UNIT I

# FOUNDATIONS OF HCI

**The Human: I/O channels – Memory – Reasoning and problem solving; The computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms.**

## 1. HUMAN

### 1.1 Introduction

Human-computer interaction (commonly referred to as HCI) researches the design and use of computer technology, focused on the interfaces between people (users) and computers. Researchers in the field of HCI both observe the ways in which humans interact with computers and design technologies that let humans interact with computers in novel ways.

**User**

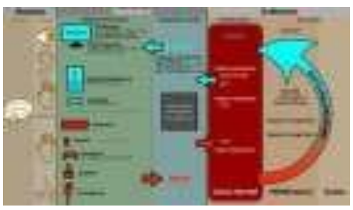
By "user", we may mean an individual user, a group of users working together. An appreciation of the way people's sensory systems (sight, hearing, touch) relay information is vital. Also, different users form different conceptions or mental models about their interactions and have different ways of learning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

**Computer**

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

**Interaction**

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



**The Goals of HCI**

The goals of HCI are to produce usable and safe systems, as well as functional systems. In order o produce computer systems with good usability, developers must attempt to: understand the factors that determine how people use technology, develop tools and techniques to enable building suitable systems, achieve efficient, effective, and safe interaction put people first.

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

**Usability**

Usability is one of the key concepts in HCI. It is concerned with making systems easy to learn and use. A usable system is:

- easy to learn
- easy to remember how to use
- effective to use
- efficient to use
- safe to use
- enjoyable to use

**Factors in HCI**

There are a large number of factors which should be considered in the analysis and design of a system using HCI principles. Many of these factors interact with each other, making the analysis even more complex. The main factors are listed in the table below:

**Organisation Factors**

- Training, job design, politics, roles, work organisation
- Environmental Factors
- Noise, heating, lighting, ventilation
- Health and Safety Factors

**The User**

- Cognitive processes and capabilities
- Motivation, enjoyment, satisfaction, personality, experience
- Comfort Factors
- Seating, equipment, layout.

**User Interface**

Input devices, output devices, dialogue structures, use of colour, icons, commands, navigation, graphics, natural language, user support, multimedia,

**Task Factors** : Easy, complex, novel, task allocation, monitoring, skills

**Constraints** : Cost, timescales, budgets, staff, equipment, buildings

**System Functionality:** Hardware, software, application

**Productivity Factors :** Increase output, increase quality, decrease costs, decrease errors, increase innovation

**Disciplines contributing to HCI**

The field of HCI covers a wide range of topics, and its development has relied on contributions from many disciplines. Some of the main disciplines which have contributed to HCI are:

**Computer Science**

- technology
- software design, development & maintenance
- User Interface Management Systems (UIMS) & User Interface Development Environments (UIDE)
- prototyping tools
- graphics

**Cognitive Psychology**

- information processing
- capabilities
- limitations
- cooperative working
- performance prediction

**Social Psychology**

- social & organizational structures

**Ergonomics/Human Factors**

- hardware design
- display readability

**Linguistics**

- natural language interfaces

**Artificial Intelligence**

- **intelligent software**

**Engineering & Design**

- graphic design

- engineering principles

## 1.2 INPUT–OUTPUT CHANNELS

A person_s interaction with the outside world occurs through information being received and sent: input and output. In an interaction with a computer the user receives information that is output by the computer, and responds by providing input to the computer – the user_s output becomes the computer_s input and vice versa.

For example, sight may be used primarily in receiving information from the computer, but it can also be used to provide information to the computer, for example by fixating on a particular screen point when using an eyegaze system. Input in the human occurs mainly through the senses and output through the motor control of the effectors.

There are five major senses: **sight, hearing, touch, taste and smell**. Of these, the first three are the most important to HCI. **Taste and smell** do not currently play a significant role in HCI, and it is not clear whether they could be exploited at all in general computer systems, although they could have a role to play in more specialized systems (smells to give warning of malfunction, for example) or in augmented reality systems. **vision, hearing** and touch are central.

There are a number of effectors, including the limbs, fingers, eyes, head and vocal system. In the interaction with the computer, the fingers play the primary role, through typing or mouse control, with some use of voice, and eye, head and body position.

Imagine using a personal computer (PC) with a mouse and a keyboard. The application you are using has a graphical interface, with menus, icons and windows. In your interaction with this system you receive information primarily by sight, from what appears on the screen.

### 1.2.1 Vision

Human vision is a highly complex activity with a range of physical and perceptual limitations, We can roughly divide visual perception into two stages: the physical reception of the stimulus from the outside world, and the processing and interpretation of that stimulus. On the one hand the physical properties of the eye and the visual system mean that there are certain things that cannot be seen by the human; on the other the interpretative capabilities of visual processing allow images to be constructed from incomplete information. We need to understand both stages as both influence what can and cannot be perceived visually by a human being, which in turn directly affects the way that we design computer systems. We will begin by looking at the eye as a physical receptor, and then go on to consider the processing involved in basic vision.

**The human eye**

Vision begins with light. The eye is a mechanism for receiving light and transforming it into electrical energy. Light is reflected from objects in the world and their image is focused upside down on the back of the eye. The receptors in the eye transform it into electrical signals which are passed to the brain.

The eye has a number of important components. The cornea and lens at the front of the eye focus the light into a sharp image on the back of the eye, the retina. The retina is light sensitive and contains two types of photoreceptor: rods and cones.

**Rods** are highly sensitive to light and therefore allow us to see under a low level of illumination. They are unable to resolve fine detail and are subject to light saturation. This is the reason for the temporary blindness we get when moving from a darkened room into sunlight: the rods have been active and are saturated by the sudden light. The **cones** do not operate either as they are suppressed by the rods. We are therefore temporarily unable to see at all. There are approximately 120 million rods per eye which are mainly situated towards the edges of the retina. Rods therefore dominate peripheral vision.

**Cones** are the second type of receptor in the eye. They are less sensitive to light than the rods and can therefore tolerate more light. There are three types of cone, each sensitive to a different wavelength of light. This allows color vision. The eye has approximately 6 million cones, mainly concentrated on the fovea, a small area of the retina on which images are fixated.

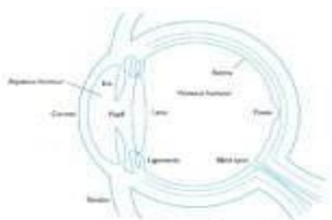


Figure: Human Eye

The retina is mainly covered with photoreceptors there is one blind spot where the optic nerve enters the eye. The blind spot has no rods or cones, our visual system compensates for this so that in normal circumstances we are unaware of it.

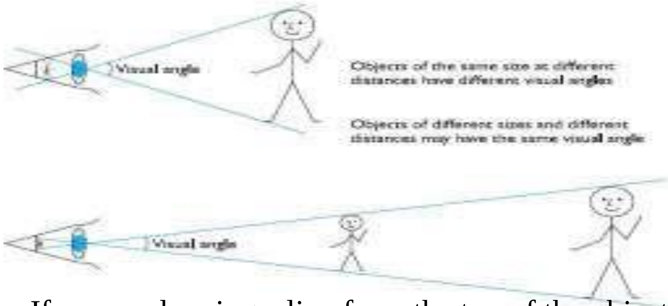
The retina also has specialized nerve cells called ganglion cells. There are two types:

X-cells, which are concentrated in the fovea and are responsible for the early detection of pattern; and Y-cells which are more widely distributed in the retina and are responsible for the early detection of movement. The distribution of these cells means that, while we may not be able to detect changes in pattern in peripheral vision, we can perceive movement.

**Visual perception**

The information received by the visual apparatus must be filtered and passed to processing elements which allow us to recognize coherent scenes, disambiguate relative distances and differentiate colour.

How does the eye perceive size, depth and relative distances? To understand this we must consider how the image appears on the retina. Reflected light from the object forms an upside-down image on the retina. The size of that image is specified as a visual angle. Figure illustrates how the visual angle is calculated.




