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?

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 cognitive model 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

DEFINING THE 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.

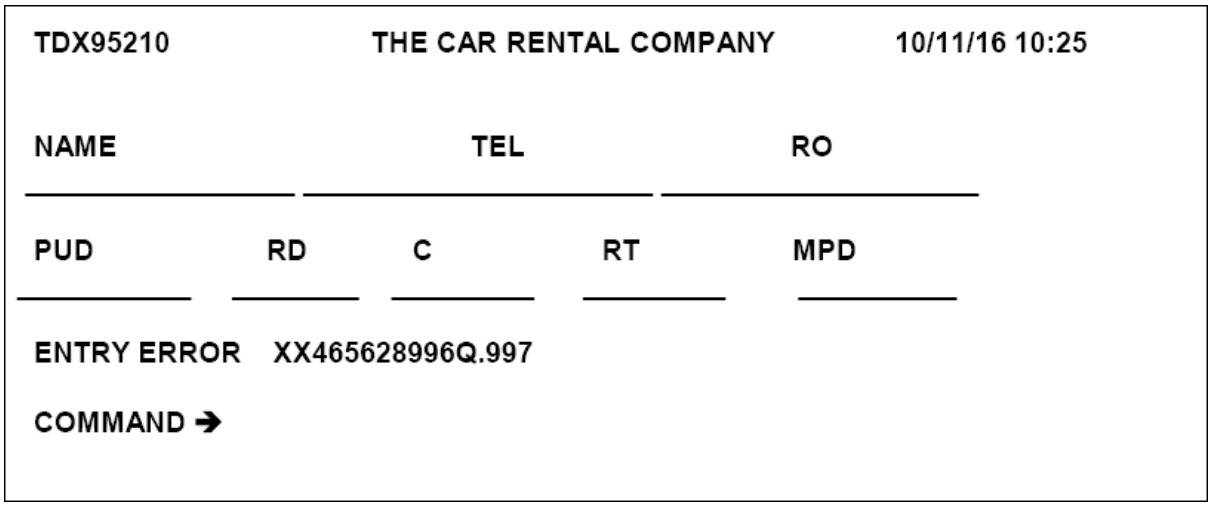
- These systems also introduced the graphical user interface as we know it a new concept was born, revolutionizing the human-computer interface.

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.

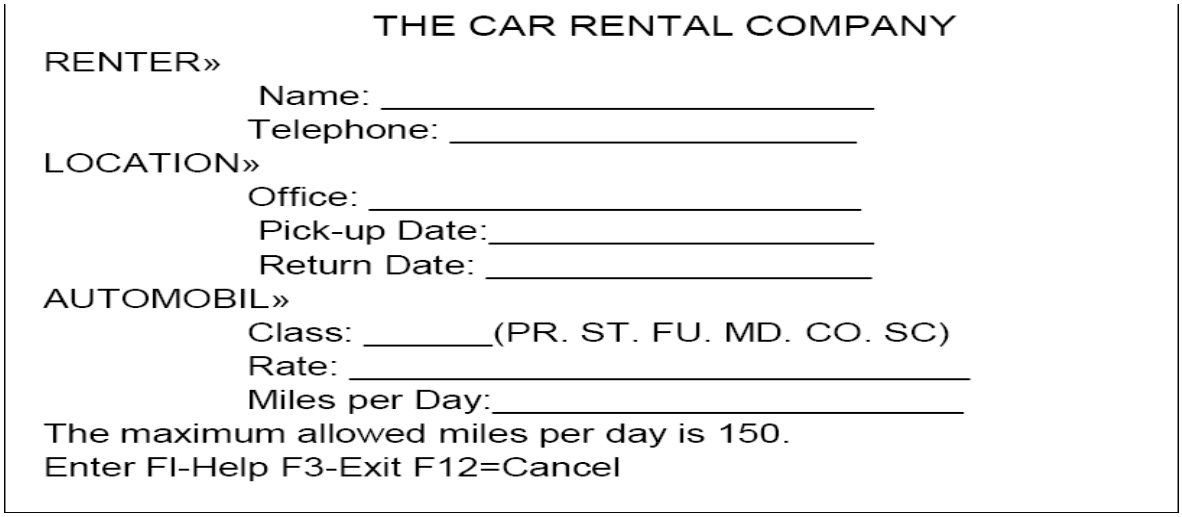
- Most early screens were monochromatic, typically presenting green text on black backgrounds.

- 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.

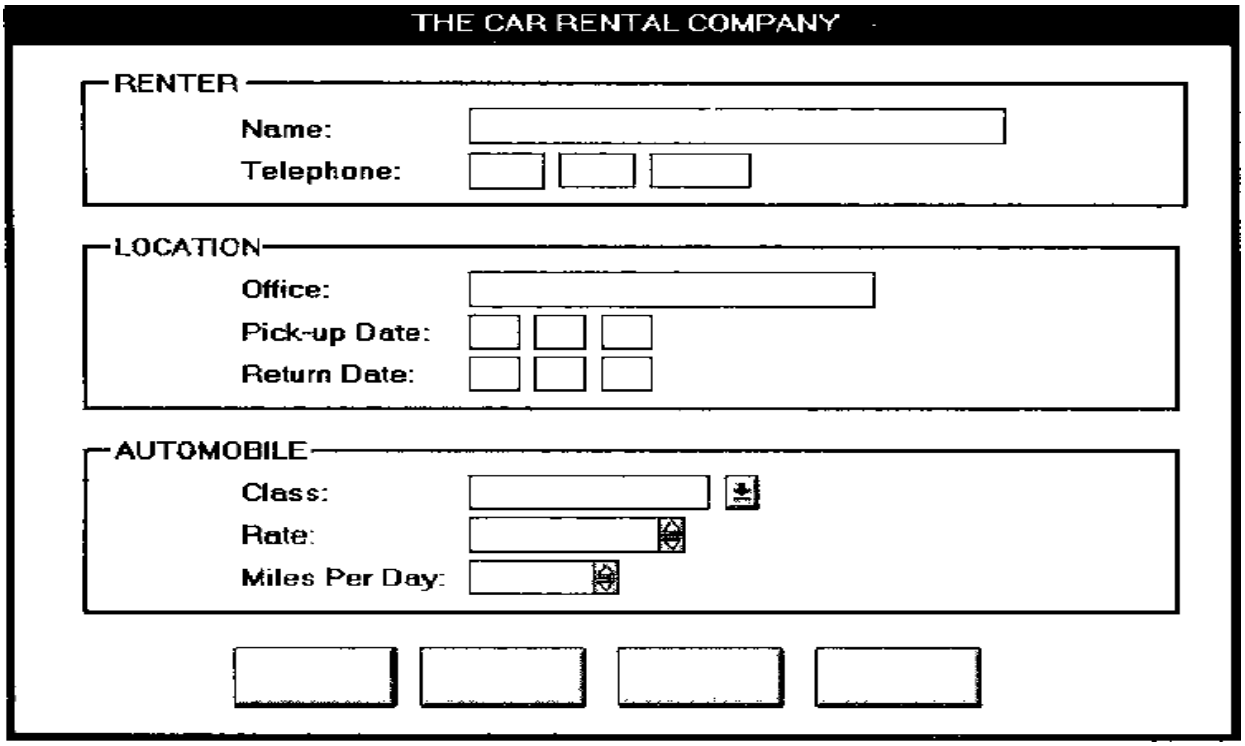
• These screens were not entirely clutter-free, however. Instructions and reminders to the user had to be inscribed on the screen in the form of prompts or completion aids such as the codes PR and Sc.

• 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

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-andclick 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.

- 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.
- The primary mechanism for performing this pick-and-click is most often the mouse and its buttons.
- The user moves the mouse pointer to the relevant element (pick) and the action is signaled (click).
- Pointing allows rapid selection and feedback. The hand and mind seem to work smoothly and efficiently together.
- 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.

PRINCIPLES FOR THE XEROX STAR

- The design of the Xerox STAR was guided by a set of principles that evolved over its lengthy development process. These principles established the foundation for graphical interfaces.
- Displaying objects that are selectable and manipulable must be created.
- A design challenge is to invent a set of displayable objects that are represented meaningfully and appropriately for the intended application.
- It must be clear that these objects can be selected, and how to select them must be Self-evident.
- When they are selected should also be obvious, because it should be clear that the selected object will be the focus of the next action. Standalone icons easily fulfilled this requirement.
- The handles for windows were placed in the borders.
- Visual order and viewer focus: Attention must be drawn, at the proper time, to the important and relevant elements of the display. Effective visual contrast between various components of the screen is used to achieve this goal. Animation is also used to draw attention, as is sound.

Feedback must also be provided to the user. Since the pointer is usually the focus of viewer attention, it is a useful mechanism for providing this feedback (by changing shapes).

- Revealed structure: The distance between one's intention and the effect must be minimized.

Most often, the distance between intention and effect is lengthened as system power increases. The relationship between intention and effect must be, tightened and made as apparent as possible to the user. The underlying structure is often revealed during the selection process.

- Consistency: Consistency aids learning. Consistency is provided in such areas as element location, grammar, font shapes, styles, and sizes, selection indicators, and contrast and emphasis techniques.
- Appropriate effect or emotional impact: The interface must provide the appropriate emotional effect for the product and its market. Is it a corporate, professional, and secure business system? Should it reflect the fantasy, wizardry, and bad puns of computer games?
- A match with the medium: The interface must also reflect the capabilities of the device on which it will be displayed. Quality of screen images will be greatly affected by a device's resolution and color-generation capabilities.

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-III

Microsoft entered the marketplace in August 1981 by releasing version 1.0 of the operating system Microsoft DOS (MS-DOS), a 16-bit command-line operating system.
Bill Gates and Paul Allen founded Microsoft and windows operating system has been its primary product.

Microsoft Windows is a multitasking operating system developed by Microsoft Corporation which uses Graphical User Interface to interact with the users. Microsoft was originally named "Traf-O-Data" in 1972, was renamed as "Micro-soft" in November 1975, then "Microsoft" on November 26, 1976

we will also clear you about the latest OS release of Windows is "Windows 10" which was launched in the year 2015.

below is how Microsoft windows evolved over time:

Windows 1.0 – Nov 1985
Windows 2.0 – Dec 1987
Windows 3.0 – May 1990
Windows 95 – Aug 1995
Windows 98 – June 1998
Windows ME – Sep 2000
Windows XP – Oct 2001
Windows Vista – Nov 2006
Windows 7 – July 2009
Windows 8.0 – Oct 2012
Windows 8.1 – Oct 2013
Windows 10 – July 2015

Main Components of Windows:

After learning about the Introduction to Windows, we are now going to study about the main components of Windows. The main components of the Windows Operating System are the following:

Configuration and maintenance
User interface
Applications and utilities
Windows Server components
File systems
Core components
Services
DirectX
Networking
Scripting and command-line
Kernel
NET Framework
Security
Deprecated components and apps

APIs

Features of Windows:

Windows Search: We can have numerous files and contents located on our system and sometimes we may run out of memory about the exact location of our file. Windows Search is a search function included with Windows that allows the user to search their entire computer.
Windows File Transfer: We may have the need to transfer in or transfer out the files and contents from our machine to other devices such as other computers or mobiles and tablets. We can do this by using an Easy Transfer Cable, CDs or DVDs, a USB flash drive, wireless Bluetooth, a network folder, or an external hard disk.

Windows Updates: Windows includes an automatic update feature with the intended purpose of keeping its operating system safe and up-to-date.
Windows taskbar: At the bottom most part of your windows, you will see a row which is known as the taskbar. It has the currently running applications, you can also pin applications that you frequently use by using an option Pin to Taskbar". The taskbar is the main navigation tool for Windows
Remote Desktop Connection: This feature of windows allows you to connect to another system and work remotely on another system.

the number of menu choices presented on a screen: -

Without logical groupings of elements, limit choices to 4 to 8. - With logical groupings of elements, limit choices to 18 to 24. Provide decreasing direction menus.

Consistency:

Provide consistency with the user's expectations.
Provide consistency in menu:
Formatting, including organization, presentation, and choice ordering.
Phrasing, including titles, choice descriptions, and instructions.
Choice selection methods.
Navigation schemes.

