MAHARANA PRATAP GROUP OF INSTITUTIONS
KOTHI MANDHANA, KANPUR
(Approved by AICTE, New Delhi and Affiliated to Dr. AKTU, Lucknow)

Digital Notes

[Department of Computer Science and Engineering]

Subject Name: Human Computer Interaction

Subject Code : RCS073

Course : B. Tech

Branch : CSE

Semester : VII (A1,A2)

Prepared by : Ms P Kamle

Reference No./CS/P Kamle/RCS073/ 4/7

Unit – 1

INTRODUCTION

Human–computer interaction (HCI), alternatively man–machine interaction (MMI) or computer–human interaction (CHI) is the study of interaction between people (users) and computers.

•With today's technology and tools, and our motivation to create really effective and usable interfaces and screens, why do we continue to produce systems that are inefficient and confusing or, at worst, just plain unusable? Is it because:

1.We don't care?
2.We don't possess common sense?
3.We don't have the time?
4.We still don't know what really makes good design?

Human-computer interaction (HCI) is a multidisciplinary field of study focusing on the design of computer technology and, in particular, the interaction between humans (the users) and computers. While initially concerned with computers, HCI has since expanded to cover almost all forms of information technology design.

DEFINITION

•"Human-computer interaction is 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."

GOALS

•A basic goal of HCI is
–to improve the interactions between users and computers
–by making computers more usable and receptive to the user's needs.
•A long term goal of HCI is
–to design systems that minimize the barrier between the human's cognitivemodel of what they want
–to accomplish and the computer's understanding of the user's task

WHY IS HCI IMPORTANT

•User-centered design is getting a crucial role!
•It is getting more important today to increase competitiveness via HCI studies (Norman,1990)
•High-cost e-transformation investments
•Users lose time with badly designed products and services
•Users even give up using bad interface
–Ineffective allocation of resources

What is USER INTERFACE

- User interface, design is a subset of a field of study called *human-computer interaction* (HCI).
- Human-computer interaction is the study, planning, and design of how people and computers work together so that

a person's needs are satisfied in the most effective way.

- HCI designers must consider a variety of factors:

– what people want and expect, physical limitations and abilities people possess,

--how information processing systems work,

– what people find enjoyable and attractive.

– Technical characteristics and limitations of the computer hardware and software must also be considered.

- The *user interface* is to

– the part of a computer and its software that people can see, hear, touch, talk to, or otherwise understand or direct.

- The user interface has essentially two components: input and output.
- *Input* is how a person communicates his / her needs to the computer.

– Some common input components are the keyboard, mouse, trackball, one's finger, and one's voice.

- *Output* is how the computer conveys the results of its computations and requirements to the user.

– Today, the most common computer output mechanism is the display screen, followed by mechanisms that take advantage of a person's auditory capabilities: voice and sound.

- The use of the human senses of smell and touch output in interface design still remain largely unexplored.

- Proper interface design will provide a mix of well-designed input and output mechanisms that satisfy the user's needs, capabilities, and limitations in the most effective way possible.
- The best interface is one that it not noticed, one that permits the user to focus on the information and task at hand, not the mechanisms used to present the information and perform the task.

THE IMPORTANCE OF GOOD DESIGN

With today's technology and tools, and our motivation to create really effective and usable interfaces and screens, why do we continue to produce systems that are inefficient and confusing or, at worst, just plain unusable? Is it because:

- We don't care?
- We don't possess common sense?
- We don't have the time?
- We still don't know what really makes good design?
- But we never seem to have time to find out what makes good design, nor to properly apply it.

After all, many of us have other things to do in addition to designing interfaces and screens.

- So we take our best shot given the workload and time constraints imposed upon us. The result, too often, is woefully inadequate.
- Interface and screen design were really a matter of common sense, we developers would have been producing *almost identical* screens for representing the real world.
- Example bad designs

– Closed door with complete wood

– suggestion : glass door

THE IMPORTANCE OF THE USER INTERFACE

• A well-designed interface and screen is terribly important to our users. It is their window to view the capabilities of the system.
• It is also the vehicle through which many critical tasks are presented. These tasks often have a direct impact on an organization's relations with its customers, and its profitability.
• A screen's layout and appearance affect a person in a variety of ways. If they are confusing and inefficient, people will have greater difficulty in doing their jobs and will make more mistakes.
• Poor design may even chase some people away from a system permanently. It can also lead to aggravation, frustration, and increased stress.

The Benefits of Good Design

• Poor clarity forced screen users to spend one extra second per screen.
– Almost one additional year would be required to process all screens.
– Twenty extra seconds in screen usage time adds an additional 14 person years.

• The benefits of a well designed screen have also been under experimental scrutiny for many years.
– One researcher, for example, attempted to improve screen clarity and readability by making screens less crowded.
– Separate items, which had been combined on the same display line to conserve space, were placed on separate lines instead.
– The result screen users were about 20 percent more productive with the less crowded version.
• Proper formatting of information on screens does have a significant positive effect on performance.
– In recent years, the productivity benefits of well-designed Web pages have also been scrutinized.
• Training costs are lowered because training time is reduced.
• support line costs are lowered because fewer assist calls are necessary.
• Employee satisfaction is increased because aggravation and frustration are reduced.
• Ultimately, that an organization's customers benefit because of the improved service they receive.
• Identifying and resolving problems during the design and development process also has significant economic benefits
• How many screens are used each day in our technological world?
• How many screens are used each day in your organization? Thousands? Millions?
• Imagine the possible savings. Proper screen design might also, of course, lower the costs of replacing "broken" PCs.

A BRIEF HISTORY OF THE HUMAN-COMPUTER INTERFACE

• The need for people to communicate with each other has existed since we first walked upon this planet.
• The lowest and most common level of communication modes we share are movements and gestures.
• Movements and gestures are language independent, that is, they permit people who do not speak the same language to deal with one another.
• The next higher level, in terms of universality and complexity, is spoken language.
• Most people can speak one language, some two or more. A spoken language is a very efficient mode of communication if both parties to the communication understand it.
• At the third and highest level of complexity is written language. While most people speak, not all can write.
• But for those who can, writing is still nowhere near as efficient a means .of communication as speaking.
• In modem times, we have the typewriter, another step upward in communication complexity.
• Significantly fewer people type than write. (While a practiced typist can find typing faster and more efficient than handwriting, the unskilled may not find this the case.)
• Spoken language, however, is still more efficient than typing, regardless' of typing skill level.
• Through its first few decades, a computer's ability to deal with human communication was inversely related to what was easy for people to do.
-- The computer demanded rigid, typed input through a keyboard; people responded slowly using this device and with varying degrees of skill.
-- The human-computer dialog reflected the computer's preferences, consisting of one style or a combination of styles using keyboards, commonly referred to as Command Language, Question and Answer, Menu selection, Function Key Selection, and Form Fill-In.
• Throughout the computer's history, designers have been developing, with varying degrees of success, other human-computer interaction methods that utilize more general, widespread, and easier-to-learn capabilities: voice and handwriting.
--Systems that recognize human speech and handwriting now exist, although they still lack the universality and richness of typed input.

INTRODUCTION OF THE GRAPHICAL USER INTERFACE

• The Xerox systems, Altus and STAR, introduced the mouse and pointing and selecting as the primary human-computer communication method.
• The user simply pointed at the screen, using the mouse as an intermediary.
• These systems also introduced the graphical user interface as we know it a new concept was born, revolutionizing the human-computer interface.

A BRIEF HISTORY OF SCREEN DESIGN

• While developers have been designing screens since a cathode ray tube display was first attached to a computer, more widespread interest in the application of good design principles to screens did not begin to emerge until the early 1970s, when IBM introduced
its 3270 cathode ray tube text-based terminal.
• A 1970s screen often resembled the one pictured in Figure.
It usually consisted of many fields (more than are illustrated here) with very cryptic and often unintelligible captions.

• It was visually cluttered, and often possessed a command field that challenged the user to remember what had to be keyed into it.
• Ambiguous messages often required referral to a manual to interpret.
• Effectively using this kind of screen required a great deal of practice and patience.
• Most early screens were monochromatic, typically presenting green text on black backgrounds.
• At the turn of the decade guidelines for text-based screen design were finally made widely available and many screens began to take on a much less cluttered look through concepts such as grouping and alignment of elements, as illustrated in Figure 1.2.
• User memory was supported by providing clear and meaningful field captions and by listing commands on the screen, and enabling them to be applied, through function keys. Messages also became clearer.
• These screens were not entirely clutter-free, however. Instructions and reminders to the user had to be inscribed on the screen in the form of prompts or completion aids such as the codes PR and Sc.
• Not all 1980s screens looked like this, however. In the 1980s, 1970s-type screens were still being designed, and many still reside in systems today.
• The advent of graphics yielded another milestone in the evolution of screen design, as illustrated in Figure above
While some basic "design principles did not change, groupings and alignment, for example, Borders were made available to visually enhance groupings, and buttons and menus for implementing commands replaced function keys.
• Multiple properties of elements were also provided, including many different font sizes and styles, line thicknesses, and colors.
• The entry field was supplemented by a multitude of other kinds of controls, including list boxes, drop-down combination boxes, spin boxes, and so forth.
• These new controls were much more effective in supporting a person's memory, now simply allowing for selection from a list instead of requiring a remembered key entry.
• Completion aids disappeared from screens, replaced by one of the new listing controls. Screens could also be simplified, the much more powerful computers being able
to quickly present a new screen.
• In the 1990s, our knowledge concerning what makes effective screen design continued
to expand. Coupled with ever-improving technology, the result was even greater improvements in the user-computer screen interface as the new century dawned.

THE POPULARITY OF GRAPHICS

• A graphical screen bore scant resemblance to its earlier text-based colleagues.
• Older text-based screen possessed a one dimensional
• Graphic screens assumed a three-dimensional look.
• Controls appeared to rise above the screen and move when activated.
• Information could appear, and disappear, as needed.
• Text could be replaced by graphical images called icons.

• These icons could represent objects or actions
• selection fields such as radio buttons, check boxes, list boxes, and palettes coexisted with the reliable old text entry field

- More sophisticated text entry fields with attached or dropdown menus of.

- Objects and actions were selected through use of pointing mechanisms.
- Increased computer power.
- User's actions to be reacted to quickly, dynamically, and meaningfully.
- WIMP interface: windows, icons, menus, and pointers.
- Graphic presentation is much more effective than other presentation methods.
- Properly used, it reduces the requirement for perceptual and mental information recoding and reorganization, and also reduces the memory loads.
- It permits faster information transfer between computers and people by permitting more visual comparisons of amounts, trends, or relationships; more compact representation of information;
- Graphics also can add appeal or charm to the interface and permit greater customization to create a unique corporate or organization style.

GRAPHICAL SYSTEMS ADVANTAGES AND DISADVANTAGES

- Reduce the memory requirements .
- More effective use of one's information.
- Dramatically reduce system learning requirements.
- Experience indicates that for many people they have done all these things.

ADVANTAGES

□ Symbols recognized faster than text
□ Faster learning
□ Faster use and problem solving
□ Easier remembering
□ More natural
□ Exploits visual/spatial cues
□ Fosters more concrete thinking
□ Provides context
□ Fewer errors
□ Increased feeling of control

□ Immediate feedback
□ Predictable system responses
□ Easily reversible actions
□ Less anxiety concerning use
□ More attractive
□ May consume less space
□ Replaces national languages
□ Easily augmented with text displays
□ Smooth transition from command language system

DISADVANTAGES

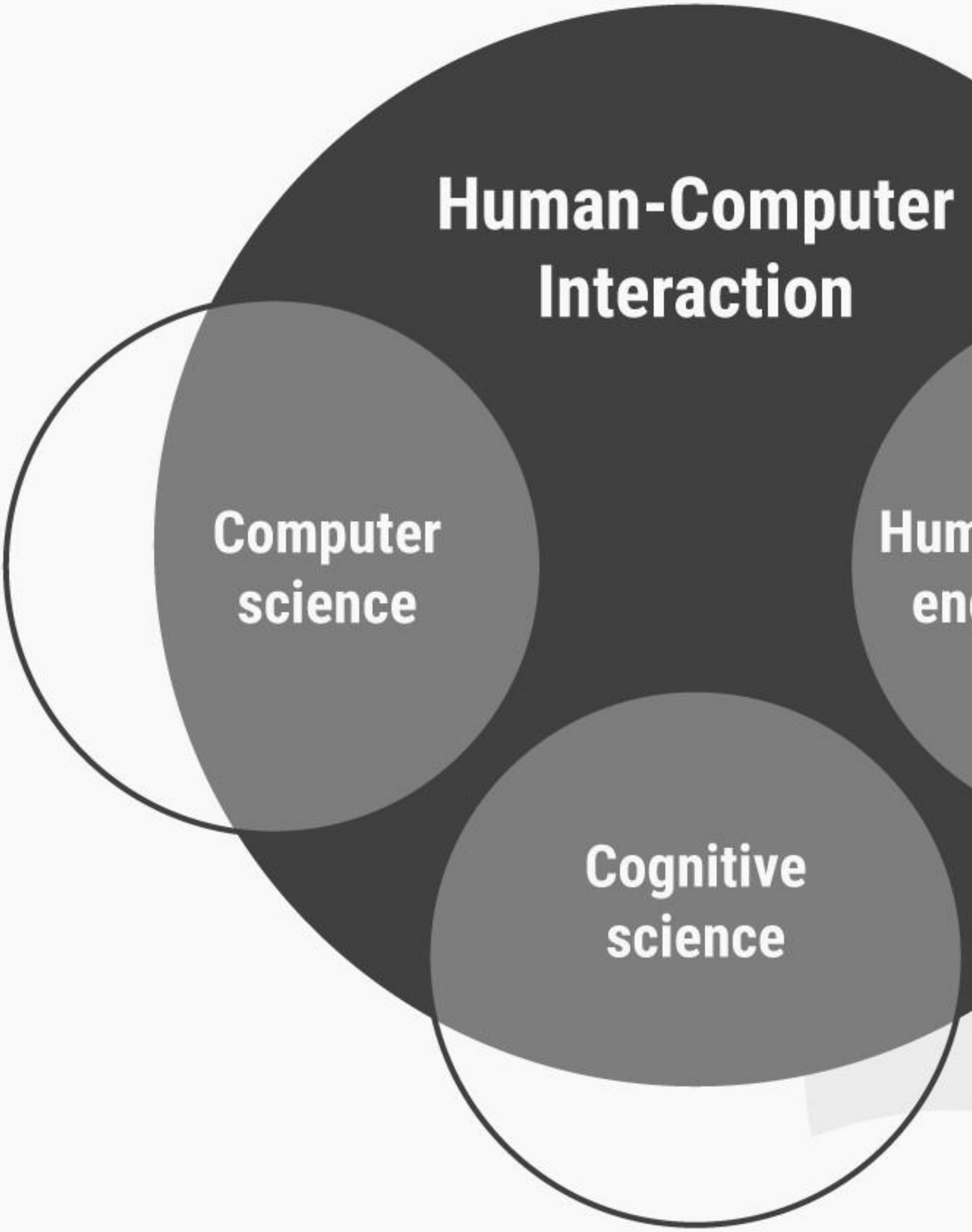
□ Greater design complexity.
□ Learning still necessary
□ Replaces national languages
□ Easily augmented with text displays
□ Smooth transition from command language system

- Lack of experimentally-derived design guidelines
- use a pointing device may also have to be learned
- Working domain is the present
- Human comprehension limitations
- Window manipulation requirements
- Production limitations
- Few tested icons exist
- Inefficient for touch typists
- Inefficient for expert users
- Not always the preferred style of interaction
- Not always fastest style of interaction
- Increased chances of clutter and confusion
- May consume more screen space
- Hardware limitations

What is Human-Computer Interaction (HCI)?

Human-computer interaction (HCI) is a multidisciplinary field of study focusing on the design of computer technology and, in particular, the interaction between humans (the users) and computers. While initially concerned with computers, HCI has since expanded to cover almost all forms of information technology design.**The Meteoric Rise of HCI .**HCI surfaced in the 1980s with the advent of personal computing, just as machines such as the Apple Macintosh, IBM PC 5150 and Commodore 64 started turning up in homes and offices in society-changing numbers. For the first time, sophisticated electronic systems were available to general consumers for uses such as word processors, games units and accounting aids. Consequently, as computers were no longer room-sized, expensive tools exclusively built for experts in specialized environments, the need to create human-computer interaction that was also easy and efficient for less experienced users became increasingly vital. From its origins, HCI would expand to incorporate multiple disciplines, such as computer science, science and human-factors engineering.