If we are drawing a line from the top of the object to a central point on the front of the eye and a second line from the bottom of the object to the same point, the visual angle of the object is the angle between these two lines. Visual angle is affected by both the size of the object and its distance from the eye. Therefore if two objects are at the same distance, the larger one will have the larger visual angle. Similarly, if two objects of the same size are placed at different distances from the eye, the furthest one will have the smaller visual angle. The visual angle indicates how much of the field of view is taken by the object. The visual angle measurement is given in either degrees or minutes of arc, where 1 degree is equivalent to 60 minutes of arc, and 1 minute of arc to 60 seconds of arc.

**Perceiving brightness**

An aspect of visual perception is the perception of brightness. Brightness is in fact a subjective reaction to levels of light. It is affected by luminance which is the amount of light emitted by an object. The luminance of an object is dependent on the amount of light falling on the object_s surface and its reflective properties. Luminance is a physical characteristic and can be measured using a photometer. Contrast is related to luminance: it is a function of the luminance of an object and the luminance of its background.

**Perceiving colour** A third factor that we need to consider is perception of colour. Colour is usually regarded as being made up of three components: **hue, intensity and saturation**. Hue is determined by the spectral wavelength of the light. Blues have short wavelengths, greens medium and reds long. Approximately 150 different hues can be discriminated by the average person. Intensity is the brightness of the color, and saturation is the amount of whiteness in the color. By varying these two, we can perceive in the region of 7 million different colors.

**The capabilities and limitations of visual processing**

Visual processing involves the transformation and interpretation of a complete image, from the light that is thrown onto the retina. Visual processing compensates for the movement of the image on the retina which occurs as we move around and as the object which we see moves. Although the retinal image is moving, the image that we perceive is stable. Similarly, colour and brightness of objects are perceived as constant, in spite of changes in luminance.

This ability to interpret and exploit our expectations can be used to resolve ambiguity. For example, consider the image shown in Figure is an ambiguous shape

Now consider Figure_s below. The context in which the object appears allows our expectations to clearly disambiguate the interpretation of the object, as either a B or a 13.

ABC

12 13 14

Consider Figure below, which line is longer? Most people when presented with this will say that the top line is longer than the bottom. In fact, the two lines are the same length. This may be due to a false application of the law of size constancy: the top line appears like a concave edge, the bottom like a convex edge.

**Reading**

There are several stages in the reading process. First, the visual pattern of the word on the page is perceived. It is then decoded with reference to an internal representation of language. The final stages of language processing include syntactic and semantic analysis and operate on phrases or sentences.

During reading, the eye makes jerky movements called saccades followed by fixations. Perception occurs during the fixation periods, which account for approximately 94% of the time elapsed. The eye moves backwards over the text as well as forwards, in what are known as regressions. If the text is complex there will be more regressions.

Adults read approximately 250 words a minute. It is unlikely that words are scanned serially, character by character, since experiments have shown that words can be recognized as quickly as single characters. Instead, familiar words are recognized using word shape. This means that removing the word shape clues (for example, by capitalizing words) is detrimental to reading speed and accuracy. The speed at which text can be read is a measure of its legibility. Experiments have shown that standard font sizes of 9 to 12 points are equally legible, given proportional spacing between lines . Similarly line lengths of between 2.3 and 5.2 inches (58 and 132 mm) are equally legible. However, there is evidence that reading from a computer screen is slower than from a book . This is thought to be due to a number of factors including a longer line length, fewer words to a page, orientation and the familiarity of the medium of the page. These factors can of course be reduced by careful design of textual interfaces. a negative contrast (dark, characters on a light screen) provides higher luminance and, therefore, increased acuity, than a positive contrast. This will in turn increase legibility. Experimental evidence suggests that in practice negative contrast displays are preferred and result in more accurate performance.

### 1.2.2 Hearing

The sense of hearing is often considered secondary to sight, but we tend to underestimate the amount of information that we receive through our ears. hearing begins with vibrations in the air or sound waves. The ear receives these vibrations and transmits them, through various stages, to the auditory nerves. The ear comprises three sections, commonly known as the

**Outer ear, middle ear and inner ear.**

The outer ear is the visible part of the ear. It has two parts: the pinna, which is the structure that is attached to the sides of the head, and the auditory canal, along which sound waves are passed to the middle ear. The outer ear serves two purposes. First, it protects the sensitive middle ear from damage. The auditory canal contains wax which prevents dust, dirt and over-inquisitive insects reaching the middle ear. It also maintains the middle ear at a constant temperature. Secondly, the pinna and auditory canal serve to amplify some sounds.

The middle ear is a small cavity connected to the outer ear by the tympanic membrane, or ear drum, and to the inner ear by the cochlea. Within the cavity are the ossicles, the smallest bones in the body. Sound waves pass along the auditory canal and vibrate the ear drum which in turn vibrates the ossicles, which transmit the vibrations to the cochlea, and so into the inner ear. This _relay_ is required because, unlike the air-filled outer and middle ears, the inner ear is filled with a denser cochlean liquid. If passed directly from the air to the liquid, the transmission of the sound waves would be poor. By transmitting them via the ossicles the sound waves are concentrated and amplified.

**Processing sound**

Processing sound has a number of characteristics which we can differentiate. Pitch is the frequency of the sound. A low frequency produces a low pitch, a high frequency, a high pitch. Loudness is proportional to the amplitude of the sound; the frequency remains constant.

Timbre relates to the type of the sound: sounds may have the same pitch and loudness but be made by different instruments and so vary in timbre. We can also identify a sound_s location, since the two ears receive slightly different sounds, owing to the time difference between the sound reaching the two ears and the reduction in intensity caused by the sound waves reflecting from the head.

The human ear can hear frequencies from about 20 Hz to 15 kHz. It can distinguish frequency changes of less than 1.5 Hz at low frequencies but is less accurate at high frequencies. Different frequencies trigger activity in neurons in different parts of the auditory system, and cause different rates of firing of nerve impulses. The auditory system performs some filtering of the sounds received, allowing us to ignore background noise and concentrate on important information. The exception is multimedia, which may include music, voice commentary and sound effects. However, the ear can differentiate quite subtle sound changes and can recognize familiar sounds without concentrating attention on the sound source.

### 1.2.3 Touch

Touch provides us with vital information about our environment. It tells us when we touch something hot or cold, and can therefore act as a warning. It also provides us with feedback when we attempt to lift an object, for example. Consider the act of picking up a glass of water. If we could only see the glass and not feel when our hand made contact with it or feel its shape, the speed and accuracy of the action would be reduced. This is the experience of users of certain virtual reality games: they can see the computer-generated objects which they need to manipulate but they have no physical sensation of touching them. Watching such users can be an informative and amusing experience! Touch is therefore an important means of feedback, and this is no less so in using computer systems. Feeling buttons depress is an important part of the task of pressing the button. Also, we should be aware that, although for the average person, haptic perception is a secondary source of information, for those whose other senses are impaired, it may be vitally important. For such users, interfaces such as braille may be the primary source of information in the interaction. The apparatus of touch differs from that of sight and hearing in that it is not localized. The skin contains three types of sensory receptor: thermo receptors respond to heat and cold, nociceptors respond to intense pressure, heat and pain, and mechanoreceptors respond to pressure.

### 1.2.4 Movement

A simple action such as hitting a button in response to a question involves a number of processing stages. The stimulus (of the question) is received through the sensory receptors and transmitted to the brain. The question is processed and a valid response generated. The brain then tells the appropriate muscles to respond. Each of these stages takes time, which can be roughly divided into reaction time and movement time.

Movement time is dependent largely on the physical characteristics of the subjects: their age and fitness, for example. Reaction time varies according to the sensory channel through which the stimulus is received. A person can react to an auditory signal in approximately 150 ms, to a visual signal in 200 ms and to pain in 700 ms.

A second measure of motor skill is accuracy. One question that we should ask is whether speed of reaction results in reduced accuracy. This is dependent on the task and the user. In some cases, requiring increased reaction time reduces accuracy. This is the premise behind many arcade and video games where less skilled users fail at levels of play that require faster responses. Speed and accuracy of movement are important considerations in the design of interactive systems, primarily in terms of the time taken to move to a particular target on a screen. The target may be a button, a menu item or an icon, for example. The time taken to hit a target is a function of the size of the target and the distance that has to be moved. This is formalized in Fitts_ law . There are many variations of this formula, which have varying constants, but they are all very similar. One common form is

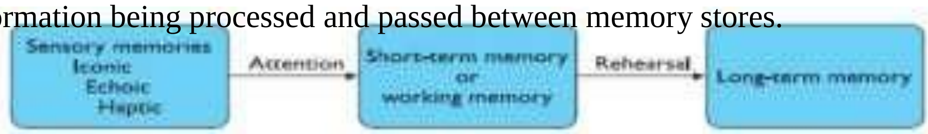
**Movement time = a + b log2(distance/size +1)**

where a and b are empirically determined constants.

