feel and smell: Force feedback gives different amounts of resistance to an input device depending on the state of the virtual operation. Haptic devices are various forms of force, resistance and texture influencing our physical senses.

Physical controls: Not only the function of controls, but also the physical design is important and needs to suit the situation in which it is used: kitchen equipment, for example, needs controls that can be cleaned easily.

Environment and bio-sensing: There are many sensors in our environment monitoring our behavior. Their measurements range from temperature and movement to the user's emotional state.

Paper: printing and scanning .

Printing : The most common printers nowadays are dot-based. In order of increasing resolution, familiar types are dot-matrix printers, ink-jet printers and laser printers.

MEMORY

Fonts and page description languages :Some printers print ASCII-characters and bitmaps by itself. Many more complex documents are translated into suitable bitmaps by the computer. More sophisticated printers can accept a page description language, e.g. PostScript. The programming-language for printing includes standard-programming constructs, which means that less data has to be sent to the printer in comparison to using a bitmap.

Screen and page: There are many differences (e.g. size, color depth, resolution etc.) between a paper print and a computer monitor, which causes problems when designing WYSIWYG-software. Especially the correct alignment of text (in different fonts) is difficult.

Scanners and optical character recognition: Scanners produce a bitmap image from a hard, original and , scanned image using optical character recognition, transfer a page of text directly into a txt-file. There are 2 kinds of scanners: flat-bed (as in a copied machine) and hand-held (as in a fax machine, however the scanner has to be manually pulled over the paper). Scanners shine a beam of light at the page and record the intensity and color of the reaction. The resolution of the scanner can differ highly between different types.

Memory

RAM and short-term memory (STM): Most current active information is held in the random access memory (RAM). RAM is volatile: contents are lost when the power is turned off. However, there are more expensive or low-power consuming memory techniques that can hold their contents when the power is off. Disks and long-term memory (LTM) There are 2 main techniques used in disks: magnetic disks (floppy, hard disk, tape) and optical disks. (CD-ROM/DVD). In comparison to RAM, the computers LTM is rather slow. Understanding speed and capacity The capacity of RAM is limited and therefore multitask-systems tend to swap background-running programs from RAM to the hard disk. When the program is fully activated it has to be swapped back, which can cause delays (von Neumann bottleneck).

PROCESSING AND NETWORKS :

Compression:

Compression techniques can be used to reduce the amount of storage required for text, bitmaps and video. In text, logical constructions in the sentence can be replaced by a short code. In video, differences between frames can be recorded instead of the whole frames. If fractal compression is used, the quality can even improve in the process.

Storage format and standards: The basic standard for text storage is the ASCII character codes, which assign to each standard printable character and several control characters an internationally recognized 7 bit code. UNICODE is an extended version of this system and can also code for foreign characters. However, this is all unformatted text. All editors which produce formatted texts have their own file format. Also for images there exists a wide range of formats. Methods of access Standard database access is by special key fields with an associated index. The user has to know the key before the system can find the information. Indices on databases are limited due to the storage costs, privacy and security. The users mistakes in searching can be compensated by using forgiving systems, for example by matching a key to a database index which corresponds closely.

Processing and networks :

Effects of finite processor speed :The processing speed of an interactive system can affect the user by being too slow (which can be avoided by using buffers) or too fast. The faults can be functional, in which the program does the wrong action. Slow responses from the system can also cause the so called cursor tracking and icon wars. If the system is too fast, the user will not have enough time to interpret the system ís output.

Limitations on interactive performance :Several factors that can limit the speed of an interactive system. They can be: Computation bound: Make sure the user has an indication of the system's progress. Storage channel bound: Select the best fitting kind of memory and access technique.

Graphics bound: The actual time of graphic operations can differ much from the estimates

Network capacity

PROCESSING AND NETWORKS

Network computing :Networked systems have an effect on interactivity, because the large distances may cause a noticeable delay in response from the system. The actions of other users may also influence your own interaction with the connected computers.

2.3 The interaction

Introduction

There are a number of ways in which the user can communicate with the system: batch input, direct manipulation, virtual reality etc.

Models of interaction ,:

The terms of interaction, Purpose of an interactive system: Aid the user in accomplishing goals from some application domain. Domain: An area of expertise and knowledge in some real-world activity. Tasks: Operations to manipulate the concepts of a domain. Goal: Desired output from a performed task. Intention: Specific action required to meet the goal. Task analyses: Identification of the problem space for the user of an interactive system in terms of domain, goals, intention and tasks. System's language: Core language, describes computational attributes of the domain relevant to the System state. User's language: Task language, describes psychological attributes of the domain relevant to the User state. System: Computerized application.