Display

If continual or frequent references to menu options are necessary, permanently display the menu in an area of the screen that will not obscure other screen data.
If only occasional references to menu options are necessary, the menu may be presented on demand
Critical options should be continuously displayed,

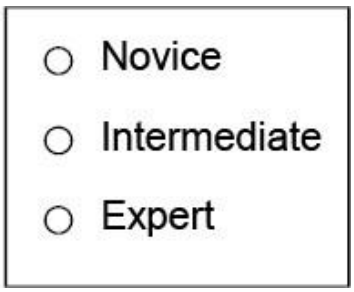
Presentation

Ensure that a menu and its choices are obvious to the user by presenting them with a unique and consistent structure, location, and/or display technique.
Ensure that other system components do not possess the same visual qualities as menu choices.
Organization
Provide a general or main menu.

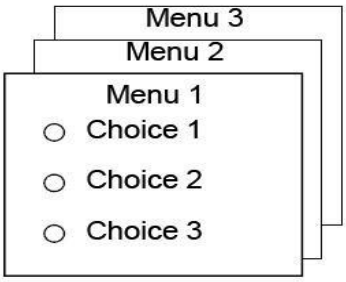
Display:
All relevant alternatives.
Only relevant alternatives.
Delete or gray-out inactive choices
Match the menu structure to the structure of the task.
Organization should reflect the most efficient sequence of steps to accomplish a person's most frequent or most likely goals.
Minimize number of menu levels within limits of clarity.
For Web sites, restrict it to two levels (requiring two mouse clicks) for fastest performance.

STRUCTURES OF MENUS

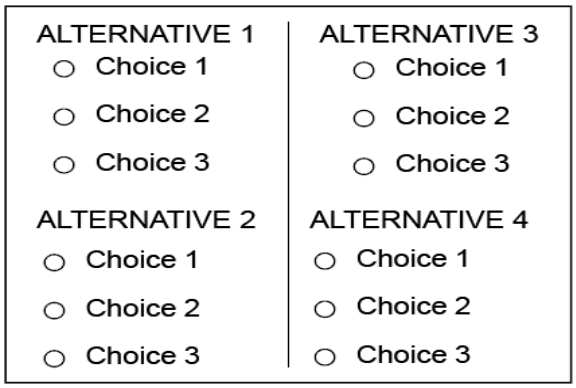
• **Single Menus**
– No other menus will follow necessitating additional user choices



• **Sequential Linear Menus**

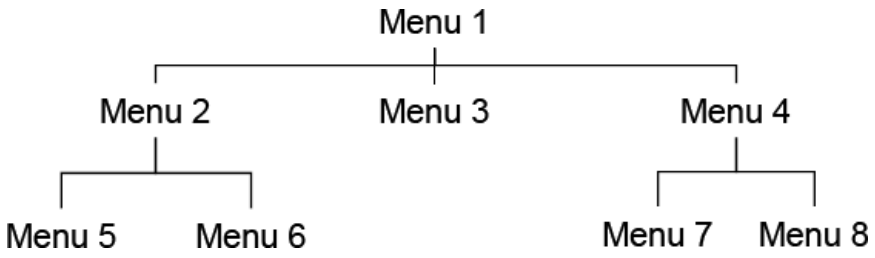


• **Simultaneous Menus**



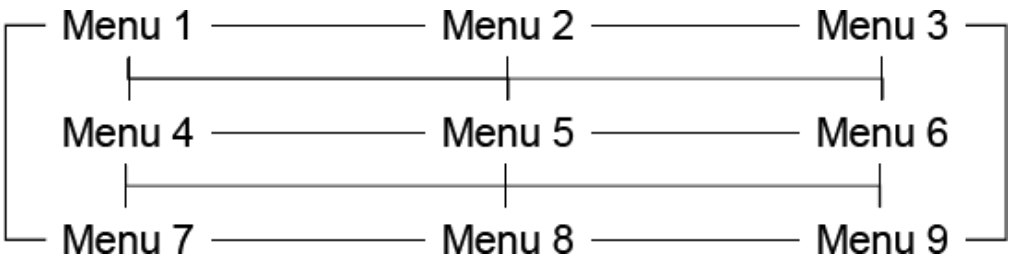
• Hierarchical Menus

– 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

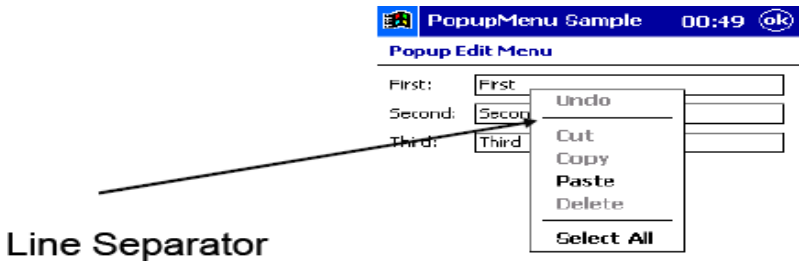
– Semantic similarity

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

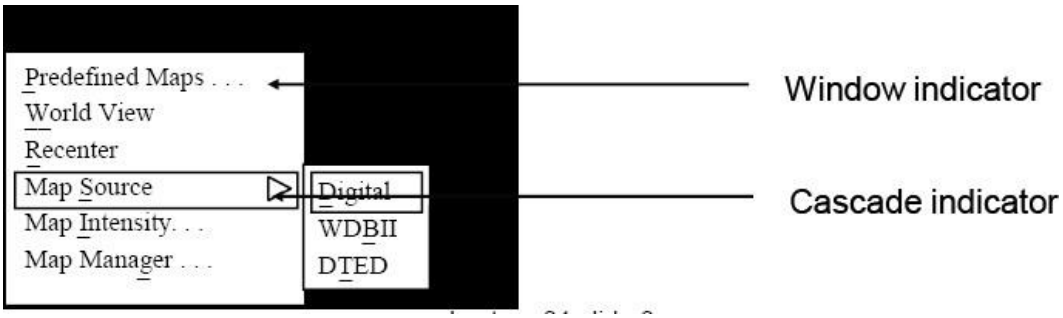
LINE SEPARATOR

• 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

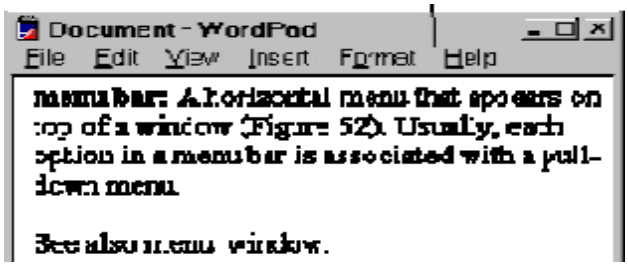
– Use to designate two opposite commands that are accessed frequently
– Use when the menu item displayed will clearly indicate that the opposite condition currently exists
– Provide a meaningful, fully spelled-out description of action
– Begin with a clear verb
– Use mixed-case letter

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

Type equation here.

- 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 cType equation here.hoice 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

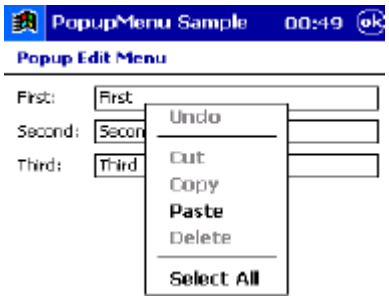
• Keyboard Equivalents and Accelerators

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

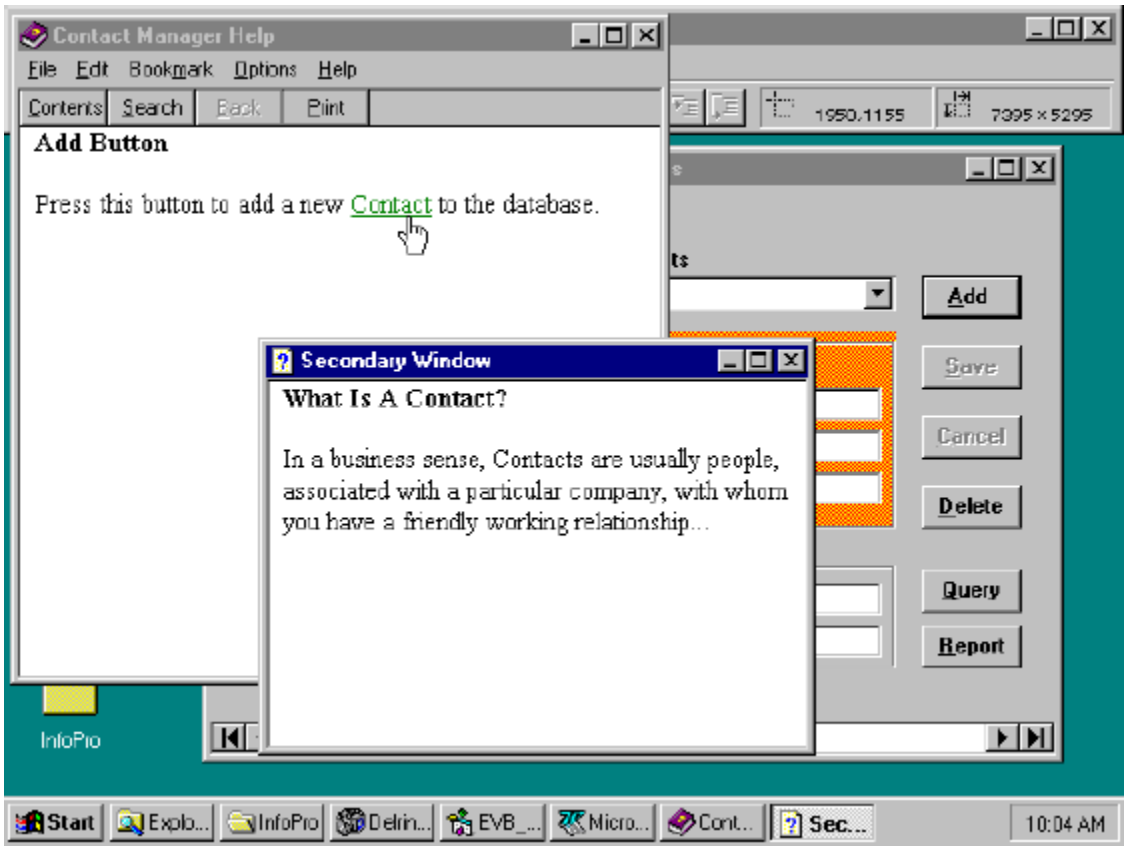
• Use overlapping windows for:
– Switching between tasks
– Tasks necessitating a greater amount of window manipulation
– Expert or experienced users
– Unpredictable display contents

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 ofsecondary 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

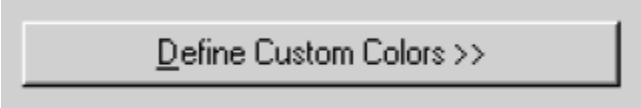
• Modeless window
– Switching between the box and its associated is permitted