**1.3 HUMAN MEMORY**

Our memory contains our knowledge of actions or procedures. It allows us to repeat actions, to use language, and to use new information received via our senses. It also gives us our sense of identity, by preserving information from our past experiences.

Memory is the second part of our model of the human as an information-processing system. Memory is associated with each level of processing. Bearing this in mind, we will consider the way in which memory is structured and the activities that take place within the system. It is generally agreed that there are three types of memory or memory function: sensory buffers, short-term memory or working memory, and long-term memory. There is some disagreement as to whether these are three separate systems or different functions of the same system. It is sufficient to note three separate types of memory. These memories interact, with information being processed and passed between memory stores.




.

Figure: A model of the structure of memory

**1.3.1 Sensory memory**

The sensory memories act as buffers for stimuli received through the senses. A sensory memory exists for each sensory channel: iconic memory for visual stimuli, echoic memory for aural stimuli and haptic memory for touch. These memories are constantly overwritten by new information coming in on these channels.

The existence of echoic memory is evidenced by our ability to ascertain the direction from which a sound originates. This is due to information being received by both ears. Since this information is received at different times, we must store the stimulus in the meantime. Echoic memory allows brief _play-back _of information. Information is passed from sensory memory into short-term memory by attention, thereby filtering the stimuli to only those which are of interest at a given time.

Attention is the concentration of the mind on one out of a number of competing stimuli or thoughts. It is clear that we are able to focus our attention selectively, choosing to attend to one thing rather than another. This is due to the limited capacity of our sensory and mental processes.

### 1.3.2 Short-term memory

Short-term memory or working memory acts as a _scratch-pad_for temporary recall of information. It is used to store information which is only required fleetingly. Short-term memory can be accessed rapidly, in the order of 70 ms. It also decays rapidly, meaning that information can only be held there temporarily, in the order of 200 ms. Short-term memory also has a limited capacity. There are two basic methods for measuring memory capacity. The first involves determining the length of a sequence which can be remembered in order. The second allows items to be freely recalled in any order.

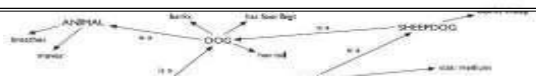
### 1.3.3 Long-term memory

If short-term memory is our working memory or _scratch-pad_, long-term memory is our main resource. Here we store factual information, experiential knowledge, procedural rules of behavior – in fact, everything that we _know_. It differs from short-term memory in a number of significant ways. First, it has a huge, if not unlimited, capacity. Secondly, it has a relatively slow access time of approximately a tenth of a second. Thirdly, forgetting occurs more slowly in long-term memory, if at all.

Long-term memory is intended for the long-term storage of information. Information is placed there from working memory through rehearsal. Unlike working memory there is little decay: long-term recall after minutes is the same as that after hours or days.

**Long-term memory structure**

There are two types of long-term memory: episodic memory and semantic memory. Episodic memory represents our memory of events and experiences in a serial form. It is from this memory that we can reconstruct the actual events that took place at a given point in our lives. Semantic memory, on the other hand, is a structured record of facts, concepts and skills



that we have acquired. The information in semantic memory is derived from that in our episodic memory.

**Long-term memory processes**

This process can be optimized in a number of ways. Ebbinghaus performed numerous experiments on memory, using himself as a subject. In these experiments he tested his ability to learn and repeat nonsense syllables, comparing his recall minutes, hours and days after the learning process. He discovered that the amount learned was directly proportional to the amount of time spent learning. This is known as the total time hypothesis. However, experiments by Baddeley and others suggest that learning time is most effective if it is distributed over time.

There are two main theories of forgetting: decay and interference. The first theory suggests that the information held in long-term memory may eventually be forgotten. Ebbinghaus concluded from his experiments with nonsense syllables that information in memory decayed logarithmically, that is that it was lost rapidly to begin with, and then more slowly. Jost‗s law, which follows from this, states that if two memory traces are equally strong at a given time the older one will be more durable.

The second theory is that information is lost from memory through interference. If we acquire new information it causes the loss of old information. This is termed retroactive interference. A common example of this is the fact that if you change telephone numbers, learning your new number makes it more difficult to remember your old number. This is because the new association masks the old. However, sometimes the old memory trace breaks through and interferes with new information. This is called proactive inhibition.

Forgetting is also affected by emotional factors. In experiments, subjects given emotive words and non-emotive words found the former harder to remember in the short term but easier in the long term.

First, proactive inhibition demonstrates the recovery of old information even after it has been ‗lost‗ by interference. Secondly, there is the ‗tip of the tongue‗ experience, which indicates that some information is present but cannot be satisfactorily accessed. Thirdly, information may not be recalled but may be recognized, or may be recalled only with prompting. This leads us to the third process of memory: information retrieval. Here we need to distinguish between two types of information retrieval, recall and recognition. In recall the information is reproduced from memory. In recognition, the presentation of the information provides the knowledge that the information has been seen before. Recognition is the less complex cognitive activity since the information is provided as a cue.

**1.4 THINKING: REASONING AND PROBLEM SOLVING**

Humans, on the other hand, are able to use information to reason and solve problems, and indeed do these activities when the information is partial or unavailable. Human thought is conscious and self-aware: while we may not always be able to identify the processes we use, we can identify the products of these processes, our thoughts. In addition, we are able to think about things of which we have no experience, and solve problems which we have never seen before.

Thinking can require different amounts of knowledge. Some thinking activities are much directed and the knowledge required is constrained. Others require vast amounts of knowledge from different domains. For example, performing a subtraction calculation requires a relatively small amount of knowledge, from a constrained domain, whereas understanding newspaper headlines demands.

**Reasoning**

Reasoning is the process by which we use the knowledge we have to draw conclusions or infer something new about the domain of interest. There are a number of different Types of reasoning: deductive, inductive and abductive. We use each of these types of reasoning in everyday life, but they differ in significant ways.

**Deductive reasoning**

Deductive reasoning derives the logically necessary conclusion from the given premises.

For example,

If it is Friday then she will go to work

It is Friday

Therefore she will go to work.

Inductive reasoning

Induction is generalizing from cases we have seen to infer information about cases we have not seen. Induction is a useful process, which we use constantly in learning about our environment. We can never see all the elephants that have ever lived or will ever live, but we have certain knowledge about elephants which we are prepared to trust for all practical purposes, which has largely been inferred by induction. Even if we saw an elephant without a trunk, we would be unlikely to move from our position that _All elephants have trunks_, since we are better at using positive than negative evidence.

**Abductive reasoning**

The third type of reasoning is abduction. Abduction reasons from a fact to the action or state that caused it. This is the method we use to derive explanations for the events we observe. For example, suppose we know that Sam always drives too fast when she has been drinking. If we see Sam driving too fast we may infer that she has been drinking. Of course, this too is unreliable since there may be another reason why she is driving fast: she may have been called to an emergency.

**Problem solving**

Human problem solving is characterized by the ability to adapt the information we have to deal with new situations often solutions seem to be original and creative. There are a number of different views of how people solve problems.

The Gestalt view that problem solving involves both reuse of knowledge and insight. This has been largely superseded but the questions it was trying to address remain and its influence can be seen in later research. In the 1970s by Newell and Simon, was the problem space theory, which takes the view that the mind is a limited information processor.

**Gestalt theory**

Gestalt psychologists were answering the claim, made by behaviorists, that problem solving is a matter of reproducing known responses or trial and error. This explanation was considered by the Gestalt school to be insufficient to account for human problem-solving behavior. Instead, they claimed, problem solving is both productive and reproductive. Reproductive problem solving draws on previous experience as the behaviorists claimed, but productive problem solving involves insight and restructuring of the problem. Indeed, reproductive problem solving could be a hindrance to finding a solution, since a person may _fixate_on the known aspects of the problem and so be unable to see novel interpretations that might lead to a solution. Gestalt psychologists backed up their claims with experimental evidence.