The Multidisciplinary Field



Human-Computer Interaction

Computer science

Hum en

Cognitive science

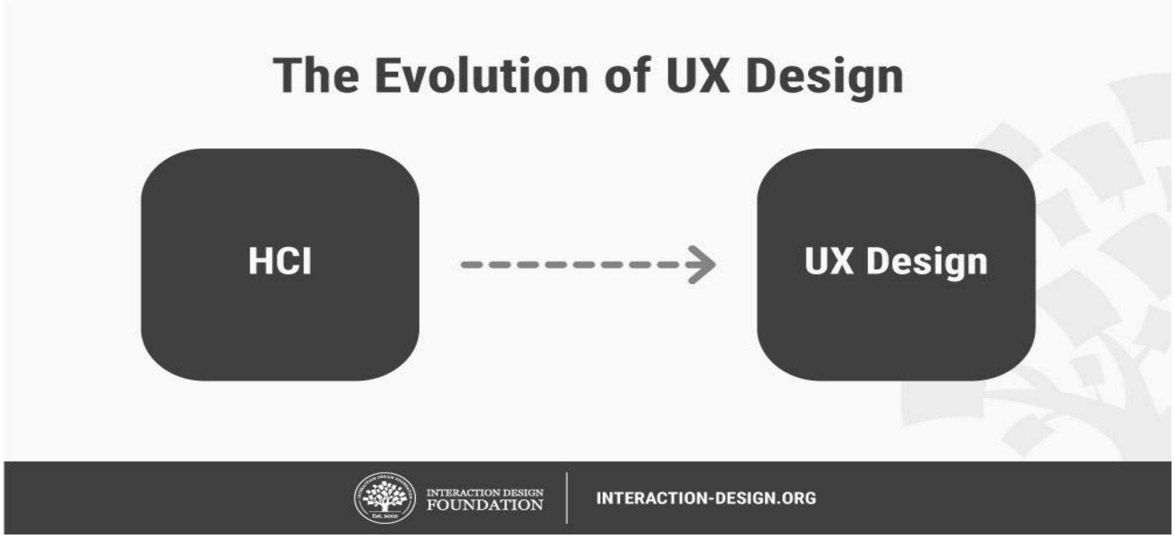
HCI soon became the subject of intense academic investigation. Those who studied and worked in HCI saw it as a crucial instrument to popularize the idea that the interaction between a computer and the user should resemble a human-to-human, open-ended dialogue. Initially, HCI researchers focused on improving the usability of desktop computers (i.e., practitioners concentrated on how easy computers are to learn and use). However, with the rise of technologies such as the Internet and the smartphone, computer use would increasingly move away from the desktop to embrace the mobile world. Also, HCI has steadily encompassed more fields:

"…it no longer makes sense to regard HCI as a specialty of computer science; HCI has grown to be broader, larger and much more diverse than computer science itself. HCI expanded from its initial focus on individual and generic user behavior to include social and organizational computing, accessibility for the elderly, the cognitively and physically impaired, and for all people, and for the widest possible spectrum of human experiences and activities. It expanded from desktop office applications to include games, learning and education, commerce, health and medical applications, emergency planning and response, and systems to support collaboration and community. It expanded from early graphical user interfaces to include myriad interaction techniques and devices, multi-modal interactions, tool support for model-based user interface specification, and a host of emerging ubiquitous, handheld and context-aware interactions."

— John M. Carroll, author and a founder of the field of human-computer interaction.

The UX Value of HCI and Its Related Realms

HCI is a broad field which overlaps with areas such as user-centered design (UCD), user interface (UI) design and user experience (UX) design. In many ways, HCI was the forerunner to UX design.



Despite that, some differences remain between HCI and UX design. Practitioners of HCI tend to be more academically focused. They're involved in scientific research and developing empirical understandings of users. Conversely, UX designers are almost invariably industry-focused and involved in building products or services—e.g., smartphone apps and websites. Regardless of this divide, the practical considerations for products that we as UX professionals concern ourselves with have direct links to the findings of HCI specialists about users' mindsets. With the broader span of topics that HCI covers, UX designers have a wealth of resources to draw from, although much research remains suited to academic audiences. Those of us who are designers also lack the luxury of time which HCI specialists typically enjoy. So, we must stretch beyond our industry-dictated constraints to access these more academic findings. When you do that well, you can leverage key insights into achieving the best designs for your users. By “collaborating” in this way with the HCI world, designers can drive impactful changes in the market and society.

2. Human Computer Interaction - brief intro

BY JOHN M. CARROLL

Human-computer interaction (HCI) is an area of research and practice that emerged in the early 1980s, initially as a specialty area in computer science embracing cognitive science and human factors engineering. 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.

2.1 Where HCI came from

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.

Figure 2.1 A-B: Personal computing rapidly pushed computer use into the general population, starting in the later 1970s. However, the non-professional computer user was often subjected to arcane commands and system dialogs.

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 programme of 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.

Figure 2.2: The Model Human Processor was an early cognitive engineering model intended to help developers apply principles from cognitive psychology.

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.

Introduction

HCI (human-computer interaction) is the study of how people interact with computers and to what extent computers are or are not developed for successful interaction with human beings.

As its name implies, HCI consists of three parts: the user, the computer itself, and the ways they work together.

User

By "user", we may mean an individual user, a group of users working together. An appreciation of the way people's sensory systems (sight, hearing, touch) relay information is vital. Also, different users form different conceptions or mental models about their interactions and have different ways of learning and keeping knowledge and. In addition, cultural and national differences play a part.

Computer

When we talk about the computer, we're referring to any technology ranging from desktop computers, to large scale computer systems. For example, if we were discussing the design of a Website, then the Website itself would be referred to as "the computer". Devices such as mobile phones or VCRs can also be considered to be “computers”.

Interaction

There are obvious differences between humans and machines. In spite of these, HCI attempts to ensure that they both get on with each other and interact successfully. In order to achieve a usable system, you need to apply what you know about humans and computers, and consult with likely users throughout the design process. In real systems, the schedule and the budget are important, and it is vital to find a balance between what would be ideal for the users and what is feasible in reality.

The Goals of HCI

The goals of HCI are to produce usable and safe systems, as well as functional systems. In order o produce computer systems with good usability, developers must attempt to:

- understand the factors that determine how people use technology
- develop tools and techniques to enable building suitable systems
- achieve efficient, effective, and safe interaction
- put people first

Underlying the whole theme of HCI is the belief that people using a computer system should come first. Their needs, capabilities and preferences for conducting various tasks should direct developers in the way that they design systems. People should not have to change the way that they use a system in order to fit in with it. Instead, the system should be designed to match their requirements.

THE CONCEPT OF DIRECT MANIPULATION

• The system is portrayed as an extension of the real world: It is assumed that a person is already familiar with the objects and actions in his or her environment of interest.

• The system simply replicates them and portrays them on a different medium, the screen.
• A person has the power to access and modify these objects, among which are windows.

• A person is allowed to work in a familiar environment and in a familiar way, focusing on the data, not the application and tools.

• The physical organization of the system, which most often is unfamiliar, is hidden from view and is not a distraction.
• Continuous visibility of objects and actions: Like one's desktop, objects are continuously visible. Reminders of actions to be performed are also obvious, labeled buttons replacing complex syntax and command names.

• Cursor action and motion occurs in physically obvious and natural ways. One problem in direct manipulation, however, is that there is no direct analogy on the desk for all necessary windowing operations.

• A piece of paper on one's desk maintains a constant size, never shrinking or growing. Windows can do both. Solving this problem required embedding a control panel, a familiar concept to most people, in a window's border.

• This control panel is manipulated, not the window itself. Actions are rapid and incremental with visible display of results , the results of actions are immediately displayed visually on the screen in their new and current form.

• Auditory feedback may also be provided. The impact of a previous action is quickly seen, and the evolution of tasks is continuous and effortless. Incremental actions are easily reversible.

EARLIER DIRECT MANIPULATION SYSTEMS

• The concept of direct manipulation actually preceded the first graphical system. The earliest full-screen text editors possessed similar characteristics.

• Screens of text resembling a piece of paper on one's desk could be created (extension of real world) and then reviewed in their entirety (continuous visibility).
• Editing or restructuring could be easily accomplished (through rapid incremental actions) and the results immediately seen.

• Actions could be reversed when necessary. It took the advent of graphical systems to crystallize the direct manipulation concept, however.

INDIRECT MANIPULATION

In practice, direct manipulation of all screen objects and actions may not be feasible because of the following:

- The operation may be difficult to conceptualize in the graphical system.

- The graphics capability of the system may be limited.

- The amount of space available for placing manipulation controls in the window border may be limited.

- It may be difficult for people to learn and remember all the necessary operations and actions.

- When this occurs, indirect manipulation is provided. Indirect manipulation substitutes words and text, such as pull-down or pop-up menus, for symbols, and substitutes typing for pointing.

- Most window systems are a combination of both direct and indirect manipulation. A menu may be accessed by pointing at a menu icon and then selecting it (direct manipulation).

- The menu itself, however, is a textual list of operations (indirect manipulation). When an operation is selected from the list, by pointing or typing, the system executes it as a command.

- Which style of interaction-direct manipulation, indirect manipulation, or a combination of both-is best, under what conditions and for whom, remains a question whose answer still eludes us.

CHARACTERISTICS OF THE GRAPHICAL USER INTERFACE

A graphical system possesses a set of defining concepts. Included are sophisticated visual Presentation, pick-and click interaction, a restricted set of interface options, visualization, object orientation, extensive use of a person's recognition memory, and concurrent performance of functions

Sophisticated Visual Presentation:

Visual presentation is the visual aspect of the interface. It is what people see on the screen.

- The sophistication of a graphical system permits displaying lines, including drawings and icons.

- It also permits the displaying of a variety of character fonts, including different sizes and styles.

- The display of 16 million or more colors is possible on some screens. Graphics also permit animation and the presentation of photograph and motion video.

The meaningful interface elements visually presented to the user in a graphical System include windows (primary, secondary, or dialog boxes), menus (menu bar, pull down, pop-up, cascading), icons to represent objects such as programs or files, assorted screen-based controls (text boxes, list boxes, combination boxes, settings, scroll bar and buttons), and a mouse pointer and cursor.

-- The objective is to reflect visually on screen the real world of the user as realistically, meaningfully, simply, and clearly possible.

A graphical system possesses a set of defining concepts. Included are sophisticated visual presentation, pick-and click interaction, a restricted set of interface options, visualization, object orientation, extensive use of a person's recognition memory, and concurrent performance of functions.

Restricted Set of Interface Options: The array of alternatives available to the user is what is presented on the screen or may be retrieved through what is presented on the screen, nothing less, nothing more. This concept fostered the acronym WYSIWYG.

Pick-and-Click Interaction: Elements of a graphical screen upon which some action is to be performed must first identified.
o The motor activity required of a person to identify this element for a proposed action is commonly referred to as pick, the signal to perform an action as cue .
o The primary mechanism for performing this pick-and-click is most often the mouse and its buttons.

o The user moves the mouse pointer to the relevant element (pick) and the action is signaled (click).

o Pointing allows rapid selection and feedback. The hand and mind seem to work smoothly and efficiently together.

o The secondary mechanism for performing these selection actions is the keyboard most systems permit pick-and-click to be performed using the keyboard as well.

Visualization: Visualization is a cognitive process that allows people to understand .Information that is difficult to perceive, because it is either too voluminous or too abstract
Presenting specialized graphic portrayals facilitates visualization.
• The best visualization method for an activity depends on what People are trying to learn from the data.

• The goal is not necessarily to reproduce a really graphical image, but to produce one that conveys the most relevant information.

• Effective visualizations can facilitate mental insights, increase productivity, and for faster and more accurate use of data.

Object Orientation: A graphical system consists of objects and actions. Objects are what people see on screen. They are manipulated as a single unit.
• Objects can be composed of sub objects. For example, an object may be a document. The document's sub objects may be a paragraph, sentence, word, and letter.

• A collection is the simplest relationship-the objects sharing a common aspect.

• A collection might be the result of a query or a multiple selection of objects. Operations can be applied to a collection of objects.
• A constraint is a stronger object relationship. Changing an object in a set affects some other object in the set.

• A document being organized into pages is an example of a constraint. A composite exists when the relationship between objects becomes so significant that the aggregation itself can be identified as an object.

- Examples include a range of cells organized into a spreadsheet, or a collection of words organized into a paragraph.

- A container is an object in which other objects exist. Examples include text in a document or documents in a folder.

A container often influences the behavior of its content. It may add or suppress certain properties or operations of objects placed within it, control access to its content, or control access to kinds of objects it will accept. These relationships help define an object's type. Similar traits and behaviors exist in objects of the same object type.

Another important object characteristic is persistence. Persistence is the maintenance of a state once it is established. An object's state (for example, window size, cursor location, scroll position, and so on) should always be automatically preserved when the user changes it.

Use of Recognition Memory : Continuous visibility of objects and actions encourages use of a person's more powerful recognition memory. The "out of sight, out of mind" problem is eliminated

CONCURRENT PERFORMANCE OF FUNCTIONS

Graphic systems may do two or more things at one time. Multiple programs may run simultaneously. When a system is not busy on a primary task, it may process background tasks (cooperative multitasking).When applications are running as truly separate tasks, the system may divide the processing power into time slices and allocate portions to each application.

Data may also be transferred between programs. It may be temporarily stored on a "clipboard" for later transfer or be automatically swapped between programs.

THE GRAPHICAL USER INTERFACE

- A user interface is a collection of techniques and mechanisms to interact with something.
- In a graphical interface the primary interaction mechanism is a pointing device of some kind.
- This device is the electronic equivalent to the human hand. What the user interacts with is a collection of elements referred to as objects.
- They can be seen, heard, touched, or otherwise perceived.
- Objects are always visible to the user and are used to perform tasks.
- They are interacted with as entities independent of all other objects.
- People perform operations, called actions, on objects. The operations include accessing and modifying objects by pointing, selecting, and manipulating. All objects have standard resulting behaviors.

THE WEB USER INTERFACE

The expansion of the World Wide Web since the early 1990s has been truly amazing. Once simply a communication medium for scientists and researchers, its many and pervasive tentacles have spread deeply into businesses, organizations, and homes around the world.

Unlike earlier text-based and GUI systems that were developed and nurtured in an organization's Data Processing and Information Systems groups, the Web's roots were sown in a market-driven society thirsting for convenience and information.

Web interface design is essentially the design of navigation and the presentation of information. It is about content, not data.

Proper interface design is largely a matter of properly balancing the structure and relationships of menus, content, and other linked documents or graphics. The design goal is to build a hierarchy of menus and pages that feels natural, is well structured, is easy to use, and is truthful.

The Web is a navigation environment where people move between pages of information, not an application environment. It is also a graphically rich environment.

Web interface design is difficult for a number of reasons. First, its underlying design language, HTML, was never intended for creating screens to be used by the general population.

Its scope of users was expected to be technical. HTML was limited in objects and interaction styles and did not provide a means for presenting information in the most effective way for people.

Next, browser navigation retreated to the pre-GUI era. This era was characterized by a "command" field whose contents had to be learned, and a navigational organization and structure that lay hidden beneath a mostly dark and blank screen.

GUIs eliminated the absolute necessity for a command field, providing menus related to the task and the current contextual situation.

Browser navigation is mostly confined to a "Back" and "Forward" concept, but "back-to where" and "forward-towhere" is often unremembered or unknown.

Web interface design is also more difficult because the main issues concern information Architecture and task flow, neither of which is easy to standardize.

It is more difficult because of the availability of the various types of multimedia, and the desire of many designers to use something simply because it is available.

It is more difficult because users are ill defined, and the user's tools so variable in nature.

The ultimate goal of a Web that feels natural, is well structured, and is easy to use will reach fruition.

THE POPULARITY OF THE WEB

While the introduction of the graphical user interface revolutionized the user interface, the Web has revolutionized computing.

It allows millions of people scattered across the globe to communicate, access information, publish, and be heard.

- It allows people to control much of the display and the rendering of Web pages.
- Aspects such as typography and colors can be changed, graphics turned off, and decisions made whether or not to transmit certain data over non secure channels or whether to accept or refuse cookies.
- Web usage has reflected this popularity. The number of Internet hosts has risen dramatically:
- In 1984, hosts online exceeded 1,000;
- in 1987, 10,000;
- in 1989, 100,000,
- in 1990, 300,000;

- in 1992 hosts exceeded one million.
- Commercialization of the Internet saw even greater expansion of the growth rate. In 1993, Internet traffic was expanding at a 341,634 percent annual growth rate. In 1996, there were nearly 10 million hosts online and 40 million connected people (PBS Timeline).
- User control has had some decided disadvantages for some Web site owners as well.
- Users have become much more discerning about good design.
- Slow download times, confusing navigation, confusing page organization, disturbing animation, or other undesirable site features often results in user abandonment of the site for others with a more agreeable interface.
- People are quick to vote with their mouse, and these warnings should not go unheeded.

GUI VERSUS WEB PAGE DESIGN

- GUI and Web interface design do have similarities. Both are software designs, they are used by people, they are interactive, they are heavily visual experiences presented through screens, and they are composed of many similar components.
- Significant differences do exist.

CONCEPT GUI WEB

- User hardware variations limited
- User hardware characteristics well defined.
- Screens appear exactly as specified.
- User hardware variations enormous.
- Screen appearance influenced by hardware being used.

