on, Penguin,

l analysis for

in Computer

ccessfully.

V: Tools for

through: a

' Inspection

evaluation.

, Usability

teraction,

hysiology

r-Verlag,

y, equip-

10

Universal design

OVERVIEW

- Universal design is about designing systems so that they can be used by anyone in any circumstance.
- Multi-modal systems are those that use more than one human 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.

10.1 INTRODUCTION

We have already discussed the importance of designing for the user, considering human abilities and requirements. But is it possible to generalize about people and, if not, how do we address the issue of human diversity in our designs?

The discussion that we had on human psychology in Chapter 1 talked about general human abilities and, in reality, people are much more varied than the discussion suggests. People have different abilities and weaknesses; they come from different backgrounds and cultures; they have different interests, viewpoints and experiences; they are different ages and sizes. All of these things have an impact on the way in which an individual will use a particular computing application and, indeed, on whether or not they can use it at all. Given such diversity, we cannot assume a 'typical' user or design only for people like ourselves.

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. An example of the former might be an interface that has both visual and audio access to commands; an example of the latter, a website that provides text alternatives for graphics, so that it can be read using a screen reader.

In this chapter, we will look at universal design in more detail. We will begin by examining seven principles of universal design. We will then look at multi-modal technology and how it can help to provide redundancy in interaction. Having identified some of the available technologies at our disposal, we will look in more detail at the particular areas of human diversity that we need to address.

10.2 UNIVERSAL DESIGN PRINCIPLES

We have defined universal design as 'the process of designing products so that they can be used by as many people as possible in as many situations as possible'. But what does that mean in practice? Is it possible to design anything so that anyone can use it – and if we could, how practical would it be? Wouldn't the cost be prohibitive? In reality, we may not be able to design everything to be accessible to everyone, and we certainly cannot ensure that everyone has the same experience of using a product, but we can work toward the aim of universal design and try to provide an *equivalent* experience.

Although it may seem like a huge task, universal design does not have to be complex or costly. In fact, if you are observant, you will see many examples of design that attempt to take account of user diversity. Next time you cross the road, look at the pavement. The curb may be lowered, to enable people who use wheelchairs to cross more easily. The paving near the curb may be of a different texture – with raised bumps or ridges – to enable people who cannot see to find the crossing point. Notice how many modern buildings have automatic doors that open on approach. Or lifts that offer both visual and auditory notification of the floor reached. And, whilst these designs make the crossing, the building and the lift more accessible to people who have disabilities, notice too how they also help other users. The parent with a child in a buggy, or the traveller with wheeled luggage, can cross the road more easily; the shopper with heavy bags, or the small child, can enter the building; and people are less likely to miss their floor because they weren't paying attention. Universal design is primarily about trying to ensure that you do not exclude anyone through the design choices you make but, by giving thought to these issues, you will invariably make your design better for everyone.

In the late 1990s a group at North Carolina State University in the USA proposed seven general principles of universal design [333]. 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.

Principle one is *equitable use*: the design is useful to people with a range of abilities and appealing to all. No user is excluded or stigmatized. Wherever possible, access should be the same for all; where identical use is not possible, equivalent use should be supported. Where appropriate, security, privacy and safety provision should be available to all.

Principle two is *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.

Principle three is that the system be *simple and intuitive to use*, regardless of the knowledge, experience, language or level of concentration of the user. The design needs to support the user's expectations and accommodate different language and literacy skills. It should not be unnecessarily complex and should be organized to facilitate access to the most important areas. It should provide prompting and feedback as far as possible.

Principle four is *perceptible information*: the design should provide effective communication of information regardless of the environmental conditions or the user's abilities. Redundancy of presentation is important: information should be represented in different forms or modes (e.g. graphic, verbal, text, touch). Essential information should be emphasized and differentiated clearly from the peripheral content. Presentation should support the range of devices and techniques used to access information by people with different sensory abilities.

Principle five is *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. Systems should fail safe from the user's perspective and users should be supported in tasks that require concentration.

Principle six is *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. Repetitive or sustained actions should be avoided.

Principle seven requires *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. Important elements should be on the line of sight for both seated and standing users. All physical components should be comfortably reachable by seated or standing users. Systems should allow for variation in hand size and provide enough room for assistive devices to be used.

These seven principles give us a good starting point in considering universal design. They are not all equally applicable to all situations, of course. For example, principles six and seven would be vital in designing an information booth but less important in designing word-processing software. But they provide a useful checklist of considerations for designers, together with guidelines on how each principle can be achieved. It is interesting to note that these principles are closely related to the ones we met in Chapter 7, in the context of general user-centered design rules, indicating again that universal design is fundamentally good design for all.

10.3 MULTI-MODAL INTERACTION

As we have seen in the previous section, providing access to information through more than one mode of interaction is an important principle of universal design. Such design relies on *multi-modal interaction*.

As we saw in Chapter 1, there are five senses: sight, sound, touch, taste and smell.

Sight is the predominant sense for the majority of people, and most interactive systems consequently use the visual channel as their primary means of presentation, through graphics, text, video and animation.

However, sound is also an important channel, keeping us aware of our surroundings, monitoring people and events around us, reacting to sudden noises, providing clues and cues that switch our attention from one thing to another. It can also have an emotional effect on us, particularly in the case of music. Music is almost completely an auditory experience, yet is able to alter moods, conjure up visual images, evoke atmospheres or scenes in the mind of the listener.

Touch, too, provides important information: tactile feedback forms an intrinsic part of the operation of many common tools – cars, musical instruments, pens, anything that requires holding or moving. It can form a sensuous bond between individuals, communicating a wealth of non-verbal information.

Taste and smell are often less appreciated (until they are absent) but they also provide useful information in daily life: checking if food is bad, detecting early signs of fire, noticing that manure has been spread in a field, pleasure.

Examples of the use of sensory information are easy to come by (we looked at some in Chapter 1), but the important point is that our everyday interaction with each other and the world around us is multi-sensory, each sense providing different information that informs the whole. Since our interaction with the world is improved by multi-sensory input, it makes sense that interactive systems that utilize more than one sensory channel will also provide a richer interactive experience. 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. As systems become more complex, the visual channel may be overloaded if too much information is presented all at once. This may lead to frustration or errors in use. 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. However, it should always be remembered that multi-modal interaction is not just about enhancing the richness of the interaction, but also about redundancy. Redundant systems provide the *same* information through a range of channels, so, for example, information presented graphically is also captioned in readable text or speech, or a verbal narrative is provided with text captions. The aim is to provide at least an equivalent experience to all, regardless of their primary channel of interaction.

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.

The use of sound is an obvious area for further exploitation. There is little doubt that we use hearing a great deal in daily life, and so developing its application to the interface may be beneficial. 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. However, taste and smell pose more serious problems for us. They are the least used of our senses, and are used more for receiving information than for communicating it. There are currently very few ways of implementing devices that can generate tastes and smells, and so these two areas are not supported. Whether this is a serious omission remains to be seen, but the tertiary nature of those senses tends to suggest that their incorporation, if it were possible, would focus on specialist applications, for example, in enhancing virtual reality systems.