• 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

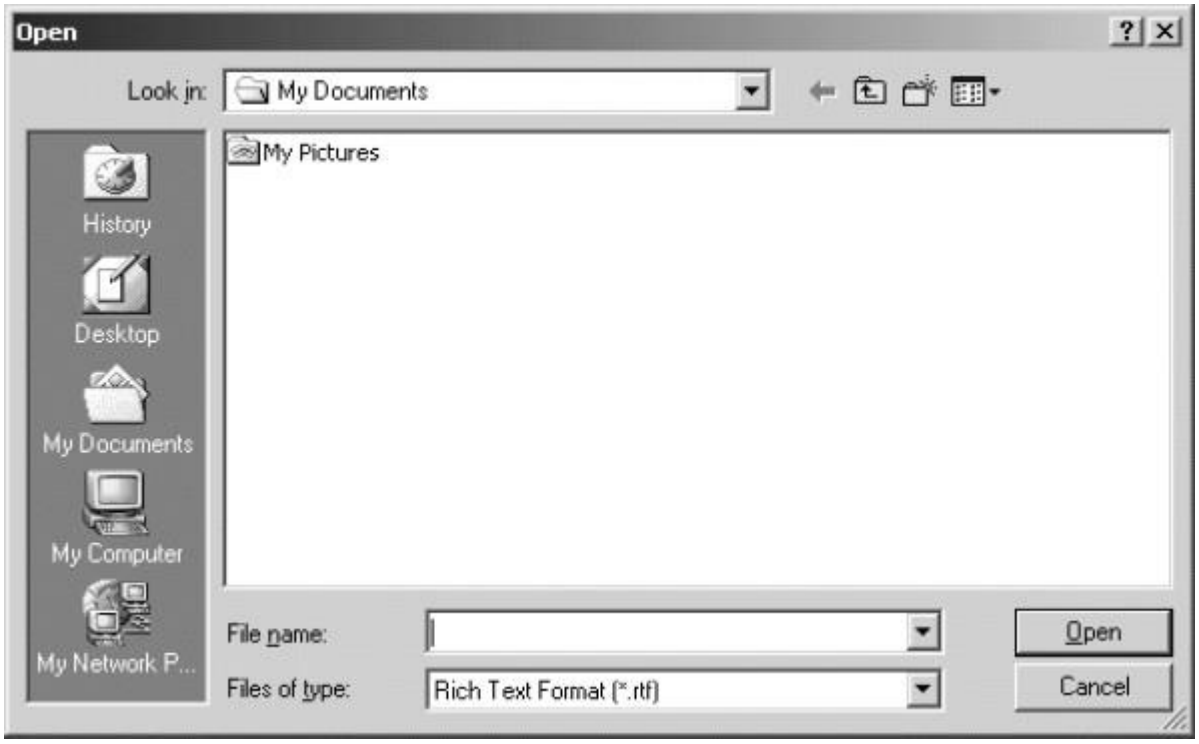
• Relevant information

• Title Bar

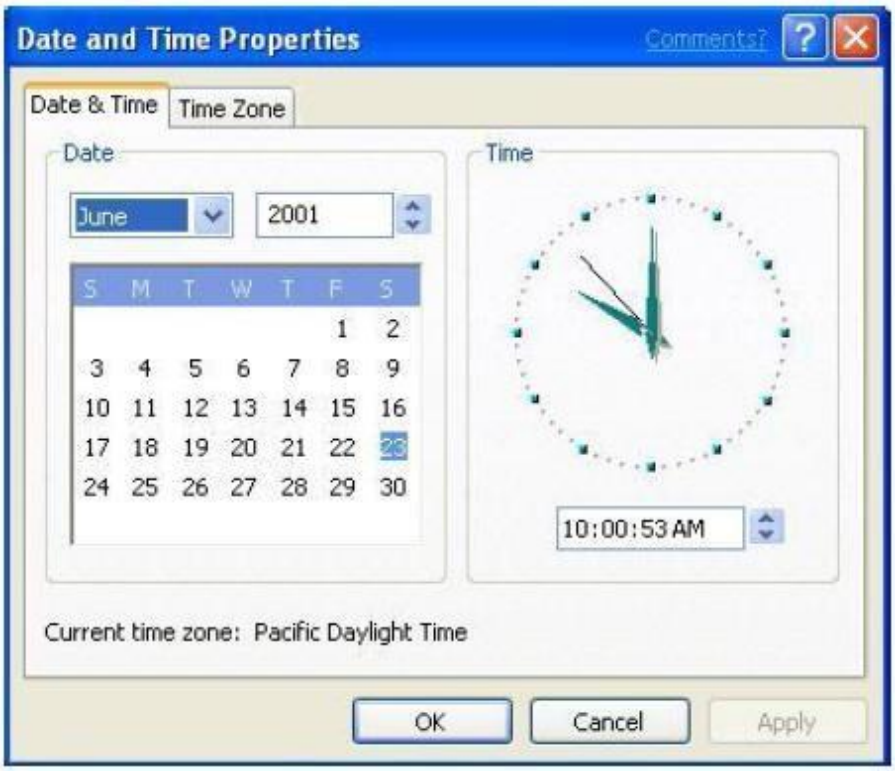
• Unfolding
– To provide advanced options at the same level in a complex dialog
– Provide a command button with an expanding dialog symbol >>
– Expand to right or downward



- 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



Property Sheets and Property Inspectors



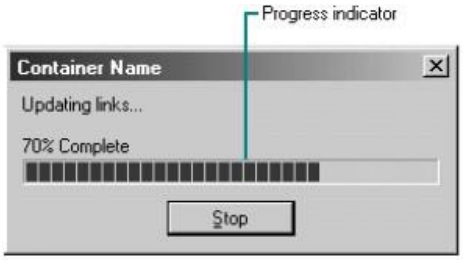
- Property sheets
  - Use for presenting the complete set of properties for an object
  - Categorize and group within property pages, as necessary
  - Command buttons to include
- Ok
- Cancel
- Apply
- Reset
- Others as necessary
  - For single property sheets, place the command on the sheet
  - For tabbed property pages, place the commands outside the tabbed pages

Property Inspectors

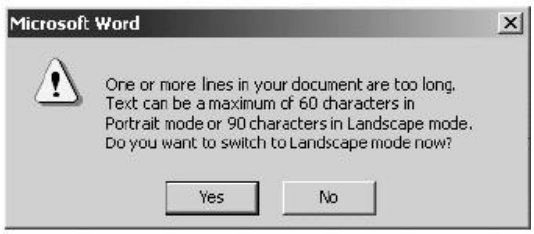
- **Property Inspectors**
  - Use for displaying only the most common or frequently accessed object properties
  - Properties of an object are displayed by using a dynamic viewer or browser that reflects the properties of the current selection
  - Property value in the selected object should be changed as soon as the user makes the change in the related property control

- 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



Progress message box

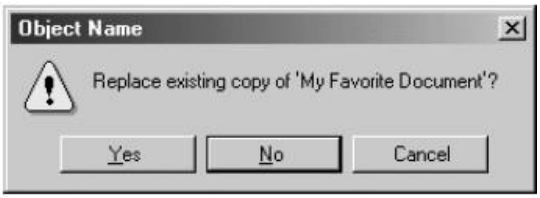


Yes No message box

Information

Warning

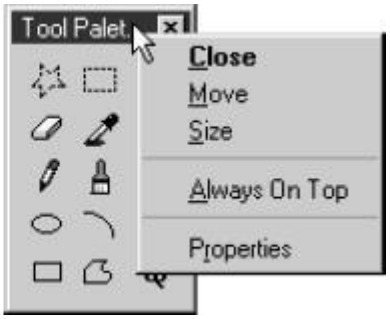
Critical



Message box choices

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
  - – Requires moving the hand far from the keyboard to use
  - – Very fatiguing to use for extended period of time

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

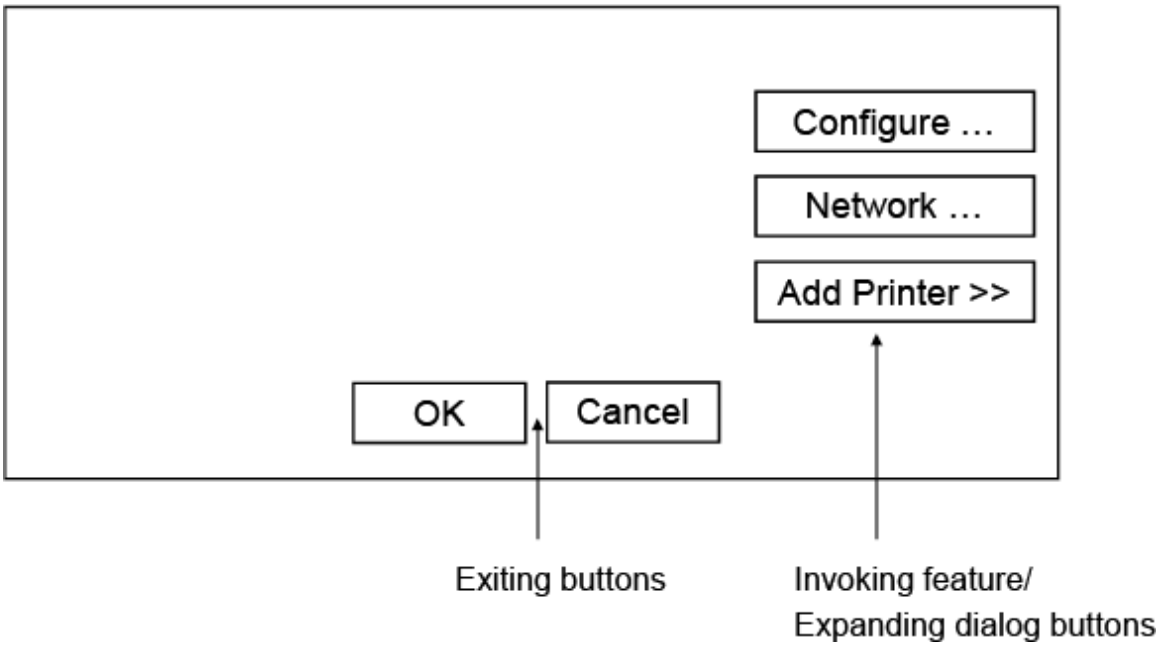
- Description
  - – A square or rectangular-shaped control with a label inside that indicates action to be accomplished
  - – The label may consist of text, graphics, or both
- Command Buttons
- Toolbars

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



Command Buttons
(Location and Layout)

- If a button has a *contingent* relationship to another control, position it adjacent to the related control

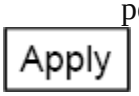
- Buttons found on more than one window should be consistently positioned

Command Buttons (Organization)

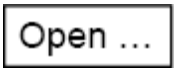
- Most frequent actions to the left or top
- Keep related buttons grouped together
- Exception: Buttons containing excessively long labels may be wider
- Windows Recommends
  - An affirmative action the left or above
  - The default first
  - OK and Cancel next to each other
  - Help last

Command Buttons (Intent Indicators)

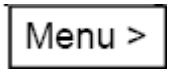
- No intent indicator is necessary, when a button causes an action to be immediately performed



- When a button leads to a cascading dialog, include and ellipsis (…)

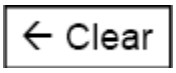


- When a button leads to a menu, include a triangle pointing in the direction the menu will appear after the label



- 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

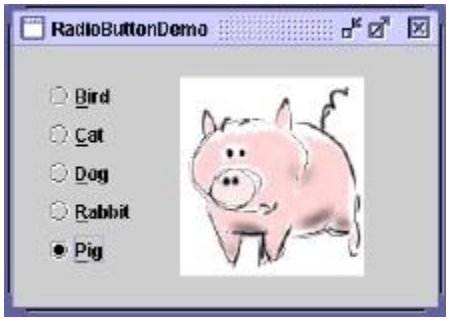
• Call attention to text box data through a highlighting technique

• 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

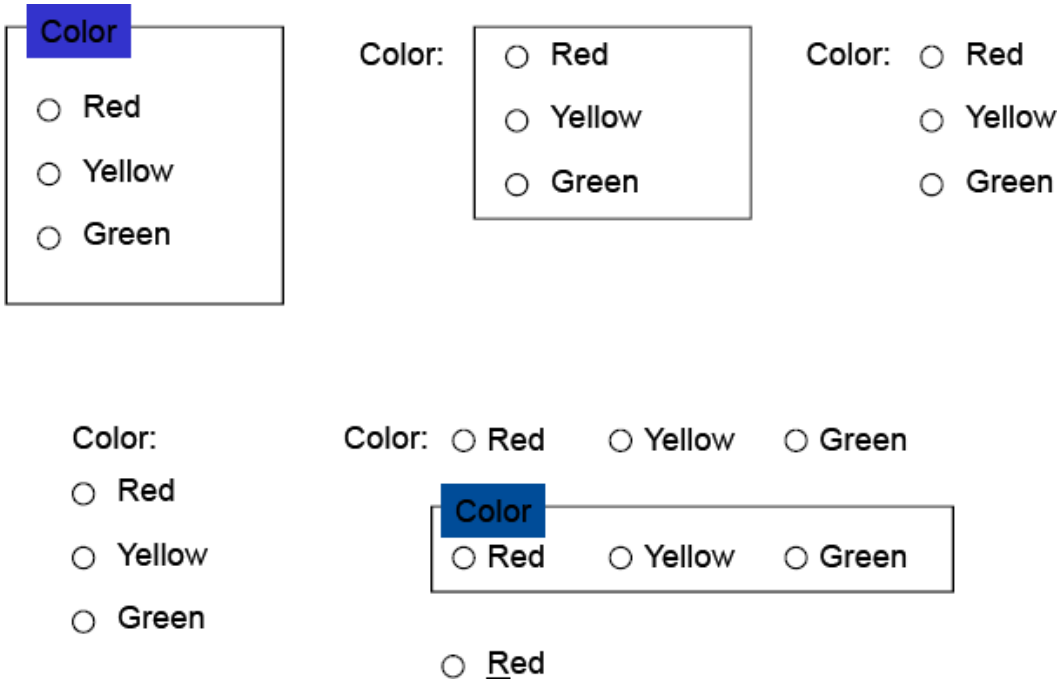
– Position the caption to the left of the choice

– 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!!