The execution-evaluation cycle The plan formulated by the user is executed by the computer. When finished, the user evaluates the results and determines the further actions. Both execution and evaluation can be divided into the following subsections: 1. Establishing the goal

FRAMEWORKS AND HCI

1. Forming the intention (more specific than goal) 2. Specifying the action sequence (based on intention) 3. Executing the action 4.Perceiving the system state 5. Interpreting the system state 6. Evaluating the system state with respect to the goals and intentions
   7. Gulf of execution: Difference between the user's formalization of the actions and the actions allowed by the system. 8. Gulf of evaluation: Distance between the physical presentation of the system state and the expectation of the user.
   On the user-side, communication is in task-language and on the system side, in core language. The user's formulation of the desired task needs to be articulated in the input-language. The task is phrased in terms of certain psychological attributes that highlight the important features of the domain for the user which, if mapped clearly onto the input language, simplify the articulation of a task. Direct manipulation can also facilitate the articulation. The responses of the input are translated to stimuli for the system. Once a state transition has occurred

within the system, the execution phase is completed and the evaluation begins by translating the system's responses into stimuli for the output component. Finally, the response from output is translated to stimuli for the user.

Ergonomics: The user side of the interface, covering both input and output and the user's immediate context. Dialog design and interface styles. Presentation and screen design ,

Frequency: The most often used controls can be accessed most easily.

The physical environment of the interaction :The system's design needs to fit the users size, position (sitting/standing), comfort and safety.

Health issues : Physical position. Temperature. Lighting. Noise. Time.

The use of color (guidelines) : Color used in displays should be as distinct as possible and the distinction should not be affected by changes in contrast. Blue should not be used to display critical information. If color is used as an indicator, it should not be the only cue: additional coding information should be included. The colors should correspond to common conventions and user expectations. Color conventions are culture-determined. Ergonomics and HCI Ergonomics contribution to HCI is in determining constraints on the way we design systems and suggesting detailed and specific guidelines and standards. Ergonomic factors are in general well established and understood and are therefore used as the basis for standardizing hardware designs.

Interaction styles : Command line interface CLI provides a means of expressing instructions to the computer directly, using function keys, single characters, abbreviations or whole-word commands. They are flexible (parameters) and can be combined to apply a number of tools to the same data. Commands should be remembered by the user, the CLI offers no ques.

ELEMENTS OF THE WIMP-INTERFACE: 18 Menus: A set of menu options available for the user is displayed on the screen. The user can select an option (recognition!) using either mouse or keyboard. The menus can be presented text-based and graphical.

Ergonomics: The study of the physical characteristics of the interaction. , Arrangement of controls and displays In appropriate placement of controls and displays can lead to inefficiency, frustration and sometimes dangerous situations.

Organization of controls: INTERACTION STYLES

Functional: functionally related controls are grouped together.

Sequential: Controls are organized to reflect the order of their use in a typical interaction.

Natural language :The ambiguity of natural language makes it very hard for a machine to understand. However, systems can be built to understand restricted subsets of a language, which is relatively successful.

Question/answer and query dialog :The user is asked a series of questions and so is led through the interaction step by step. These interfaces are easy to learn and use, but are limited in their functionality and power. Query languages are used to construct queries to retrieve information from a database. They require specifications from the user in a strict syntax.

Form-fills and spreadsheets : Primarily used for data entry but can also be useful in data retrieval applications. Most form-filling interfaces assist the user during the interaction. Spreadsheets are a sophisticated variation of form filling. The user can enter and alter values and formulae in any order and the system will maintain consistency amongst the values displayed, ensuring that all formulae are obeyed. The WIMP interface :Windows, icons, menus and pointers: the default interface style for the majority of computer systems today.

Point-and-click interfaces :The PCI is closely related to the WIMP-style: pointing and clicking are the only actions required to access information.

Three-dimensional interfaces: The simplest technique is where ordinary WIMP elements are given a 3D appearance. A more complex technique uses interfaces with 3D workspaces. The objects displayed are flat, but are displayed in perspective: they shrink when they are further away. The most complex 3D-workspace is virtual reality.