Even if we do not use other senses in our systems, it is certainly worth thinking about the nature of these senses and what we gain from them as this will improve our understanding of the strengths and weaknesses of visual communication [96].

The next sections of this chapter will look at some of the alternative modes of human–computer communication, concentrating particularly on sound, touch, handwriting and gesture. We will consider how each mode can be used to create richer interaction and provide redundancy.

10.3.1 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 keyclicks, reduces errors [237]. 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. The dual presentation of information through sound and vision supports universal design, by enabling access for users with visual and hearing impairments respectively. It also enables information to be accessed in poorly lit or noisy environments. Sound can convey transient information and does not take up screen space, making it potentially useful for mobile applications.

However, in spite of this, the auditory channel is comparatively little used in standard interfaces, and where it is used it is often peripheral to the interaction. Information provision is predominantly visual. There is a danger that this will overload the visual channel, demanding that the user attend to too many things at once and select appropriate information from a mass of detail in the display. Reliance on visual information forces attention to remain focussed on the screen, and the persistence of visual information means that even detail that is quickly out of date may remain on display after it is required, cluttering the screen further. It also presents significant problems for people with visual impairment, whose access to applications can be severely restricted by solely visual output. More widespread effective use of sound in the interface would alleviate these problems. There are two types of sound that we could use: speech and non-speech.

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 process seems so effortless that we often do not appreciate its complex structures, and it is not until we attempt to learn a new language later in life, or to make explicit the rules of the one we speak, that the difficulties inherent in language understanding become apparent. 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. We will use English to illustrate but most other languages have similar issues.

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. Language is more than simple sounds, however. Emphasis, stress, pauses and pitch can all be used to alter the meaning and nature of an utterance, a common example being the rise in pitch at the end of a sentence to indicate a question in English. This alteration in tone and quality of the phonemes is termed *prosody* and is used, in addition to the actual words, to convey a great deal of meaning and emotion within a sentence. Prosodic information gives language its richness and texture, but is very difficult to quantify. Owing to the manner in which sound is produced in the vocal tract, mouth and nose of the speaker, the limitation in response speed means that phonemes sound differently when preceded by different phonemes. This is termed *co-articulation*, and the resulting differences in sound can be used to construct a set of *allophones*, which represent all the different sounds within the language. Ignoring prosodic information, the concatenation of allophones together should produce intelligible, articulate speech. However, depending on the analysis of language used, and the regional accent studied, there are between 120 and 130 allophones. These, in turn, can be formed into *morphemes*, which represent the smallest unit of language that has meaning. They are the basic building blocks of language rather than of speech. Morphemes can be either parts of words or whole words, and they are built into sentences using the rules of grammar of the language.

Even being able to decompose sentences into their basic parts does not mean that we can then understand them: the syntax (structure) only serves as a standard foundation upon which the semantics (meaning) is based. We are rarely aware of the complex structure of speech, and concentrate on extracting the meaning from the sentences we hear, rather than decomposing the sounds into their constituent parts.

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.

The complexity of language is one barrier to success, but there are other, more practical, problems also associated with the automatic recognition of the spoken word. Background noise can interfere with the input, masking or distorting the information, while speakers can introduce redundant or meaningless noises into the information stream by repeating themselves, pausing or using 'continuation' noises such as 'ummm' and 'errr' to fill in gaps in their usual speech. Variations between individuals also cause problems; people have unique voices, and systems that are successful are tuned to be sensitive to minute variations in tone and frequency of the speaker's voice – new speakers present different inflections to the system, which then fails to perform as well. A more serious problem is caused by regional accents, which vary considerably. This strong variation upsets the trained response of the recognition system. More serious still is the problem posed by different languages: everything from phonemes up can be different.

The phonetic typewriter

One early successful speech-based system is the *phonetic typewriter*. This uses a neural network that clusters similar sounds together (see Figure 10.1).

Designed to produce typed output from speech input in Finnish, it is trained on one particular speaker, and then generalizes to others. However, its performance with speakers other than the one on which it was trained is noticeably poorer, and it relies on a large dictionary of minute variations to supplement its general transcription mechanism. Without the dictionary, it achieves a significantly lower recognition rate.

One reason that the phonetic typewriter was able to achieve acceptable levels of recognition and transcription is that Finnish is a phonetic language, that is one which is spelt as it sounds. There are other phonetic languages, for example Welsh, but most languages do not have such a straightforward mapping between sound and text. Think of English words such as 'wait' and 'weight' or 'one' and 'won', for example.

Puzzle: How do you pronounce 'ghuti'? (Answer on the web pages!)

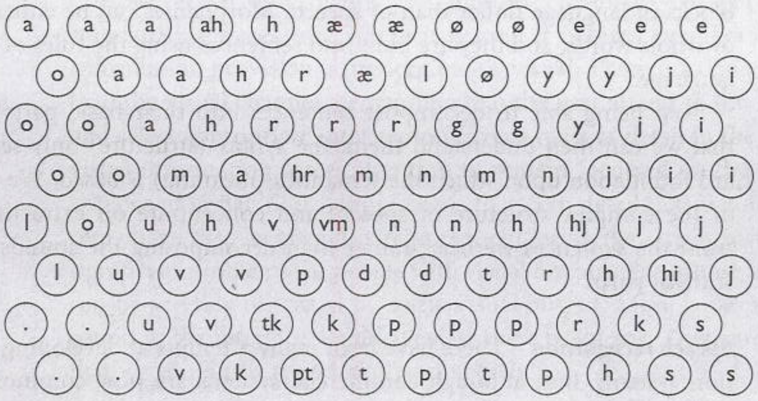


Figure 10.1 The phonetic typewriter

Speech recognition offers another mode of communication that may be used to supplement existing channels or be the primary one. When a user's hands are already occupied, such as in a factory, speech may prove to be the ideal input medium. Speech input does not require the use of a cumbersome keyboard and so in lightweight mobile situations there is a potential role for such systems. It also provides an alternative means of input for users with visual, physical and cognitive impairment as we will see later. Single-user, limited vocabulary systems can work satisfactorily, but the current success rate of recognition for general users and unconstrained language is still low.

Despite its limitations there are commercial systems employing speech recognition. Speech-based word processors are easily available and several computers use speech input as a marketing feature. Telephone-based systems also use speech, but they face a more difficult task as they must be speaker independent. At the simplest end, some systems ask you to speak an extension number, but, as tone dialing becomes universal, the advantage of this over typing the number is dubious! Other systems make more active use of voice, including information systems for airline bookings. These more sophisticated systems work because they are *interactive*: the system reflects back to the user its interpretation of the speech input, allowing the user to enter into a dialog to correct any errors. This is precisely what happens in normal conversation – we don't get it right all the time.