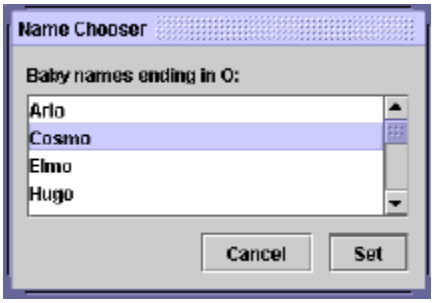
Palettes

- 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!

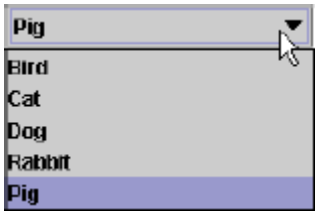
Combination Entry/Selection Controls and Other Operable Controls

- 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



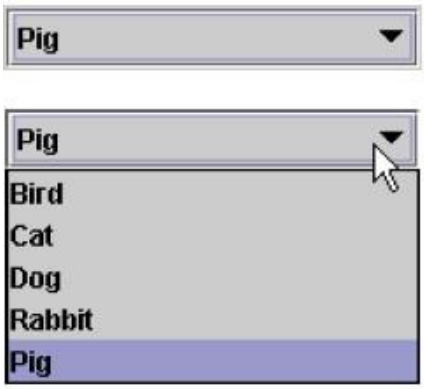
• A single rectangular text box entry field, beneath which is a larger rectangular list box (resembling a drop-down list box)

• 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 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)

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

Multimedia:

Multimedia is a form of communication that combines different content forms such as text, audio, images, animations, or video into a single presentation, in contrast to traditional mass media, such as printed material or audio recordings. Popular examples of multimedia include video podcasts, audio slideshows and Animated videos.

Multimedia can be recorded for playback on computers, laptops, smartphones, and other electronic devices.

multimedia tools provide a unique opportunity to increase expansivity, usability, and enjoyment of computer interfaces, and this should be taken into account when designing them

A Multimedia Application is an Application which uses a collection of multiple media sources e.g. text, graphics, images, sound/audio, animation and/or video. Hypermedia can be considered as one of the multimedia applications.

examples of recent applications of multimedia are electronic books and newspapers, electronic classroom presentation technologies, full-motion videoconferencing, sophisticated imaging, and graphics design tools.

In education, multimedia is used to produce computer-based training courses (popularly called CBTs) and reference books like encyclopedia and almanacs. A CBT lets the user go through a series of presentations, text about a particular topic, and associated illustrations in various information formats.

why should we use multimedia instruction?Multimedia assignments or projects allow your students to present their newly attained knowledge through images, audio and video instead of just textually. Find or create online games, tutorials or quizzes to facilitate an interactive learning environment.

Components of Multimedia

Following are the common components of multimedia:

- **Text**- All multimedia productions contain some amount of text. The text can have various types of fonts and sizes to suit the profession presentation of the multimedia software.
- **Graphics**- Graphics make the multimedia application attractive. In many cases people do not like reading large amount of textual matter on the screen. Therefore, graphics are used more often than text to explain a concept, present background information etc. There are two types of Graphics:
    - **Bitmap images**- Bitmap images are real images that can be captured from devices such as digital cameras or scanners. Generally bitmap images are not editable. Bitmap images require a large amount of memory.

- **Vector Graphics**- Vector graphics are drawn on the computer and only require a small amount of memory. These graphics are editable.

- **Audio**- A multimedia application may require the use of speech, music and sound effects. These are called audio or sound element of multimedia.Speech is also a perfect way for teaching. Audio are of analog and digital types. Analog audio or sound refers to the original sound signal. Computer stores the sound in digital form. Therefore, the sound used in multimedia application is digital audio.
- **Video**- The term video refers to the moving picture, accompanied by sound such as a picture in television. Video element of multimedia application gives a lot of information in small duration of time. Digital video is useful in multimedia application for showing real life objects. Video have highest performance demand on the computer memory and on the bandwidth if placed on the internet. Digital video files can be stored like any other files in the computer and the quality of the video can still be maintained. The digital video files can be transferred within a computer network. The digital video clips can be edited easily.
- **Animation**- Animation is a process of making a static image look like it is moving. An animation is just a continuous series of still images that are displayed in a sequence. The animation can be used effectively for attracting attention. Animation also makes a presentation light and attractive. Animation is very popular in multimedia application

Applications of Multimedia

Following are the common areas of applications of multimedia.

- **Multimedia in Business**- Multimedia can be used in many applications in a business. The multimedia technology along with communication technology has opened the door for information of global wok groups. Today the team members may be working anywhere and can work for various companies. Thus the work place will become global. The multimedia network should support the following facilities:
    - Voice Mail
    - Electronic Mail
    - Multimedia based FAX
    - Office Needs
    - Employee Training
    - Sales and Other types of Group Presentation
    - Records Management
- **Multimedia in Marketing and Advertising**- By using multimedia marketing of new products can be greatly enhanced. Multimedia boost communication on an affordable cost opened the way for the marketing and advertising personnel. Presentation that have flying banners, video transitions, animations, and sound effects are some of the elements used in composing a multimedia based advertisement to appeal to the consumer in a way never used before and

promote the sale of the products.

- **Multimedia in Entertainment**- By using multimedia marketing of new products can be greatly enhanced. Multimedia boost communication on an affordable cost opened the way for the marketing and advertising personnel. Presentation that have flying banners, video transitions, animations, and sound effects are some of the elements used in composing a multimedia based advertisement to appeal to the consumer in a way never used before and promote the sale of the products.
- **Multimedia in Education**- Many computer games with focus on education are now available. Consider an example of an educational game which plays various rhymes for kids. The child can paint the pictures, increase reduce size of various objects etc apart from just playing the rhymes.Several other multimedia packages are available in the market which provide a lot of detailed information and playing capabilities to kids.
- **Multimedia in Bank**- Bank is another public place where multimedia is finding more and more application in recent times. People go to bank to open saving/current accounts, deposit funds, withdraw money, know various financial schemes of the bank, obtain loans etc. Every bank has a lot of information which it wants to impart to in customers. For this purpose, it can use multimedia in many ways. Bank also displays information about its various schemes on a PC monitor placed in the rest area for customers. Today on-line and internet banking have become very popular. These use multimedia extensively. Multimedia is thus helping banks give service to their customers and also in educating them about banks attractive finance schemes.
- **Multimedia in Hospital**- Multimedia best use in hospitals is for real time monitoring of conditions of patients in critical illness or accident. The conditions are displayed continuously on a computer screen and can alert the doctor/nurse on duty if any changes are observed on the screen. Multimedia makes it possible to consult a surgeon or an expert who can watch an ongoing surgery line on his PC monitor and give online advice at any crucial juncture.

  In hospitals multimedia can also be used to diagnose an illness with CD-ROMs/ Cassettes/ DVDs full of multimedia based information about various diseases and their treatment.Some hospitals extensively use multimedia presentations in training their junior staff of doctors and nurses. Multimedia displays are now extensively used during critical surgeries.
- **Multimedia Pedagogues**- Pedagogues are useful teaching aids only if they stimulate and motivate the students. The audio-visual support to a pedagogue can actually help in doing so. A multimedia tutor can provide multiple numbers of challenges to the student to stimulate his interest in a topic. The instruction provided by pedagogue have moved beyond providing only button level control to intelligent simulations, dynamic creation of links, composition and collaboration and system testing of the user interactions.
- **Communication Technology and Multimedia Services**- The advancement of high computing abilities, communication ways and relevant standards has started the beginning of an era where you will be provided with multimedia facilities at home. These services may include:
    - Basic Television Services
    - Interactive entertainment
    - Digital Audio

- Video on demand
- Home shopping
- Financial Transactions
- Interactive multiplayer or single player games
- Digital multimedia libraries
- E-Newspapers, e-magazines

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

- **Synthetics** 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

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

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

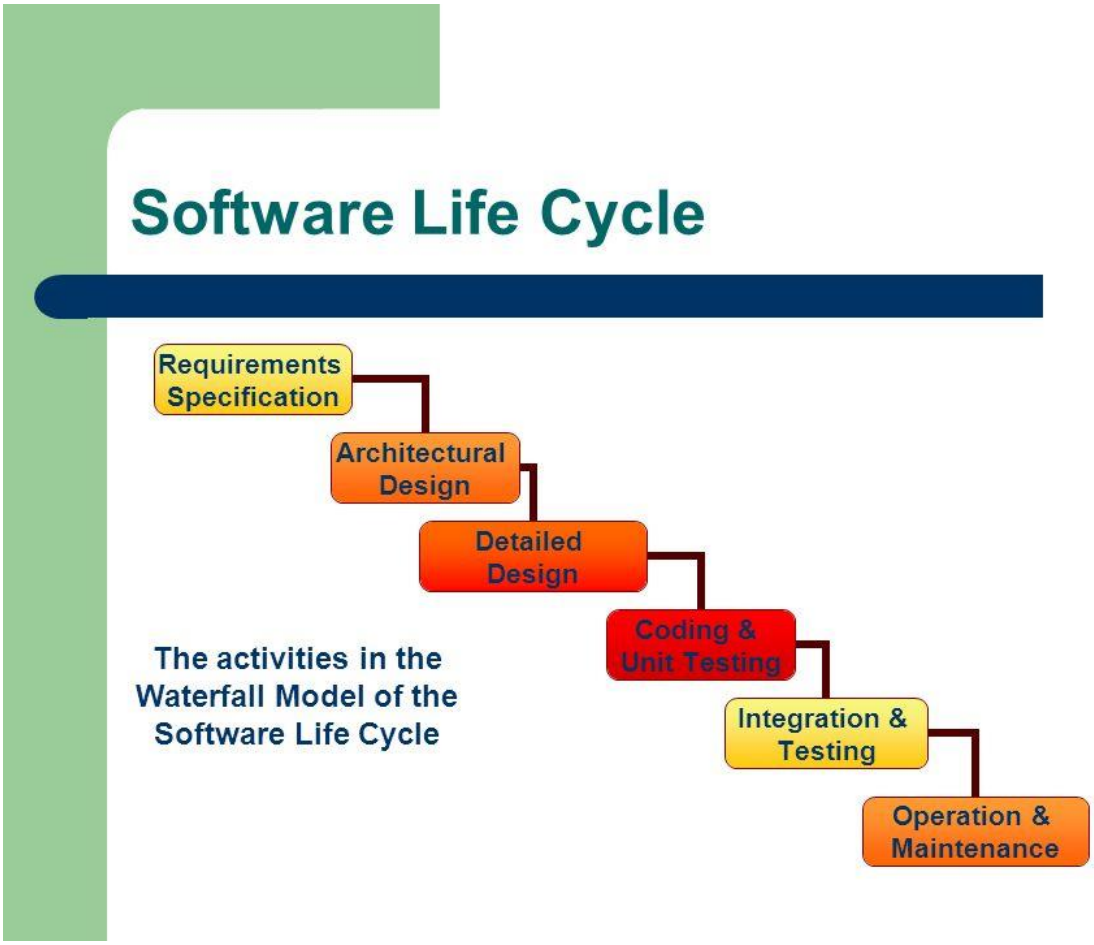
- Adjacent colors only differing in the amount of blue they posses
- Single color distinctions for color-deficient user
- Using colors in unexpected ways
- Using color to improve readability of densely packed text
  —Recommend to use space lines

UNIT -4

HCI in the software process:

- **HCI** is used in **software process** to interact product with user and give them a simple and comfort able interface. ... **Software** developers check the envoirment that it is simple and interactive for selected envoirment and it is only possible human interaction.

- The **importance of human computer interaction** in the **development** process of **software** projects. ... In design and **development** phases of **software** projects, the properties of human which is an important agent for interaction -- such as behavioral, cognitive, perceptive, efficiency and physical factors have to be considered.



- There are **5** important **principles** to take into consideration which are: balance, rhythm and repetition, emphasis, proportion and scale, and last but not least, harmony.

•Requirements specification

designer and customer try capture what the system is expected to provide can be expressed in natural language or more precise languages, such as a task analysis would provide –Informal design and scenario

based design will result in better requirements analysis

Detailed design

– Detailed design of the interface

– Move from informal to formal specification

• Separation of layers

– A layered approach to software development

will provide for more flexibility

• Data

• Logic

• Interface

Testing

• Testing is not only about functionality of

code

• Usability testing

– There are some basics that are nearly

always important

• Layout

• Language

• Number of click/steps to perform task

• Choose rather than remember

Iterative design and prototyping

• Iterative design overcomes inherent problems of incomplete requirements

Requirements

Design

Implement

Review

• Prototypes

– simulate or animate some features of intended

system

– different types of prototypes

• throw-away

• incremental

• evolutionary

• Management issues

– time

– planning

– non-functional features

– contracts

Techniques for prototyping

Storyboards need not be computer-based can be animated

Limited functionality simulations some part of system functionality provided by designers tools like HyperCard are common for these Wizard of Oz technique

Warning about iterative design design inertia – early bad decisions stay bad diagnosing real usability problems in prototypes…. and not just the symptoms

Waterfall or Prototype

• Waterfall

– Interaction paradigm

'standard' and well

understood?