Elements of the WIMP-interface :The elements of the WIMP interfaces are called widgets: the toolkit for interaction between user and system.

ELEMENTS OF THE WIMP-INTERFACE :

Windows: Windows are areas on the screen that behave as if they were independent terminals in their own right: it can contain any information and can be resized or moved around. Some systems allow windows within windows.

Icons :An icon is a small picture used to represent a closed window.

Pointers: The different shapes of the cursor are often used to distinguish modes. Cursors are also used to give information about the systems activity (hour-glass). In essence pointers are nothing more than small bitmap images with a hotspot: the location to which they point.

Menus : A menu presents a choice of operations or services that can be performed by the system at a given time. Menus provide information cues in the form of an ordered list of operations that can be scanned and selected by using the pointer. There are two types: pop-up menus, that represent context-dependent options, and pull-down menus, that are always visible. The right grouping of the menu-items is the most difficult part of designing a menu.

Buttons: Buttons are individual and isolated regions within a display that can be selected by the user to invoke a specific action. Radio buttons are used for selecting one option from a group. When there are multiple options selectable, check boxes are more common.

Toolbars: Mostly equivalent to menus, except for that a toolbar can also hold buttons.

Palettes: Palettes are mechanisms for making the set of possible modes and the active mode visible to the user (collection of icons).

Dialog boxes: Dialog boxes are information windows used by the system to bring the user's attention to some important information, possibly an error or a warning used to prevent a possible error, or as a sub dialog for a very specific task.

INTERACTIVITY

Interactivity : Interactivity is essential in determining the feel of a WIMP environment. In WIMP environments, the user takes the initiative, with many options and many applications simultaneously available. The exceptions to this are the pre emptive parts of the interface, where the system can take the initiative for various reasons (e.g. the need for specific information). In modern systems, preemptive parts should be avoided as much as possible.

The context of the interaction :The presence of other people in a work environment affects the performance of the worker in any task, for example, by competition-behaviour. However, when it comes to acquisition of new skills, the presence of others can inhibit performance (fear of failure). In order to perform well, users must be motivated. If the (computer) system makes it difficult for the user to perform a certain task, he might get frustrated and his productivity could drop. The user may also lose motivation if a system is introduced that does not match the actual requirements of the job to be done. In that case the user will reject the system, be resentful and unmotivated or adapt the intended interaction to his own requirements. A well designed system, however, may also work motivating on the user.

Experience engagement and fun: It is no longer sufficient that users can use a system, they have to want to use it as well.

Understanding experience: In education, there is the zone of proximal development, in which you do things with some support that you cannot do yourself. Learning is optimal in this zone.

Designing experience

Physical design and engagement Designers constraints: Ergonomic Physical Legal and safety Context and environment Aesthetic

EXPERIENCE ENGAGEMENT AND FUN

Economic Fluidity: The extent to which the physical structure and manipulation of the device naturally relate to the logical functions it supports.

Managing value: If we want people to want to use a device or application, we need to understand their personal values. In the development of software we should take into account that the user wants to see the gains from the new technique as soon as possible and not after a long time of using it.

2.4 Paradigms for interaction

Time sharing :Time sharing means that a single computer could support multiple users. The introduction of time sharing meant the end of batch-processing, in which complete jobs processed individually.

Video display units :The earliest applications of display screen images were developed in military applications. However, it took until 1962 to develop Sketchpad, a simulation language for visual models. It demonstrated that computers could be used to create visual models of abstractions.

Programming toolkits :The idea of building components of a computer system that will allow you to rebuild a more complex system is called bootstrapping and has been used to a great extent in all of computing. The power of programming toolkits is that small, well understood components can be composed in fixed ways in order to create larger tools. Once these larger tools become understood, they can continue to be composed with other tools, and the process continues.

Personal computing :As technology progresses, it is now becoming more difficult to distinguish between what constitutes a personal computer or workstation and what constitutes a mainframe. Some examples of the first personal-computing applications are LOGO and NLS.

Window systems and the WIMP interface Humans are capable of doing multiple tasks at the same time. The personal computer needs to be just as flexible in order to be an effective dialog partner. The modern PC is, and by using windows it can present messages to the user in the context of their task, so the user is able to distinguish the messages from different tasks.