**Problem space theory**

Newell and Simon proposed that problem solving centers on the problem space. The problem space comprises problem states, and problem solving involves generating these states using legal state transition operators. The problem has an initial state and a goal state and people use the operators to move from the former to the latter. Such problem spaces may be huge, and so heuristics are employed to select appropriate operators to reach the goal. One such heuristic is means–ends analysis. In means–ends analysis the initial state is compared with the goal state and an operator chosen to reduce the difference between the two.

Newell and Simon_s theory, and their General Problem Solver model which is based on it, have largely been applied to problem solving in well-defined domains, for example solving puzzles. These problems may be unfamiliar but the knowledge that is required to solve them is present in the statement of the problem and the expected solution is clear. In real-world problems finding the knowledge required to solve the problem may be part of the problem, or specifying the goal may be difficult.

**Analogy in problem solving**

A third element of problem solving is the use of analogy. Similarities between the known domain and the new one are noted and operators from the known domain are transferred to the new one.

**Skill acquisition**

The entire problem solving that we have considered so far has concentrated on handling unfamiliar problems. A commonly studied domain is chess playing. It is particularly suitable since it lends itself easily to representation in terms of problem space theory. The initial state is the opening board position; the goal state is one player checkmating the other; operators to move states are legal moves of chess. It is therefore possible to examine skilled behavior within the context of the problem space theory of problem solving.

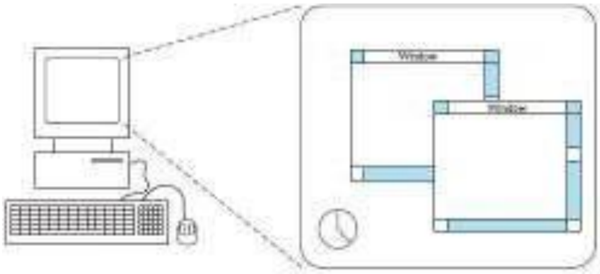
**Errors and mental models**

Human capability for interpreting and manipulating information is quite impressive. Some are trivial, resulting in no more than temporary inconvenience or annoyance. Others may be more serious, requiring substantial effort to correct.

**1.5 THE COMPUTER**

**A typical computer system**

There is the computer _box_itself, a keyboard, a mouse and a colour screen. The screen layout is shown alongside it. Data have to be entered into and obtained from a system, and there are also many different types of user, each with their own unique requirements.



**Levels of interaction – batch processing**

There was minimal interaction with the machine: the user would simply dump a pile of punched cards onto a reader, press the start button, and then return a few hours later. This still continues today although now with pre-prepared electronic files or possibly machine-read forms. With batch processing the interactions take place over hours or days. In contrast the typical desktop computer system has interactions taking seconds or fractions of a second. The field of Human– Computer Interaction largely grew due to this change in interactive pace.

**Richer interaction – everywhere, everyway**

Information appliances are putting internet access or dedicated systems onto the fridge, microwave and washing machine: to automate shopping, give you email in your kitchen or simply call for maintenance when needed. We carry with us WAP phones and smartcards, have security systems that monitor us and web cams that show our homes to the world.

### 1.5.1 Elements

A computer system comprises various elements, each of which affects the user of the system.

- Input devices for interactive use, allowing text entry, drawing and selection from the screen:
    - text entry: traditional keyboard, phone text entry, speech and handwriting
    - pointing: principally the mouse, but also touchpad, stylus and others
    - 3D interaction devices.
- Output display devices for interactive use:
    - different types of screen mostly using some form of bitmap display
    - large displays and situated displays for shared and public use
    - digital paper may be usable in the near future.
- Virtual reality systems and 3D visualization which have special interaction and display devices.
- Various devices in the physical world:
    - physical controls and dedicated displays
    - sound, smell and haptic feedback
    - sensors for nearly everything including movement, temperature, bio-signs.
- Paper output and input: the paperless office and the less-paper office:
    - different types of printers and their characteristics, character styles and fonts
    - Scanners and optical character recognition.

## 1.6 Memory

### 1.6.1 Short term memory (RAM)

At the lowest level of computer memory are the registers on the computer chip, but these having little impact on the user except in so far as they affect the general speed of the computer? Most currently active information is held in silicon-chip random access memory (RAM). Different forms of RAM differ as to their precise access times, power consumption and characteristics. Typical access times are of the order of 10 nanoseconds, that is a hundred-millionth of a second, and information can be accessed at a rate of around 100 Mbytes (million bytes) per second. Typical storage in modern personal computers is between 64 and 256 Mbytes. Most RAM is volatile, that is its contents are lost when the power is turned off. However, many computers have small amount of non-volatile RAM, which retains its contents, perhaps with the aid of a small battery. This may be used to store setup information in a large computer, but in a pocket organizer will be the whole memory. Non- volatile RAM is more expensive so is only used where necessary, but with many notebook computers using very low-power static RAM, the divide is shrinking. By strict analogy, non- volatile RAM ought to be classed as LTM, but the important thing we want to emphasize is the gulf between STM and LTM in a traditional computer system. In PDAs the distinctions become more confused as the battery power means that the system is never completely off, so RAM memory effectively lasts forever. Some also use flash memory, which is a form of silicon memory that sits between fixed content ROM (read-only memory) chips and normal RAM. Flash memory is relatively slow to write, but once written retains its content even with no power whatsoever. These are sometimes called silicon disks on PDAs. Digital cameras typically store photographs in some form of flash media and small flash-based devices are used to plug into a laptop or desktop‘s USB port to transfer data.

### 1.6.2 Disks and long-term memory (LTM)

For most computer users the LTM consists of disks, possibly with small tapes for backup. The existence of backups, and appropriate software to generate and retrieve them, is an important area for user security. However, we will deal mainly with those forms of storage that impact the interactive computer user. There are two main kinds of technology used in disks: magnetic disks and optical disks. The most common storage media, floppy disks and hard (or fixed) disks, are coated with magnetic material, like that found on an audio tape, on which the information is stored. Typical capacities of floppy disks lie between 300 kbytes and 1.4 Mbytes, but as they are removable, you can have as many as you have room for on your desk. Hard disks may store from under 40 Mbytes to several gigabytes (Gbytes) that is several thousand million bytes. With disks there are two access times to consider, the time taken to find the right track on the disk, and the time to read the track. The former dominates random reads, and is typically of the order of 10 ms for hard disks. The transfer rate once the track is found is then very high, perhaps several hundred kilobytes per second. Various forms of large removable media are also available, fitting somewhere between floppy disks and removable hard disks, and are especially important for multimedia storage. Optical disks use laser light to read and (sometimes) write the information on the disk. There are various high capacity specialist optical devices, but the most common is the CD-ROM, using the same technology as audio compact discs. CD-ROMs have a capacity of around 650 megabytes, but cannot be written to at all. They are useful for published material such as online reference books, multimedia and software distribution. Recordable CDs are a form of WORM device (write-once read-many) and are more flexible in that information can be written, but (as the name suggests) only once at any location – more like a piece of paper than a blackboard. They are obviously very useful for backups and for producing very secure audit information. Finally, there are fully rewritable optical disks, but the rewrite time is typically much slower than the read time, so they are still primarily for archival not dynamic storage. Many CD-ROM reader/writers can also read DVD format, originally developed for storing movies. Optical media are more robust than magnetic disks and so it is easier to use a jukebox arrangement, whereby many optical disks can be brought online automatically as required. This can give an online capacity of many hundreds of gigabytes. However, as magnetic disk capacities have grown faster than the fixed standard of CD-ROMs, some massive capacity stores are moving to large disk arrays.