– The problem is well

understood?

– Data centric systems

• Information systems

• Data warehouse

• Prototype

– The interaction paradigm

new or poorly

understood?

– The problem definition is

incomplete or poorly

defined?

– Interface centric systems

• games

• Modelling

• Design tools

Waterfall or prototype

• It doesn't have to be a one or the other

decision

• Many systems are a blend

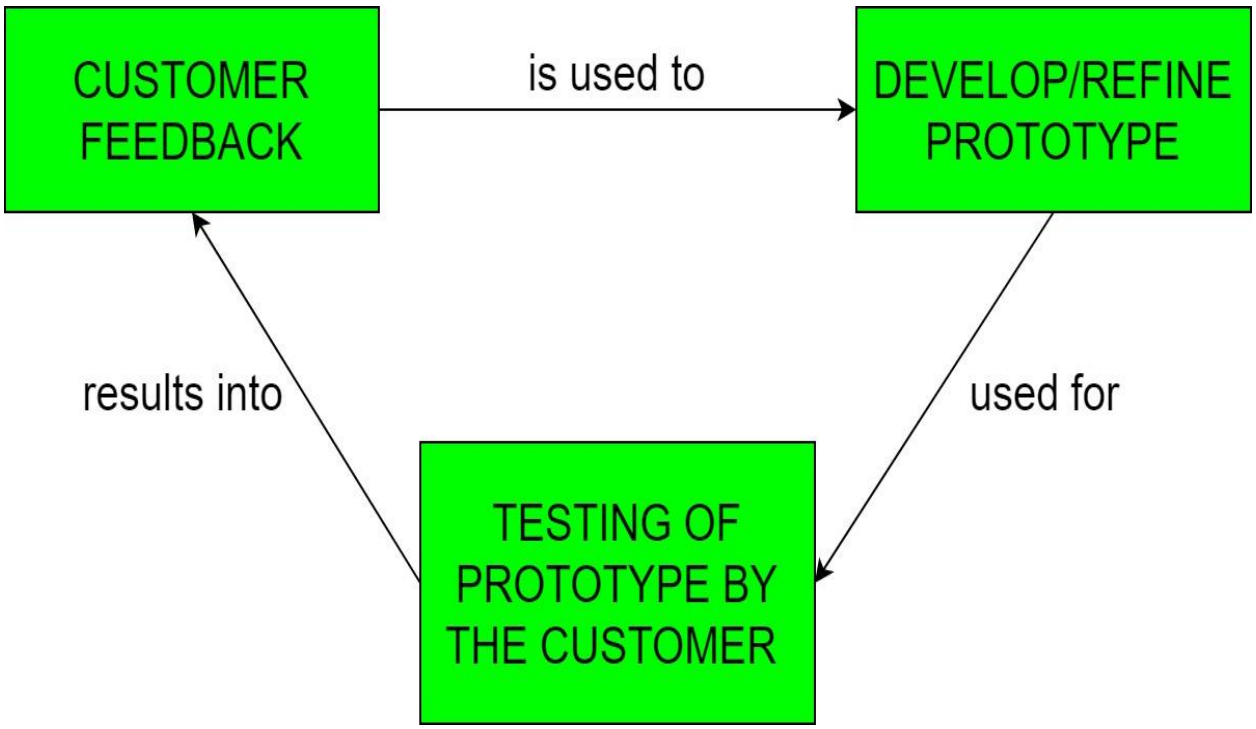
– With some parts are prototyped to elicit

requirements

• There isn't one 'best way'

• Nor is there a 'silver bullet'

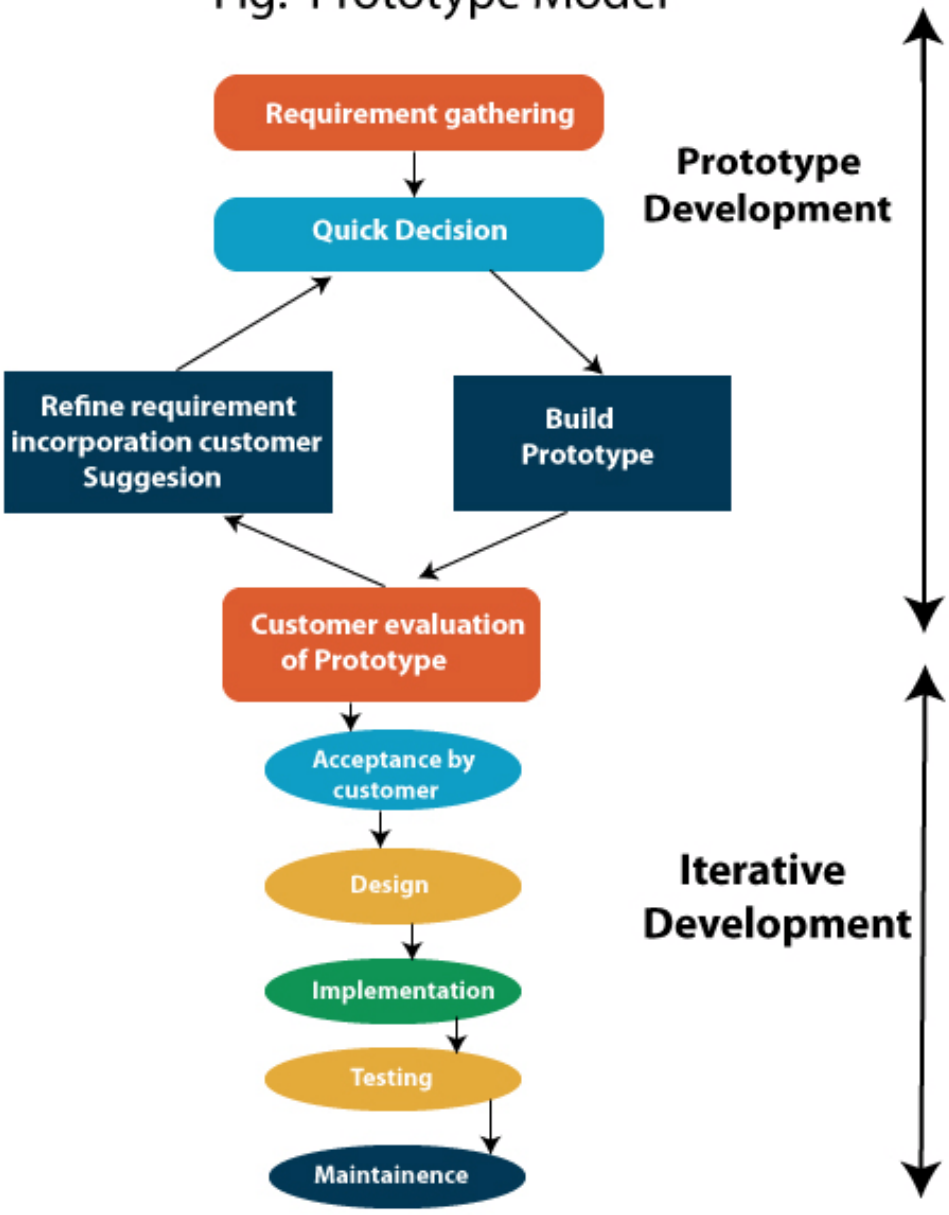
- Prototyping is defined as the process of developing a working replication of a product or system that has to be engineered. It offers a small scale facsimile of the end product and is used for obtaining customer feedback as described below:



- The prototype model requires that before carrying out the development of actual software, a working prototype of the system should be built. A prototype is a toy implementation of the system. A prototype usually turns out to be a very crude version of the actual system, possible exhibiting limited functional capabilities, low reliability, and inefficient performance as compared to actual software

- In many instances, the client only has a general view of what is expected from the software product. In such a scenario where there is an absence of detailed information regarding the input to the system, the processing needs, and the output requirement, the prototyping model may be employed.

Fig: Prototype Model



Advantage of Prototype Model

- Reduce the risk of incorrect user requirement
- Good where requirement are changing/uncommitted
- Regular visible process aids management
- Support early product marketing
- Reduce Maintenance cost.

- Errors can be detected much earlier as the system is made side by side.

Disadvantage of Prototype Model

- An unstable/badly implemented prototype often becomes the final product.
- Require extensive customer collaboration
- Costs customer money
- Needs committed customer
- Difficult to finish if customer withdraw
- May be too customer specific, no broad market
- Difficult to know how long the project will last.
- Easy to fall back into the code and fix without proper requirement analysis, design, customer evaluation, and feedback.
- Prototyping tools are expensive.
- Special tools & techniques are required to build a prototype.
- It is a time-consuming process.

HCI Pattern Evaluation Techniques:

Evaluation role is to access designs and test systems to ensure that they actually behave as we expect and meet user requirements. Ideally, evaluation should occur throughout the design life cycle, with the results of the evaluation feeding back into modifications to the design.

The three main types of evaluation methods are goal-based, process-based and outcomes-based. Goal-based evaluations measure if objectives have been achieved (We highly recommend S.M.A.R.T. Goals). Process-based evaluations analyze strengths and weaknesses.Review-based evaluation Results from the literature used to support or refute parts of design. Care needed to ensure results are transferable to new design.

Aim of evaluation is to test the functionality and usability of the design and to identify and rectify any problems. A design can be evaluated before any implementation work has started, to minimize the cost of early design errors.Evaluation is a process that critically examines a program. It involves collecting and analyzing information about a program's activities, characteristics, and outcomes. Its purpose is to make judgments about a program, to improve its effectiveness, and/or to inform programming decisions

Human Computer Interaction

Evaluation TechniquesEvaluation Techniques

 Evaluation

 tests usability and functionality of system occurs in laboratory, field and/or in collaboration with users

 evaluates both design and implementation should be considered at all stages in the design life cycle

Goals of Evaluation assess extent of system functionality assess effect of interface on user identify specific problems Evaluating Designs

Cognitive Walkthrough

 Proposed by Polson et al.

 evaluates design on how well it supports user in learning task

 usually performed by expert in cognitive psychology

 Expert 'walks through' design to identify potential problems using psychological principles

 forms used to guide analysisCognitive Walkthrough (ctd)

 For each task walkthrough considers

 what impact will interaction have on user?

 what cognitive processes are required?

 what learning problems may occur?

 Analysis focuses on goals and knowledge: does the design lead the user to generate the correct goals?

Heuristic Evaluation

 Proposed by Nielsen and Molich.

 usability criteria (heuristics) are identified

 design examined by experts to see if these are violated

 Example heuristics

 system behaviour is predictable

 system behaviour is consistent

 feedback is provided

 Heuristic evaluation `debugs' design.Review-based evaluation Results from the literature used to support or refute parts of design. Care needed to ensure results are transferable to new design.

 Model-based evaluation

- Cognitive models used to filter design options e.g. GOMS prediction of user performance.
- Design rationale can also provide useful evaluation information Evaluating through user Participation Laboratory studies
- **Advantages:**
- specialist equipment available
- uninterrupted environment
- **Disadvantages:**
- lack of context difficult to observe several users cooperating
- **Appropriate**
- if system location is dangerous or impractical for constrained single user systems to allow controlled manipulation of useField Studies
- **Advantages:**
- natural environment
- context retained (though observation may alter it)
- longitudinal studies possible
- **Disadvantages:**
- distractions
- noise
- **Appropriate**
- where context is crucial for longitudinal studiesEvaluating Implementations Requires an artefact: simulation, prototype, full implementation Experimental evaluation controlled evaluation of specific aspects of interactive behaviour evaluator chooses hypothesis to be tested a number of experimental conditions are considered which differ only in the value of some controlled variable.
- changes in behavioural measure are attributed to different conditions

Experimental factors

Subjects

- who – representative, sufficient sample
- **Variables**

- things to modify and measure
- **Hypothesis**
- what you'd like to show
- **Experimental design**
- how you are going to do it Variables
- **independent variable (IV)**
- characteristic changed to produce different conditions
- e.g. interface style, number of menu items
- **dependent variable (DV)**
- characteristics measured in the experiment
- e.g. time taken, number of errors.