GRAPHICAL USER INTERFACE

- User hardware variations limited
- User hardware characteristics well defined.
- Screens appear exactly as specified.
- Data and applications
- Typically created and used by known and trusted sources.
- Properties generally known.
- Typically placed into system by users or known people and organizations.
- Typically organized in a meaningful fashion.
- A notion of private and shared data exists:

- Install, configure, personalize, start, use, and upgrade programs.
- Open, use, and close data files.
- Fairly long times spent within an application. Familiarity with applications often achieved.
- Controlled and constrained by program.
- Windows, menus, controls, data, tool bars, messages, and so on.
- Many transient, dynamically appearing and disappearing.
- Presented as specified by designer. Generally standardized by toolkits and style guides
- Through menus, lists, trees, dialogs, and wizards. Not a strong and visible concept.
- Constrained by design.
- Generally standardized by toolkits and

- style guides.User Focus • Data and applications • Information and navigation
- Enables maintenance of a better sense of context. Restricted navigation paths.
- Multiple viewable windows Interactions such as clicking menu choices, pressing buttons ,selecting list choices ,and cut/copy/paste occur within context of active program.
- Nearly instantaneous.
- Typically prescribed and constrained by toolkit.
- Visual creativity allowed but difficult.
- Little significant personalization.
- Unlimited capability proportional to sophistication of hardware and software. Targeted to a specific audience with specific tasks. Only limited by the amount of programming undertaken to support it
- Major objective exists within and across applications. Aided by platform toolkit and design guidelines. Universal consistency in GUI products generally created through toolkits and design guidelines.
- Integral part of most systems and applications. Accessed through standard mechanisms. Documentation, both online and offline,
- Usually provided.
- Personal support desk also usually provided
- Seamless integration of all applications into the platform environment a major objective.
- Toolkits and components are key elements in accomplishing this objective
- Tightly controlled in business systems, proportional to degree of willingness to invest resources and effort

WEB

- User hardware variations enormous.
- Screen appearance influenced by hardware being used.
- Information and navigation
- Full of unknown content.
- Source not always trusted.
- Often not placed onto the Web by users or known people and organizations.
- Highly variable organization.
- Privacy often suspect
- Link to a site, browse or read pages, fill out forms, register for services, participate in transactions, download and save things.
- Movement between pages and sites very rapid. Familiarity with many sites not established.
- Infinite and generally unorganized.
- Two components, browser and page.
- Within page, any combination of text, images, audio, video, and animation.
- May not be presented as specified by the designer dependent on browser, monitor, and user specifications.
- Little standardization
- Through links: bookmarks, and typed URLs. Significant and highly visible concept.
- Few constraints ,frequently causing a lost “sense of place”
- Few standards.
- Typically part of page design, fostering an lack of consistency
- Poorer maintenance of a sense of context. Single-page entities.
- Unlimited navigation paths.
- Contextual clues become limited or are difficult to find.
- Basic interaction is a single click. This can cause extreme changes in context, which may not be noticed.

- Quite variable, depending on transmission speeds, page content, and so on. Long times can upset the user
- Fosters a more artistic, individual, and unrestricted presentation style.
- Complicated by differing browser and display capabilities, and bandwidth limitations.
- Limited personalization available.
- Limited by constraints imposed by the hardware, browser, software, client support, and user willingness to allow features because of response time, security, and privacy concerns
- No similar help systems.
- The little available help is built into the page. Customer service support, if provided, oriented to product or service offered.
- Apparent for some basic functions within most Web sites (navigation, printing,and so on.)
- Sites tend to achieve individual distinction rather than integration.
- Susceptible to disruptions caused by user, telephone line and cable providers, Internet service providers, hosting servers, and remotely accessed sites.

PRINCIPLES OF USER INTERFACE DESIGN

- An interface must really be just an extension of a person. This means that the system and its software must reflect a person's capabilities and respond to his or her specific needs.
- It should be useful, accomplishing some business objectives faster and more efficiently than the previously used method or tool did.
- It must also be easy to learn, for people want to do, not learn to do.
- Finally, the system must be easy and fun to use, evoking a sense of pleasure and accomplishment not tedium and frustration.
- The interface itself should serve as both a connector and a separator
- a connector in that it ties the user to the power of the computer, and a separator in that it minimizes the possibility of the participants damaging one another.
- While the damage the user inflicts on the computer tends to be physical (a frustrated pounding of the keyboard), the damage caused by the computer is more psychological.
- Throughout the history of the human-computer interface, various researchers and writers have attempted to define a set of general principles of interface design.
- What follows is a compilation of these principles. They reflect not only what we know today, but also what we think we know today.
- Many are based on research, others on the collective thinking of behaviorists working with user interfaces.
- These principles will continue to evolve, expand, and be refined as our experience with Gills and the Web increases.

GENERAL PRINCIPLES

- The design goals in creating a user interface are described below.
- They are fundamental to the design and implementation of all effective interfaces, including GUI and Web ones.
- These principles are general characteristics of the interface, and they apply to all aspects.
- The compilation is presented alphabetically, and the ordering is not intended to imply degree of importance.

Aesthetically Pleasing

Provide visual appeal by following these presentation and graphic design principles:

- Provide meaningful contrast between screen elements.

- Create groupings.
- Align screen elements and groups.
- Provide three-dimensional representation.
- Use color and graphics effectively and simply.

Clarity

The interface should be visually, conceptually, and linguistically clear, including

- Visual elements
- Functions
- Metaphors
- Words and Text

Compatibility

Provide compatibility with the following:

- The user
- The task and job
- The Product

Adopt the User's Perspective

Configurability

Permit easy personalization, configuration, and reconfiguration of settings.

- Enhances a sense of control
- Encourages an active role in understanding

Comprehensibility

A system should be easily learned and understood: A user should know the following:

- What to look at
- What to do
- When to do it
- Where to do it
- Why to do it
- How to do it

The flow of actions, responses, visual presentations, and information should be in a sensible order that is easy to recollect and place in context.

Consistency

A system should look, act, and operate the same throughout. Similar components should:

- Have a similar look.
- Have similar uses.
- Operate similarly.
- The same action should always yield the same result
- The function of elements should not change.

- The position of standard elements should not change.

Control

The user must control the interaction.

- Actions should result from explicit user requests.
- Actions should be performed quickly.

- Actions should be capable of interruption or termination.
- The user should never be interrupted for errors
- The context maintained must be from the perspective of the user.

- The means to achieve goals should be flexible and compatible with the user's skills, experiences, habits, and preferences.

- Avoid modes since they constrain the actions available to the user.

- Permit the user to customize aspects of the interface, while always providing a Proper set of defaults

Directness

Provide direct ways to accomplish tasks.

- Available alternatives should be visible.
- The effect of actions on objects should be visible.

Flexibility

A system must be sensitive to the differing needs of its users, enabling a level and type of performance based upon:

- Each user's knowledge and skills.
- Each user's experience.
- Each user's personal preference.
- Each user's habits.
- The conditions at that moment.

Efficiency

Minimize eye and hand movements, and other control actions.

- Transitions between various system controls should flow easily and freely.
- Navigation paths should be as short as possible.
- Eye movement through a screen should be obvious and sequential.

Anticipate the user's wants and needs whenever possible.

Familiarity

- Employ familiar concepts and use a language that is familiar to the user.
- Keep the interface natural, mimicking the user's behavior patterns.
- Use real-world metaphors.

Forgiveness

- Tolerate and forgive common and unavoidable human errors.
- Prevent errors from occurring whenever possible.
- Protect against possible catastrophic errors.
- When an error does occur, provide constructive messages.
-

Predictability

- The user should be able to anticipate the natural progression of each task.
- Provide distinct and recognizable screen elements.
- Provide cues to the result of an action to be performed.
- All expectations should be fulfilled uniformly and completely.

Recovery

A system should permit:

- Commands or actions to be abolished or reversed.
- Immediate return to a certain point if difficulties arise.

Ensure that users never lose their work as a result of:

- An error on their part.
- Hardware, software, or communication problems

Responsiveness

The system must rapidly respond to the user's requests Provide immediate acknowledgment for all user actions:

- Visual.
- Textual
- Auditory.

Transparency

Permit the user to focus on the task or job, without concern for the mechanics of the interface.

- Workings and reminders of workings inside the computer should be invisible to the user.

Simplicity

Provide as simple an interface as possible.

Five ways to provide simplicity:

- Use progressive disclosure, hiding things until they are needed
- Present common and necessary functions first
- Prominently feature important functions
- Hide more sophisticated and less frequently used functions.
- Provide defaults.
- Minimize screen alignment points.
- Make common actions simple at the expense of uncommon actions being made harder.
- Provide uniformity and consistency.

UNIT-II

OBSTACLES AND PITFALLS IN DEVELOPMENT PATH

- No body ever gets it right for the first time
- Development is chock full of surprises.
- Good design requires living in a sea of changes.
- Designers need good tools.
- Performance design goals
- People may make mistakes while using a good system also

COMMON PITFALLS

- No early analysis and understanding the users needs and expectations.
- A focus on using design features or components .
- No usability testing.
- No common design team vision.
- Poor communication

COMMON USABILITY PROBLEMS

- Ambiguous menus and icons.
- Languages that permit only single direction movement through a system.
- Input and direct manipulation limits.
- Complex linkage.
- Inadequate feedback.
- Lack of system anticipation.
- Inadequate error messages.

IRRITATING CHARACTERS

- Visual clutter
- Impaired information readability
- Incomprehensible components
- Annoying distractions.
- Confusing navigation.
- Inefficient operations
- Inefficient page scrolling.
- Information overload

DESIGN TEAM

- Development
- Human factors
- Visual Design
- Usability assesment
- Documentation
- Training

HUMAN INTERACTION WITH COMPUTERS

Understanding How People Interact with Computers Characteristics of computer systems, past and present, that have caused, and are causing, people problems. We will then look at the effect these problems have –

- Why people have trouble with computers
- Responses to poor design

- People and their tasks

Why People Have Trouble with Computers

- Extensive technical knowledge but little behavioral training.
- With its extensive graphical capabilities.
- Poorly designed interfaces.
- What makes a system difficult to use in the eyes of its user?
- Use of jargon
- Non-obvious design
- Fine distinctions
- Disparity in problem-solving strategies
- an "error-preventing" strategy
- Design inconsistency

PSYCHOLOGICAL

Typical psychological responses to poor design are:

- Confusion: Detail overwhelms the perceived structure. Meaningful patterns are difficult to ascertain, and the conceptual model or underlying framework cannot be understood or established.
- Annoyance: Roadblocks that prevent a task being completed, or a need from being satisfied, promptly and efficiently lead to annoyance. Inconsistencies in design,
slow computer reaction times, difficulties in quickly finding information, outdated information, and visual screen distractions are a few of the many things that may annoy users.
- Frustration: An overabundance of annoyances, an inability to easily convey one's intentions to the computer, or an inability to finish a task or satisfy a need can cause frustration. Frustration is heightened if an unexpected computer response cannot be undone or if what really took place cannot be determined: Inflexible and unforgiving systems are a major source of frustration.
- Panic or stress: Unexpectedly long delays during times of severe or unusual pressure may introduce panic or stress. Some typical causes are unavailable systems or long response times when the user is operating under a deadline or dealing with an irate customer.
- Boredom: Boredom results from improper computer pacing (slow response times or long download times) or overly simplistic jobs.
- These psychological responses diminish user effectiveness because they are severe blocks to concentration.

--Thoughts irrelevant to the task at hand are forced to the user’s attention, and necessary concentration is impossible.

--The result, in addition to higher error rates, is poor performance, anxiety, and dissatisfaction Physical.

- Psychological responses frequently lead to, or are accompanied by, the following physical reactions.
- Abandonment of the system: The system is rejected and other information sources are relied upon. These sources must, of course, be available and the user must have the discretion to perform the rejection.

In business systems this is a common reaction of managerial and professional personnel. With the Web, almost all users can exercise this option.

- Partial use of the system: Only a portion of the system's capabilities are used, usually those operations that are easiest to perform or that provide the most benefits. Historically, this has been the most common user reaction to most computer systems. Many aspects of many systems often go unused.
- Indirect use of the system: An intermediary is placed between the would-be user and the computer. Again, since this requires high status and discretion, it is another typical response of managers or others with authority.

• Modification of the task: The task is changed to match the capabilities of the system. This is a prevalent reaction when the tools are rigid and the problem is unstructured, as in scientific problem solving.
• Compensatory activity: Additional actions are performed to compensate for system inadequacies. A common example is the manual reformatting of information to match the structure required by the computer. This is a reaction common to workers whose discretion is limited, such as clerical personnel.
• Misuse of the system: The rules are bent to shortcut operational difficulties. This requires significant knowledge of the system and may affect system integrity.
• Direct programming: The system is reprogrammed by its user to meet specific needs. This is a typical response of the sophisticated worker.
• These physical responses also greatly diminish user efficiency and effectiveness. They force the user to rely upon other information sources, to fail to use a system's complete capabilities, or to perform time-consuming "work-around" actions

IMPORTANT HUMAN CHARACTERISTICS IN DESIGN
• Importance in design are perception, memory, visual acuity, foveal and peripheral vision, sensory storage, information processing, learning, skill, and individual differences.
• Perception
• Proximity
• Similarity
• Matching patterns
• Succinctness
• Closure
• Unity
• Continuity
• Balance
• Expectancies
• Context
• Signals versus noise

• Memory: Memory is not the most stable of human attributes, as anyone who has forgotten why they walked into a room, or forgotten a very important birthday, can attest.
• -*Short-term,* or working, memory.
- *Long-term* memory
- Mighty memory
- Sensory Storage

• Mental Models: As a result of our experiences and culture, we develop mental models of things and people we interact with.
• A mental model is simply an internal representation of a person's current understanding of something. Usually a person cannot describe this mental mode and most often is unaware it even exists.
• Mental models are gradually developed in order to understand something, explain things, make decisions, do something, or interact with another person.
• Mental models also enable a person to predict the actions necessary to do things if the action has been forgotten or has not yet been encountered.
• Movement Control : Once data has been perceived and an appropriate action decided upon, a response must be made.
• In many cases the response is a movement. In computer systems, movements include such activities as pressing keyboard keys, moving the screen pointer by pushing a mouse or rotating a trackball, or clicking a mouse button

THE IMPLICATIONS IN SCREEN DESIGN

- Learning: Learning, as has been said, is the process of encoding inlong-term memory information that is contained in short-term memory.
- It is a complex process requiring some effort on our part. Our ability to learn is important-it clearly differentiates people from machines.
- Given enough time people can improve the performance in almost any task. Too often, however, designers use our learning ability as an excuse to justify complex design.
- A design developed to minimize human learning time can greatly accelerate human performance.
- People prefer to stick with what they know, and they prefer to jump in and get started. Unproductive time spent learning is something frequently avoided.
- Skill: The goal of human performance is to perform skillfully. To do so requires linking inputs and responses into a sequence of action.
- The essence of skill is performance of actions or movements in the correct time sequence with adequate precision. It is characterized by consistency and economy of effort.
- Economy of effort is achieved by establishing a work pace that represents optimum efficiency.
- It is accomplished by increasing mastery of the system through such things as progressive learning of shortcuts, increased speed, and easier access to information or data.
- Skills are hierarchical in nature, and many basic skills may be integrated to form increasingly complex ones. Lower-order skills tend to become routine and may drop out of consciousness.
- System and screen design must permit development of increasingly skillful performance.
- Individual Differences: In reality, there is no average user. A complicating but very advantageous human characteristic is that we all differ-in looks, feelings, motor abilities, intellectual abilities, learning abilities and speed, and so on.
- In a keyboard data entry task, for example, the best typists will probably be twice as fast as the poorest and make 10 times fewer errors.
- Individual differences complicate design because the design must permit people with widely varying characteristics to satisfactorily and comfortably learn the task or job, or use the Web site.
- In the past this has usually resulted in bringing designs down to the level of lowest abilities or selecting people with the minimum skills necessary to perform a job.
- But technology now offers the possibility of tailoring jobs to the specific needs of people with varying and changing learning or skill levels. Multiple versions of a system can easily be created.
- Design must provide for the needs of all potential users

HUMAN CONSIDERATIONS IN DESIGN

- The User's Knowledge and Experience

The knowledge possessed by a person, and the experiences undergone, shape the design of the interface in many ways. The following kinds of knowledge and experiences should be identified.

- Computer Literacy - Highly technical or experienced, moderate computer experience, or none
- System Experience - High, moderate, or low knowledge of a particular system and its methods of interaction
- Application Experience - High, moderate, or low knowledge of similar systems
- Task Experience - Other Level of knowledge of job and job tasks
- Systems Use - Frequent or infrequent use of other systems in doing job
- Education - High school, college, or advanced degree
- Reading Level - Less than 5th grade, 5th-12th, more than 12th grade
- Typing Skill - Expert (135 WPM), skilled (90 WPM), good (55 WPM), average (40 WPM), or "hunt and peck" (10 WPM).
- Native Language or Culture- English, another, or several.