**1.6.3 Compression**

In fact, things are not quite so bad, since compression techniques can be used to reduce the amount of storage required for text, bitmaps and video. All of these things are highly redundant. Consider text for a moment. In English, we know that if we use the letter ‗q‘ then ‗u‘ is almost bound to follow. At the level of words, some words like ‗the‘ and ‗and‘ appear frequently in text in general, and for any particular work one can find other common terms (this book mentions ‗user‘ and ‗computer‘ rather frequently). Similarly, in a bitmap, if one bit is white, there is a good chance the next will be as well. Compression algorithms take advantage of this redundancy. For example, Huffman encoding gives short codes to frequent words and runlength encoding represents long runs of the same value by length value pairs. Text can easily be reduced by a factor of five and bitmaps often compress to 1% of their original size. For video, in addition to compressing each frame, we can take advantage of the fact that successive frames are often similar. We can compute the difference between successive frames and then store only this – compressed, of course. More sophisticated algorithms detect when the camera pans and use this information also. These differencing methods fail when the scene changes, and so the process periodically has to restart and send a new, complete (but compressed) image. For storage purposes this is not a problem, but when used for transmission over telephone lines or networks it can mean glitches in the video as the system catches up. With these reductions it is certainly possible to store low-quality video at 64 Kbyte/s; that is, we can store five hours of highly compressed video on our 1 Gbyte hard disk. However, it still makes the humble video cassette look very good value. Probably the leading edge of video still and photographic compression is fractal compression. Fractals have been popularized by the images of the Mandelbrot set (that swirling pattern of computer-generated colors seen on many T-shirts and posters).Fractals refer to any image that contains parts which, when suitably scaled, are similar to the whole. If we look at an image, it is possible to find parts which are approximately self-similar, and these parts can be stored as a fractal with only a few numeric parameters. Fractal compression is especially good for textured features, which cause problems for other compression techniques. The decompression of the image can be performed to any degree of accuracy, from a very rough soft-focus image, to one more detailed than the original. The former is very useful as one can produce poor-quality output quickly, and better quality given more time. The latter is rather remarkable – the fractal compression actually fills in details that are not in the original.

**1.6.4 Storage format and standards**

The most common data types stored by interactive programs are text and bitmap images, with increasing use of video and audio, and this subsection looks at the ridiculous range of file storage standards. We will consider database retrieval in the next subsection. The basic standard for text storage is the ASCII (American standard code for information interchange) character codes, which assign to each standard printable character and several control characters an internationally recognized 7 bit code (decimal values 0–127), which can therefore be stored in an 8 bit byte, or be transmitted as 8 bits including parity. Many systems extend the codes to the values 128–255, including line-drawing characters, mathematical symbols and international letters such as ‗æ'. There is a 16 bit extension, the UNICODE standard, which has enough room for a much larger range of characters including the Japanese Kanji character set. As we have already discussed, modern documents consist of more than just characters. The text is in different fonts and includes formatting information such as centering, page headers and footers. On the whole, the storage of formatted text is vendor specific, since virtually every application has its own file format. This is not helped by the fact that many suppliers attempt to keep their file formats secret, or update them frequently to stop others' products being compatible. With the exception of bare ASCII, the most common shared format is rich text format (RTF), which encodes formatting information including style sheets. However, even where an application will import or export RTF, it may represent a cut-down version of the full document style. RTF regards the document as formatted text, that is it concentrates on the appearance. Documents can also be regarded as structured objects: this book has chapters containing sections, subsections . . . paragraphs, sentences, words and characters. There are ISO standards for document structure and interchange, which in theory could be used for transfer between packages and sites, but these are rarely used in practice. Just as the PostScript language is used to describe the printed page, SGML (standard generalized markup language) can be used to store structured text in a reasonably extensible way. You can define your own structures (the definition itself in SGML), and produce documents according to them. XML (extensible markup language), a lightweight version of SGML, is now used extensively for web-based applications. For bitmap storage the range of formats is seemingly unending. The stored image needs to record the size of the image, the number of bits per pixel, possibly a color map, as well as the bits of the image itself. In addition, an icon may have a ‗hot-spot' for use as a cursor. If you think of all the ways of encoding these features, or leaving them implicit, and then consider all the combinations of these different encodings, you can see why there are problems. And all this before we have even considered the effects of compression! There is, in fact, a whole software industry producing packages that convert from one format to another. Given the range of storage standards (or rather lack of standards), there is no easy advice as to which is best, but if you are writing a new word processor and are about to decide how to store the document on disk, think, just for a moment, before defining yet another format.

1. Short-term memory: RAM
2. Long-term memory: magnetic and optical disks
3. capacity limitations related to document and video storage
4. Access methods as they limit or help the user.

**1.7 Processing & Networks**

The effects when systems run too slow or too fast, the myth of the infinitely fast machine

limitations on processing speed

Networks and their impact on system performance.

**1.7.1 Effects of finite processor speed**

Speed of processing can seriously affect the user interface. These effects must be taken into account when designing an interactive system. There are two sorts of faults due to processing speed: those when it is too slow, and those when it is too fast!

Example of the former above. This was a functional fault, in that the program did the wrong thing. The system is supposed to draw lines from where the mouse button is depressed to where it is released. However, the program gets it wrong – after realizing the button is down, it does not check the position of the mouse fast enough, and so the user may have moved the mouse before the start position is registered. This is a fault at the implementation stage of the system rather than of the design. But to be fair, the programmer may not be given the right sort of information from lower levels of system software.

A second fault due to slow processing is where, in a sense, the program does the right thing, but the feedback is too slow, leading to strange effects at the interface. In order to avoid faults of the first kind, the system buffers the user input; that is, it remembers key presses and mouse buttons and movement. Unfortunately, this leads to problems of its own. One example of this sort of problem is cursor tracking, which happens in character-based text editors. The user is trying to move backwards on the same line to correct an error, and so presses the cursor-left key. The cursor moves and when it is over the correct position, the user releases the key. Unfortunately, the system is behind in responding to the user, and so has a few more cursor-left keys Moore‘s law.

Everyone knows that computers just get faster and faster. However, in 1965 Gordon Moore, co-founder of Intel, noticed a regularity. It seemed that the speed of processors, related closely to the number of transistors that could be squashed on a silicon wafer, was doubling every 18 months – exponential growth. One of the authors bought his first ‗proper' computer in 1987; it was a blindingly fast 1.47 MHz IBM compatible (Macs were too expensive). By 2002 a system costing the same in real terms would have had a 1.5 GHz processor – 1000 times faster or 210 in 15 years, that is 10 to 18 months. There is a similar pattern for computer memory, except that the doubling time for magnetic storage seems to be closer to one year. For example, when the first edition of this book was written one of the authors had a 20 Mbyte hard disk; now, 11 years later, his disk is 30 Gbytes – around210 times more storage in just 10 years. The effects of this are dramatic. If you took a young baby today and started recording a full audio video diary of every moment, day and night, of that child‘s life, by the time she was an old lady her whole life experience would fit into memory the size of a small grain of dust.

The computer to process – the cursor then overshoots. The user tries to correct this by pressing the cursor-right key, and again overshoots. There is typically no way for the user to tell whether the buffer is empty or not, except by interacting very slowly with the system and observing that the cursor has moved after every keypress. A similar problem, icon wars, occurs on window systems. The user clicks the mouse on a menu or icon, and nothing happens; for some reason the machine is busy or slow. So the user clicks again, tries something else – then, suddenly, all the buffered mouse clicks are interpreted and the screen becomes a blur of flashing windows and menus. This time, it is not so much that the response is too slow – it is fast enough when it happens – but that the response is variable. The delays due to swapping programs in and out of main memory typically cause these problems. Furthermore, a style of interaction that is optimal on one machine may not be so on a slower machine. In particular, mouse-based interfaces cannot tolerate delays between actions and feedback of more than a fraction of a second, otherwise the immediacy required for successful interaction is lost. If these responses cannot be met then a more old-fashioned, command-based interface may be required. Whereas it is immediately obvious that slow responses can cause problems for the user, it is not so obvious why one should not always aim for a system to be as fast as possible. However, there are exceptions to this – the user must be able to read and understand the output of the system. For example, one of the authors was once given a demonstration disk for a spreadsheet. Unfortunately, the machine the demo was written on was clearly slower than the author‘s machine, not much, at worst half the speed, but different enough. The demo passed in a blur over the screen with nothing remaining on the screen long enough to read. Many high-resolution monitors suffer from a similar problem when they display text. Whereas older character-based terminals scrolled new text from the bottom of the screen or redrew from the top, bitmap screens often ‗flash‘ up the new page, giving no indication of direction of movement. A final example is the rate of cursor flashing: the rate is often at a fixed.