The metaphor :Metaphors are used quite successful to teach new concepts in terms of one s which are already understood. This also works with computers: many of the tasks on a computer are presented as metaphors of tasks . However, the metaphor is inadequate for promoting (and even gets in line with) a full understanding of the computer. Furthermore, metaphors portray a cultural bias and therefore it is difficult to create a metaphor that is internationally understood.

Direct manipulation Features: Visibility of the objects of interest. Incremental action at the interface with rapid feedback on all actions. Reversibility of all actions, so that users are encouraged to explore without severe penalties. Syntactic correctness of all actions, so that every user action is a legal operation. Replacement of complex command languages with actions to manipulate directly the visible objects. Psychological approach: model-world metaphor (direct engagement): In a system built on the model-world metaphor, the interface is itself a world where the user can act, and which changes state in response to user actions. The world of interest is explicitly represented and there is no intermediary between user and world. Appropriate use of the model-world metaphor can create the sensation in the user of acting upon the objects and task domains themselves: direct engagement. From the user's perspective, the interface is the system. A consequence of DM is that there is no longer a clear distinction between input and output. Widgets become interaction objects, with input and output. A way to bring DM into practice is through WYSIWYG-interfaces.

PARADIGMS FOR INTERACTION

Language versus action : The interface can be seen as the mediator between the user and the system. The user gives instructions to the interface and it is then the responsibility of the interface to see that those instructions are carried out. The user-system communication is by means of indirect language instead of direct actions. Two meaningful interpretations to this: Users are required to understand how the underlying system functions and the interface as interlocutor need not perform much translation. Users are not required to understand the underlying system: the interface serves a more active role by translating and interpreting the userís input to correct system commands.

Hypertext: Hypertext is based on the memory ex-technique: a storage and retrieval apparatus used to link different texts together. The name hypertext points to the nonlinear structure of the information.

Multi-modality: A multi-modal interactive system is a system that relies on the use of multiple human communication channels. Each different channel for the user is referred to as a modality of interaction. not all systems are multi-modal, however. Genuine multi-modal systems rely to a greater extent on simultaneous use of multiple communication channels for both input and output.

Computer-supported cooperative work: The main distinction between CSCW systems and interactive systems designed for a single user is that designers can no longer neglect the society within which many users operate. CSCW systems are built to allow interaction between humans via the computer and so the needs of the many must be represented in the one product. CSCW can be synchronous (users have to be online at the same time) and asynchronous (users don't have to be online at the same time).

The world wide web: WWW is not the same as internet. Internet is just the connection between different computers. WWW is the graphic top-layer which is very popular for exchanging information in the HTML-markup notation. It even took the introduction of the WWW to make the internet popular and currently the web is one of the major reasons for buying computers.

Agent-based interfaces Software agents perform actions for the user. The major problem is to specify the users task correctly to the user in a suitable language. Some agents use AI.

PARADIGMS FOR INTERACTION :to learn from the user. Some agents have an embodiment: a representation in the interface (e.g. an icon).

Ubiquitous computing :The intention of UC is to create a computing infrastructure that permeates our physical environment so much that we do not notice the computer any longer. On a small scale, this is already put into practice (watches, PDAs etc.), however a major breakthrough will still take some time. Sensor-based and context-aware interaction Sensor based interaction is simply the future-idea of the computer adjusting to our behavior and performing on background using the information gathered from sensors. In context-aware computing the interaction is implicit than in ordinary interface use. The computer or sensor-enhanced environment is using heuristics and other semi-intelligent means to predict what would be useful for the user. CA-applications should follow the principles of appropriate intelligence: Be right as often as possible, and useful when acting on these correct predictions. Do not cause inordinate problems in

in the event of an action resulting from a wrong prediction.

2.5 Interaction design basics:

Introduction: Interaction design is about how the artifact produced is going to affect the way people work: the design of interventions.

What is design? Design: achieving goals within constraints. Goals: the purpose of the design we are intending to produce Constrain: the limitations on the design process by external factors Trade-offs: choosing which goals or constraints can be relaxed so that others can be met.

The golden rule of design Understand your material: computers (limitations, capacities, tools, platforms) and people (psychological, social aspects, human error)

To err is human It is the nature of humans to make mistakes and systems should be designed to reduce the likelihood of those mistakes and to minimize the consequences when mistakes happen.

The central message: the user During design, always concentrate on the user.

The process of design Requirements: Through observations and interviews, the features of the system to be designed are mapped.

.