JOB/TASK/NEED

- Type of System Use - Mandatory or discretionary use of the system.
- Frequency of Use - Continual, frequent, occasional, or once-in-a-lifetime use of system
- Task or Need importance - High, moderate, or low importance of the task being performed
- Task Structure - Repetitiveness or predictability of tasks being automated, high, moderate, or low
- Social Interactions - Verbal communication with another person required or not required
- Primary Training - Extensive or formal training, self training through manuals, or no training
- Turnover Rate - High, moderate, or low turnover rate for jobholders
- Job Category - Executive, manager, professional, secretary, clerk
- Lifestyle - For Web e-commerce systems, includes hobbies, recreational pursuits, and economic status

PSYCHOLOCICAL CHARCTERISTICS

- Attitude - Positive, neutral, or negative feeling toward job or system
- Motivation - Low, moderate, or high due to interest or fear
- Patience - Patience or impatience expected in accomplishing goal
- Expectations - Kinds and reasonableness
- Stress Level - High, some, or no stress generally resulting from task performance
- Cognitive Style - Verbal or spatial, analytic or intuitive, concrete or abstract.

PHYSICAL CHARACTRISTICS

- Age Young middle aged or elderly
- Gender Male or Female
- Handness Left, right or ambidextrous
- Disabilities Blind, defective vision, deafness, motor handicap

HUMAN INTERACTION SPEEDS

- The speed at which people can perform using various communication methods has been studied by a number of researchers.
- Reading: The average adult, reading English prose in the United States, has a reading speed in the order of 250-300 words per minute. Proof reading text on paper has been found to occur at about 200 words per minute, on a computer monitor, about 180 words per minute.
- One technique that has dramatically increased reading speeds is called Rapid Serial Visual Presentation, or RSVP. In this technique single words are presented one at a time in the center of a screen. New words continually replace old words at a rate set by the reader. For a sample of people whose paper document reading speed was 342 words per minute. (With a speed range of 143 to 540 words per minute.) Single words were presented on a screen in sets at a speed sequentially varying ranging from 600 to 1,600 words per minute. After each set a comprehension test was administered.

READING

- Prose text - 250-300 words per minute.
- Proof reading text on paper - 200 words per minute.
- Proofreading text on a monitor - 180 words per minute.

LISTENING

- Speaking to a computer: 150-160 words per minute.
- After recognition corrections: 105 words per minute.

KEYING

- Typewriter

Fast typist: 150 words per minute and higher
Average typist: 60-70 words per minute

- Computer

Transcription: 33 words per minute
Composition: 19 words per minute

- Two finger typists

Memorized text: 37 words per minute
Copying text: 27 words per minute

- Hand printing

Memorized text: 31 words per minute.
Copying text: 22 words per minute.

UNDERSTAND THE BUSINESS FUNCTION

- Business definition and requirements analysis

--Direct methods
--Indirect methods
--Requirements collection guidelines

- Determining basic business functions

--Developing conceptual modes
--Understanding mental models
--Users new mental model

- Design standards or style guides

--Value of standards and guidelines
--Document design
--Design support and implementation

- System training and documentation

-- Training
--Documentation D

DIRECT METHODS

- Individual Face-to-Face Interview
- Telephone Interview or Survey
- Traditional Focus Group
- Facilitated Team Workshop
- Observational Field Study
- User-Interface Prototyping
- Usability Laboratory Testing
- Card Sorting for Web Sites
- A technique to establish groupings of information for Web sites

INDIRECT METHODS

- MIS Intermediary
- Paper Surveyor Questionnaire
- Electronic Surveyor Questionnaire
- Electronic Focus Group
- Marketing and Sales
- Support Line
- E-Mail or Bulletin Board
- User Group
- Competitor Analyses
- Trade Show
- Other Media Analysis
- System Testing

DETERMINING BASIC BUSINESS FUNCTIONS

- Major system functions are listed and described, including critical system inputs and outputs. A flowchart of major functions is developed. The process the developer will use is summarized as follows:

Gain a complete understanding of the user's mental model based upon:

- The user's needs and the user's profile.
- A user task analysis.
- Develop a conceptual model of the system based upon the user's mental model. This includes:
- Defining objects.

- Developing metaphors.

UNDERSTANDING THE USER'S MENTAL MODEL

- The next phase in interface design is to thoroughly describe the expected system user or users and their current tasks.
- The former will be derived from the kinds of information collected in Step 1 "Understand the User or Client," and the requirements analysis techniques described above.
- A goal of task analysis, and a goal of understanding the user, is to gain a picture of the user's mental model.
- A mental model is an internal representation of a person's current conceptualization and understanding of something.
- Mental models are gradually developed in order to understand, explain, and do something.
- Mental models enable a person to predict the actions necessary to do things if the actions have been forgotten or have not yet been encountered.

PERFORMING A TASK ANALYSIS

- User activities are precisely.
- Task analysis involves breaking down the user's activities to the individual task level.
- Knowing why establishes the major work goals;
- Complete description of all user tasks and interactions.
- Work activities are studied using the techniques just reviewed;
- Direct observation, interviews, questionnaires, or obtaining measurements of actual current system usage.
- Listing of the user's current tasks.
- Another result is a list of objects the users see as important to what they do

DEVELOPING CONCEPTUAL MODELS

- The output of the task analysis is the creation, by the designer, of a conceptual model for the user interface.
- A conceptual model is the general conceptual framework through which the system's functions are presented.
- Such a model describes how the interface will present objects, the relationships between objects, the properties of objects, and the actions that will be performed.
- A conceptual model is based on the user's mental model. Since the term mental model refers to a person's current level of knowledge about something, people will always have them

DEVELOPING CONCEPTUAL MODELS

- Since mental models are influenced by a person’s experiences, and people have different experiences, no two user mental models are likely to be exactly the same.
- Each person looks at the interface from a slightly different perspective. The goal of the designer is to facilitate for the user the development of useful *mental model of the system.*

• This is accomplished by presenting to the user a *meaningful conceptual model of the system* .
• When the user then encounters the system, his or her *existing mental model* will, hopefully, mesh well with the system's conceptual model.
• As a person works with a system, he or she then develops a *mental model of the system.*
• The system mental model the user derives is based upon system's behavior, including factors such as the system inputs, actions, outputs (including screens and messages), and its feedback and guidance characteristics, all of which are components of the conceptual model.
• Documentation and training also playa formative role. Mental models will be developed regardless of the particular design of a system, and then they will be
Modified with experience.
• What must be avoided in design is creating for the user a conceptual model that leads to the creation of a false mental model of the system, or that inhibits the user from creating a meaningful or efficient mental model.

Guidelines for Designing Conceptual Models

• Reflect the user's mental model, not the designer's.
• Draw physical analogies or present metaphors.
• Comply with expectancies, habits, routines, and stereotypes.
• Provide action-response compatibility.
• Make invisible parts and process of a system visible.
• Provide proper and correct feedback.
• Avoid anything unnecessary or irrelevant.
• Provide design consistency.
• Provide documentation and a help system that will reinforce the conceptual model.
• Promote the development of both novice and expert mental models.

Defining Objects

• Determine all objects that have to be manipulated to get work done.
Describe:
- The objects used in tasks.
- Object behavior and characteristics that differentiate each kind of object.
- The relationship of objects to each other and the people using them.
- The actions performed.
- The objects to which actions apply.
- State information or attributes that each object in the task must preserve, display,or allow to be edited.

• Identify the objects and actions that appear most often in the workflow.
• Make the several most important objects very obvious and easy to manipulate

Developing Metaphors

• Choose the analogy that works best for each object and its actions.
• Use real-world metaphors.
• Use simple metaphors.
• Use common metaphors.
• Multiple metaphors may coexist.
• Use major metaphors, even if you can't exactly replicate them visually.
• Test the selected metaphors

Eight Golden Rulesof User Interface Design attributed to Ben Shneiderman.

Ben Shniderman consolidated known tacit knowledge and practice guidelines used by Graphic Interfacedesigners into a set of eight general guidelines for the use of computer science specialist who were being introduced to Visual Graphic designers'work ofinteractive interface for any

software program. There is ample empirical evidence published in HCI literature which collaborates and consolidates the applicability of the eight guide norms. These are more guidelines rather than ‘ rules’. . Words like ‘Principles’, ‘Rules’ have later got added on while describing these general axioms. They are very useful for designers as well as software engineersinvolved in design of interfaces. Using these eight points it is possible to distinguish a good interface design from a bad one especially from the Human- Computer interaction point of view. These have been put forth in a concise and understandable manner by Ben Shneiderman.
It needs to be noted that apart from these eight there are many more similar useful pointers available in HCI and Usability literature. While merely or blindly applying these eight guidelines is not necessarily going to grantee a good interface design , they are useful in heuristic evaluation to identify GUIs that fall out of normal ‘ pattern’ and rate them as good or bad.
The Eight ‘rules’ in point form reproduced from published literature are as follows.

1. Strive for Consistency
2. Cater to Universal Usability
3. Offer Informative feedback
4. Design Dialogs to yield closure
5. Prevent Errors
6. Permit easy reversal of actions
7. Support internal locus of control
8. Reduce short term memory

Explanations & Examples

1. Strive for Consistency

Users need to be able to do the same thing the same way that they have been doing.- every time.
Interfaces need toexhibit ‘consistent’ quality across screens/ applications both visually as well as behaviorally.
Consistency leads to a pattern which is easier to handle cognitively.
Consistency such as ‘similar sequence of actions in similar situations’ makesit easy to learn.
Consistency can be achieved through graphical elements such as fonts, colour, shape, position being consistently same in all menus&screens, across, categories. For a particular software.
For example is the ON button is on the right in the first screen and moves to middle in the third screen then positional inconsistency has occurred.
GUI designers use a background grid to place interactive elements in a consistent and orderly way so as to make them appear both physically as well a visually at the same place across the entire softwarepackage.
In case certain exceptions are required to be made, they should be comprehensible, distinct and limited in number.
(Illustrations)

2. Cater to wide range& type of Users

Universal designstrives to cater to as wide a range of human users of different characteristics (age, culture, educational level, disability) with a single design. While this may not be feasible or possible in all contexts,Shniderman’s rule none the less needs to be followed so as not to leave out taking into consideration a section of users, otherwise competent, who cannot use the interface due to no fault of theirs.
Users are classified as Novice, Intermediate and Experts. Experts tend to use lesser interactions at a faster pace. Abrivations short cuts keys etc are some of the techniques usewd. Interfacses need to cater to all levels of users novice to experts.

3. Offer Informative feedback

Interfaces need tonot just be to communicate but also inform the ‘user ‘in terms of learning & feed -back which tells them that they are proceeding in the right direction. For every action of the user there needs to be a feedback – only then ‘ interaction’ (in HCI) is said to take place. Specific error messages composed in a appositive tone give affirmative feedback without having to feel punitive.

Unless the user gets a feed- back he/she cannot proceed or is unsure of the correctness of the action.

4. Design Dialogs to yield closure

In an interaction - dialogue needs to have a closure which is recognized by the user as end of an action. Sequence of actions need to proceed in a dialogue by engaging the user in a step by step manner. Like in a mathematical expression, every enclosing bracket needs a corresponding closing bracket. So also subsequence of actions needs to be grouped with intermittent closing of each sub group followed finally by a closer action of the group.

Ex: A message at the end og a sequence of events. Of sending a SMS.

Example:

Press ON button
□ Look at the green lamp.
□ If greenglows press push button and yellow lamp will glow.
□ Push 3rd button and continue till green lamp stops glowing.

End of task.

Notice the yellow lamp feed back being not closed?

What happens to it. Did it stop glowing? When or why is it glowing when the task is over ?

Are some of the questions that may arise due to non closure of dialogues.A closed dialogue:
□ Look at the green lamp.
□ If greenglows press 2ndpush button and yellow lamp will glow.
□ Press 3rd button and continue with other 3 buttons till green lamp stops glowing.

5. Prevent Errors

Interfaces need to minimize errors. Human Computer dialogue can be designed to minimize and prevent errors made by users . There cloud be many reasons for users errors but the user himself or herself is not one of them. Users can make errors while interacting with computers as well as while inputting information.

Even if the user makes an error the system needs top be designed to detect it, take corrective or precautionary steps to arrest it. It also needs to offer a way out for recovery. A default system unchanged message needs to be communicated to the user if an error has happened .

6. Permit easy reversal of actions

Interactions need to build in retracing backwards /reverse actions if need be so as give relief from anxiety to the user. The system should encourage exploration without techno fear. One way to do this is to provide a re traceable path backwards of all actions and permit their nullification.

Ex: Word based applications have reversal in both the direction – backwards (last action) and forward (post action)

7. Support internal locus of control
Make aware and allow user to always feel in control of the system and of the situation. User should believe that they are controlling the system and not the other way around. . The bearing of where the user presently is helps the user to orient or reorient the interaction. The user should never allow feeling lost.

8. Reduce short term memory
Care not load the cognitive short term memory of the user by expecting user to remember several sequences , actions and their consequences at a time. Means loading their short term memory while interacting. Millers 7 chunks of information is often prescribed as a solution to limit short term memory. In psychological experiments it has been found that the short term memory can hold 7 +- 2 bits called chunks of information. Long sequential actins requiring more than 7 chunks need to be broken down into smaller chunks.
Example:

Norman's seven principles
Norman proposed that actions are performed in cycles such as (i) establishing a goal ; (ii) Executing the action; (iii) Evaluating the results.
In 1988 Donald Norman proposed seven rules to simplify design work :
He intended them to be used to transform difficult tasks into simple ones.
1. Use both knowledge in world & knowledge in the head
2. Simplify task structures.
3. Make things visible
4. Get the mapping right (User mental model = Conceptual Model = Designed Model)
5. Convert constrains into advantages (Physical constraints, Cultural constraints, Technological constraints)
6. Design for Error
7. When all else fails – Standardize.

Models of Interaction.

Norman's HCI model consists of three types:
User's Mental Model ; System Image Model ; Conceptual Model.
The User's Mental Model is the model of a machine's working that a user creates whenlearning and using a computer. It is not technically accurate. It may also be not stable over time. User's mental models keep changing, evolving as learningcontinues.
In a way Mental Models are models people have of themselves, others and environment.
The mental model of a device is formed by interpreting its perceived actions and is a visiblestructure.

The System image Model is the visible physical part of the computing system / device.
The Conceptual Model. This is the technically accurate model of the computer / device / system created by designers / teachers/researchers. For their specific internal technical use. Users too have a Conceptual model but it is their mental model unless the user is a technically qualified as the evaluator. In a way as far a the user is concerned mental models and conceptual models are inherent to each other. Designer's too have Mental models of the system. So a Conceptual model of the system needs to be as close as possible to the System's Image Model.
A good device / system will emerge when the starting point of the design process is the user-his/her mental model' (in turn derived through user research- task analysis , walk thoughts Contextual inquiry etc) being the basis of the system image and its conceptual model.
The Conceptualisation of the Designer had in his/her mind is called the *design model* .
The User model (what the user develops in the self to explain the operation of the system) and the system image (the system's appearance, operation way it responds is usually a blend of the users mental model and conceptual model all rolled into one. (unless the user happens to be an expert)
Ideally, the design model and user model have to be as close as possible for the systems acceptance. The designer must ensure that the system image is consistent with and operates according to the proper conceptual model.
As a basis for his Interaction Model Norman proposed the following levels of abstraction of knowledge of the user :

Task Level
Goal Level
Semantic level
Syntax level
Lexical level
Physical Level.

Task Level: task level is to analyse the user's needs and to structure the task domain in such a way, that a computer system can play a part in it. The task level describes the structure of the tasks which can be delegated to the computer system.

Semantic level describes the set of objects, attributes, and operations, the system and the user can communicate. Semantics is about how the user interprets it and makes meanings out of the system.

Syntactic level describes which conceptual entities and operations may be referred to in a particular command context or system state.

Interaction level describes the translation of commands and objects into the associated physical actions and the structure of the interaction, including typing / mouse usage rules.

Common Interaction Styles in GUIs:

- Command Line
- Menus
- Natural language (Speech, Gestural,)
- Query- Dialogue (Question Answer)
- Forms (Filling)
- WIMP (Windows- Icons- Menus- Pointer)
- Point & Click
- Three Dimensional
- Virtual Reality

UNIT-III

STRUCTURES OF MENUS

- **Single Menus**

– No other menus will follow necessitating additional user choices

- **Sequential Linear Menus**
- **Simultaneous Menus**

- **Hierarchical Menus**

– When many relationships exist between menu alternatives, and some menu options are only appropriate depending upon a previous menu selection, a hierarchical structure is the best solution.