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. However, there are as many problems in speech synthesis as there are in recognition. The most difficult problem is that we are highly sensitive to variations and intonation in speech, and are therefore intolerant of imperfections in synthesized speech. We are so used to hearing natural speech that we find it difficult to adjust to the monotonic, non-prosodic tones that synthesized speech can produce. In fact, most speech synthesizers can deliver a degree of prosody, but in order to decide what intonation to give to a word, the system must have an understanding of the domain. So an effective automatic reader would also need to be able to understand natural language, which is difficult. However, for 'canned' messages and responses, the prosody can be hand coded yielding much more acceptable speech.

Synthesized speech also brings other problems. Being transient, spoken output cannot be reviewed or browsed easily. It is intrusive, requiring either an increase in noise in the office environment or the use of headphones, either of which may be too large a price to pay for the benefits the system may offer.

However, there are some application areas in which speech synthesis has been successful. For users who are blind or partially sighted, synthesized speech offers an output medium which they can access. Screen readers are software packages that read the contents of a computer screen, using synthesized speech. Modern screen readers read more than simply the text on the screen. They read exactly what they find including icons, menus, punctuation and controls. They also read events, such as dialog boxes opening, so that they can be used with graphical interfaces.

Speech synthesis is also useful as a communication tool to assist people with physical disabilities that affect their speech. Here speech synthesis needs to produce output that is as natural as possible with as little input effort from the user as possible, perhaps using a simple switch. Human conversation is rapid and complex, making this a significant challenge. Most communication tools of this type use predefined messages, enabling the user to select a message appropriate to the context quickly and easily.

Design Focus

Designing websites for screen readers

While screen readers provide users with visual impairments access to standard interfaces and software, web access can be more problematic. Screen readers can only read textual elements of web pages, so graphics and scripts cannot be interpreted. It is therefore important that web designers take account of these limitations and design sites to be accessible to people using screen readers. For example, HTML 'alt' tags should always be used for necessary graphics, and text alternative menus and navigation controls provided.

In addition, most read text across the page, so text arranged in columns can become garbled. For example, consider text arranged in two columns, such as the opening lines of these nursery rhymes:

Jack and Jill went up the hill	Mary had a little lamb
To fetch a pail of water	Its fleece was white as snow

With some screen readers, this text would be read as 'Jack and Jill went up the hill Mary had a little lamb To fetch a pail of water Its fleece was white as snow' – clearly nonsense. Where possible, text should make sense when read across a page.

Users of screen readers may also find it difficult to follow links embedded in text, particularly where there are several in a block of text. It can therefore be helpful to provide links to the main sections in a clear location where they will be read horizontally, such as at the top of the page.

If you want to experience something of what it is like to access the web using a screen reader, try the simulation produced by the Web Accessibility in Mind project, available at www.webaim.org/simulations/screenreader (accessed March 2003). Their site also contains a wealth of information about web accessibility.

Another useful resource to help you design accessible websites is the Web Accessibility Initiative's (WAI) checklist for accessibility. This is available on their website www.w3.org (accessed March 2003).

A tool that you can use to test your websites for accessibility is Bobby from Watchfire (bobby.cast.org, accessed March 2003). Bobby is a web-based accessibility evaluation tool that highlights problem areas of a given website and encourages accessible solutions. It provides suggestions for improvement using the guidelines provided by the WAI and the US Government's Section 508 legislation. Using Bobby, web designers can test their sites for accessibility and get advice for resolving any problems before the site goes live.

Used as a supplement to other output channels, speech can also enhance applications where the user's visual attention is focussed elsewhere, such as warnings in aircraft cockpits and, more recently, in cars. We will return to some of these applications later in the chapter.

Uninterpreted speech Speech does not have to be recognized by a computer to be useful in the interface. Fixed pre-recorded messages can be used to supplement or replace visual information. Recordings have natural human prosody and pronunciation, although quality is sometimes low. Segments of speech can be used together to construct messages, for example the announcements in many airports and railway stations.

Recordings of users' speech can also be very useful, especially in collaborative applications, for example many readers will have used voicemail systems. Also, recordings can be attached to other artifacts as *audio annotations* in order to communicate with others or to remind oneself at a later time. For example, audio annotations can be attached to Microsoft Word documents.

When recordings are replayed, they can be digitally speeded up. If you simply play an audio recording faster, the pitch rises – and human speech ends up sounding rather like Mickey Mouse. However, *digital signal-processing* techniques can accelerate a recording while keeping the same pitch. Speech can be played back at up to twice the normal rate without any loss of comprehensibility. This can be used in a telephone help desk where a pre-recorded message asks the enquirer to state his problem. The problem can then be replayed at an accelerated rate to the operator, reducing the operator time per enquiry. The utility of such methods needs careful analysis, however. The operator may often begin to act on a message while it is still playing, hence reducing any gain from faster playback. Furthermore, reduced interactivity may lead to more misunderstandings, and the enquirer's waiting time may be increased.

DESIGN FOCUS

Choosing the right kind of speech

If you include speech input in an interface you must decide what level of speech interaction you wish to support:

recording simply recording and replaying messages or annotations;

transcription turning speech into text as in a word processor;

control telling the computer what to do: for example, 'print this file'.

Each level has its own problems and advantages; for example, control only requires a limited vocabulary, but is more dangerous: 'I said print not delete . . . !' However, the biggest problem arises if you try to mix these levels. In text we use quotes to make such distinctions, but they are hard in speech: 'insert the word "delete" before the word "before"'.

In fact, for general interface use, speech is best mixed with other modes of communication as happens in everyday life. For example, in a word processor you may use a tablet and pen to ring a word and then say 'move this word to here' as you tap the pen at the target location. This is exactly what you would do when talking through corrections to a document with someone.

Non-speech sound

We have considered the use of speech in the interface, but non-speech sounds can offer a number of advantages. As speech is serial, we have to listen to most of a sentence before we understand what is being said. Non-speech sounds can often be assimilated much more quickly. Speech is language dependent – a speech-based system requires translation for it to be used for another language group. The meaning of non-speech sounds can be learned regardless of language. Speech requires the user's attention. Non-speech sound can make use of the phenomenon of auditory adaptation: background sounds are ignored unless they change or cease. However, a disadvantage is that non-speech sounds have to be learned, whereas the meaning of a spoken message is obvious (at least to a user who is familiar with the language used).

Non-speech sound can be used in a number of ways in interactive systems. It is often used to provide transitory information, such as indications of network or system changes, or of errors. It can also be used to provide status information on background processes, since we are able to ignore continuous sounds but still respond to changes in those sounds. Users of early home computers with their noisy power supplies, and computer operators listening to the chatter of the printer and the spinning of disks and tape drives, both report that they are able to tell what stage a process is at by the characteristic sounds that are made.

Non-speech sound can also be used to provide a second representation of actions and objects in the interface to support the visual mode and provide confirmation for the user. It can be used for navigation round a system, either giving redundant supporting information to the sighted user or providing the primary source of information for the visually impaired. Experiments on auditory navigation [290] have demonstrated that auditory clues are adequate for a user to locate up to eight targets on a screen with reasonable speed and accuracy. This suggests that there is little reason for ignoring the role of sound in interfaces on the grounds that it may be too vague or inaccurate.