SCREEN DESIGN AND LAYOUT

Global structure - hierarchical organization

Overall structure of an application: the way the various screens, pages or physical device states link to one another. This can be done using hierarchy: humans tend to be better at using this structure, as long as the hierarchy does not go to deep.

Global structure – dialog

Dialog: the pattern of non-hierarchical interaction occurring when the user performs a certain action, e.g. deleting a file.

Wider still Style issues: we should conform to platform standards Functionality issues: the program should conform to standard functions. Navigation issues: we may need to support linkages between applications

Screen design and layout

Tools for layout Grouping and structure: if things logically belong together, then we should normally visually group them together. Order of groups and items: the order on the screen should follow the natural order for the user. Decoration: decorations can be used to emphasize grouping. Alignment: the proper use of alignment can help the user to find information in lists and columns quickly.

White space: white space can be used to separate blocks, highlight structures etc.

User actions and control For entering information, the same criteria dictate the layout. It is also very important that the interface gives a clear clue what to do. A uniform layout is then helpful. Affordance (things may (by their shape for example) suggest what to do with them) is, sometimes, helpful as well. It is, however, not appropriate to depict a real-world object in a context where its normal affordances do not work.

ITERATION AND PROTOTYPING(HILL-CLIMBING APPROACH, LOCAL & GLOBAL MAXIMA, :

The way of presenting information on screen depends on the kind of information, the technologies available to present it and the purpose for which it is used. We have an advantage when presenting information in an interactive system in that it is easy to allow the user to choose among several representations, thus making it possible to achieve different goals. In an ideal design, the interface is both usable and aesthetically pleasing. However, the looks of the interface should never come to the disadvantage of the usability. This is mostly the case with the excessive use of color and 3D. Localization/internationalization: the process of making software suitable for different cultures and languages.

Iteration and proto typing(hill-climbing approach, local & global maxima,

Formative evaluation: intended to improve designs. Summative evaluation: verify whether the product is good enough. In order for prototyping methods to work, you need to understand what is wrong and how to improve it, and you also need a good starting point. If the design is very complex, it is sometimes wise to start with various alternatives and to drop them one by one during the design process.

HCI in the software process

Introduction

Software engineering: the sub discipline that addressses the management and technical issues of the development of software systems.

Software life cycle: the activities that take place from the initial concept for a software system up until its eventual phasing out and replacement. HCI aspects are relevant within all the activities of the software life cycle.

The software life cycle :

Activities in the life cycle : Requirements specification: capture a description of what the eventual system will be expected to provide. Requirements, formulated in natural language, are translated to a more formal and un ambigious language. Architectural design: how does the system provide the services expected from it. In this part, the system is decomposed into components that can be brought in from existing products or that can be developed from scratch . Detailed design: a refinement of the component description provided by the architectural design, made for each component seperately. Coding and unit testing: implementing the detailed design in an executable programming language and testing the different components. Integration and testing: integrating the di§erent components into a complete system and testing it as a whole. Sometimes also certify the system according to ISO-standards. Maintenance: all the work on the system after the system is released.

USABILITY ENGINEERING

Validation and verification :

Verification (designing the thing right) will most often occur within a single life-cycle activity or between two adjacent activities. Validation of a design (designing the right thing) demonstrates that within the various activities the customer requirements are satisfied. Because verification proofs are between rather formal languages, the proofs are rather formal too. The validation proof, however, is not: there is a gap between the real world and structured design, known as the formality gap. The consequence is, that there is always a certain subjectivity involved with validation. Management and contractual issues In management, the technical view on the software lifecycle is sometime insufficient: a much wider perspective must be adopted which takes into account the marketability of a system, it training needs, the availability of skilled personnel or possible subcontractors, and other topics outside the activities for the development if the isolated system. In managing the development process, the temporal relationship between the various activities is more important, as are the intermediate deliverables which represent the technical content, as the designer must demonstrate to the customer that progress is being made. The technical perspective of the life cycle is described in stages of activity, whereas the managerial perspective is described in temporally bound phrases: input and output of documentation.

Interactive systems and the software life cycle The life cycle for development described above presents the process of design in a somewhat pipeline order. In reality, the actual process is iterative: work in one design activity affects work in any other activity both before or after it in the life cycle. All of the requirements for an interactive system cannot be determined from the start. During the design process, the system is made more usable by having the potential user test the prototypes and observe his behaviour. In order to do this, clear understanding of human task performance and cognitive processes is very important.