- **Connected Menus**

– This menu gives you a full control over the navigation flow

- **Event-Trapping Menus**

□ Provide ever-present background of control over the system’s state and parameters while the user is working on a foreground task

– Serve three functions

- Immediately change some parameter in the current environment (bold text)
- Take user out of current environment to perform function (spell check)
- Exit and allow user to go to new environment (exit)

FUNCTIONS OF MENUS

- Navigation to a New Menu
- Execute an Action or Procedure
- Displaying Information
- Data or Parameter Input

CONTENT OF MENUS

- Menu Context

– Provides information to keep the user oriented

- Menu Title

– Provides the context for the current set of choices

- Choice Descriptions:

– Descriptions can range from a mnemonic, numeric or alphabetized listing

- Completion Instructions

– Tell users how to indicate their choices

FORMATTING OF MENUS

- Consistency

– Provide consistency in menu

- Organization, presentation, and choice ordering
- Display

– Frequent references

- Permanently display the menu in an area of the screen that will not obscure other screen data

– Occasional references

- Presentation

– Should be obvious with a unique and consistent structure

- Organization

– Provide a main menu

– Display

- All relevant alternatives (gray-out inactive choices)

– Minimize number of menu levels

– Number of menu choices presented on a screen

- 4-8 choices without logical grouping of elements
- 18-24 choices with logical groupings of elements with no more than 10 items within a group

– Never require menus to be scrolled

- Complexity

– Provide both simple and complex menus

- Item Arrangement

– Orient for top-to-bottom reading

– Left justify descriptions

– Organize for left to right reading

- Ordering

– Numeric order

– Sequence/Frequency of occurrence

– Importance

– Semantic similarity

GROUPINGS

- Create grouping of items that are logical, unique, meaningful and mutually exclusive
- Present no more than six or seven groupings on screen
- Separate grouping created through either

– Wider spacing, or a thin ruled line

- Provide immediate access to critical or frequently chosen items

LINE SEPARATOR

- Separate vertically arrayed grouping with subtle solid lines
- Separate vertically arrayed subgroupings with subtle dotted or dashed lines
- For independent groupings

– Extend the line to the left and right menu borders

PHRASING THE MENU

- Menu Titles: Should be Short, Simple, Distinctive title
- Menu Choice Description:

– Can be single, compound or multiple words

– Use task-oriented not data-oriented wording

– Must never use the same wording as its menu title

– Identical choices on different menus should be worded identically

- Keyboard Accelerators

– Ctrl +B or (Ctrl +B)

- Keyboard Equivalents

– Normal, Bold, Italic

- Intent Indicators

– To a cascade indicator: place a triangle or right pointing solid arrow following the choice

– To a window indicator: place ellipsis (...) immediately follow the choice

SELECTING MENU CHOICES

- Initial Cursor Positioning
- Choice Selection
  - Pointers
  - Keyboards
  - Selection/Execution
  - Combining techniques
- Defaults
  - Provide a default whenever possible (as Bold Text)
- Unavailable Choices
  - Should be dimmed or “grayed out”

MARK TOGGLES OR SETTING

- Purpose
  - Use to designate that an item or feature is active over a relatively long period of time
  - Use to provider a reminder that an item or feature is active or inactive
  - Position the indicator to the left of the option
  - For situations where several nonexclusive choices may be selected, consider including one alternative that deselects all items an reverts the state to the normal condition

Bold old Ctrl+B
Italic Ctrl+I

TOGGLED MENU ITEMS

- Purpose
  - Use to designate two opposite commands that are accessed frequently
  - Use when the menu item displayed will clearly indicate that the opposite condition currently exists
  - Provide a meaningful, fully spelled-out description of action
  - Begin with a clear verb

– Use mixed-case letter view	view
Hide Grid	Show grid

KINDS OF GRAPHICAL MENUS

- Menu Bar
- Pull-Down Bar
- Cascading Menu Bar
- Pop-Up Menu
- Iconic Menu

Menu Bar

- Advantage
  - Always visible
  - Easy to browse
  - Do not obscure the screen working area
  - Allow for use of keyboard equivalents
- Disadvantage
  - Consume a full row of screen space
  - Require looking away from the main working area to find
  - Require moving pointer from the main working area to select
  - Horizontal orientation is less efficient for scanning
- All primary windows must have a menu bar
- All menu bars must have an associated pull-down menu containing at least two choices
- Do not allow the user to turn off the display of the menu bar
- Locate at the top of the screen, just below the screen title
- Use single-word choices whenever possible
- Order choice left-to-right with

– Most frequent choices to left/ related information grouped together
- Help, when included should be located at the right side
- Layout: x File xxx Edit xxx Options Help x

- Separate the bar from the remainder of the screen by

– A different background or Solid lines above and below
- Use reverse color selection cursor to surround the choice

Pull-Down Menu

- Proper Usage

– A small number of items
– Items best represented textually
– Items whose content rarely changes
- Advantages

– No window space is consumed when they r not used
– Allow for display of both keyboard equivalents and accelerators
– Vertical orientation permits more choices to be displayed
- Disadvantage

– Require searching and selecting
– Require moving the pointer out of working area to select
– May obscure the screen working area
- Gray-out or dim items that can not be chosen
- Position the pull-down directly below the selected menu bar choice
- Restrict to no more than 5-10 choices
- Place frequent or critical items at the top
- Multicolumn menus are not desirable
- Alight the first character of the pull-down descriptions under the second character of the applicable menu bar choice
- If a menu item establishes or changes the attributes of data or properties of the interface, mark the pull down choice or choices whose state is current or active "On"
- Grouping:
- Mark Toggles or Setting
- Cascade and Leading to other windows indicator
- Keyboard Equivalents and Accelerators

Cascading Menus

- **Advantage:**

– Top-level menus are simplified because some choices are hidden
– More first-letter mnemonics are available because menus possess fewer alternatives
– High-level command browsing is easier because subtopics are hidden
- **Disadvantage**

– Access to submenu items requires more steps
– Access to submenu items require a change in pointer movement
- Place an arrow or right-pointing triangle to the right of each menu
- Leave the choice leading to the cascading menu highlighted
- Do not exceed three menu levels (two cascades)

Pop Up Menu

- Choices may be also presents alternatives or choices within the context of the task
- Pop-up menus may be requested when the mouse pointer is positioned over a designated or hot area of screen (a window border) or over a designed icon
- **Advantage**

– They do not use window space when not displayed

– They appear in the working area

- **Disadvantage**

– They existence must be learned and remembered

– May obscure the screen working area

– Require a special action to see the menu (Mouse click)

Iconic Menu

- Use to remind user of the functions, commands, application choices
- Create icons that

– Help enhance recognition and hasten option selection

– Meaningful and clearly represent choices

SELECT THE PROPER KINDS OF WINDOWS

A window is an area of the screen that contains a particular view of some area of the computer or some portion of a person's dialog with the computer.

Content

- A window's characteristics
- A window's components
- A window's presentation styles
- The types of windows available
- Organizing window system functions
- A window's operations
- Web system frames and pop-up windows

Window Characteristics

- A name or title, allowing it to be identified
- A size in height and width (which can vary)
- Only active windows can have their contents altered
- A window may be partially or fully hidden behind another window
- Information with in a window may extend beyond window's display area
- Presentation is arranged in relation to other windows (tiled, overlapping, or cascading)
- Methods for manipulation of the window on the screen
- Its highlight, that is, the part that is selected

Windows are useful in the following

- Presentation of Different Levels of Information
- Presentation of Multiple Kinds of Information
- Sequential Presentation of Levels or Kinds of Information
- Access to Different Sources of Information
- Combining Multiple Sources of Information
- Perform More Than One Task
- Reminding
- Monitoring
- Multiple Representations of the Same Task

Components of a Window

- Frame (Border)
- Title Bar
- Title Bar Icon
- Window Sizing Buttons
- What's This Button
- Menu Bar
- Status Bar
- Scroll Bars
- Split Box(Split Bar)
- Toolbar
- Command Area
- Size Grip
- Work Area

Window Presentation Styles

- Tiled Windows

– They are easier, according to studies, for novice or inexperienced people to learn
– Yield better user performance for tasks where the data requires little window manipulation to complete the task
– Only a limited number can be displayed in the screen area available
– As windows are opened or closed, existing windows change in size. This can be annoying
– As the number of displayed windows increases, each window can get very tiny

- Overlapped Windows

– Visually, their look is 3-D, resembling the desktop that is familiar to the user
– Windows can maintain larger sizes
– Windows can maintain consistent sizes, position
– They are operationally much more complex than tiled windows.

More control functions require greater user attention and manipulation

– Windows themselves can be lost behind other windows and be resumed not to exist

- Cascading Windows (Special type of overlapping window)

– No window is ever completely hidden
– Bringing any window to the front is easier
– It provides simplicity in visual presentation and cleanness

Picking a Presentation Style

• Use tiled window for:
– Single task activities
– Data that needs to be seen simultaneously
– Tasks requiring little window manipulation
– Novice or inexperienced users
• Use overlapping windows for:
– Switching between tasks
– Tasks necessitating a greater amount of window manipulation
– Expert or experienced users
– Unpredictable display contents

Type of Windows

• Primary Window
– Should represent an independent function or application
– Use to present constantly used window components and controls
– Use for presenting information that is continually updated (Date and time)
– Often called main window or application window
– Do not divide independent function into two or more primary windows.
• Secondary Windows
– A dependent secondary

• It can only be displayed from a command on the interface of its primary window
– A independent secondary

• Can be opened independently of a primary window (property sheet)
• Microsoft Windows possesses several types of secondary type of secondary windows called
– Dialog boxes
– Property sheet
– Property inspectors
– Message boxes
– Palette windows
– Pop-up windows

• Secondary Windows

Modal and Modeless

• Modal window
– Will not permit interaction with another window until the current dialog is completed
– Remain displayed until the appropriate action is taken after which it is removed
– Modal dialog boxes typically request critical information or actions
• Modeless window
– Switching between the box and its associated is permitted
• Cascading
– To provide advanced options at a lower level in a complex dialog
– Provide a command button leading to the next dialog box with …
– Provide no more than two cascades in a given path
– Don not cover previous critical information
• Relevant information
• Title Bar
• Unfolding
– To provide advanced options at the same level in a complex dialog
– Provide a command button with an expanding dialog symbol >>
– Expand to right or downward
□ Dialog Boxes

• Use for presenting brief messages
• Use for requesting specific, temporary actions

• Use for performing actions that
– Take a short time to complete
– Are not frequently changed
• Usually be those that do not occur frequently
• Command button to include
– OK
– Cancel
– Others as necessary

• Message Boxes
• If a message requires no choices to be made but only acknowledgement, include an ok button and optionally a help menu
• If the message requires the user to make a choice, include a command button for each option
• Include OK and Cancel buttons only when the user has the option of continuing or stopping the action
• Use Yes and No buttons when the user must decide how to continue
• If the choices are too ambiguous, label the command buttons with the names of specific actions, for example, **Save and Delete**

Palette and Pop-Up Windows
• Palette windows are modeless secondary windows that present a set of controls.
• Palette windows are distinguished by their visual appearance, a collection of images, colors or patterns
• The title bar for a palette window is shorter and includes only a close button
• Use pop-up windows to display
– Additional information when an abbreviated form of the information is the main presentation
– Textual labels for graphical controls
– Context-sensitive Help information
– Pop-up windows do not contain standard secondary widow components such as a title bar and close button

Select the Proper Device-Based Controls
Device-based controls, often called input devices, are the mechanisms through which people communicate their desires to the system.

Identify the characteristics and capabilities of device-based control
• Trackball
• Joystick
• Graphic tablet
• Light pen
• Touch screen
• Voice
• Mouse
• Keyboard

Trackball
• Description
– A ball that rotates freely in all directions in its socket
• Advantages
– Direct relationship between hand and pointer movement in terms of direction and speed
– Does not obscure vision of screen
– Does not require additional desk space (if mounted on keyboard)
• Disadvantage
– Movement indirect, in plane different from screen
– Requires hand to be removed from keyboard keys
– Requires different hand movements

– May be difficult to control
– May be fatiguing to use over extended time

Joystick

• Advantages

– Direct relationship between hand and pointer movement in terms of direction and speed
– Does not obscure vision of screen
– Does not require additional desk space (if mounted on keyboard)

• Disadvantage

– Movement indirect, in plane different from screen
– Requires hand to be removed from keyboard keys
– Requires different hand movements
– May be difficult to control
– May be fatiguing to use over extended time
– May be slow and inaccurate.

Graphic (*Touch*) Tablet

• Description

– Pressure-,heat-,light-, or light-blockage-sensitive horizontal surfaces that lie on the desktop or keyboard
– May be operated with fingers, light pen, or objects like pencil

• Advantages

– Direct relationship between hand and pointer movement in terms of direction and speed
– Does not obscure vision of screen
– More comfortable horizontal operating plane

• Disadvantage

– Movement is indirect, in a plane different from screen
– Requires hand to be removed from keyboard
– Requires different hand movements to use
– Finger may be too large fro accuracy with small objects

Touch Screen

• Advantages

– Direct relationship between hand and pointer movement in terms of direction and speed
– Movement is direct, in the same plane as screen
– Requires no additional desk space

• **Disadvantage**

– Finger may obscure part of screen
– Finger may be too large for accuracy with small objects
– Requires moving the hand far from the keyboard to use
– Very fatiguing to use for extended period of time
– May Damage the screen

Light Pen

• Description

– A special surface on a screen sensitive to the touch of a special stylus or pen

• Advantage

– Direct relationship between hand and pointer movement in terms of direction, distance, and speed
– Movement is direct, in the same plane as screen
– Requires minimal additional desk space
– Stands up well in high-use environments
– More accurate than finger touching

• Disadvantage

– Hand may obscure part of screen
– Requires picking it to use

Voice

- Description
  - Automatic speech recognition by the computer
- **Advantage**
  - Simple and direct
  - Useful for people who cannot use a keyboard
  - Useful when the user’s hands are occupied
- **Disadvantage**
  - High error rates due to difficulties in
    - Recognizing boundaries between spoken words
    - Blurred word boundaries due to normal speech patterns
  - Slower throughput than with typing
  - Difficult to use in noisy environment
  - Impractical to use in quiet environment

Mouse

- **Advantage**
  - Direct relationship between hand and pointer movement in terms of direction, distance, and speed.
  - Permit a comfortable hand resting position
  - Selection mechanisms are included on mouse
  - Does not obscure vision of the screen
- **Disadvantage**
  - Movement is indirect, in a plane different from screen
  - Requires hand to be removed from keyboard
  - Requires additional desk space
  - May require long movement distances
  - Requires a degree of eye-hand co ordination

Mouse Usage Guidelines

- Provide a “hot zone” around small or thin objects that might require extremely fine mouse positioning
- Never use double-clicks or double-drags as the only means of carrying out essential operations
- Do not use mouse plus keystroke combinations
- Do not require a person to point at a moving target

Keyboard

- **Advantage**
  - Familiar
  - Accurate
  - Does not take up additional desk space
  - Very useful for
    - Entering text and alphanumeric data
    - Inserting in text and alphanumeric data
    - Keyed shortcuts accelerators
    - Keyboard mnemonics equivalents
- **Disadvantage**
  - Slow for non-touch-typists
  - Slower than other devices in pointing
  - Requires discrete actions to operate
  - No direct relationship between finger or hand movement.

Keyboard Guidelines

- Provide keyboard accelerators
    - Assign single keys for frequently performed, small-scale tasks
    - Use standard platform accelerators
    - Assign Shift-key combinations for actions that extend or are complementary to the actions of key or key combination used without the Shit-key
    - Assign Ctrl-key combinations for
- Infrequent actions
- Tasks that represent larger-scale versions of the task assigned to the unmodified key
- Provide keyboard equivalents
    - Use standard platform equivalents
    - Use the first letter of the item description
    - Provide window navigation through use of keyboard keys

Selecting the Proper Device-Based Control

- Provide keyboard accelerators
    - Assign single keys for frequently performed, small-scale tasks
    - Use standard platform accelerators
    - Assign Shift-key combinations for actions that extend or are complementary to the actions of key or key combination used without the Shit-key
    - Assign Ctrl-key combinations for
- Infrequent actions
- Tasks that represent larger-scale versions of the task assigned to the unmodified key
- Provide keyboard equivalents
    - Use standard platform equivalents
    - Use the first letter of the item description
    - Provide window navigation through use of keyboard keys

Selecting the Proper Device-Based Controls

- Provide keyboards for tasks involving
    - Heavy text entry and manipulation
    - Movement through structured arrays consisting of few discrete objects
- Provide an alternative pointing device for graphical or drawing tasks
    - Mouse: pointing, selecting, drawing, and dragging
    - Joystick: selecting and tracking
    - Trackball: pointing, selecting and tracking
    - Touch screen pointing and selecting
    - Graphic tablet pointing selecting, drawing, and dragging
- Provide touch screens under the following conditions
    - The opportunity for training is minimal
    - Targets are large, discrete and spread out
    - Frequency of use is low
    - Desk space is at a premium
    - Little or no text input requirement exists
- Consider user characteristics and preferences

Provide keyboards for touch typists

- Minimize eye and hand movements between devices

Pointer Guidelines

- The pointer

– Should be visible at all times
– Should contrast well its background
– Should maintain its size across all screen locations and during movement

- Shape of pointer

– Should clearly indicate its purpose and meaning
– Should be constructed of already defined shapes
– Should not be used for any other purpose other than its already defined meaning

- Use only as many shapes as necessary to inform the user about current location and status
- Animation should not distract

Choose the Proper Screen Based Controls

Screen Based controls, often simply called controls and sometimes called widgets. By definitions, they are graphic objects that represent the properties or operations of other objects.