But what kind of non-speech sounds should we use in the interface? There are two alternatives: using sounds that occur naturally in the world and using more abstract generated sounds. We will consider an example of each type.

Auditory icons *Auditory icons* [141] use natural sounds to represent different types of objects and actions in the interface. The SonicFinder [142] for the Macintosh was developed from these ideas, to enhance the interface through redundancy. Natural sounds are used because people recognize the source of a sound and its behavior rather than timbre and pitch [364]. For example, a noise will be identified as glass breaking or a hollow pipe being tapped. Such recognition is quite sophisticated: we can identify not only the source of a sound (e.g. tapping a pipe) but characteristics of the sound source (e.g. whether the pipe is hollow or solid).

In the SonicFinder, auditory icons are used to represent desktop objects and actions. So, for example, a folder is represented by a papery noise, and throwing something in the wastebasket by the sound of smashing. This helps the user to learn the sounds since they suggest familiar actions from everyday life. However, this advantage also creates a problem for auditory icons. Some objects and actions do not have obvious, naturally occurring sounds that identify them. In these cases a sound effect can be created to suggest the action or object but this moves away from the ideal of using familiar everyday sounds that require little learning. For example, copying has no immediate analog sound and in the SonicFinder it is indicated by the sound of pouring a liquid into a receptacle, with the pitch rising to indicate the progress of the copying.

SharedARK and ARKola

Natural sounds have been used to model environments such as a physics laboratory [145], called *SharedARK* (Shared Alternate Reality Kit) and a virtual manufacturing plant, *ARKola* [147]. In SharedARK, multiple users could perform physics experiments in a virtual laboratory. Sound was used in three different ways: as confirmation of actions, for status information and as aids to navigation. Confirmatory sounds use similar principles to the SonicFinder, providing redundant information that increases feedback. Process and state information sounds exist on two levels, global and local. Global sounds represent the state of the whole system and can be heard anywhere, while local sounds are specific to particular experiments and alter when the user changes from one experiment to another. Navigational information is provided by soundholders, which are auditory landmarks. They can be placed anywhere in the system and get louder as the user moves towards them, decreasing in volume when moving away. This allows the user to wander through an arena much greater than the size of the screen without getting lost and lets him return to specific areas very easily by returning to the soundholder.

In ARKola, a soft drinks factory was modeled, with two users attempting to optimize the factory's output, working remotely from each other and using an audio/video link. Input machines supplied raw materials while output machines capped the bottles and shipped them out. Each machine had an on/off switch and a rate control, with a sound that indicated its status; for example, the bottle dispenser made the sound of clinking glass, with a rhythm that indicated its operating speed. Splashing sounds indicated spilled liquids, while breaking glass showed that bottles were being lost. The users monitored the status of the plant by listening to the auditory clues, and were able to help each other more effectively, since they found it easier to monitor their own machines without having to spend time looking at them, and could hear when something had gone wrong with their partner's part of the system.

Non-speech sounds such as this can convey a lot of meaning very economically. A file arrives in a mailbox and, being a large file, it makes a weighty sound. If it is a text file it makes a rustling noise, whereas a compiled program may make a metallic clang. The sound can be muffled or clear, indicating whether the mailbox is hidden by other windows or not, while the direction of the sound would indicate the position of the mailbox icon on the screen. If the sound then echoes, as it would in a large, empty room, the system load is low. All this information can be presented in a second or so.

Worked exercise *Think of a set of naturally occurring sounds to represent the operations in a standard drawing package (for example, draw, move, copy, delete, rotate).*

Answer This can exercise the imagination! Are there natural analogies? For example, does the physical action, say, of drawing have a sound associated with it? The sound of a pencil on paper may be appropriate but is it identifiable? Similarly, a photocopier whirring could represent the copy operation, and tearing paper delete. Rotate and move are more difficult since the physical operation is not associated with a sound. Perhaps direction and movement can be indicated by sounds becoming nearer or more distant?

Earcons

An alternative to using natural sounds is to devise synthetic sounds. *Earcons* [36] use structured combinations of notes, called *motives*, to represent actions and objects (see Figure 10.2). These vary according to rhythm, pitch, timbre, scale and volume. There are two types of combination of earcon. *Compound earcons* combine different motives to build up a specific action, for example combining the motives for 'create' and 'file'. *Family earcons* represent compound earcons of similar types. As an example, operating system errors and syntax errors would be in the 'error' family. In this way, earcons can be hierarchically structured to represent menus. Earcons are easily grouped and refined owing to their compositional and hierarchical nature, but they require learning to associate with a specific task in the interface since there is an

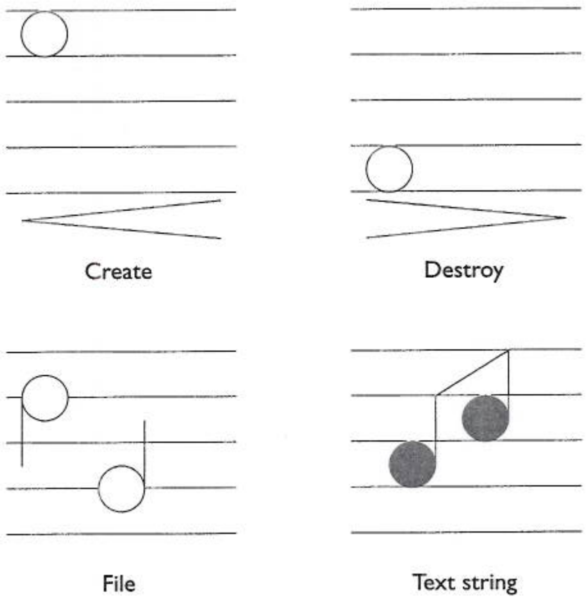


Figure 10.2 Earcons (after Blattner [36], reprinted by permission of Lawrence Erlbaum Associates, Inc.)

arbitrary mapping. Conversely, auditory icons have a semantic relationship with the function that they represent, but can suffer from there being no appropriate sound for some actions.

Earcons provide a structured approach to designing sound for the interface, but can users learn the sounds adequately, and what factors influence their use? Evidence suggests that people can learn to recognize earcons, and that the most important element in distinguishing different sounds is timbre, the characteristic quality of the sound produced by different instruments and voices [47]. Other factors such as pitch, rhythm and register should be used to supplement timbre in creating distinctive sets of musical earcons. Interestingly, the user's musical ability appears to have little effect on his ability to remember earcons: users were able to identify around 80% of earcons from hierarchically ordered sets of 30 or more, regardless of their musical background [45]. It is also possible to create compound earcons by playing sounds in parallel as well as serially. This obviously reduces the time taken to hear the sound but does not affect the user's accuracy [45].