Problems with usability engineering :The major feature of usability engineering is the assertion of explicit usability metrics early on in the design process which can be used to judge a system once it is delivered. The problem with usability metrics is that they rely on measurements of very specific user actions in very specific situations. At early stages of design, the designers do not yet have the information to set goals for measured observations. Another problem is that usability engineering provides a means of satisfying usability specifications and not necessarily usability: the usability metrices must be interpreted correctly. Iterative design and prototyping:

Iterative design: a purposeful design process which tries to overcome the inherent problems of incomplete requirement specification by cycling through several designs, incrementally improving upon the final product with each pass. On the technical side, this is described by the use of prototypes. There are 3 main approaches of prototyping: Throw-away: the knowledge gained from the prototype is used in the final design, but the prototype is discarded . Incremental: the final product is released as a series of components that have been prototyped seperately Evolutionary: the prototype is not discarded but serves as a basis for the next iteration of the design Prototypes differ according to the amount of functionality and performance they provide relative to the final product. The importance lies in its projected realism, since they are tested on real users. Since providing realism in prototypes is costly, there are several problems on the management side: Time: prototyping costs time which is taken away from the real design. Therefore, there are rapid-prototyping techniques.

Planning: Non-functional features: some of the most important features, as safety and reliability, cannot be tested using a prototype. Contracts: Prototyping cannot form the basis for a legal contract and must be supported with documentation.

DESIGN RATIONALE

Techniques for prototyping Storyboards: a graphical depiction of the outward appearance of the intended system, without any accompanying system functionality. Limited functionality simulations: Programming support for simulations means a designer can rapidly build graphical and textual interaction objects and attach some behaviour to those objects, which mimics the system's functionality. There are many techniques.

High-level programming support: High-level programming languages allow the programmer to abstract away from the hardware specifics and thinking terms that are closer to the way the input and output devices are perceived as interaction devices. This technique can also be provided by a user interface management system, in which features of the interface can be designed apart from the underlying functionality

Warning about iterative design :First, design decisions made at the beginning of the prototyping process are often wrong and design inertia can be so great as never to overcome an initial bad decision. Second, if a potential usability problem is discovered, it is important to understand and solve the reason for the problem, and not the symptoms of it.

Design Rationale :DR is the information that explains why a computer system is the way it is, including its structural and functional description. The benefits of DR: DR provides a communication mechanism among the members of the design team. DR can capture the context of a design decision in order that a different design team can determine if a similar rationale is appropriate for their product. producing a DR forces the designer deliberate more carefully about design decisions.

2.6 DESIGN RATIONALE

Process-oriented design rationale DR is often represented using the IBIS (issue-based information system), in which a hierarchical process oriented structure is used: A root issue identifies the main problem, and various descendent positions are put forth as potential solutions. The relationship between issue en position is refuted by arguments. The IBBIS can be notated textual and graphical. Design space analysis In this representation, the design space is initially structured by a set of questions representing the major issues of the design. Options provide alternative solutions to the question. Options can evoke criteria and new questions and therefore the entire representation can also be hierarchically visualized in a tree-graph.

Psychological design rationale :The purpose of PDR is to design the natural task-artifact cycle of design activity. When a new system becomes an artifact, further observation reveals that in addition to the required tasks it also supports tasks the designer never intended. Once these new tasks have been understood, they can serve as requirements for future artifacts. The first step in PDR is to identify the tasks that the proposed system will address and to characterize those tasks by questions that the user tries to answer in accomplishing them. For each question, a set of scenarios of user-system behavior is suggested to support the user in addressing the question. The initial system can then be implemented to provide the functionality suggested by the scenarios. Once this system is running, observations of its use and some designer relaxation is used to produce the actual DR for that version of the system. By forcing the designer to document the PDR, it is hoped that he will become more aware of the natural evolution of user tasks and the artifact, taking advantage of how consequences of one design can be used to improve later designs.

Universal design is the process of designing products so that they can be used by as many people as possible in as many situations as possible. Applied to HCI, this means designing interactive systems that are usable by anyone, with any range of abilities, using any technology platform. This can be achieved by designing systems either to have built in redundancy or to be compatible with assistive technologies.

Universal design principles In the late 1990ís a group at North Carolina State University proposed seven general principles of universal design, which give us a framework in which to develop interactive systems.