### 1.7.2 Limitations on Interactive performance

There are several factors that can limit the speed of an interactive system:

Computation bound

Storage channel bound

Graphics bound

Network capacity

### 1.7.3 Networked Computing

Computer systems in use today are much more powerful than they were a few years ago, which means that the standard computer on the desktop is quite capable of high-performance interaction without recourse to outside help. However, it is often the case that we use computers not in their standalone mode of operation, but linked together in networks. This brings added benefits in allowing communication between different parties, provided they are connected into the same network, as well as allowing the desktop computer to access resources remote from itself. Such networks are inherently much more powerful than the individual computers that make up the network: increased computing power and memory are only part of the story, since the effects of allowing people much more extensive, faster and easier access to information are highly significant to individuals, groups and institutions.

## 1.8 INTERACTION

Interaction involves at least two participants: the user and the system. The interface must therefore effectively translate between them to allow the interaction to be successful. This translation can fail at a number of points and for a number of reasons. The use of models of interaction can help us to understand exactly what is going on in the interaction and identify the likely root of difficulties. They also provide us with a framework to compare different interaction styles and to consider interaction problems.

### 1.8.1 The terms of interaction

The purpose of an interactive system is to aid a user in accomplishing goals from some application domain. A domain defines an area of expertise and knowledge in some real-world activity. Some examples of domains are graphic design, authoring and process control in a factory.

A domain consists of concepts that highlight its important aspects. In a graphic design domain, some of the important concepts are geometric shapes, a drawing surface and a drawing utensil. Tasks are operations to manipulate the concepts of a domain. A goal is the desired output from a performed task. For example, one task within the graphic design domain is the construction of a specific geometric shape with particular attributes on the drawing surface. A related goal would be to produce a solid red triangle centered on the canvas. An intention is a specific action required to meet the goal.

### 1.8.2 The execution–evaluation cycle

The interactive cycle can be divided into two major phases: execution and evaluation. These can then be subdivided into further stages, seven in all. The stages in Norman_s model of interaction are as follows:

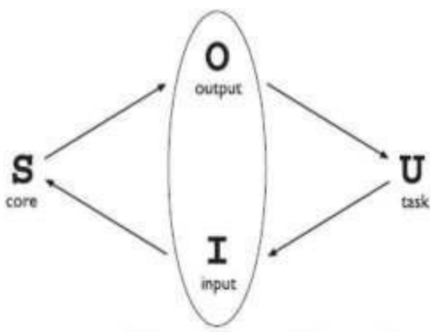
1. Establishing the goal.
2. Forming the intention.
3. Specifying the action sequence.
4. Executing the action.
5. Perceiving the system state.
6. Interpreting the system state.
7. Evaluating the system state with respect to the goals and intentions.

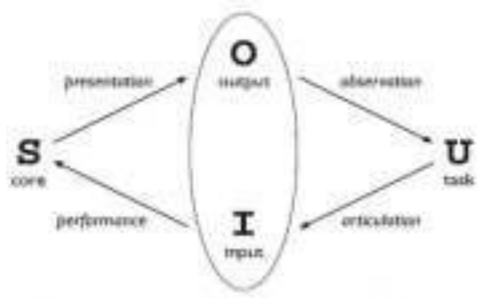
It is liable to be imprecise and therefore needs to be translated into the more specific intention, and the actual actions that will reach the goal, before it can be executed by the user. The user perceives the new state of the system, after execution of the action sequence, and interprets it in terms of his expectations. If the system state reflects the user_s goal then the computer has done what he wanted and the interaction has been successful; otherwise the user must formulate a new goal and repeat the cycle.

Norman uses this model of interaction to demonstrate why some interfaces cause problems to their users. He describes these in terms of the gulfs of execution and the gulfs of evaluation. As we noted earlier, the user and the system do not use the same terms to describe the domain and goals – remember that we called the language of the system the core language and the language of the user the task language. The gulf of execution is the difference between the user_s formulation of the actions to reach the goal and the actions allowed by the system. If the actions allowed by the system correspond to those intended by the user, the interaction will be effective. The interface should therefore aim to reduce this gulf. The gulf of evaluation is the distance between the physical presentation of the system state and the expectation of the user. If the user can readily evaluate the presentation in terms of his goal, the gulf of evaluation is small. The more effort that is required on the part of the user to interpret the presentation, the less effective the interaction.

**1.8.3 The interaction framework**

The interaction framework attempts a more realistic description of interaction by including the system explicitly, and breaks it into four main components. The nodes represent the four major components in an interactive system – the System, the User, the Input and the Output. Each component has its own language. In addition to the User_s task language and the System_s core language, which we have already introduced, there are languages for both the Input and Output components. Input and Output together form the Interface.




The general interaction framework Translations between components

The System then transforms itself as described by the operations; the execution phase of the cycle is complete and the evaluation phase now begins. The System is in a new state, which must now be communicated to the User. The current values of system attributes are rendered as concepts or features of the Output. It is then up to the User to observe the Output and assess the results of the interaction relative to the original goal, ending the evaluation phase and, hence, the interactive cycle. There are four main translations involved in the interaction: articulation, performance, presentation and observation.

**Assessing overall interaction**

The interaction framework is presented as a means to judge the overall usability of an entire interactive system. This is not surprising since it is only in attempting to perform a particular task within some domain that we are able to determine if the tools we use are adequate. For a particular editing task, one can choose the text editor best suited for interaction relative to the task. The best editor, if we are forced to choose only one, is the one that best suits the tasks most frequently performed. Therefore, it is not too disappointing that we cannot extend the interaction analysis beyond the scope of a particular task.

## 1.9 MODELS

## FRAMEWORKS AND HCI

The field of ergonomics addresses issues on the user side of the interface, covering input and output, as well as the user_s immediate context. Dialog design and interface styles can be placed particularly along the input branch of the framework, addressing both articulation and performance.

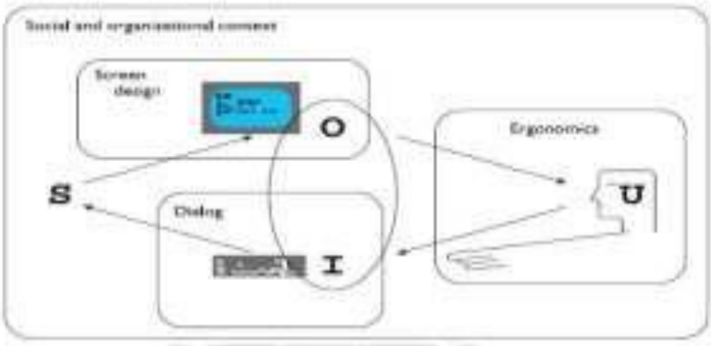



Figure: A framework for human–computer interaction.

Presentation and screen design relates to the output branch of the framework. The entire framework can be placed within a social and organizational context that also affects the interaction. Each of these areas has important implications for the design of interactive systems and the performance of the user.

## 1.10 ERGONOMICS

Ergonomics (or human factors) is traditionally the study of the physical characteristics of the interaction: how the controls are designed, the physical environment in which the interaction takes place, and the layout and physical qualities of the screen. A primary focus is on user performance and how the interface enhances or detracts from this. In seeking to evaluate these aspects of the interaction, ergonomics will certainly also touch upon human psychology and system constraints. It is a large and established field, which is closely related to but distinct from HCI, and full coverage would demand a book in its own right. Here we consider a few of the issues addressed by ergonomics as an introduction to the field. We will briefly look at the arrangement of controls and displays, the physical environment, health issues and the use of colour. These are by no means exhaustive and are intended only to give an indication of the types of issues and problems addressed by ergonomics.

### 1.10.1 Arrangement of controls and displays

The exact organization that this will suggest will depend on the domain and the application, but possible organizations include the following:

- Functional controls and displays are organized so that those that are functionally related are placed together;
- Sequential controls and displays are organized to reflect the order of their use in a typical interaction (this may be especially appropriate in domains where a particular task sequence is enforced, such as aviation);
- Frequency controls and displays are organized according to how frequently they are used, with the most commonly used controls being the most easily accessible.