10.3.2 Touch in the interface

We have already considered the importance of touch in our interaction with our environment, in Chapter 1. Touch is the only sense that can be used to both send and receive information. Although it is not yet widely used in interacting with computers, there is a significant research effort in this area and commercial applications are becoming available.

The use of touch in the interface is known as *haptic interaction*. Haptics is a generic term relating to touch, but it can be roughly divided into two areas: cutaneous perception, which is concerned with tactile sensations through the skin; and kinesthetics, which is the perception of movement and position. Both are useful in interaction but they require different technologies.

In Chapter 2, Section 2.6.3, we considered a number of examples of haptic devices, including some based on vibration against the skin (cutaneous) and others on resistance or force feedback (kinesthethic). They facilitate perception of properties such as shape, texture, resistance and temperature as well as comparative spatial properties such as size, height and position. This means haptics can provide information on the character of objects in the interface, as well as more realistic simulations of physical activities, either for entertainment or for training.

In this section, we will look in a little more detail at some of the different types of haptic devices and consider, in particular, the role of haptics in universal design. As we will see, touch can provide both a primary source of information for users with visual impairments and a richer multi-modal experience for sighted users.

One example of a tactile device is an electronic – or soft – braille display. Braille displays are made up of a number of cells (typically between 20 and 80), each containing six or eight electronically controlled pins that move up and down to produce braille representations of characters displayed on the screen. Whereas printed braille normally has six dots per cell, electronic braille typically has eight pins, with the extra two representing additional information about that cell, such as cursor position and character case.

Electronic braille displays benefit from two factors: a well-established tactile notation (braille) and a user group with expertise in using this notation. But can similar techniques be used to provide more generic tactile feedback, such as to display graphics? The problem with using raised pins for this type of display is the resolution required. Braille requires only six or eight pins; a graphical display would require many more, which raises the problem of fitting the necessary number of fast actuators (to move the pins) into a few cubic centimeters. This presents a serious engineering challenge.

The other main type of haptic device is the force feedback device, which provides kinesthetic information back to the user, allowing him to feel resistance, textures, friction and so on. One of the most commonly used examples is the PHANTOM range, from SensAble Technologies. The PHANTOM provides three-dimensional

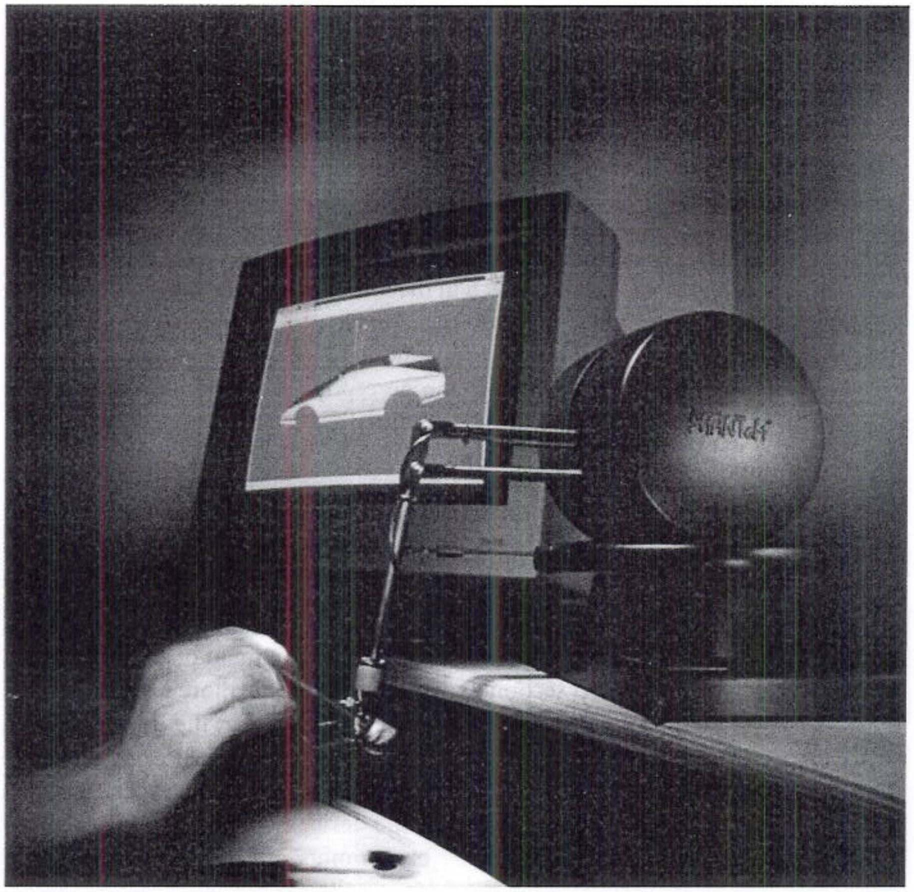


Figure 10.3 A PHANTOM Premium 1.5 haptic device. Source: Courtesy of SensAble Technologies

force feedback, allowing users to touch virtual objects. As well as offering the functionality of the mouse, in addition, the user's movement is monitored by optical sensors on the device, and these, together with models of the virtual objects, are used to calculate the forces applied back to the user. The user therefore can feel the outline and resistance of objects, their texture and position. This type of device has potential application for simulations and training situations where touch is important, such as medicine. It can also be used to provide a haptic 'image' of an interface, providing the user with information about the objects and their functionality based on how they feel. This offers another channel of information, which enhances the richness of the interaction and makes the design more universal.

At present, the hardware needed to support haptic interaction is prohibitively expensive for most users. But this is liable to change as the applications become more widespread and commercially viable.

10.3.3 Handwriting recognition

Like speech, we consider handwriting to be a very natural form of communication. The idea of being able to interpret handwritten input is very appealing, and handwriting appears to offer both textual and graphical input using the same tools. There are problems associated with the use of handwriting as an input medium, however, and in this section we shall consider these. We will first look at the mechanisms for capturing handwritten information, and then look at the problems of interpreting it.

Technology for handwriting recognition

The major piece of technology used to capture handwriting is the digitizing tablet, explained in more detail in Chapter 2. Free-flowing strokes made with a pen are transformed into a series of coordinates, approximately one every 1/50th of a second (or at the sampling rate of the digitizer). Rapid movements produce widely spaced dots, in comparison with slow movements: this introduces immediate errors into the information, since the detail of the stroke between dots is lost, as is the pressure information.

DESIGN FOCUS

Apple Newton

The Apple Newton was the first popular pen-based computer. Other systems, such as the GO PenPoint computer, were available earlier, but did not achieve a significant breakthrough. One reason is that the Newton targeted the organizer market where miniature keyboards were difficult to use (and anyway managers don't use them!). Also this niche market did not demand large amounts of text input, and the graphical interface made it easy to do non-text-based tasks.

Handwriting recognition was acceptable for a number of reasons: the base algorithm achieved a reasonable level of writer-independent recognition; the algorithm was adaptive – it learned the characteristics of the owner during use; and it was word based, so that idiosyncrasies in connected writing could be learnt for common words. But, most important, it was *interactive*. After a word was written, the Newton printed its interpretation of the word; if it was wrong you could try again or correct it letter by letter. This gave the system a chance to learn and meant that errors were not fatal!