Operable Controls

- Operable controls are those that permit the entry, selection, changing, or editing of a particular value, or cause a command to be performed.

– Buttons
– Text entry/read-only, selection, combination entry/selection
– Specialized controls

Buttons

- Description

– A square or rectangular-shaped control with a label inside that indicates action to be accomplished
– The label may consist of text, graphics, or both

- Command Buttons
- Toolbars

Command Buttons (Usage and Label)

- Use to provide fast access to frequently used or critical commands (for windows with a menu bar)
- Use to provide access to all necessary commands (for windows without a menu bar)
- Use single-word labels whenever possible (Use two –three words for clarity, if necessary)
- Use mixed-case letters with the first letter of each significant label word capitalized.
- Do not number labels
- Center the label within the button borders
- Provide consistency in button labeling across all screens
- Restrict the number of buttons on a window to six or fewer
- Provide as large as button as feasible and maintain consistent button heights and widths

Command Buttons (Location and Layout)

- Buttons exiting a dialog, and usually closing the window, should be positioned horizontally and centered across the lower part of the window
- For a button invokes a dialog or expands the dialog, position it centered and aligned vertically along the right side of the window
- Do not provide alignment with other screen controls. Maintain alignment and spacing only within the buttons themselves
- Position the buttons within windows before locate the other window controls 81

Command Buttons
(Location and Layout)

- If a button has a *contingent* relationship to another control, position it adjacent to the related control
- Buttons found on more than one window should be consistently positioned

Command Buttons (Organization)

- Most frequent actions to the left or top
- Keep related buttons grouped together
- Exception: Buttons containing excessively long labels may be wider
- Windows Recommends

– An affirmative action the left or above
– The default first
– OK and Cancel next to each other
– Help last

Command Buttons (Intent Indicators)

- No intent indicator is necessary, when a button causes an action to be immediately performed
- When a button leads to a cascading dialog, include and ellipsis (…)
- When a button leads to a menu, include a triangle pointing in the direction the menu will appear after the label
- When a button leads to and expanding dialog, include a double arrow (>>)
- When a button has a contingent relationship to another control, include a single arrow pointing at the control

Command Buttons (Expansion and Defaults)

- Gray buttons after Expansion or when not applicable
- When a window is first displayed, provide a default action, if practical
- A default should be the most likely action:

– A confirmation
– An application of the activity being performed
– A positive action such as OK
– If a destructive action is performed (such as a deletion) the default should be Cancel

- Indicate the default action by displaying the buttons with a bold or double border

Command Buttons (Keyboard Equivalents, Accelerators)

- The mnemonic should be the first character of the button's label
- If duplication exists in first characters, use another character in the label
- Designate the mnemonic character by underlining it
- Assign a keyboard accelerator to each button to facilitate keyboard selection

Command Buttons (Scrolling and Button Activation)

- Use buttons to move between multi-page forms, not scroll bars Label buttons Next and Previous
- Highlight the button in some visually distinctive manner when the point is resting on it and the button is available for selection

Toolbars (Usage, Structure and size)

- Provide easy and fast access to most frequently used commands or options across multiple screens
- Provide buttons of equal size
- Create a meaningful and unique icon
- Center the image within the button
- Create a meaningful label
- Provide the smaller size as the default size with a user option to change it

Toolbars (Organization and Location)

- Place the most frequently used actions to the left or the top

- Keep related buttons grouped together
- Separate potentially destructive buttons from frequently chosen selections
- Permit user to reconfigure the button organization
- Position main features and functions bar horizontally across top of window just below menu bar
- Position subtask and sub features bars along sides of window
- Permit the location of the bar to be changed by the user

Toolbars (Active items, Button Activation and Customization)

- Make only currently available toolbar items available
- Temporarily not available items by displaying grayed out
- Highlight the button in some visually distinctive manner when the pointer is resting on it
- Call attention to the button in another visually distinctive manner when it has been activated or pressed
- Permit toolbars to be turned off by user
- Allow the customizing of toolbars

Text Entry/Read-Only Controls (Captions)

For entry boxes

- Place a colon (:) immediately following the caption
- For single fields, caption can be located in front of upper left corner of the box
- For multiple fields, position the caption upper left of the box

For read-only boxes

- If the data field is long or about the same length, center the caption above the displayed text box
- If the data is alphanumeric, short, or quite variable in length, left-justify the caption above the displayed
- If the data field is numeric and variable in length, right justify the caption above the displayed

Text Entry/Read-Only Controls (Fields)

- To visually indicate that it is an enterable field, present the box in a recessed manner
- Present read-only text boxes on the window background
- Break up long text boxes through incorporation of slashes(/), dashes (-), spaces, or common delimiters
- Call attention to text box data through a highlighting technique
- Gray-out temporarily unavailable text boxes

Selection Controls

- Radio Buttons
- Check Boxes
- Palettes
- List Boxes
- List View Controls
- Drop-down/Pop-up List Boxes

Radio Buttons

- A two part control consisting of the following

– Small circles, diamonds, or rectangles

– Choice descriptions

- When a choice is selected

– The option is highlighted

– Any existing choice is automatically un highlighted and deselected

- Purpose

– To set one item from a small set of option (2 to 8)

- For mutually exclusive choices (that is, only can be selected)

• Most useful for data and choices that are
– Discrete
– Small and fixed in number
– Not easily remembered
– Most easily understood when the alternatives can be seen together and compared to one another
– Never change in content
• Do not use
– For commands

Radio Buttons (Defaults and Structure)

• If there is a default selection, designate it as the default and display its button filled in. Else, display all the buttons without setting a dot

• When a multiple selection includes choices, display the buttons in another unique manner, such as gray shadow

• Left-align the buttons and choice descriptions

• A columnar orientation is the preferred unless vertical space on the screen is limited

• Enclose the buttons in a border to visually strengthen the relationship

Radio Buttons (Organization, Related Control)

• Arrange selection in expected order or follow other patterns (frequency of occurrence, sequence of use, or importance)

• Position any control related to a radio button immediately to the right of the choice description. End the label with an arrow

Radio Buttons (Captions)

• Display full spelled out in mixed-case letters, capitalizing the first letter of all significant words
• Columnar orientation
– With a control border, position the caption:
• Upper-left-justified within the border
• Alternatively, to the left of the topmost choice description with (:)
– Without a control border position the caption:
• Left-justified above the choice description with (:)
• Alternatively, the caption may be located to the left of the topmost choice description with (:)
• Horizontal orientation
– Position the caption to the left of the choice
– Alternatively, with a control border, left-justified within the border

Radio Buttons (Keyboard Equivalents and Selection and Indication)

• Assign a keyboard mnemonic to each choice description by underlining the applicable letter in the choice description

• Highlight the selection choice in some visually distinctive way when the cursor’s resting on it

• When a choice is selected, distinguish it visually from the unselected choices

• If there is a default choice, display the selected choice as set in the control

Radio Buttons

Check Boxes

• Each option acts as a switch and can be either “on” or “off”

– When an option is selected, a mark (X) appears within the square box, or the box is highlighted in some other manner

– Otherwise the square is unselected or empty (off)
• Each box can be
– Switched on or off independently
– Used alone or grouped in sets

!!Other properties are similar to the radio button's properties!!

Palettes

• A control consisting of a series of graphical alternatives. The choices themselves are descriptive, being composed of colors, patterns, or images
• To set one of a series of mutually exclusive options presented graphically or pictorially
• Usually consume less screen space than textual equivalents
• Do not use
– Where the alternatives cannot be meaningfully and clearly represented pictorially
– Where words are clearer than images
– Where the choices are going to change
• Create boxes of equal size
• Position the boxes adjacent to, or butted up against another
• A columnar orientation is the preferred manner
• Top to button, Left to right ordering by expected order, frequency of occurrence, sequence of use or alphabetically
• Display it less brightly than the other choices, if a choice is not available
• Highlight the choice in some visually distinctive way when the pointer is resting
• When a choice is selected, distinguish it visually from the unselected choices

List Boxes

• A permanently displayed box-shaped control containing a list of attributes or objects from which
– A single selection is made (mutually exclusive), or
– Multiple selections are made (non-mutually exclusive)
• Unlimited number of choices
• If the list content change, items will be hard to find
• Good for data that are
– Best represented textually
– Not frequently selected
– Large in number
– Fixed in list length
• Clearly and meaningfully describe the choices available
• Present in mixed case
• Left-align into columns
• Require no more than 40 page-downs to search a list
– If more are required, provide a method for using criteria
• Must be long enough to display 6-8 choices
– If it is the major control within a window, the box may be larger
• When box can't made wide enough to display longest entry
– Break the long entries with an ellipsis (…)
– Provide horizontally scrolling
• Order in a logical and meaningful way to permit easy browsing (allow user to change the sort order will be great)
• If a particular choices is not available in the current context, omit , gray or dim it
• Enclose the choices in a box with a solid border
• Use mixed-case
• Preferred position of the control caption is above upper-left
• When a list box is disabled, display its caption as gray out
• Highlight the selection choice when the pointer is resting on

Single-Selection List Boxes
• If presented with an associated text box control
– Position the list box below and as close as possible to the text box
– The list box caption should be worded similarly to the text box caption
– If the related text box and the list box are very close, the caption may be omitted from the list box
• When the list box is first displayed
– Present the currently active choice highlighted or marked with a circle or diamond to the left of the entry
– If a choice has not been previously selected, provide a default choice and display it in the same manner that is used in selecting it

Multiple-Selection List Boxes
• Mark the selected choice with an X or check mark to the left of the entry
• Consider providing a summary list box
– Position it to the right of the list box
– Use the same color for the summary list box
• Consider providing a display-only text control indicating how many choices have been selected
– Position it justified upper-right above the list box
• Provide command buttons for *Select All* and *Deselect All*
• When the list box is first displayed
– Display the currently active choices
– Mark with and X or check mark to the left of the entry

Drop-Down/Pop-up List Boxes
• Unlimited number of choices
• When displayed, all choices may not always be visible, requiring scrolling
• Use drop-down/pop-up when
– Screen space or layout consideration makes radio buttons or single-selection list boxes impractical
– Do not use a drop-down list if it important that all options be seen together.
• Provide a visual cue that a box is hidden by including a downward pointing arrow, or other meaningful image
• !Other properties are the same as List boxes!

Combination Entry/Selection Controls and Other Operable Controls
• Spin Boxes
• Combo Boxes
• Drop-down/Pop-up Combo Boxes
• Slider

Spin Boxes
• A single line field followed by two small, vertically arranged buttons (pointing up and pointing down arrow)
• Selection/entry is made by
– Using the mouse to point at one of directional buttons
– Keying a value directly into field itself
• Consumes little screen spaces
• Useful only for certain kinds of data
• Proper usage for
– For mutually exclusive choices
– Where screen is space is limited
– Small in number
– Infrequently changed, selected

- To reduce the size of potentially long lists, break the listing into subcomponents (break a date into dd mm yy)
- When first displayed, present a default choice in the box
- The spin box should be wide enough to display the longest entry or choice
- Caption is mixed-case letters
- Position the caption to the left of the box

– Alternatively, left-justified above the box

- For numeric values

– Show a larger value using the up arrow

Combo Boxes

- A single rectangular text box entry field, beneath which is a larger rectangular list box (resembling a drop-down list box)
- The text box permits a choice to be keyed within it
- As text is typed into the text box, the list scrolls to the nearest match
- Also, when an item in the list box is selected, that item is placed within the text box

Drop-down/Pop-up combo Boxes

- A single rectangular text box with a small button to the side and an associated hidden list of options
- Selection are made by using the mouse or keyboard
- The information keyed doesn’t not have to match
- Unlimited number of entries and choices
- Flexible, permitting selection or typed entry
- Requiring scrolling
- Proper usage

– Where screen is limited

– For data and choices that are

- Best represented textually
- Frequently changed
- Large in number

Drop-down/Pop-up combo Boxes

- Provide a visual cue that a list box is hidden by including a downward-pointing
- *Other properties are the same as Drop-down/Pop-up List Box!!*

Slider

- A scale exhibiting degrees of a quality on a continuum
- To make a setting when a continuous qualitative adjustment is acceptable
- Spatial representation of relative setting
- Not as precise as an alphanumeric indication
- Proper usage:

– When an object has a limited range of possible settings

– When the range of values is continuous

– When graduations are relatively fine

Custom Controls

- Presentation controls

– Provide details about other screen elements or controls or assist in giving the **screen structure**

- Static Text Fields
- Group boxes
- Column Headings
- ToolTips

- Balloon Tips
- Progress indicators

Task Best Control If screen Space Constraints Exist

□ Mutually Exclusive Radio Buttons Drop-down/Pop-up List Box

□ Not Mutually Exclusive Check Boxes Multiple-Selection List Box

□ Select or Type a Value

□ Text Entry Field

□ Radio Buttons with “Other”

□ Drop-down Combo Box

□ Setting a Value within a Range

□ Spin Button Text Box

Suggested Uses for Graphical Controls
IF: USE:
•Mutually exclusive alternative
•Best represented verbally
•Very limited in number (2 to 8)
AND:
•Typed entry is never necessary
•Content can never change
•Adequate screen space is available
Radio Buttons
OR:
•Typed entry is never necessary
•Content can never change
•Adequate screen space is not available
Drop-down/Pop-up List Box
OR:
•Typed entry may be necessary
•Content can change
•Adequate screen space is available
Combo box
Suggested Uses for Graphical Controls

IF: USE:
OR:
• Type entry may be necessary
•Content can change
•Adequate screen space is not available
Drop-down/Pop-up Combo Box
Suggested Uses for Graphical Controls
IF: USE:
•Mutually exclusive alternative
•Best represented verbally
•Potentially large in number (9 or more)
AND:

•Typed entry is never necessary
•Content can never change
•Adequate screen space is available
Single-Selection List Box
OR:
•Typed entry is never necessary
•Content can never change
•Adequate screen space is not available
Drop-down/Pop-up List Box
OR:
•Typed entry may be necessary
•Content can change
•Adequate screen space is available
Combo box
Suggested Uses for Graphical Controls
IF: USE:
OR:
• Typed entry may be necessary
•Content can change
•Adequate screen space is not available
Drop-down/Pop-up Combo Box
Suggested Uses for Graphical Controls
IF: USE:
•Mutually exclusive alternative
•Best represented graphically
•Content rarely changes
•Small or large number of items
Palette
IF: USE:
•Mutually exclusive alternatives
•Not frequently selected
•Content does not change
•Predictable, consecutive data
•Typed entry sometimes desirable
And:
•Adequate screen space is not available Spin Box
OR:
•Adequate screen space is not available Combo Box
Suggested Uses for Graphical Controls
IF: USE:
•Mutually exclusive alternative
•Continuous data with a limited range of setting
•Value increases/decreases in a well-known, predictable way
•Spatial representation enhances comprehension
Slider
IF: USE:
•Nonexclusive alternatives
•Best represented verbally

•Typed entry is never necessary
•Content can never change
•Adequate screen space is available
And:
•Very limited in number (2 to 8) Check Boxes
OR:
•Potentially large in number (9 or more) Multiple-Selection List Box
• Revision

Write Clear Text and Message

Words

Do not use technical words, made-up words or terms
filespec, abend, or spool, Ungroup or dearchive

- Do not use abbreviations or acronyms

– Always use the fully spelled-out form the first time it is encountered in the interface

- Consider the usage of contradictions or short forms (won't vs will not, un- -ness), Complete words is preferred
- Positive terms (avoid the prefix "ir-" "in-" "dis-" and "un-")
- Simple action words ("Project status listing" → "List")
- Consistency
- Multiple-word phrases are more readable if the entire phrase is on one line
- Abbreviation, mnemonics, and acronyms should not include punctuation

Sentences and Messages

- Brief and simple
- Directly and immediately usable (Should not search through reference)
- Affirmative statement is easier to understand than negative statements
- Active voice is usually easier to understand than passive voice
- Main topic at the beginning
- Use the same grammatical structure for elements of sentences
- Imply that the system is awaiting the user's direction, not that the system is directing the user
- Negative tones or actions, or threats are not very friendly *("Numbers are illegal" vs "Months must be entered by name")*
- Encouraging message would be better than insulting message
- Should remain factual and informative, and should not attempt humor or punishment

Messages

- Screen messages is classified into two categories

- System messages:
- generated by the system to keep the user informed of the system's state and activities
- Instructional messages (prompting message) :
- tell the user how to work with, or complete the screen displayed