**The physical environment of the interaction**

Physical issues in the layout and arrangement of the machine interface, ergonomics is concerned with the design of the work environment itself. This will depend largely on the domain and will be more critical in specific control and operational settings than in general computer use. The physical environment in which the system is used may influence how well it is accepted and even the health and safety of its users. It should therefore be considered in all design. The first consideration here is the size of the users. Obviously this is going to vary considerably. All users should be comfortably able to see critical displays. For long periods of use, the user should be seated for comfort and stability. Seating should provide back support. If required to stand, the user should have room to move around in order to reach all the controls.

**1.10.2 Health issues**

There are a number of factors that may affect the use of more general computers. Again these are factors in the physical environment that directly affect the quality of the interaction and the user_s

**Performance:**

Users should be able to reach all controls comfortably and see all displays. Users should not be expected to stand for long periods and, if sitting, should be provided with back support. If a particular position for a part of the body is to be adopted for long periods (for example, in typing) support should be provided to allow rest.

**Temperature**

Extremes of hot or cold will affect performance and, in excessive cases, health. Experimental studies show that performance deteriorates at high or low temperatures, with users being unable to concentrate efficiently.

**Lighting** The lighting level will again depend on the work environment. Adequate lighting should be provided to allow users to see the computer screen without discomfort or eyestrain. The light source should also be positioned to avoid glare affecting the display.

**Noise** Excessive noise can be harmful to health, causing the user pain, and in acute cases, loss of hearing. Noise levels should be maintained at a comfortable level in the work environment.

This does not necessarily mean no noise at all. Noise can be a stimulus to users and can provide needed confirmation of system activity.

**Time** The time users spend using the system should also be controlled. it has been suggested that excessive use of CRT displays can be harmful to users, particularly pregnant women.

**The use of color**

Colors used in the display should be as distinct as possible and the distinction should not be affected by changes in contrast. Blue should not be used to display critical information. If color is used as an indicator it should not be the only cue: additional coding information should be included.

The colors used should also correspond to common conventions and user expectations. Red, green and yellow are colors frequently associated with stop, go and standby respectively. Therefore, red may be used to indicate emergency and alarms; green, normal activity; and yellow, standby and auxiliary function. These conventions should not be violated without very good cause.

**Ergonomics and HCI**

Ergonomics is a huge area, which is distinct from HCI but sits alongside it. Its contribution to HCI is in determining constraints on the way we design systems and suggesting detailed and specific guidelines and standards. Ergonomic factors are in general well established and understood and are therefore used as the basis for standardizing hardware designs.

**1.11 INTERACTION STYLES**

Interaction can be seen as a dialog between the computer and the user. The choice of interface style can have a profound effect on the nature of this dialog. There are a number of common interface styles including

- command line interface
- menus
- natural language
- question/answer and query dialog
- form-fills and spreadsheets
- WIMP
- point and click
- Three-dimensional interfaces.

**1.11.1 Command line interface**

The command line interface was the first interactive dialog style to be commonly used and, in spite of the availability of menu-driven interfaces, it is still widely used. It provides a means of expressing instructions to the computer directly, using function keys, single characters, abbreviations or whole-word commands. In some systems the command line is the only way of communicating with the system, especially for remote access using telnet. Menu-based interfaces, providing accelerated access to the system_s functionality for experienced users. Command line interfaces are powerful in that they offer direct access to system functionality and can be combined to apply a number of tools to the same data. They are also flexible: the command often has a number of options or parameters that will vary its behavior in some way, and it can be applied to many objects at once, making it useful for repetitive tasks. Flexibility and power brings with it difficulty in use and learning.

Commands must be remembered, as no cue is provided in the command line to indicate which command is needed. They are therefore better for expert users than for novices. This problem can be alleviated a little by using consistent and meaningful commands and abbreviations. The commands used should be terms within the vocabulary of the user rather than the technician. Unfortunately, commands are often obscure and vary across systems, causing confusion to the user and increasing the overhead of learning.

**1.11.2 Menus**

In a menu-driven interface, the set of options available to the user is displayed on the screen, and selected using the mouse, or numeric or alphabetic keys. Since the options are visible they are less demanding of the user, relying on recognition rather than recall. Menu options still need to be meaningful and logically grouped to aid recognition. Often menus are hierarchically ordered and the option required is not available at the top layer of the hierarchy. The grouping and naming of menu options then provides the only cue for the user to find the required option. Such systems either can be purely text based, with the menu options being presented as numbered choices, or may have a graphical component in which the menu appears within a rectangular box and choices are made, perhaps by typing the initial letter of the desired selection, or by entering the associated number, or by moving around the menu with the arrow keys.

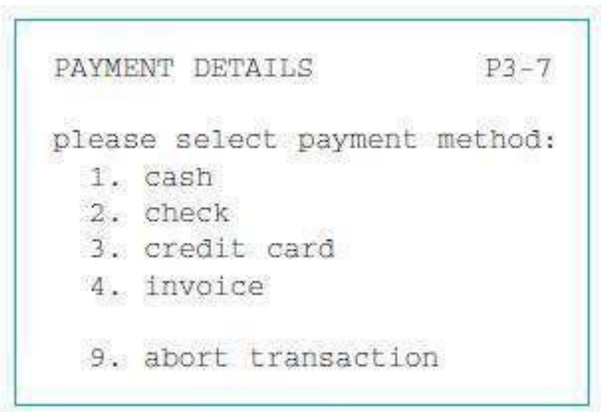


Figure Menu-driven interface

### 1.11.3 Natural language

Users, unable to remember a command or lost in a hierarchy of menus, may long for the computer that is able to understand instructions expressed in everyday words! Natural language understanding, both of speech and written input, is the subject of much interest and research. the ambiguity of natural language makes it very difficult for a machine to understand. Language is ambiguous at a number of levels. First, the syntax, or structure, of a phrase may not be clear. If we are given the sentence―The boy hit the dog with the stick‖

### 1.11.4 Question/answer and query dialog

Question and answer dialog is a simple mechanism for providing input to an application in a specific domain. The user is asked a series of questions (mainly with yes/no responses, multiple choice, or codes) and so is led through the interaction step by step These interfaces are easy to learn and use, but are limited in functionality and power. As such, they are appropriate for restricted domains (particularly information systems) and for novice or casual users.

**Form-fills and spreadsheets**

Form-filling interfaces are used primarily for data entry but can also be useful in data retrieval applications. The user is presented with a display resembling a paper form, with slots to fill in .Often the form display is based upon an actual form with which the user is familiar, which makes the interface easier to use. The user works through the form, filling in appropriate values. The data are then entered into the application in the correct place. Most form-filling interfaces allow easy movement around the form and allow some fields to be left blank. They also require correction facilities, as users may change their minds or make a mistake about the value that belongs in each field. The dialog style is useful primarily for data entry applications and, as it is easy to learn and use, for novice users.

Spreadsheets are a sophisticated variation of form filling. The spreadsheet comprises a grid of cells, each of which can contain a value or a formula. The formula can involve the values of other cells (for example, the total of all cellsin this column). The user can enter and alter values and formulae in any order and the system will maintain consistency amongst the values displayed, ensuring that all formulae are obeyed. The user can therefore manipulate values to see the effects of changing different parameters. Spreadsheets are an attractive medium for interaction: the user is free to manipulate values at will and the distinction between input and output is blurred, making the interface more flexible and natural.

**The WIMP interface**

WIMP stands for windows, icons, menus and pointers (sometimes windows, icons, mice and pull-down menus), and is the default interface style for the majority of interactive computer systems in use today, especially in the PC and desktop workstation arena. Examples of WIMP interfaces include Microsoft Windows for IBM PC compatibles, MacOS for Apple Macintosh compatibles and various X Windows-based systems for UNIX.