In fact, although it has survived, the Apple Newton, like many devices employing novel input techniques, did not achieve the level of success one might have envisaged. This may be because it arrived at the same time as portable computers became *really* portable, and perhaps because the Apple Newton was only suitable for large pockets (of both a sartorial and financial nature). Smaller organizers with both pen-based input and small keyboards are now available, and it remains to be seen whether these achieve the market breakthrough this technology promises.

Digitizing tablets have been refined by incorporating a thin screen on top to display the information, producing *electronic paper*. Advances in screen technology mean that such devices are small and portable enough to be realistically useful in handheld organizers such as the Apple Newton. Information written onto the digitizer can simply be redisplayed, or stored and redisplayed for further reference. However, while this has limited use in itself, systems are most useful when they are able to interpret the strokes received and produce text. It is this recognition that we will look at next.

Recognizing handwriting

The variation between the handwriting of individuals is large (see Figure 10.4); moreover, the handwriting of a single person varies from day to day, and evolves over the years.

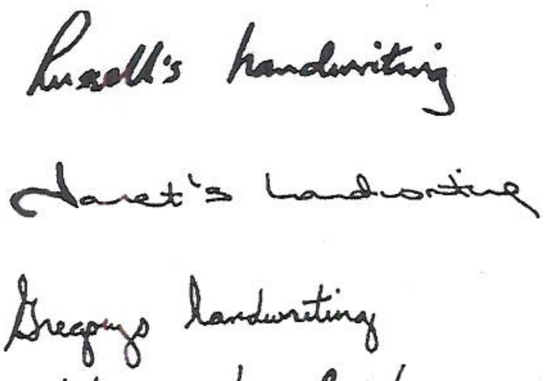


Figure 10.4 Handwriting varies considerably

These problems are reminiscent of those already discussed in speech recognition, and indeed the recognition problem is not dissimilar. The equivalent of co-articulation is also prevalent in handwriting, since different letters are written differently according to the preceding and successive ones. This causes problems for recognition systems, which work by trying to identify the lines that contain text, and then to segment the digitized image into separate characters. This is so difficult to achieve reliably that there are no systems in use today that are good at general cursive script recognition. However, when letters are individually written, with a small separation, the success of systems becomes more respectable, although they have to be trained to recognize the characteristics of the different users. If tested on an untrained person, success is limited again. Many of the solutions that are being attempted in speech recognition are also being tried in handwriting recognition systems, such as whole-word recognition, the use of context to disambiguate characters, and neural networks, which learn by example.

10.3.4 Gesture recognition

Gesture is a component of human–computer interaction that has become the subject of attention in multi-modal systems. Being able to control the computer with certain movements of the hand would be advantageous in many situations where there is no possibility of typing, or when other senses are fully occupied. It could also support communication for people who have hearing loss, if signing could be 'translated' into speech or vice versa. But, like speech, gesture is user dependent, subject to variation and co-articulation. The technology for capturing gestures is expensive, using either computer vision or a special dataglove (see Chapter 2). The dataglove provides easier access to highly accurate information, but is a relatively intrusive technology, requiring the user to wear the special Lycra glove. The interpretation of the sampled data is very difficult, since segmenting the gestures causes problems. A team from Toronto [131] has produced a gesture recognition system that translates hand movements into synthesized speech, using five neural networks working in parallel to learn and then interpret different parts of the inputs.

The Media Room at MIT uses a different approach in order to incorporate gestures into the interaction. The Media Room has one wall that acts as a large screen, with smaller touchscreens on either side of the user, who sits in a central chair. The user can navigate through information using the touchscreens, or by joystick, or by voice. Gestures are incorporated by using a position-sensing cube attached to a wristband worn by the user. The *put that there* system uses this gestural information coupled with speech recognition to allow the user to indicate what should be moved where by pointing at it. This is a much more natural form of interaction than having to specify verbally what it is that has to be moved and describing where it has to go, as well has having the advantage of conciseness. Such a short, simple verbal statement is much more easily interpreted by the speech recognition system than a long and complex one, with the resolution of ambiguity done by interpreting the other mode of interaction, the gesture. Each modality supports the other.

10.4 DESIGNING FOR DIVERSITY

We noted in Chapter 1 that, although we can make general observations about human capabilities, users in fact have different needs and limitations. Interfaces are usually designed to cater for the 'average' user, but unfortunately this may exclude people who are not 'average'. As we saw in the introduction to this chapter, people are diverse and there are many factors that must be taken into account if we are to come close to universal design.

In this section, we will consider briefly some of these factors and the particular challenges that each raises. We will consider three key areas: disability, age and culture.

10.4.1 Designing for users with disabilities

It is estimated that at least 10% of the population of every country has a disability that will affect interaction with computers. Employers and manufacturers of computing equipment have not only a moral responsibility to provide accessible products, but often also a legal responsibility. In many countries, legislation now demands that the workplace must be designed to be accessible or at least adaptable to all – the design of software and hardware should not unnecessarily restrict the job prospects of people with disabilities.

We will look briefly at sensory, physical and cognitive impairments and the issues they raise for interface design.

Visual impairment

The sensory impairment that has attracted the most attention from researchers, perhaps because it is potentially also one of the most debilitating as far as interaction is concerned, is visual impairment. The rise in the use of graphical interfaces reduces the possibilities for visually impaired users. In text-based interaction, screen readers using synthesized speech or braille output devices provided complete access to computers: input relied on touch-typing, with these mechanisms providing the output. However, today the standard interface is graphical. Screen readers and braille output are far more restricted in interpreting the graphical interface, as we saw in Section 10.3.1, meaning that access to computers, and therefore work involving computers, has been reduced rather than expanded for visually impaired people.

There are two key approaches to extending access: the use of sound and the use of touch. We have already considered these in Section 10.3 so we will summarize only briefly here.

A number of systems use sound to provide access to graphical interfaces for people with visual impairment. In Section 10.3.1 we looked at a range of approaches to the use of sound such as speech, earcons and auditory icons. All of these have been used in interfaces for blind users.

So

Sou
user
audi
four
the
incr
cell
conf
item
movi
click

Soun
user
the
tone

Soun
the r
audit
notio

Figur

Soundtrack

Soundtrack is an early example of a word processor with an auditory interface, designed for users who are blind or partially sighted [118]. The visual items in the display have been given auditory analogs, made up of tones, with synthesized speech also being used. A two-row grid of four columns is Soundtrack's main screen (see Figure 10.5); each cell makes a different tone when the cursor is in it, and by using these tones the user can navigate around the system. The tones increase in pitch from left to right, while the two rows have different timbres. Clicking on a cell makes it speak its name, giving precise information that can reorient a user who is lost or confused. Double clicking on a cell reveals a submenu of items associated with the main screen item. Items in the submenu also have tones; moving down the menu causes the tone to fall whilst moving up makes it rise. A single click causes the cell to speak its name, as before, whilst double clicking executes the associated action.