System Messages

- Status messages
- Providing information concerning the progress of a lengthy operation
- Usually contains a progress indicator and a short message
- Informational messages (notification messages)
- This kind of message is usually identified by an "I" icon to the left of the message
- Warning messages
- They are usually identified by an "!"
- The user must determine whether the situation is in fact a problem and may be asked to advise the system whether or not to proceed (A deletion request by a user is any action that commonly generates a warning messages)

System Messages

- Critical messages (Action messages)
- Call attention to conditions that require a user action before the system can proceed
- Some products use a "Do Not" symbol while others use a "Stop" sign. An X in a circle used by Microsoft Windows
- Question messages
- A question message asks a question and offers a choice of options for selection
- It is designated by a "?" icon proceeding the message text

Writing Message Box Text

- Title bar: Clearly identify the source of the message
- The name of the object to which it refers
- The name of the application to which it refers
- Do not include an indication of message type
- Use mixed case in the headline style

- Message box: Provide a clear and concise description of the condition of the condition of the condition causing the message box to be displayed
  - Use complete sentences with ending punctuation
  - Show only message box about the cause of condition in single message
  - Make the solution an option offered in the message
  - Use the word “Please” conservatively
- Do not exceed two or three lines
- Center the message text in window
- Include the relevant icon identifying the type of message

Message Box Controls

- Command Buttons:
  - If a message requires no choices to be made, include an *OK* button
  - If a message requires a choice to be made
    - *OK* and *Cancel* buttons only when the user has the option to continue or cancel
    - *Yes* and *No* buttons when the user must decide how to continue
    - If these choices are too ambiguous, label with the name of specific actions
  - If a message describes an interrupted process, provide *Stop* button
  - If a message offer a chance to cancel a process, provide a *Cancel* button
  - If more details about a message must be presented, provide a *Help* button
  - Display only one message box for a specific condition
- Close Box:
  - Enable the title bar Close only if the message includes a *Cancel* button

Instructional Messages

- Provide instructional information at the depth of detail needed by the user
  - Accessing instruction through a *Help* function is the best solution
- Location it at strategic position on the screen
- Display it in a manner that visually differentiates it from other screen elements
- In writing, follow all relevant writing guideline for words, sentences, and messages

ERROR!
PLEASE HIT YOUR BACK BUTTON AND ENTER A SEARCH
THE SEARCH FIELD DID NOT CONTAIN AN ENTRY
PLEASE CLICK THE BACK BUTTON AND TYPE A SEARCH VALUE

Text for Web Pages

- Words

— Avoid using words that are specific to the Web (A few Web-specific terms are “This Web site”, “Click here” and “Follow this link”)

— A good test of this guideline is to print out a page, read it, and see if it makes as much sense on paper as it does on screen

- Error Messages

— Provide helpful error messages for:

- Incomplete or incorrectly keyed, entered, or selected data
- Requests for documents that do not exist or cannot be found

— Present them in a visually distinctive and noticeable manner

- Instructions

— Make sure instruction are detailed enough to be understood without being specific to one browser version or brand

— Don’t use “Return To”

— Describe where an “Up” button leads where the user will go

Text for Web Pages

- Presentation

— Provide text that contrasts highly with the background

- Writing

— Write objectively

— Use the inverted pyramid organization

— Be concise, using only about half the number of words of conventional text

— Each paragraph should be short and Contain only one main idea

— Make text more scannable by using bulleted listings, tables, headings and bold types

— Too many links within text can disrupt reading continuity and content understanding

— Place them at the beginning or end of paragraphs or section of text

— Test for readability by printing out text to carefully proofread it

Links

- Create wording that make link that user can predict where the link leads

  - Descriptive
  - Differentiable
  - Predictive
- Create links that are brief and to the point, avoiding wordiness
- Write text containing embedded links as if there were no links in it
  - Choose the most relevant words or phrase as the active link
  - Link must no t be continued over two lines
- Standalone links should not exceed one sentence in length
  - Paragraph with embedded links are sometimes useful for a variety of reasons
  - Paragraph with embedded links are sometimes useful for a variety of reasons
  - Embedded Links

Paragraph with embedded links are sometimes useful for a variety of reasons

Link Titles

- A short explanation of a link before the user selects the link
- Provide link titles that describes
  - The name of site the link will lead to
  - The name of subsection the link will lead to
  - The kind of information to be found at the destination
  - Warning about possible problems to be encountered at the other end
- Restrict them to no more than 60 characters

Page Title

- Should contain many keywords at possible
- Provide a page title
  - That possess meaningful keywords
  - Whose first word is its most important descriptor
  - That makes sense when viewed completely out of context
  - That is different from other page titles
  - Is written in mixed case using the headline
- Do not highlight keywords

Heading and Headlines

- Used to scan to find screen content of interest.
- Their wording must provide a strong clue as to the content they relate to
- Heading should be descriptive and straightforward
- No clever, cute or funny headline
- Skip leading articles (the and a)

Some Words to Forget

Create Meaningful Graphics, Icons and Images

Creating Images

- Create familiar and concrete shapes
- Create visually and conceptually distinct shapes
  - Incorporate unique features of an object
  - Do not display within a border
- Clearly reflect object represented
- Simple reflect object represented, avoiding excessive detail
- Create as a set, communicating relationships to one another through common shapes
- Provide consistency in icon type
- Create shapes of the proper emotional tone

Creating Images

- Create familiar and concrete shapes
- Create visually and conceptually distinct shapes
  - Incorporate unique features of an object
  - Do not display within a border
- Clearly reflect object represented
- Simple reflect object represented, avoiding excessive detail
- Create as a set, communicating relationships to one another through common shapes

- Provide consistency in icon type
- Create shapes of the proper emotional tone

Icons

- Icons are most often used to represent objects and actions with which users can interact
- Icons may stand alone on a desktop or in a window, or be grouped together in a toolbar
- A secondary use of a icon is to reinforce important information, a warning icon in a dialog message box

Characteristics of Icons

- **Syntactics** refers to a icon's physical structure

– Shape, Color, Size

– Similar shapes and colors can be used to classify a group of related icons

- **Semantics** is the icon's meaning

– What does it refer – a file, a waste basket, or some other objects?

- **Pragmatics** is how the icons are physically produced and depicted

– Is the screen resolution sufficient to illustrate ?

- Syntactics, semantics and pragmatics determine an icon's effectiveness and usability

Influences on Icon Usability

- Provide icons that are

– Familiar

– Clarity

– Simple

– Consistent

– Directness of the meaning

– Efficient

– Discriminable from others

- Also consider the

– Context in which the icon is used

- Expectancies of users
- Complexity of task

Choosing Icons

- A Successful Icon
  - Looks different from all other icons
  - Is obvious what it does or represents
  - Is recognizable when no larger than 16 pixels square
  - Look as good in black and white as in color
- Size
  - 16x16, 24x24, 26x26, 32x32 pixels 16-and-256 color version
  - Use colors from the system palette
- Provide as large a hot zone as possible
  - With stylus or pen: 15 pixels square
  - With mouse: 20 pixels square
  - With finger: 40 pixels square

Choosing Images

- Use existing icons when available
- Use images for nouns, not verbs
- Use traditional images
- Consider user cultural and social norms

Creating Images

- Create familiar and concrete shapes
- Create visually and conceptually distinct shapes
  - Incorporate unique features of an object
  - Do not display within a border
- Clearly reflect object represented
- Simple reflect object represented, avoiding excessive detail

- Create as a set, communicating relationships to one another through common shapes
- Provide consistency in icon type
- Create shapes of the proper emotional tone

Creating Images

- Create familiar and concrete shapes
- Create visually and conceptually distinct shapes

– Incorporate unique features of an object

– Do not display within a border

- Clearly reflect object represented
- Simple reflect object represented, avoiding excessive detail
- Create as a set, communicating relationships to one another through common shapes
- Provide consistency in icon type
- Create shapes of the proper emotional tone

Drawing Images

- Providing consistency in shape over varying sizes
- Do not use triangular arrows in design to avoid confusion with other system symbols
- When icons are used to reflect varying attributes, express these attributes as meaning meaningfully as possible
- Provide proper scale and orientation
- Use perspective and dimension whenever possible
- Accompany icon with a label to assure intended meaning

Icon Animation and Audition

- Animation

– Use

- To provide feedback
- For visual interest

  - Make it interruptible or independent of user’s primary interaction
  - Do not use it for decoration
  - Permit it to be turned off by the user
  - For fluid animation, present images at 16++ frames /second
- Auditions
  - Consider auditory icons

The design Process

- Define the icon’s purpose and use
- Collect, evaluate, and sketch ideas
- Draw in black and white
- Draw using an icon-editing utility or drawing package
- Test for users
  - Expectations
  - Recognition
  - Learning
- Test for clarity
- Register new icons in the system’s registry

Graphics in Web

- Use Graphics to
  - Supplements the textual content, not as a substitute for it
  - Convey information that can’t be effectively accomplished using text
  - Enhance navigation through
    - Presenting a site overview
    - Identifying site pages
    - Identifying content areas
  - Limit the use of graphics that take long time to load
  - Coordinate the graphics with all other page elements

Images

- Use standard images, image internationalization
- Provide descriptive text or labels with all images
- Distinguish navigational images from decorative images
- Minimize

— The number of presented images

— The size of presented images

— Image animation

— Number of colors

- GIF, JPEG is prefer

Photographs/Pictures

- Use when every aspect of the images is relevant
- Use JPEG format
- On the initial page

— Display a small version

- A thumbnail
- Zoom-in on most relevant detail

— Link to larger photos showing as much detail as needed

Video

- To show the proper way to perform a task
- To provide a personal message
- To grab attention
- Never automatically download a video into a page
- Provide controls (playing, pausing, and stopping)

- Considering using

– Existing video

– Audio only

– A slide show with audio

Diagrams

- To show the structure of objects
- To show the relationship of objects
- To show the flow of a process or task
- To reveal a temporal or spatial order

Animation

- To explain ideas involving a change in

– Time

– Position

- To illustrate the location or state of a process
- To show continuity in transitions
- To enrich graphical representations
- To aid visualization of 3-D structures
- Provide a freeze frame and stop mode
- Avoid distracting animation

Audition

- Uses as a supplement to text and graphics
- To establish atmosphere
- To create a sense of place
- To teach
- To sample

- The content should be simple
- Provide audio controls

Combining Mediums

- Use sensory combination that work best together
  - Auditory text with visual graphics
  - Screen text with visual graphics
- Both the visual and auditory information should be totally relevant to the task being performed
- Visual and auditory textual narrative should be presented simultaneously
- Considering downloading times when choosing a media
- Testing
  - Legibility
  - Comprehensibility
  - Acceptance

Choose the Proper Colors

Color Uses

- Use color to assist in formatting
  - Relating elements into grouping
  - Breaking apart separate groupings of information
  - Highlighting or calling attention to important information
- Use color as visual code to identify
  - Screen captions and data
  - Information from different sources
  - Status of information
- Use color to
  - Realistically portray natural objects

– Increase screen appeal

Possible Problems with Color

- High Attention-Getting Capacity

– Viewer might associate, tie together, screen elements of same color

– Result in confusing, slower reading

- Interference with Use of Other Screens

- Varying Sensitivity of the Eye to Different Colors

– Viewing red and blue → Eye fatigue

- Color-Viewing Deficiencies

- Cross-Disciplinary and Cross-Cultural Differences

– For financial mangers - Corporate qualities or reliability

– For health care professionals – Death

– For nuclear reactor monitors – Coolness or water

– For American movie audiences – Tenderness or Pornography

Choosing Colors for Categories of Information

- Color chosen to organize information or data on a screen must aid the transfer of information from the display to the user, Some examples of using color code

– If decisions are made based on the status of information on the screen, color-code the types of status the information

– Screen searching is performed to locate information of particular kind, color-code for contrast

– If the sequence of information use is constrained or ordered, use color to identify the sequence

– If the information on a screen is crowded, use color to provide visual grouping

- Never rely on color as the only way of identifying a screen element

- Always consider how spatial formatting, highlighting, and messages may also be useful

Color in Context

- Color are subject to contextual effects

- Small adjacent colored images may appear to the eye to merge or mix

- A color on a dark background will look lighter and brighter than the same color on a light background

- Colors also change as light levels change

Usage

- Design for monochrome first or in shades of black, white and gray

- Doing this will permit the screen to be effectively used:

– By people with a color-viewing deficiency

– On monochrome displays

– In conditions where ambient lighting distorts the perceived color

– If the color ever fails

- Use colors conservatively

– Do not use color where other identification techniques, such as location, are available

Discrimination and Harmony

- Select 4-5 colors for best absolute discrimination

– Red, yellow, green, blue, and brown

- Select 6-7 colors for best comparative discrimination

– Orange, yellow-green, cyan, violet, and magenta

- Choose harmonious colors

– One color plus two colors on either side of its complement

– Three colors at equidistant point around the color circle

- For extended viewing or older viewers, use brighter colors

Emphasis

- To draw attention or to emphasize elements, use bright or highlighted colors or use less bright colors for deemphasize

– The perceived brightness of colors from most to least is white, yellow, green, blue, red

- To emphasize separation, use contrasting colors

– Red and green, blue and yellow

- To convey similarity, use similar colors

– Orange and yellow, blue and violet

Common Meanings

- To indicate that actions are necessary, use warm colors

– Red, orange, yellow

- To provide status or background, use cool colors

– Green, blue, violet, purple

- Conform to human expectation

– Red: Stop, fire, hot, danger

– Yellow: Caution, slow, test

– Green: Go, OK, clear, vegetation, safety

– Blue: Cold, water, calm, sky, neutrality

– Gray, White: Neutrality

– Warm colors: Action, response required, spatial closeness

– Cool colors: Status, background information, spatial remoteness

- Typical implications of color with dramatic portrayal are

– High illumination: Hot, active, comic situations

– Low illumination: Emotional, tense, tragic, romantic situations

– High saturation: Emotional, tense, hot, comic situations

– Warm colors: Active, leisure, recreation, comic situations

– Cool colors: Efficiency, work, tragic and romantic situations

- Proper use of color also requires consideration of the experiences and expectation of the screen viewers

Location and Ordering

- In the center of the visual field, use red and green

- For peripheral viewing, use blue, yellow, black, and white
- Use adjacent colors that differ by hue and value or lightness for a sharp edge and maximum differentiation
- Order colors by their spectral position
  - Red, orange, yellow, green, blue, indigo, violet

Foregrounds and Backgrounds

- Foregrounds
  - Use colors that highly contrast with the background color
  - For text or data
    - Black on light-color background of low intensity (no bright white)
    - Desaturated spectrum colors such as white, yellow, or green on dark background
    - Warmer more active colors
  - To emphasize an element, highlight it in a light value of the foreground color, pure white, or yellow
  - To deemphasize and element, lowlight it in a dark value of the foreground color

Foregrounds and Backgrounds

- Backgrounds
  - Use colors that do not compete with the foreground
  - Use
    - Light-colored backgrounds of low intensity: Off-white or light gray
    - Desaturated colors
    - Cool, dark colors such as blue or black
    - Colors on the spectral extreme end
  - Blue, black, gray, brown, red, green, and purple

Foregrounds and Backgrounds

- Backgrounds
  - Use colors that do not compete with the foreground

– Use

- Light-colored backgrounds of low intensity: Off-white or light gray
- Desaturated colors
- Cool, dark colors such as blue or black
- Colors on the spectral extreme end

– Blue, black, gray, brown, red, green, and purple
Color spectrum → http://www.brobstsystems.com/colors.htm

Gray Scale

- For fine discrimination use a black-gray-white scale

– Recommend values

- White: Screen background, text located in any black area
- Light gray: Background of a Pushbutton area
- Medium gray: Icon background area, Menu drop shadow, Window drop shadow, Inside area of system icons, Filename bar
- Dark gray: Window boarder
- Black: Text, Window title bar, Icon border, Icon elements, Ruled lines

Text in Color

- Text in color is not as visible as it is in black
- When switching text from black to color

– Double the width of lines

– Use bold or larger type:

- If originally 8 to 12 points, increase by 1 to 2 points
- If originally 14 to 24 points, increase by 2 to 4 points

Text in Color

- Text in color is not as visible as it is in black
- When switching text from black to color

– Double the width of lines

– Use bold or larger type:

- If originally 8 to 12 points, increase by 1 to 2 points
- If originally 14 to 24 points, increase by 2 to 4 points
- Check legibility by squinting at text

– Too-light type will recede or even disappear

Choosing color for web pages

- Always minimize the number of presented colors for faster downloading
- Always consider color in context, never in isolation
- Use similar or same color schemes throughout a Web site → help the user maintain a sense of place
- Foreground colors should be a different as possible from background colors
- The most recommended foreground text color is black presented on a light-colored background of low intensity (off white or light gray)
- Use dark backgrounds when establishing contrast between an area of the screen and the main screen body Choosing color for web pages
- High intensity colors as back-ground such as red, magenta and bright green) must be avoided
- When choosing foreground and background colors, ensure that contrasting combinations are selected
- Use a uniform color in large screen areas