1. Equitable use: the design is useful to people with a range of abilities and appealing to all. No user is excluded or stigmatized. Wherever possible, access should be the same for all. Where appropraite, security, privacy and safety provision should be available to all. 2. Flexibility in use: the design allows for a range of ability and preference, through choice of methods of use and adaptivity to the user's pace, precision and custom. 3. Simple and intuitive to use, regardless of the users (intellectual/physical) properties. It should provide prompting and feedback as far as possible. 4. Perceptive information: the design should provide effective communication of information regardless of the environmental conditions or the user's abilities. 5. Tolerance for error: minimizing the impact and damage caused by mistakes or unintended behavior.

2.7 MULTI-MODAL INTERACTION

Low physical effort: systems should be designed t be comfortable to use, minimizing physical effort and fatigue.

Size and space for approach and use: the placement of the system should be such that it can be reached and used by any user regardless of body size, posture or mobility.

Multi-modal interaction :Since our daily interaction with the world around us is multi-modal, interaction channels hat use more than 1 sensory channel also provide a richer interactive experience. The use of multiple sensory channels increases the bandwidth of the interaction between human and computer and also makes the interaction look more like a natural human-human interaction.

Sound in the interface There is experimental evidence that the addition of audio confirmation of modes reduces errors. There are 2 types of sound available: speech and non-speech. Speech in the interface Structure of speech :

The English language is made up of 40 phonemes: atomic elements of speech that represent a distinct sound. However, the sounds do not make the language entirely: the alteration of tone and quality in phonemes, prosody, gives additional emotion and meaning to a sentence. Also, the sound of a phoneme is influenced by its preceding phoneme, which is called co-articulation. The result of prosody and co articulation on phonemes can be used to construct a set of allophones, which represent all the different sounds of a language. These can be combined into morphemes: the smallest still meaningful elements of a language, being either words or part of words. Speech recognition Speech recognition has not yet been very successful due to the complexity of language, but also because background noise interferes with the input, the users provide gap-fillers in their speech and different speakers produce different sounds. However, despite it's limitations, speech recognition is becoming available in commercial products. Speech synthesis Speech synthesis has also not yet been very successful, mostly because we are sensitive to variation and intonation in speech which can barely be accomplished by the computer. Also, being, transient, spoken output cannot be reviewed or browsed easily. However, for users with certain visual or speech disabilities, the current techniques already work well.

Un interpreted speech

Speech does not have to be interpreted by a computer to be useful in the interface: recordings of speech can be a useful output.

DESIGNING FOR DIVERSITY

non-speech sound:

Non-speech sounds can often be assimilated more quickly than speech sounds, and are language-independent. It also requires less of the users attention. A disadvantage is that the meaning of the sounds has to be learned. There are two kinds of usable non-speech sounds: sounds that occur naturally in the world (example: Sonic Finder) and using more abstract generated sounds (example: Ear cons).

Touch in the interface:

The use of touch in the interface is known as haptic interaction (cutaneous perception [tactile sensations through the skin] and kinesthetics [the perception of movement and position]). Touch can provide a primary source of information for users with visual impairments and a richer multi-modal experience for sighted users. The main devices are the electronic braille and the force feedback device.

Handwriting recognition Handwriting is mostly captured using a digitizer tablet or electronic paper. Recognition is difficult due to the differences between various person's handwriting. Individually written characters are better recognized than longer strings.

Gesture recognition:

Gesture is user-dependent, subject to variation and co-articulation and therefore difficult to recognize by a computer. The current systems mostly use data-gloves to capture the gestures.

Designing for diversity

Visual impairment: there are two main approaches: the use of sound and the use of touch.

Hearing impairment:

This does not have much influence on the use of a certain interface.

Physical impairment:

For most of this kind of users, the precision required for mouse control is very difficult. This can sometimes be solved by applying speech input, keyboard driver attached to the head.

Speech impairment:

The use of a normal interface is not a problem. To assist the disabled, multimedia systems provide synthetic speech and text based communication- and conference systems.

Dyslexia: to minimize the amount of text the user needs to process, speech input and output can replace reading and writing. Specially designed spelling correction programs can check the user's input.

Autism: Communication and social interaction are major areas of difficulty for people with autism. Because computers interact rather impersonal, people with autism can use them well as a communications medium.

Designing for different age groups :

Older people: the lack of mobility of many elderly might catch their interest in e-mail and instant-messaging. However, due to the disabilities lots of these users have special design needs:

1.designs must be clear, 2.simple and 3. errors.