Soundtrack allows text entry by speaking the words or characters as they are entered, with the user having control over the degree of feedback provided. It was found that users tended to count the different tones in order to locate their position on the screen, rather than just listen to the tones themselves, although one user with musical training did use the pitch.

Soundtrack provides an auditory solution to representing a visually based word processor, though the results are not extensible to visual interfaces in general. However, it does show that the human auditory system is capable of coping with the demands of highly interactive systems, and that the notion of auditory interfaces is a reasonable one.

File Menu	Edit Menu	Sound Menu	Format Menu
Alert	Dialog	Document1	Document2

Figure 10.5 The screen division in Soundtrack. Source: Courtesy of Alistair D. N. Edwards

Soundtrack (see the box above) was an early example of the use of non-speech sound to provide an auditory interface to a word processor. A major limitation of this application was the fact that it was a specialized system; it could not be used to augment commercially available software. Ideally, users with disabilities should have

Design Focus

Mathematics for the blind

Solve the following equation: $3(x-2)+4=7-2(3-x)$.

Did you do it in your head or use a piece of paper? When an equation is even slightly complex the instant response of a sighted person is to reach for paper and pencil. The paper acts as an *external memory*, allowing you to record and recall previous steps in a calculation. Blind children learning mathematics have to perform nearly all such calculations in their head, putting them at a severe disadvantage.

Mathtalk is a system developed as part of a European project to create a mathematics workstation for blind people [330]. It uses speech synthesis to speak formulae, and keyboard input to navigate and manipulate them. The first stage, simply speaking a formula out loud, is complex in itself. Given the spoken equation 'three *x* plus four equals seven', how do you know whether this is '$3x+4=7$' or '$3(x+4)=7$'? To make it unambiguous one could say the latter as 'three open bracket *x* plus four close bracket equals seven', but this soon becomes very tedious. In fact, when reading mathematics people use several cues in their speech: longer and shorter gaps between terms, and prosody: rising and falling pitch (see Figure 10.6). The Mathtalk system includes a set of rules for generating such patterns suitable for most equations.

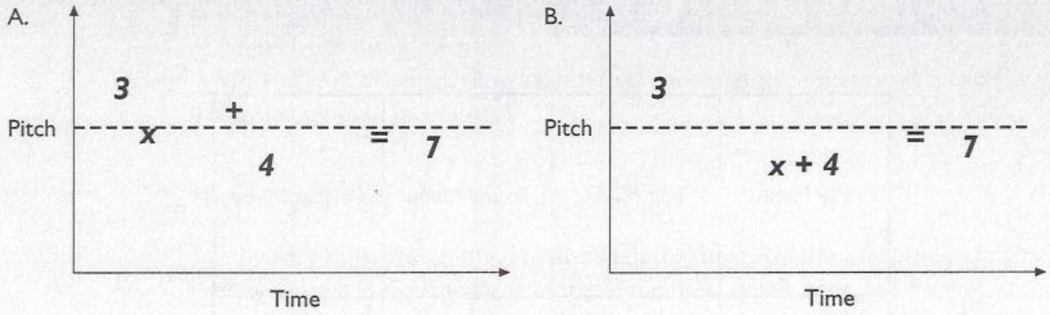


Figure 10.6 Pausing and pitch help distinguish between two expressions

Visual interaction with paper isn't just at the level of reading and writing whole equations. Recall from Chapter 1 that reading usually includes *regressions* where our eyes move backwards as well as forwards through text. Also, when seeing graphical material (remember that mathematics makes heavy use of brackets, symbols, superscripts, etc.), we rely on getting a quick feel for the material at a glance before examining it in detail. Both of these factors are crucial when reading an equation and so Mathtalk supports rapid keyboard-based navigation *within* each equation, and *algebra earcons*, short motives based on the rise and fall of the prosody of an equation.

Notice that Mathtalk uses *keyboard* input combined with speech output. Speech input is slow and error-prone compared with a keyboard. Braille output can also be used for mathematics, but only a small percentage of blind people read braille. Choosing the right input and output devices requires a deep knowledge of the user population and careful analysis of the intended tasks.

access to the same applications as anyone else. Outspoken is a Macintosh application that uses synthetic speech to make other Macintosh applications available to visually impaired users. A common problem with this and other screen readers and talking browsers (see Section 10.3.1) is the sheer amount of information represented. Browsing is difficult and all of the information must be held in the head of the user, putting a heavy load on memory.

A more recent development is the use of touch in the interface. As we saw in Section 10.3.2, there are two key approaches to this, both of which can be used to support people with visual impairment. Tactile interaction is already widely used in electronic braille displays, which represent what is on the screen through a dynamic braille output. It could also be used to provide more information about graphics and shape, if the engineering challenges of building higher resolution tactile devices can be overcome. Force feedback devices also have the potential to improve accessibility to users with visual impairment, since elements in the interface can be touched, and edges, textures and behavior used to indicate objects and actions. A limitation of this technology at present is that objects must be rendered using specialist software in order for the devices to calculate the appropriate force to apply back to the user. This again represents a move away from use of generic applications to specialist applications. However, it is likely that major applications will become 'haptic enabled' in the near future.

Hearing impairment

Compared with a visual disability where the impact on interacting with a graphical interface is immediately obvious, a hearing impairment may appear to have little impact on the use of an interface. After all, it is the visual not the auditory channel that is predominantly used. To an extent this is true, and computer technology can actually enhance communication opportunities for people with hearing loss. Email and instant messaging are great levellers and can be used equally by hearing and deaf users alike.