**Point-and-click interfaces**

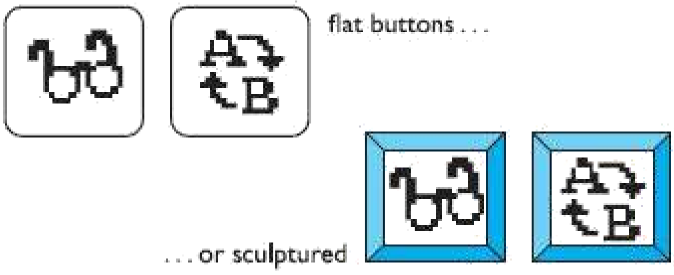
This point-and-click interface style is obviously closely related to the WIMP style. It clearly overlaps in the use of buttons, but may also include other WIMP elements. The philosophy is simpler and more closely tied to ideas of hypertext. In addition, the point-and-click style is not tied to mouse-based interfaces, and is also extensively used in touch screen information systems. In this case, it is often combined with a menu-driven interface. The point-and-click style has been popularized by World Wide Web pages, which incorporate all the above types of point-and-click navigation: highlighted words, maps and iconic buttons.

### 1.11.5 Three-dimensional interfaces

There is an increasing use of three-dimensional effects in user interfaces. The most obvious example is virtual reality, but VR is only part of a range of 3D techniques available to the interface designer. The simplest technique is where ordinary WIMP elements, buttons, scroll bars, etc., are given a 3D appearance using shading, giving the appearance of being sculpted out of stone. By unstated convention, such interfaces have a light source at their top right. Where used judiciously, the raised areas are easily identifiable and can be used to highlight active areas. Some interfaces make indiscriminate use of sculptural effects, on every text area, border and menu, so all sense of differentiation is lost.

## 1.12 ELEMENTS INTERACTIVITY

Dialog design is focused almost entirely on the choice and specification of appropriate sequences of actions and corresponding changes in the interface state. It is typically not used at a fine level of detail and deliberately ignores the _semantic_level of an interface: for example, the validation of numeric information in a forms-based system.




It is worth remembering that interactivity is the defining feature of an interactive system. This can be seen in many areas of HCI. For example, the recognition rate for speech recognition is too low to allow transcription from tape, but in an airline reservation system, so long as the system can reliably recognize yes and no it can reflect back its understanding of what you said and seek confirmation. Speech-based input is difficult, speech-based interaction easier. Also, in the area of information visualization the most exciting developments are all where users can interact with visualization in real time, changing parameters and seeing the effect.

Interactivity is also crucial in determining the feel_of a WIMP environment. All WIMP systems appear to have virtually the same elements: windows, icons, menus, pointers, dialog boxes, buttons, etc. In fact, menus are a major difference between the MacOS and Microsoft Windows environments: in MacOS you have to keep the mouse depressed throughout menu selection; in Windows you can click on the menu bar and a pull-down menu appears and remains there until an item is selected or it is cancelled. Similarly the detailed behavior of buttons is quite complex.

## 1.13 PARADIGMS

### 1.13.1 Paradigms of Interaction

**The paradigms of interaction are**

**Time sharing**

Major contributions to come out of this new emphasis in research were the concept of time sharing, in which a single computer could support multiple users. The human (or more accurately, the programmer) was restricted to batch sessions, in which complete jobs were submitted on punched cards or paper tape to an operator who would then run them individually on the computer. Time-sharing systems of the 1960s made programming a truly interactive venture and brought about a subculture of programmers known as _hackers_ – single-minded masters of detail who took pleasure in understanding complexity. Though the purpose of the first interactive time-sharing systems was simply to augment the programming capabilities of the early hackers, it marked a significant stage in computer applications for human use. Rather than rely on a model of interaction as a pre-planned activity that resulted in a complete set of instructions being laid out for the computer to follow, truly interactive exchange between programmer and computer was possible. The computer could now project itself as a dedicated partner with each individual user and the increased throughput of information between user and computer allowed the human to become a more reactive and spontaneous collaborator.

**Video display units**

In mid-1950s researchers were experimenting with the possibility of presenting and manipulating information from a computer in the form of images on a video display unit (VDU). These display screens could provide a more suitable medium than a paper printout for presenting vast quantities of strategic information for rapid assimilation. The earliest applications of display screen images were developed in military applications, most notably the Semi-Automatic Ground Environment (SAGE) project of the US Air Force.

**Programming toolkits**

Douglas Engelbart_s ambition since the early 1950s was to use computer technology as a means of complementing human problem-solving activity.

**Personal computing**

Programming toolkits provide a means for those with substantial computing skills to increase their productivity greatly. One of the first demonstrations that the powerful tools of the hacker could be made accessible to the computer novice was a graphics programming language for children called LOGO. A child could quite easily pretend they were inside_the turtle and direct it to trace out simple geometric shapes, such as a square or a circle. By typing in English phrases, such as go forward or Turn left, the child/programmer could teach the turtle to draw more and more complicated figures. By adapting the graphical programming language to a model which children could understand and use, Paper demonstrated a valuable maxim for interactive system development – no matter how powerful a system may be, it will always be more powerful if it is easier to use.

**Window systems and the WIMP interface**

Humans are able to think about more than one thing at a time, and in accomplishing some piece of work, they frequently interrupt their current train of thought to pursue some other related piece of work. A personal computer system which forces the user to progress in order through all of the tasks needed to achieve some objective, from beginning to end without any diversions, does not correspond to that standard working pattern.

One presentation mechanism for achieving this dialog partitioning is to separate physically the presentation of the different logical threads of user–computer conversation on the display device. The window is the common mechanism associated with these physically and logically separate display spaces.

**The metaphor**

Papert used the metaphor of a turtle dragging its tail in the dirt. Children could quickly identify with the real-world phenomenon and that instant familiarity gave them an understanding of how they could create pictures. The danger of a metaphor is usually realized after the initial honeymoon period. When word processors were first introduced, they relied heavily on the typewriter metaphor. The keyboard of a computer closely resembles that of a standard typewriter, so it seems like a good metaphor from which to start.

**Hypertext**

Hypertext is text which is not constrained to be linear. Hypertext is text which contains links to other texts. The term was coined by Ted Nelson around 1965. HyperMedia is a term used for hypertext which is not constrained to be text: it can include graphics, video and sound, for example. Apparently Ted Nelson was the first to use this term too. Hypertext and HyperMedia are concepts, not products.

**Multi-modality**

Genuine multi-modal systems rely to a greater extent on simultaneous use of multiple communication channels for both input and output. Humans quite naturally process information by simultaneous use of different channels. We point to someone and refer to them as you‗, and it is only by interpreting the simultaneous use of voice and touch that our directions are easily articulated and understood. Designers have wanted to mimic this flexibility in both articulation and observation by extending the input and output expressions an interactive system will support. So, for example, we can modify a gesture made with a pointing device by speaking, indicating what operation is to be performed on the selected object.

**Computer-supported cooperative work**

Personal computing provides individuals with enough computing power so that they were liberated from dumb terminals which operated on a time-sharing system. It is interesting to note that as computer networks became widespread, individuals retained their powerful workstations but now wanted to reconnect themselves to the rest of the workstations in their immediate working environment, and even throughout the world! One result of this reconnection was the emergence of collaboration between individuals via the computer – called computer-supported cooperative work, or CSCW.

**The World Wide Web**

WWW or "Web" is a global information medium which users can read and write via computers connected to the Internet. The term is often mistakenly used as a synonym for the

Internet itself, but the Web is a service that operates over the Internet, just as e-mail also does. The history of the Internet dates back significantly further than that of the World Wide Web.

The internet is simply a collection of computers, each linked by any sort of data connection, whether it be slow telephone line and modem or high-bandwidth optical connection. The computers of the internet all communicate using common data transmission protocols (TCP/IP) and addressing systems (IP addresses and domain names). This makes it possible for anyone to read anything from anywhere, in theory, if it conforms to the protocol. The web builds on this with its own layer of network protocol (http), a standard markup notation (such as HTML) for laying out pages of information and a global naming scheme (uniform resource locators or URLs). Web pages can contain text, color images, movies, sound and, most important, hypertext links to other web pages. Hypermedia documents can therefore be _published_by anyone who has access to a computer connected to the internet.

**Ubiquitous computing**

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. Ubiquitous computing is also known as pervasive computing, every ware and ambient intelligence.