Children: Especially younger children have special design-needs, not only in software, but also in documentation and in hardware, since they do not often have a well-developed hand-eye coordination and may have trouble using a keyboard.

Designing for cultural differences Different areas of misunderstanding include the different meaning of symbols, the use of gestures and the direction of reading and the universal meaning of colours.

3.CONCLUSIONS

The means by which humans interact with computers continues to evolve rapidly. Human–computer interaction is affected by the forces shaping the nature of future computing. These forces include:

- Decreasing hardware costs leading to larger memory and faster systems
- Miniaturization of hardware leading to portability
- Reduction in power requirements leading to portability
- New display technologies leading to the packaging of computational devices in new forms
- Specialized hardware leading to new functions
- Increased development of network communication and distributed computing
- Increasingly widespread use of computers, especially by people who are outside of the computing profession
- Increasing innovation in input techniques (e.g., voice, gesture, pen), combined with lowering cost, leading to rapid computerization by people previously left out of the "computer revolution."
- Wider social concerns leading to improved access to computers by currently disadvantaged groups

The future for HCI, based on current promising research, is expected to include the following characteristics:

- *Ubiquitous communication*. Computers are expected to communicate through high speed local networks, nationally over
  - wide-area networks, and portably via infrared, ultrasonic, cellular, and other technologies. Data and computational services will be portably accessible from many if not most locations to which a user travels.
  - *High-functionality systems*. Systems can have large numbers of functions associated with them. There are so many systems that most users, technical or non-technical, do not have time to learn them in the traditional way (e.g., through thick manuals).
  - *Mass availability of computer graphics*. Computer graphics capabilities such as image processing, graphics transformations, rendering, and interactive animation are becoming widespread as inexpensive chips become available for inclusion in general workstations and mobile devices.
  - *Mixed media*. Commercial systems can handle images, voice, sounds, video, text, formatted data. These are exchangeable over communication links among users. The separate worlds of consumer electronics (e.g., stereo sets, VCRs, televisions) and computers are partially merging. Computer and print worlds are expected to cross-assimilate each other.

- *High-bandwidth interaction*. The rate at which humans and machines interact is expected to increase substantially due to the changes in speed, computer graphics, new media, and new input/output devices. This can lead to some qualitatively different interfaces, such as virtual reality or computational video.
- *Large and thin displays*. New display technologies are finally maturing, enabling very large displays and displays that are thin, lightweight, and low in power consumption. This is having large effects on portability and will likely enable the development of paper-like, pen-based computer interaction systems very different in feel from desktop workstations of the present.
- *Information utilities*. Public information utilities (such as home banking and shopping) and specialized industry services (e.g., weather for pilots) are expected to proliferate. The rate of proliferation can accelerate with the introduction of high-bandwidth interaction and the improvement in quality of interfaces.

REFERENCES

[1] Card, Stuart K.; Thomas P. Moran; Allen Newell (July 1980). "The keystroke-level model for user performance time with interactive systems". *Communications of the ACM* 23 (7): 396–410. doi:10.1145/358886.358895.

[2] Jump up^ Carlisle, James H. (June 1976). "Evaluating the impact of office automation on top management communication".*Proceedings of the June 7–10, 1976, National Computer Conference and Exposition*. pp. 611–616.doi:10.1145/1499799.1499885. Use of 'human-computer interaction' appears in references

[3] Jump up^ Suchman, Lucy (1987). *Plans and Situated Action. The Problem of Human-Machine Communication*. New York, Cambridge: Cambridge University Press. Retrieved 7 March2015.

[4] Jump up^ Dourish, Paul (2001). *Where the Action Is: The Foundations of Embodied Interaction*. Cambridge, MA: MIT Press.

[5] ^ Jump up to:*a b c* Hewett; Baecker; Card; Carey; Gasen; Mantei; Perlman; Strong; Verplank. "ACM SIGCHI Curricula for Human-Computer Interaction". ACM SIGCHI. Retrieved

[6] Human computer interaction-brief intro
JM Carroll - 2013 - free-knowledge.org

[7] Human-computer interaction: Psychological aspects of the human use of computing GM Olson, JS Olson - Annual review of psychology, 2003 - annualreviews.org

[8]Creating a design science of human-computer interaction
JM Carroll - Interacting with computers, 1993 - Elsevier

[9] Creating a design science of human-computer interaction
JM Carroll - 1992 - Springer