Hypothesis

- **prediction of outcome**
- framed in terms of IV and DV
- e.g. “error rate will increase as font size decreases”
- **null hypothesis:**
- states no difference between conditions
- aim is to disprove this
- e.g. null hyp. = “no change with font size”

Observational Methods

Think Aloud, Cooperative evaluation, Protocol analysis,

Automated analysis, Post-task walkthroughs

Think Aloud

- user observed performing task
- user asked to describe what he is doing and why,

what he thinks is happening etc.

- **Advantages**
- simplicity - requires little expertise
- can provide useful insight
- can show how system is actually use
- **Disadvantages**
- subjective
- selective
- act of describing may alter task performance

Cooperative evaluation

- variation on think aloud
- user collaborates in evaluation
- both user and evaluator can ask each other questions throughout
- **Additional advantages**
- less constrained and easier to use
- user is encouraged to criticize system clarification possible

Protocol analysis

- paper and pencil – cheap, limited to writing speed
- audio – good for think aloud, difficult to match with other protocols
- video – accurate and realistic, needs special equipment, obtrusive
- computer logging – automatic and unobtrusive, large amounts of data difficult to analyze
- user notebooks – coarse and subjective, useful insights, good for longitudinal studies
- Mixed use in practice.
- audio/video transcription difficult and requires skill.
- Some automatic support tools available

Automated analysis – EVA

- **Workplace project**

 Post task walkthrough

 user reacts on action after the event

 used to fill in intention

 Advantages

 analyst has time to focus on relevant incidents

 avoid excessive interruption of task

 Disadvantages

 lack of freshness may be post-hoc interpretation of eventS

 Examples Noldus Pocket Observer XT (http://www.noldus.com)Post-task Walkthroughs

 Transcript played back to participant for comment immediately fresh in mind delayed evaluator has time to identify questions useful to identify reasons for actions and alternatives considered

 necessary in cases where think aloud is not possible

Questionnaires Interviews

 analyst questions user on one-to-one basis usually based on prepared questions informal, subjective and relatively cheap

 Advantages

 can be varied to suit context

 issues can be explored more fully

 can elicit user views and identify unanticipated problems

 Disadvantages

 very subjective

 time consuming Questionnaires

 Set of fixed questions given to users

 Advantages

 quick and reaches large user group

 can be analyzed more rigorously

 Disadvantages

- less flexible
- less probing Questionnaires (ctd)
- Need careful design
- what information is required?
- how are answers to be analyzed?
- Styles of question
- general
- open-ended
- scalar
- multi-choice
- ranked Physiological methods Eye tracking

Physiological measurement

eye tracking head or desk mounted equipment tracks the position

of the eye .eye movement reflects the amount of cognitive processing a display requires measurements include

fixations: eye maintains stable position. Number and duration indicate level of difficulty with display

Saccades: rapid eye movement from one point of interest to another

scan paths: moving straight to a target with a short fixation at the target is optimal physiological measurements .emotional response linked to physical changes these may help determine a user's reaction to an interface

- **measurements include:**
- heart activity, including blood pressure, volume and pulse.
- activity of sweat glands: Galvanic Skin Response (GSR)
- electrical activity in muscle: electromyogram (EMG)
- electrical activity in brain: electroencephalogram (EEG)
- some difficulty in interpreting these physiological responses - more research needed

Choosing an Evaluation Method

 when in process: design vs. implementation

 style of evaluation: laboratory vs. field

 how objective: subjective vs. objective

 type of measures: qualitative vs. quantitative

 level of information: high level vs. low level

 level of interference: obtrusive vs. unobtrusive

 resources available: time, subjects, equipment, expertise

Evaluation Techniques

What is evaluation?

Goals of evaluation

Evaluation through expert analysis

Evaluation through user participation

Choosing an evaluation method

Universal design

Universal design principles

Multi-modal interaction

7. Evaluation Techniques

 Evaluation tests the usability, functionality and acceptability of an interactive system.

 Evaluation may take place:

– in the laboratory

– in the field.

 Some approaches are based on expert evaluation:

– analytic methods

– review methods

– model-based methods.

 Some approaches involve users:

– experimental methods

– observational methods

– query methods.

 An evaluation method must be chosen carefully and must be suitable for the job.

What Is Evaluation?

Even if such a process is used, we still need to assess our designs and test our systems to ensure that they actually behave as we expect and meet user requirements. This is the role of evaluation.

Goals of Evaluation

Evaluation has three main goals:

 to assess the extent and accessibility of the system's functionality,

 to assess users' experience of the interaction, and to identify any specific problems with the system.

Evaluation at this level may also include measuring the user's performance with the system, to assess the effectiveness of the system in supporting the task. In addition to evaluating the 1system design in terms of its functional capabilities, it is important to assess the user's experience of the interaction and its impact upon him.

The final goal of evaluation is to identify specific problems with the design. These may be aspects of the design which, when used in their intended context, cause unexpected results, or confusion amongst users.

Evaluation Through Expert Analysis

If the design itself can be evaluated, expensive mistakes can be avoided, since the design can be altered prior to any major resource commitments. Typically, the later in the design process that an error is discovered, the more costly it is to put right and, therefore, the less likely it is to be rectified.

These depend upon the designer, or a human factors expert, taking the design and assessing the impact that it will have upon a typical user. The basic intention is to identify any areas that are likely to cause difficulties because they violate known cognitive principles, or ignore accepted empirical results.

We will consider four approaches to expert analysis:

- cognitive walkthrough,
- heuristic evaluation,
- the use of models and
- use of previous work.

Cognitive walkthrough

The origin of the cognitive walkthrough approach to evaluation is the code walk through familiar in software engineering. Walkthroughs require a detailed review of a sequence of actions. In the code walkthrough, the sequence represents a segment of the program code that is stepped through by the reviewers to check certain characteristics(for example, that coding style is adhered to, conventions for spelling variables versus procedure calls, and to check that system wide invariants are not violated).

In the cognitive walkthrough, the sequence of actions refers to the steps that an interface will require a user to perform in order to accomplish some known task.

To do a walkthrough (the term walkthrough from now on refers to the cognitive walkthrough, and not to any other kind of walkthrough), you need four things:

1. A specification or prototype of the system. It doesn't have to be complete, but it should be fairly detailed. Details such as the location and wording for a menu can make a big difference.

2. A description of the task the user is to perform on the system. This should be are presentative task that most users will want to do.

3. A complete, written list of the actions needed to complete the task with the proposed system.

24. An indication of who the users are and what kind of experience and knowledge the evaluators

can assume about them.

Heuristic evaluation

A heuristic is a guideline or general principle or rule of thumb that can guide a design decision or be used to critique a decision that has already been made. *Heuristic evaluation*, developed by Jakob Nielsen and Rolf Molich, is a method for structuring the critique of a system using a set of relatively simple and general heuristics.

The general idea behind heuristic evaluation is that several evaluators independently critique a system to come up with potential usability problems. It is important that there be several of these evaluators and that the evaluations be done independently.

Nielsen's ten heuristics are:

1. **Visibility of system status**
2. **Match between system and the real world**
3. **User control and freedom**
4. **Consistency and standards**
5. **Error prevention**
6. **Recognition rather than recall**
7. **Flexibility and efficiency of use**
8. **Aesthetic and minimalist design**
9. **Help users recognize, diagnose and recover from errors**
10. **Help and documentation**

Once each evaluator has completed their separate assessment, all of the problems are collected and the mean severity ratings calculated. The design team will then determine the ones that are the most important and will receive attention first.

Model-based evaluation

A third expert-based approach is the use of models. Certain cognitive and design models provide a means of combining design specification and evaluation into the same framework.

Dialog models can also be used to evaluate dialog sequences for problems, such as unreachable states, circular dialogs and complexity. Models such as state transition networks are useful for evaluating dialog designs prior to implementation.

Using previous studies in evaluation

Experimental psychology and human–computer interaction between them possess a wealth of experimental results and empirical evidence. Some of this is specific to a particular domain, but much deals with more generic issues and applies in a variety of situations.

3A final approach to expert evaluation exploits this inheritance, using previous results as evidence to support (or refute) aspects of the design. It is expensive to repeat experiments continually and an expert review of relevant literature can avoid the need to do so. It should be noted that experimental results cannot be expected to hold arbitrarily across contexts.

Evaluation through user Participation

The techniques we have considered so far concentrate on evaluating a design or system through analysis by the designer, or an expert evaluator, rather than testing with actual users. However, useful as these techniques are for filtering and refining the design, they are not a replacement for actual usability testing with the people for whom the system is intended: the users.

These include:

- empirical or experimental methods,
- observational methods,
- query techniques, and
- methods that use physiological monitoring, such as eye tracking and measures of heart rate and skin conductance.

Styles of evaluation

Before we consider some of the techniques that are available for evaluation with users, we will distinguish between two distinct evaluation styles: those performed under laboratory

conditions and those conducted in the work environment or ‘in the field’.

Laboratory studies

In the first type of evaluation studies, users are taken out of their normal work environment to take part in controlled tests, often in a specialist usability laboratory (although the ‘lab’ may simply be a quiet room). This approach has a number of benefits and disadvantages.

A well-equipped usability laboratory may contain sophisticated audio/visual recording and analysis facilities, two-way mirrors, instrumented computers and the like, which cannot be replicated in the work environment.

There are, however, some situations where laboratory observation is the only option, for example, if the system is to be located in a dangerous or remote location, such as a space station. Also some very constrained single-user tasks may bead equate performed in a laboratory.

Field studies

The second type of evaluation takes the designer or evaluator out into the user’s work environment in order to observe the system in action. Again this approach has its pros and cons. 4High levels of ambient noise, greater levels of movement and constant interruptions, such as phone calls, all make field observation difficult. However, the very ‘open’ nature of the situation means that you will observe interactions between systems and between individuals that would have been missed in a laboratory study.

Empirical methods: experimental evaluation

One of the most powerful methods of evaluating a design or an aspect of a design is to use a controlled experiment. This provides empirical evidence to support a particular claim or hypothesis. It can be used to study a wide range of different issue sat different levels of detail. Any experiment has the same basic form. The evaluator chooses a hypothesis to test, which can be determined by measuring some attribute of participant behavior.

Participants

The choice of participants is vital to the success of any experiment. In evaluation experiments, participants should be chosen to match the expected user population as closely as possible. If participants are not actual users, they should be chosen to be of a similar age and level of education as the intended user group.

A second issue relating to the participant set is the sample size chosen. Often this is something that is determined by pragmatic considerations: the availability of participants is limited or resources are scarce.

Variables

Experiments manipulate and measure variables under controlled conditions, in order to test the hypothesis. There are two main types of variable: those that are 'manipulated' or changed (known as the independent variables) and those that are measured (the dependent variables). Independent variables are those elements of the experiment that are manipulated to produce different conditions for comparison. Examples of independent variables in evaluation experiments are interface style, level of help, number of menu items and icon design.

Dependent variables, on the other hand, are the variables that can be measured in the experiment, their value is 'dependent' on the changes made to the independent variable.

The dependent variable must be measurable in some way, it must be affected by the independent variable, and, as far as possible, unaffected by other factors. Common choices of dependent variable in evaluation experiments are the time taken to complete a task, the number of errors made, user preference and the quality of the user's performance.

Hypotheses

A hypothesis is a prediction of the outcome of an experiment. It is framed in terms of the independent and dependent variables, stating that a variation in the independent Variable will cause a difference in the dependent variable. The aim of the experiment is to show that this prediction is correct.

Experimental design

In order to produce reliable and generalizable results, an experiment must be carefully designed. We have already looked at a number of the factors that the experimenter must consider in the design, namely the participants, the independent and dependent variables, and the hypothesis.

The first phase in experimental design is to choose the hypothesis: to decide exactly what it is you are trying to demonstrate.

The next step is to decide on the *experimental method* that you will use. There are two main methods: *between-subjects* and *within-subjects*.

In a between-subjects(or *randomized*) design, each participant is assigned to a different condition. There are at least two conditions: the experimental condition (in which the variable has been manipulated) and the control, which is identical to the experimental condition except for this manipulation.

There may, of course, be more than two groups, depending on the number of independent variables and the number of levels that each variable can take.

The advantage of a between-subjects design is that any learning effect resulting from the user performing in one condition and then the other is controlled: each user performs under only one condition.

The disadvantages are that a greater number of participants are required, and that significant variation between the groups can negate any results. Also, individual differences between users can bias the results.