– Large areas of the same color download faster

- For smaller element, the more contrast is required
- Use flat Web-safe colors
- Select color that can be easily reproduced in black and white

Use of Color to Avoid

- Relying exclusively on color (Spatial Formatting and component locations)
- Too many colors at one time
- Highly saturated, spectrally extreme colors together

  - Red/blue and yellow/purple
  - Yellow/blue, green/blue and red/green
- Low-brightness color for extended viewing or older viewer
- Colors of equal brightness
- Colors lacking contrast
- Fully saturated colors for frequently read screen components

Use of Color to Avoid

- Pure blue for text, thin lines, and small shapes
- Colors in small areas
- Colors for fine details
  - Black, gray, and white will provide better resolution
  - Other colors for large area or attracting attention
- Non-opponent colors
  - Red/yellow or green/blue
  - Recommend: Red/green or yellow/blue
- Red and green in the periphery of large-scale displays
  - Yellow and blue are much better

Use of Color to Avoid

- Adjacent colors only differing in the amount of blue they posses
- Single color distinctions for color-deficient user
- Using colors in unexpected ways
- Using color to improve readability of densely packed text
  - Recommend to use space lines

UNIT-4

Specification Methods

- Design requires a good notation to record and discuss alternate possibilities:
  - The default language for specifications in any field is natural language, e.g., English

  - Communication medium, e.g., sketchpad, or blackboard
- Natural-language specifications tend to be:
  - lengthy
  - vague
  - ambiguous
- Therefore often are difficult to prove:
  - correct
  - consistent
  - complete
- Backus-Naur Form (a.k.a. Backus Normal Form or BNF)
  - high-level components are described as nonterminals
  - specific strings are described as terminals
- Grammars Example

<Telephone book entry>::= <Name><Telephone number>
<Name> ::= <Last name>, <First name>
<Last name> ::= <string>
<First name> ::= <string>
<string> ::= <character>|<character><string>
<character> ::= A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z
<Telephone number>::= (<area code>) <exchange>-<local number>
<area code>::= <digit><digit><digit>
<exchange>::= <digit><digit><digit>
<local number>::= <digit><digit><digit><digit><digit>::= 0|1|2|3|4|5|6|7|8|9

- Examples of acceptable entries
  - WASHINGTON, GEORGE (301) 555-1234
  - BEEF, STU (726) 768-7878
  - A, Z (999) 111-1111
- Multiparty grammars

<Session> ::= <U: Opening> <C: Responding>
<U: Opening> ::= LOGIN <U: Name>
<U: Name> ::= <U: string>
<C: Responding> ::= HELLO [<U: Name.]
U: User C: Computer

- Multiparty grammars are effective for text oriented command sequences

- **Transition Diagram**

– a set of *nodes* that represents system states and a set of *links* between the nodes that represents possible transitions

- State Charts

Interface-Building Tools:
Features of Interface-Building Tools.

- User Interface Independence

– Separate interface design from internals

– Enable multiple user interface strategies

– Enable multiple platform support

– Establish user interface architect role

– Enforce standards

- Methodology & Notation

– Develop design procedures

– Find ways to talk about design

– Create project management

- Rapid Prototyping

– Try out ideas very early

– Test, revise, test, revise,...

– Engage end users, managers, and others

- Software Support

– Increase productivity

– Offer some constraint & consistency checks

– Facilitate team approaches

– Ease maintenance

User interface mockup tools

- Examples

– Paper and pencil

– Word processors

- Slide-show software
- Macromedia Director, Flash mx, or Dreamweaver

- Visual Editing

- Microsoft Visual Studio
- Borland JBuilder

- **Finding the right tool is a tradeoff between six main criteria:**

- Part of the application built using the tool.
- Learning time
- Building time
- Methodology imposed or advised
- Communication with other subsystems
- Extensibility and modularity

- **The windowing system layer**

- Sometimes working at a low-level is required.
- E.g., new platform
- The while(true) main loop

- **The GUI toolkit layer**

- Widgets, such as windows, scroll bars, pull-down or pop-up menu, etc.
- Difficult to use without an interface

- **The application framework and specialized language layer**

□ Application frameworks are based on object-oriented programming

- Can quickly build sophisticated interfaces
- Require intensive learning

□ Specialized language layers lighten the programming burden

- Tcl (and its toolkit Tk)
- Perl/Tk
- Python/Tk
- Visual Basic
- Java Script

Evaluation and Critiquing Tools
Tullis' Display Analysis Program, Version 4.0:

- Takes alphanumeric screen designs and produces display-complexity metrics plus some advice:

– Upper-case letters: 77% The percentage of upper-case letters is high.

- Consider using more lower-case letters, since text printed in normal upper- and lower-case letters is read about 13% faster than text in all upper case. Reserve all upper-case for items that need to attract attention.

– Maximum local density = 89.9% at row 9, column 8. Average local density = 67.0%

- The area with the highest local density is identified...you can reduce local density by distributing the characters as evenly as feasible over the entire screen.

– Total layout complexity = 8.02 bits Layout complexity is high.

- This means that the display items (labels and data) are not well aligned with each other...Horizontal complexity can be reduced by starting items in fewer different columns on the screen (that is, by aligning them vertically).

- **Doctor HTML - Web Page Analyzer:**

– Did not find the required open and close HEAD tag. You should open and close the HEAD tag in order to get consistent performance on all browsers.

– Found extra close STRONG tags in the document. Please remove them.

UNIT -5:

Keyboard Layouts

QWERTY layout

— 1870 Christopher Latham Sholes

— good mechanical design and a clever placement of the letters that slowed down the users enough that key jamming was infrequent

— put frequently used letter pairs far apart, thereby increasing finger travel distances

Dvorak layout

— 1920

— reduces finger travel distances by at least one order of magnitude

— Acceptance has been slow despite the dedicated efforts of some devotees

— it takes about 1 week of regular typing to make the switch, but most users have been unwilling to invest the effort

ABCDE style

— 26 letters of the alphabet laid out in alphabetical order nontypists will find it easier to locate the keys

Additional keyboard issues

— IBM PC keyboard was widely criticized because of the placement of a few keys

- backslash key where most typists expect SHIFT key
- placement of several special characters near the ENTER key

— Number pad layout

— wrist and hand placement

Keys

— 1/2 inch square keys

— 1/4 inch spacing between keys

— slight concave surface

— matte finish to reduce glare finger slippage

— 40- to 125-gram force to activate

— 3 to 5 millimeters displacement

— tactile and audible feedback important

— certain keys should be larger (e.g. ENTER, SHIFT, CTRL)

— some keys require state indicator, such as lowered position or light indicator (e.g. CAPS LOCK)

— key labels should be large, meaningful, permanent

— some "home" keys may have additional features, such as deeper cavity or small raised dot, to help user locate their fingers properly (caution - no standard for this)
Function keys

— users must either remember each key's function, identify them from the screen's display, or use a template over the keys in order to identify them properly

— can reduce number of keystrokes and errors

— meaning of each key can change with each application placement on keyboard can affect efficient use

— special-purpose displays often embed function keys in monitor bezel

— lights next to keys used to indicate availability of the function, or on/off status

— typically simply labeled F1, F2, etc, though some may also have meaningful labels, such as CUT, COPY, etc.

— frequent movement between keyboard home position and mouse or function keys can be disruptive to use

— alternative is to use closer keys (e.g. ALT or CTRL) and one letter to indicate special function
Cursor movement keys

— up, down, left, right

— some keyboards also provide diagonals

— best layout is natural positions

— inverted-T positioning allows users to place their middle three fingers in a way that reduces hand and finger movement

— cross arrangement better for novices than linear or box

— typically include typamatic (auto-repeat) feature

— important for form-fillin and direct manipulation

— other movements may be performed with other keys, such as TAB, ENTER, HOME, etc.
Keyboard and keypads for small devices

— Wireless or foldable keyboards

— Virtual keyboards

— Cloth keyboards

— Soft keys

— Pens and touchscreens

Pointing Devices

Pointing devices are applicable in six types of interaction tasks:

1. Select:

– user chooses from a set of items.

– used for traditional menu selection, identification of a file in a directory, or marking of a part in an automobile design.

2. Position:

– user chooses a point in a one-, two-, three-, or higher-dimensional space

– used to create a drawing, to place a new window, or to drag a block of text in a figure.

3. Orient:

– user chooses a direction in a two-, three-, or higher-dimensional space.

– direction may simply rotate a symbol on the screen, indicate a direction of motion for a space ship, or control the operation of a robot arm.

4. Path:

– user rapidly performs a series of position and orient operations.

– may be realized as a curving line in a drawing program, the instructions for a cloth cutting machine, or the route on a map.

5. Quantify:

– user specifies a numeric value.

– usually a one-dimensional selection of integer or real values to set parameters, such as the page number in a document, the velocity of a ship, or the amplitude of a sound.

6. Text:

– user enters, moves, and edits text in a two-dimensional space. The

– pointing device indicates the location of an insertion, deletion, or change.

– more elaborate tasks, such as centering; margin setting; font sizes; highlighting, such as boldface or underscore; and page layout.

Direct-control pointing devices

lightpen

— enabled users to point to a spot on a screen and to perform a select, position, or other task

— it allows direct control by pointing to a spot on the display

— incorporates a button for the user to press when the cursor is resting on the desired spot on the screen

— lightpen has three disadvantages: users' hands obscured part of the screen, users had to remove their hands from the keyboard, and users had to pick up the lightpen

Touchscreen

— allows direct control touches on the screen using a finger

— early designs were rightly criticized for causing fatigue, hand-obscuring-the-screen, hand-off-keyboard, imprecise pointing, and the eventual smudging of the display

— lift-off strategy enables users to point at a single pixel

— the users touch the surface

— then see a cursor that they can drag around on the display

— when the users are satisfied with the position, they lift their fingers off the display to activate

— can produce varied displays to suit the task

— are fabricated integrally with display surfaces

Tablet PCs and Mobile Devices:

- Natural to point on the LCD surface
- Stylus
- Keep context in view
- Pick up & put down stylus
- Gestures and handwriting recognition

Indirect pointing devices

- mouse

— the hand rests in a comfortable position, buttons on the mouse are easily pressed, even long motions can be rapid, and positioning can be precise

- trackball

— usually implemented as a rotating ball 1 to 6 inches in diameter that moves a cursor

- joystick

— are appealing for tracking purposes

- graphics tablet

— a touch-sensitive surface separate from the screen

- touchpad

— built-in near the keyboard offers the convenience and precision of a touchscreen while keeping the user's hand off the display surface

- Human-factors variables

— speed of motion for short and long distances

— accuracy of positioning

— error rates

— learning time

— user satisfaction

- Other variables

— cost

— durability

— space requirements

— weight

— left- versus right-hand use

— likelihood to cause repetitive-strain injury

— compatibility with other systems

Comparison of pointing devices

- Some results

— direct pointing devices faster, but less accurate

— graphics tablets are appealing when user can remain with device for long periods without switching to keyboard

— mouse is faster than isometric joystick

— for tasks that mix typing and pointing, cursor keys a faster and are preferred by users to a mouse

— muscular strain is low for cursor keys

- Fitts' Law

— Index of difficulty = log2 (2D / W)

— Time to point = C1 + C2 (index of difficulty)

— C1 and C2 and constants that depend on the device

— Index of difficulty is log2 (2*8/1) = log2(16) = 4 bits

— A three-component equation was thus more suited for the high-precision pointing task:

— Time for precision pointing = C1 + C2 (index of difficulty) + C3 log2 (C4 / W)

Novel devices

- Foot controls
- Eye-tracking
- Multiple-degrees-of-freedom devices
- DataGlove
- Haptic feedback

- Bimanual input
- Ubiquitous computing and tangible user interfaces
- Handheld devices

Speech and auditory interfaces

- Speech recognition still does not match the fantasy of science fiction:

— demands of user's working memory

— background noise problematic

— variations in user speech performance impacts effectiveness

— most useful in specific applications, such as to benefit handicapped users

- Discrete word recognition

— recognize individual words spoken by a specific person; can work with 90- to 98-percent reliability for 20 to 200 word vocabularies

— Speaker-dependent training, in which the user repeats the full vocabulary once or twice

— Speaker-independent systems are beginning to be reliable enough for certain commercial applications

— been successful in enabling bedridden, paralyzed, or otherwise disabled people

— also useful in applications with at least one of the following conditions:

- speaker's hands are occupied
- mobility is required
- speaker's eyes are occupied
- harsh or cramped conditions preclude use of keyboard

— voice-controlled editor versus keyboard editor

- lower task-completion rate
- lower error rate

— use can disrupt problem solving

- Continuous-speech recognition

— Not generally available:

- difficulty in recognizing boundaries between spoken words
- normal speech patterns blur boundaries
- many potentially useful applications if perfected
- Speech store and forward

— Voice mail users can

- receive messages
- replay messages
- reply to caller
- forward messages to other users, delete messages
- archive messages
- Systems are low cost and reliable.
- Voice information systems

— Stored speech commonly used to provide information about tourist sites, government services, after-hours messages for organizations

— Low cost

— Voice prompts

— Deep and complex menus frustrating

— Slow pace of voice output, ephemeral nature of speech, scanning and searching problems

— Voice mail

— Handheld voice recorders

— Audio books

— Instructional systems

- Speech generation

— Michaelis and Wiggins (1982) suggest that speech generation is "frequently preferable" under these circumstances:

- The message is simple.
- The message is short.
- The message will not be referred to later.
- The message deals with events in time.
- The message requires an immediate response.
- The visual channels of communication are overloaded.

- The environment is too brightly lit, too poorly lit, subject to severe vibration, or otherwise unsuitable for transmission of visual information.
- The user must be free to move around.
- The user is subjected to high G forces or anoxia

Audio tones, audiolization, and music

— Sound feedback can be important:

- to confirm actions
- offer warning
- for visually-impaired users
- music used to provide mood context, e.g. in games
- can provide unique opportunities for user, e.g. with simulating various musical instruments

Displays – Small and Large

- The display has become the primary source of feedback to the user from the computer

— The display has many important features, including:

- Physical dimensions (usually the diagonal dimension and depth)
- Resolution (the number of pixels available)
- Number of available colors, color correctness
- Luminance, contrast, and glare
- Power consumption
- Refresh rates (sufficient to allow animation and video)
- Cost
- Reliability

Usage characteristics distinguish displays:

- Portability
- Privacy
- Saliency
- Ubiquity
- Simultaneity

Display technology

- Monochrome displays

— are adequate, and are attractive because of their lower cost

- RGB shadow-mask displays

— small dots of red, green, and blue phosphors packed closely

- Raster-scan cathode-ray tube (CRT)

— electron beam sweeping out lines of dots to form letters

— refresh rates 30 to 70 per second

- Liquid-crystal displays (LCDs)

— voltage changes influence the polarization of tiny capsules of liquid crystals

— flicker-free

— size of the capsules limits the resolution

- Plasma panel

— rows of horizontal wires are slightly separated from vertical wires by small glass-enclosed capsules of neon-based gases

- Light-emitting diodes (LEDs)

— certain diodes emit light when a voltage is applied

— arrays of these small diodes can be assembled to display characters

- Electronic ink

— Paper like resolution

— Tiny capsules with negatively and positively charged particles

- Braille displays

— Pins provide output for the blind

- Large displays

— Informational wall displays

— Interactive wall displays

— Multiple desktop displays

- Heads-up and helmet mounted displays

— A heads-up display can, for instance, project information on a partially silvered widescreen of an airplane or car

— A helmet/head mounted display (HMD) moves the image with the user

— 3D images

Mobile device displays

- Currently mobile devices used for brief tasks, except for game playing
- Optimize for repetitive tasks
- Custom designs to take advantage of every pixel
- DataLens allows compact overviews
- Web browsing difficult
- Okay for linear reading, but making comparisons can be difficult

Animation, image, and video

- Accelerated graphics hardware
- More information shared and downloaded on the web
- Scanning of images and OCR
- Digital video
- CDROMS and DVDs
- Compression and decompression through MPEG
- Computer-based video conferencing

Printers

- Important criteria for printers:

— Speed

— Print quality

— Cost

— Compactness

— Quiet operation

— Use of ordinary paper (fanfolded or single sheet)

— Character set

— Variety of typefaces, fonts, and sizes

— Highlighting techniques (boldface, underscore, and so on)

— Support for special forms (printed forms, different lengths, and so on)

— Reliability

- dot-matrix printers

— print more than 200 characters per second, have multiple fonts, can print boldface, use variable width and size, and have graphics capabilities

- inkjet printers

— offer quiet operation and high-quality output

- thermal printers or fax machines

— offer quiet, compact, and inexpensive output on specially coated papers

- laser printers

— operate at 30,000 lines per minute

- color printers

— allow users to produce hardcopy output of color graphics, usually by an inkjet approach with three colored and black inks

- photographic printers

— allow the creation of 35-millimeter or larger slides (transparencies) and photographic prints