The second experimental design is within-subjects (or *repeated measures*). Here each user performs under each different condition. This design can suffer from transfer of learning effects, but this can be lessened if the order in which the conditions are tackled is varied between users. There is also less chance of effects from variation between participants.

Statistical Measures

The first two rules of statistical analysis are to *look* at the data and to *save* the data. It is

easy to carry out statistical tests blindly when a glance at a graph, histogram or table of results 6would be more instructive. In particular, looking at the data can expose *outliers*, single data items that are very different from the rest.

Variables can be classified as either *discrete variables* or *continuous variables*. A discrete variable can only take a finite number of values or *levels*, for example, a screen color that can be red, green or blue.

A third sort of test is the contingency table, where we classify data by several discrete attributes and then count the number of data items with each attribute combination.

Examples of questions one might ask about the data are as follows:

Is there a difference?

How big is the difference?

How accurate is the estimate?

Identify your hypothesis, participant group, dependent and independent variables, experimental design, task and analysis approach.

Answer The following is only an example of the type of experiment that might be devised.

Participants Taken from user population.

Hypothesis Color coding will make selection more accurate.

IV (Independent Variable) Color coding.

DV (Dependent Variable) Accuracy measured as number of errors.

Design Between-groups to ensure no transfer of learning (or within-groups with appropriate safeguards if participants are scarce).

Task The interfaces are identical in each of the conditions, except that, in the second, color is added to indicate related menu items.

Analysis *t* test.

Studies of groups of users

So far we have considered the experimental evaluation of single-user systems.

Experiments to evaluate elements of group systems bring additional problems. Given the complexities of human–human communication and group working, it is hardly surprising that experimental studies of groups and of groupware are more difficult than the corresponding single-user experiments already considered.

The participant groups To organize, say, 10 experiments of a single-user system requires 10 participants.

The experimental task Choosing a suitable task is also difficult. We may want to test a variety of different task types: creative, structured, information passing, and so on. Also, the tasks must encourage active cooperation, either because the task requires consensus, or because information and control is distributed among the participants.

7***Data gathering*** Even in a single-user experiment we may well use several video cameras as well as direct logging of the application. In a group setting this is replicated for each participant. So for a three-person group, we are trying to synchronize the recording of six or more video sources and three keystroke logs.

Field studies with groups There are, of course, problems with taking groups of users and putting them in an experimental situation. If the groups are randomly mixed, then we are effectively examining the process of group formation, rather than that of a normal working group.

Observational Techniques

A popular way to gather information about actual use of a system is to observe users interacting with it.

Think aloud and cooperative evaluation

Think aloud process has a number of advantages:

- the process is less constrained and therefore easier to learn to use by the evaluator
- the user is encouraged to criticize the system
- the evaluator can clarify points of confusion at the time they occur and so maximize

the effectiveness of the approach for identifying problem areas.

The usefulness of think aloud, cooperative evaluation and observation in general is largely dependent on the effectiveness of the recording method and subsequent analysis. The record of an evaluation session of this type is known as a *protocol*, and there are a number of methods from which to choose.

Protocol Analysis

Methods for recording user actions include the following:

Paper and pencil This is primitive, but cheap, and allows the analyst to note interpretations and extraneous events as they occur. However, it is hard to get detailed information, as it is limited by the analyst's writing speed.

Audio recording This is useful if the user is actively 'thinking aloud'. However, it may be difficult to record sufficient information to identify exact actions in later analysis, and it can be difficult to match an audio recording to some other form of protocol (such as a handwritten script).

Video recording This has the advantage that we can see *what* the participant is doing (*as long as* the participant stays within the range of the camera).

Computer logging It is relatively easy to get a system automatically to record user actions at a keystroke level, particularly if this facility has been considered early in the design.

8**User notebooks** The participants themselves can be asked to keep logs of activity / problems. This will obviously be at a very coarse level – at most, records every few minutes and, more likely, hourly or less.

Automatic Protocol Analysis Tools

Analyzing protocols, whether video, audio or system logs, is time consuming and tedious by hand. It is made harder if there is more than one stream of data to synchronize. One solution to this problem is to provide automatic analysis tools to support the task.

Query Techniques

Another set of evaluation techniques relies on asking the user about the interface directly.

Query techniques can be useful in eliciting detail of the user's view of a system. They embody the philosophy that states that the best way to find out how a system meets user requirements is to 'ask the user'.

There are a number of styles of question that can be included in the questionnaire. These include the following:

General These are questions that help to establish the background of the user and his place within the user population. They include questions about age, sex, occupation, place of residence, and so on.

Open-ended These ask the user to provide his own unprompted opinion on a question, for example 'Can you suggest any improvements to the interface?'.

Scalar These ask the user to judge a specific statement on a numeric scale, usually corresponding to a measure of agreement or disagreement with the statement.

Multi-choice Here the respondent is offered a choice of explicit responses, and may be asked to select only one of these, or as many as apply.

Ranked These place an ordering on items in a list and are useful to indicate a user's preferences.

Answer Assume that all users have used both systems.

Questionnaire

Consider the following questions in designing the questionnaire:

- what information is required?
- how is the questionnaire to be analyzed?

You are particularly interested in user preferences so questions should focus on different aspects of the systems and try to measure levels of satisfaction. The use of scales will make responses for each system easier to compare.

Evaluation through monitoring physiological responses

One of the problems with most evaluation techniques is that we are reliant on

Observation and the users telling us what they are doing and how they are feeling. What if we

were able to measure these things directly? Interest has grown recently in the use of what is sometimes called objective usability testing, ways of monitoring physiological aspects of computer use.

Eye tracking for usability evaluation

There are many possible measurements related to usability evaluation including:

Number of fixations

Fixation duration

Scan path

Physiological Measurements

Physiological measurement involves attaching various probes and sensors to the user

These measure a number of factors:

Heart activity,

Activity of the sweat glands

Electrical activity in muscle

Electrical activity in the brain

Choosing an Evaluation Method

Factors Distinguishing Evaluation Techniques

We can identify at least eight factors that distinguish different evaluation techniques and therefore help us to make an appropriate choice. These are:

- the stage in the cycle at which the evaluation is carried out
- the style of evaluation
- the level of subjectivity or objectivity of the technique
- the type of measures provided
- the information provided
- the immediacy of the response
- the level of interference implied

 the resources required.

1. Design vs. implementation

2. Laboratory vs. field studies

3. Subjective vs. objective

4. Qualitative vs. quantitative measures

5. Information provided

6. Immediacy of response

7. Intrusiveness

8. Resources

Universal Design

 Universal design is about designing systems so that theycan be used by anyone in any circumstance.

 Multi-modal systems are those that use more than onehuman input channel in the interaction.

 These systems may, for example, use:

– speech

– non-speech sound

– touch

– handwriting

– gestures.

 Universal design means designing for diversity, including:

– people with sensory, physical or cognitive impairment

– people of different ages

– people from different cultures and backgrounds.

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

Universal Design Principles

In the late 1990s a group at North Carolina State University in the USA proposed seven general principles of universal design. These were intended to cover all areas of design and are equally applicable to the design of interactive systems. These principles give us a framework in which to develop universal designs.

- *equitable use*: the design is useful to people with a range of abilities and appealing to all. No user is excluded or stigmatized. Where appropriate, security, privacy and safety provision should be available to all.
- *flexibility in use*: the design allows for a range of ability and preference, through choice of methods of use and adaptivity to the user's pace, precision and custom.
- *simple and intuitive to use*, regardless of the knowledge, experience, language or level of concentration of the user.
- *perceptible information*: the design should provide effective communication of information regardless of the environmental conditions or the user's abilities. Presentation should support the range of devices and techniques used to access information by people with different sensory abilities.

tolerance for error: minimizing the impact and damage caused by mistakes or unintended behavior. Potentially dangerous situations should be removed or made hard to reach. Potential hazards should be shielded by warnings.

- *low physical effort*: systems should be designed to be comfortable to use, minimizing physical effort and fatigue. The physical design of the system should allow the user to

maintain a natural posture with reasonable operating effort.

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

Multi-Model Interaction

In addition, such multi-sensory or multi-modal systems support the principle of redundancy required for universal design, enabling users to access the system using the mode of interaction that is most appropriate to their abilities.

The majority of interactive computer systems are predominantly visual in their interactive properties; often WIMP based, they usually make use of only rudimentary sounds while adding more and more visual information to the screen.

By utilizing the other sensory channels, the visual channel can be relieved of the pressure of providing all the information required and so interaction should improve.

The use of multiple sensory channels increases the *bandwidth* of the interaction between the human and the computer, and it also makes human–computer interaction more like the interaction between humans and their everyday environment, perhaps making the use of such systems more natural.

Usable sensory inputs

In computing, the visual channel is used as the predominant channel for communication, but if we are to use the other senses we have to consider their suitability and the nature of the information that they can convey.

Sound is already used, to a limited degree, in many interfaces: beeps are used as warnings and notification, recorded or synthesized speech and music are also used. Tactile feedback, as we have already seen, is also important in improving interactivity and so this represents another sense that we can utilize more effectively.

Sound in the interface

Sound is an important contributor to usability. There is experimental evidence to suggest

that the addition of audio confirmation of modes, in the form of changes in key clicks, reduces errors. Video games offer further evidence, since experts tend to score less well when the sound is turned off than when it is on; they pick up vital clues and information from the sound while concentrating their visual attention on different things.

Speech in the interface

Language is rich and complex. We learn speech naturally as children 'by example' –by listening to and mimicking the speech of those around us. This complexity makes speech recognition and synthesis by computer very difficult.

Structure of speech If we are fully to appreciate the problems involved with the computer-based recognition and generation of speech, we need first to understand the basic structure of speech. The English language is made up of 40 *phonemes*, which are the atomic elements of speech. Each phoneme represents a distinct sound, there being 24 consonants and 16 vowel sounds.

Speech recognition There have been many attempts at developing speech recognition systems, but, although commercial systems are now commonly and cheaply available, their success is still limited to single-user systems that require considerable training.

Speech synthesis Complementary to speech recognition is speech synthesis. The notion of being able to converse naturally with a computer is an appealing one for many users, especially those who do not regard themselves as computer literate, since it reflects their natural, daily medium of expression and communication.

Un interpreted speech Speech does not have to be recognized by a computer to be useful in the interface.

UNIT 5

Cognitive models

Cognitive models represent users of interactive systems.
Hierarchical models represent a user's task and goal structure.
Linguistic models represent the user-system grammar.
Physical and device models represent human motor skills.
Cognitive modeling is an area of computer science that deals with simulating human problem-solving and mental processing in a computerized model. Such a model can be used to simulate or predict human behavior or performance on tasks similar to the ones modeled and improve human-computer interaction.
GOMS is a theory of the cognitive skills involved in human-computer tasks. ... For a given task , a particular GOMS structure can be constructed and used to predict the time required to complete the task.
In addition, the model can be used to identify and predict the effects of errors on task performance.

GOMS is a family of predictive models of human performance that can be used to improve the efficiency of human-machine interaction by identifying and eliminating unnecessary user actions. GOMS stands for (Goals, Operators, Methods, and Selection).

GOMS is a theory of the cognitive skills involved in human-computer tasks. ... For a given task , a particular GOMS structure can be constructed and used to predict the time required to complete the task.
In addition, the model can be used to identify and predict the effects of errors on task performance.

Cognitive models

• goal and task hierarchies
• linguistic
• physical and device
• architecturalCognitive models
• They model aspects of user:
– understanding
– knowledge
– intentions
– processing
• Common categorisation:
– Competence vs. Performance
– Computational flavour
– No clear divide

Goal and task hierarchies

• Mental processing as divide-and-conquer
• Example: sales report
produce report
gather data
. find book names
. . do keywords search of names database
. . . . … *further sub-goals*
. . sift through names and abstracts by hand
. . . . … *further sub-goals*
. search sales database - further sub-goals
layout tables and histograms - further sub-goals
write description - further sub-goals

goals vs. tasks

• goals – intentions
what you would like to be true
• tasks – actions
how to achieve it

• GOMS
– goals are internal
• HTA
– actions external
– tasks are abstractions

Issues for goal hierarchies

• Granularity
– Where do we start?
– Where do we stop?
• Routine learned behaviour, not problem solving
– The unit task
• Conflict
– More than one way to achieve a goal
• ErrorTechniques
• Goals, Operators, Methods and Selection (GOMS)
• Cognitive Complexity Theory (CCT)
• Hierarchical Task Analysis (HTA) -

GOMS

Goals
– what the user wants to achieve
Operators
– basic actions user performs
Methods
– decomposition of a goal into subgoals/operators
Selection
– means of choosing between competing methods

GOMS example

GOAL: CLOSE-WINDOW
. [select GOAL: USE-MENU-METHOD
. MOVE-MOUSE-TO-FILE-MENU
. PULL-DOWN-FILE-MENU
. CLICK-OVER-CLOSE-OPTION
GOAL: USE-CTRL-W-METHOD
. PRESS-CONTROL-W-KEYS]
For a particular user:
Rule 1: Select USE-MENU-METHOD unless another rule applies
Rule 2: If the application is GAME, select CTRL-W-METHOD

Cognitive Complexity Theory

• Two parallel descriptions:
– User production rules
– Device generalised transition networks
• Production rules are of the form:
– if condition then action
• Transition networks covered under dialogue models

Example: editing with vi

• Production rules are in long-term memory
• Model working memory as attribute-value mapping:
(GOAL perform unit task)
(TEXT task is insert space)

(TEXT task is at 5 23)
(CURSOR 8 7)
• Rules are pattern-matched to working memory,
e.g., LOOK-TEXT task is at %LINE %COLUMN
is true, with LINE = 5 COLUMN = 23.Active rules:
SELECT-INSERT-SPACE
INSERT-SPACE-MOVE-FIRST
INSERT-SPACE-DOIT
INSERT-SPACE-DONE

Four rules to model inserting a space

New working memory
(GOAL insert space)
(NOTE executing insert space)
(LINE 5) (COLUMN 23)
SELECT-INSERT-SPACE
matches current working memory
(SELECT-INSERT-SPACE
IF (AND (TEST-GOAL perform unit task)
(TEST-TEXT task is insert space)
(NOT (TEST-GOAL insert space))
(NOT (TEST-NOTE executing insert space)))
THEN ((ADD-GOAL insert space)
(ADD-NOTE executing insert space)
(LOOK-TEXT task is at %LINE %COLUMN)))

Notes on CCT

• Parallel model
• Proceduralisation of actions
• Novice versus expert style rules
• Error behaviour can be represented
• Measures
– depth of goal structure
– number of rules
– comparison with device description

Problems with goal hierarchies

• a post hoc technique
• expert versus novice
• How cognitive are they?

Linguistic notations

• Understanding the user's behaviour and cognitive difficulty based on analysis of language between user and system.
• Similar in emphasis to dialogue models
• Backus–Naur Form (BNF)
• Task–Action Grammar (TAG)

Backus-Naur Form (BNF)

• Very common notation from computer science
• A purely syntactic view of the dialogue
• Terminals
– lowest level of user behaviour
– e.g. CLICK-MOUSE, MOVE-MOUSE
• Nonterminals
– ordering of terminals
– higher level of abstraction
– e.g. select-menu, position-mouse

Example of BNF

• Basic syntax:

– nonterminal ::= expression

• An expression

– contains terminals and nonterminals

– combined in sequence (+) or as alternatives (|)

draw line ::= select line + choose points + last point

select line ::= pos mouse + CLICK MOUSE

choose points ::= choose one | choose one + choose points

choose one ::= pos mouse + CLICK MOUSE

last point ::= pos mouse + DBL CLICK MOUSE

pos mouse ::= NULL | MOVE MOUSE+ pos mouse

Measurements with BNF

• Number of rules (not so good)

• Number of + and | operators

• Complications

– same syntax for different semantics

– no reflection of user's perception

– minimal consistency checking

Task Action Grammar (TAG)

• Making consistency more explicit

• Encoding user's world knowledge

• Parameterised grammar rules

• Nonterminals are modified to include additional semantic features

Consistency in TAG

• In BNF, three UNIX commands would be described as:

copy ::= cp + filename + filename | cp + filenames + directory

move ::= mv + filename + filename | mv + filenames + directory

link ::= ln + filename + filename | ln + filenames + directory

• No BNF measure could distinguish between this and a less consistent grammar in which

link ::= ln + filename + filename | ln + directory + filenames

Consistency in TAG (cont'd)

• consistency of argument order made explicit using a parameter, or semantic feature for file operations

• Feature Possible values

Op = copy; move; link

• Rules

file-op[Op] ::= command[Op] + filename + filename

| command[Op] + filenames + directory

command[Op = copy] ::= cp

command[Op = move] ::= mv

command[Op = link] ::= ln

Other uses of TAG

• User’s existing knowledge

• Congruence between features and commands

• These are modelled as derived rules

Physical and device models

• The Keystroke Level Model (KLM)

• Buxton's 3-state model

• Based on empirical knowledge of human motor system

• User's task: acquisition then execution.
– these only address execution
• Complementary with goal hierarchies

Keystroke Level Model (KLM)

• lowest level of (original) GOMS
• six execution phase operators
– Physical motor:
K - keystroking
P - pointing
H - homing
D - drawing
– Mental
M - mental preparation
– System
R - response
• times are empirically determined.
Texecute = TK + TP + TH + TD + TM + TR

KLM example

GOAL: ICONISE-WINDOW
[select
GOAL: USE-CLOSE-METHOD
. MOVE-MOUSE-TO- FILE-MENU
. PULL-DOWN-FILE-MENU
. CLICK-OVER-CLOSE-OPTION
GOAL: USE-CTRL-W-METHOD
PRESS-CONTROL-W-KEY]
• compare alternatives:
• USE-CTRL-W-METHOD vs.
• USE-CLOSE-METHOD
• assume hand starts on mouse

USE-CLOSE-METHOD

P[to menu]
1.1
B[LEFT down] 0.1
M
1.35
P[to option]
1.1
B[LEFT up]
0.1
Total
3.75 s

USE-CTRL-W-METHOD

H[to kbd]
0.40
M
1.35
K[ctrlW key] 0.28
Total
2.03 s

Architectural models

• All of these cognitive models make assumptions about the architecture of the human mind.
• Long-term/Short-term memory

• Problem spaces
• Interacting Cognitive Subsystems
• Connectionist
• ACT

Display-based interaction

• Most cognitive models do not deal with
user observation and perception
• Some techniques have been extended
to handle system output
(e.g., BNF with sensing terminals, Display-TAG)
but problems persist
• Exploratory interaction versus planning

History of GOMS:

Developed in 1983 by Stuart Card, Thomas P. Moran and Allen Newell.
Explained in their book "The Psychology of Human Computer Interaction"

Goals - what the user intends to accomplish
Operators - are actions that are performed to reach the goal.
Methods - are sequences of operators that accomplish a goal.
Selections - there can be more than one
method available to accomplish a single goal;
used to describe when a user would select a
certain method over the others. These can be very wide reaching; from very
high-level goals (e.g. Write a book) to very
low-level goals (e.g. Type the word 'Red').
Higher-level goals can be divided into smaller
lower-level goals. Operators are the simple actions that are used to
accomplish your goals (e.g. 'Left-click mouse button'
or 'Press ALT').
Operators cannot break down any further: they are
atomic elements (similar to those found in a
database).
Generally, it's assumed that each operator requires a
fixed quantity of time to perform the action and this
time interval is independent of context.
For example, to double left click a mouse button takes
0.40 seconds of execution time, regardless of what
you happen to be clicking on.
Methods are procedures that describe how to accomplish goals.
A method is essentially the steps a user has to
take to complete a specified task.
For instance, one method to accomplish the goal 'Highlight Word' in a Windows text editor would be to 'move cursor' to the beginning of the word and 'hold shift and right arrow key'. Another method to accomplish the same goal could involve 'holding down the left mouse button' and 'dragging to the beginning of the word'.
Selection rules specify which method is best to use when completing a goal, based on the given context.
Since there could be several ways of achieving the same result, a selection rule utilises the user's knowledge of the best method to achieve the required goal.
Selection rules generally take the form of a conditional statement, such as "IF the word to highlight is less than five characters USE the arrow keys and shift ELSE the mouse dragging method".

GOSM :

Quantitatively, GOMS offer good predictive models of performance time and learning. For example, when choosing between two systems you can apply a GOMS model.

Application 1 has a lower start-up costs, but will be slower to perform frequent tasks; Application 2 will be faster to perform tasks, but has a longer learning time, etc.
With these quantitative predictions, you can examine such tradeoffs in the light of what is important to your company.
Qualitatively, GOMS can be used to design training programs and help systems.
This approach has been shown to be an efficient way to organise help systems, tutorials, and training programs as well as user documentation.

NGOMS:

'Natural GOMS Language' allows a more flexible representation of a task using 'human' language:
Method for goal: Deleting Icon
Step 1. Select Icon For Deletion (1.10 sec)
Step 2. Drag Icon To Trashcan
(1.10 sec)
Step 3. Update User with audio cue (0.22 sec)
This NGOMSL model predicts that it will take 2.42 seconds to delete an icon.

Problems with GOMS:

The model assumes a certain level of skill – it cannot accurately be applied to beginners.
The model doesn't take into account time for learning the system or remembering how to use it after a long period of disuse; for example, can you remember where all the options are in Windows 98?
The model removes human error from the equation; even highly skilled users make the occasional mistake!
Mental workload is not addressed in the model; it's far more taxing when remembering a longer process than a short one. E.g. It's a far less stressful task to highlight text than entering network settings manually.
Users can get tired; you're not going to be as quick typing after three hours as you were when you started!
Differences among users is not accounted for within the model – e.g. Those who are left handed aren't given special preference.
Predicting whether a system will be functional or acceptable for users is not included in the model. E.g. Just because workers can enter data quickly into a new database system, it doesn't mean it's particularly 'user friendly' and 'easy to work with'.
How computer-assisted technology integrates into everyday business is not addressed in the model. For example, entering commands using a keyboard could be the quickest method of data capture for astronauts, however, trying to type in 'zero gravity' may be difficult.

Ubiquitous computing:

Ubiquitous computing (or "ubicomp") is a concept in software engineering and computer science where computing is made to appear anytime and everywhere.
Ubiquitous computing aims to permeate and interconnect all areas of life, and thus to enable a ubiquitous flow of data, information
Ubiquitous computing is a paradigm in which the processing of information is linked with each activity or object as encountered. It involves connecting electronic devices, including embedding microprocessors to communicate information. Devices that use ubiquitous computing have constant availability and are completely connected.
Ubiquitous computing focuses on learning by removing the complexity of computing and increases efficiency while using computing for different daily activities.
The main focus of ubiquitous computing is the creation of smart products that are connected, making communication and the exchange of data easier and less obtrusive.

Key features of ubiquitous computing include:

Consideration of the human factor and placing of the paradigm in a human, rather than computing, environment
Use of inexpensive processors, thereby reducing memory and storage requirements
Capturing of real-time attributes
Totally connected and constantly available computing devices
Focus on many-to-many relationships, instead of one-to-one, many-to-one or one-to-many in the environment, along with the idea of technology, which is constantly present
Includes local/global, social/personal, public/private and invisible/visible features and considers knowledge creation, as well as information dissemination
Relies on converging Internet, wireless technology and advanced electronics
Increased surveillance and possible restriction and interference in user privacies, as the digital devices are wearable and constantly connected
As technology progresses, the reliability factor of the different equipment used may be impacted

Wood:

Ambient Wood is an innovative educational experience exploring how biological ideas can be learnt in a real environment augmented by technology.

Augmented reality is increasingly used through smartphones, tablets and head-mounted displays with the goal of enhancing the real world with virtual elements in an interactive manner.

Whereas virtual reality replaces what people see and experience, augmented reality actually adds to it. Using devices such as HTC Vive, Oculus Rift, and Google Cardboard, VR covers and replaces users' field of vision entirely, while AR projects images in front of them in a fixed area.

Augmented reality (AR) adds digital elements to a live view often by using the camera on a smartphone. ... Virtual reality (VR) implies a complete immersion experience that shuts out the physical world.

Various technologies are used in augmented reality rendering, including optical projection systems, monitors, handheld devices, and display systems, which are worn on the human body. A head-mounted display (HMD) is a display device worn on the forehead, such as a harness or helmet-mounted.

Augmented reality (AR) is a technology that lets people superimpose digital content (images, sounds, text) over a real-world environment. ... In other words, AR is on the verge of becoming a part of everyday life.

IKEA Mobile App. ...
Nintendo's Pokémon Go App. ...
Google Pixel's Star Wars Stickers. ...
Disney Coloring Book. ...
L'Oréal Makeup App. ...
Weather Channel Studio Effects. ...
U.S. Army.

There are 3 primary categories of virtual reality simulations used today: non-immersive, semi-immersive, and fully-immersive simulations.