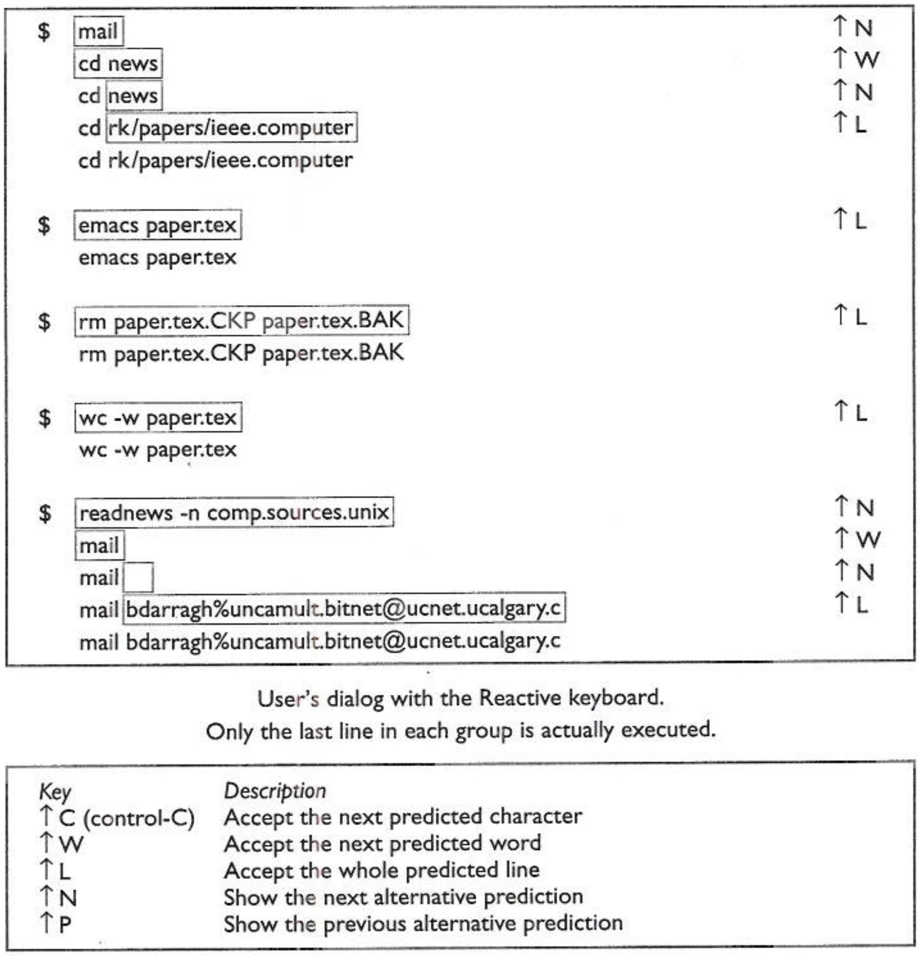
Gesture recognition has also been proposed to enable translation of signing to speech or text, again to improve communication particularly with non-signers.

However, the increase in multimedia and the use of sound in interfaces has, ironically, created some access difficulties for people with hearing problems. Many multimedia presentations contain auditory narrative. If this is not supplemented by textual captions, this information is lost to deaf users. Captioning audio content, where there is not already a graphical or textual version, also has the advantage of making audio files easier and more efficient to index and search, which in turn enhances the experience of all users – a sure sign of good universal design!

Physical impairment

Users with physical disabilities vary in the amount of control and movement that they have over their hands, but many find the precision required in mouse control difficult. Speech input and output is an option for those without speech difficulties.

An alternative is the eyegaze system (Chapter 2), which tracks eye movements to control the cursor, or a keyboard driver that can be attached to the user's head. If the user is unable to control head movement, gesture and movement tracking can be used to allow the user control. If the user has limited use of a keyboard, a predictive system, such as the Reactive keyboard [157], can help, by anticipating the commands that are being typed and offering them for execution. This can cut the typing requirement considerably. Predictions are based on what the user has typed in the current session or a previous one. The predictions therefore anticipate within the context in which the user is currently working (for example, operating system commands, programming text or free text). Figure 10.7 shows an interaction using the Reactive keyboard.



```
$  mail                                                         ↑N
   cd news                                                      ↑W
   cd news                                                      ↑N
   cd rk/papers/ieee.computer                                   ↑L
   cd rk/papers/ieee.computer

$  emacs paper.tex                                              ↑L
   emacs paper.tex

$  rm paper.tex.CKP paper.tex.BAK                               ↑L
   rm paper.tex.CKP paper.tex.BAK

$  wc -w paper.tex                                              ↑L
   wc -w paper.tex

$  readnews -n comp.sources.unix                                ↑N
   mail                                                         ↑W
   mail                                                         ↑N
   mail bdarragh%uncamult.bitnet@ucnet.ucalgary.c               ↑L
   mail bdarragh%uncamult.bitnet@ucnet.ucalgary.c
```

User's dialog with the Reactive keyboard.
Only the last line in each group is actually executed.

Key	*Description*
↑C (control-C)	Accept the next predicted character
↑W	Accept the next predicted word
↑L	Accept the whole predicted line
↑N	Show the next alternative prediction
↑P	Show the previous alternative prediction

Reactive keyboard commands

Figure 10.7 An interaction using the Reactive keyboard. Source: Courtesy of Saul Greenberg

ents to
l. If the
can be
dictive
mands
equire-
current
context
mands,
eactive

` N
` W
↑ N
↑ L

↑ L

↑ L

↑ L

↑ N
↑ W
↑ N
↑ L

of

Speech impairment

For users with speech and hearing impairments, multimedia systems provide a number of tools for communication, including synthetic speech (see Section 10.3.1) and text-based communication and conferencing systems (see Chapter 19). Textual communication is slow, which can lower the effectiveness of the communication. Predictive algorithms have been used to anticipate the words used and fill them in, to reduce the amount of typing required. Conventions can help to provide context, which is lost from face-to-face communication, for example the 'smilie' :-), to indicate a joke. Facilities to allow turn-taking protocols to be established also help natural communication [256]. Speech synthesis also needs to be rapid to reflect natural conversational pace, so responses can be pre-programmed and selected using a single switch.

Dyslexia

Users with cognitive disabilities such as dyslexia can find textual information difficult. In severe cases, speech input and output can alleviate the need to read and write and allow more accurate input and output. In cases where the problem is less severe, spelling correction facilities can help users. However, these need to be designed carefully: often conventional spelling correction programs are useless for dyslexic users since the programs do not recognize their idiosyncratic word construction methods. As well as simple transpositions of characters, dyslexic users may spell phonetically, and correction programs must be able to deal with these errors.

Consistent navigation structure and clear signposting cues are also important to people with dyslexia. Color coding information can help in some cases and provision of graphical information to support textual can make the meaning of text easier to grasp.

Autism

Autism affects a person's ability to communicate and interact with people around them and to make sense of their environment. This manifests itself in a range of ways but is characterized by the *triad of impairments*:

1. Social interaction – problems in relating to others in a meaningful way or responding appropriately to social situations.
2. Communication – problems in understanding verbal and textual language including the use of gestures and expressions.
3. Imagination – problems with rigidity of thought processes, which may lead to repetitive behavior and inflexibility.

How might universal design of technology assist people with autism? There are two main areas of interest: communication and education.

Communication and social interaction are major areas of difficulty for people with autism. Computers, on the other hand, are often motivating, perhaps because they are relatively consistent, predictable and impersonal in their responses. The user is in control. Computer-mediated communication and virtual environments have been suggested as possible ways of enabling people with autism to communicate more easily with others, by giving the user control over the situation. Some people with autism have difficulties with language and may be helped by graphical representations of information and graphical input to produce text and speech. Again this is supported by providing redundancy in the design.

Computers may also have a role to play in education of children with autism, particularly by enabling them to experience (through virtual environments and games) social situations and learn appropriate responses. This can again provide a secure and consistent environment where the child is in control of his own learning. The use of computers to support people with autism in this way is still a new research area and it is likely that new software and tools will develop in the next few years.

10.4.2 Designing for different age groups

We have considered how people differ along a range of sensory, physical and cognitive abilities. However, there are other areas of diversity that impact upon the way we design interfaces. One of these is age. In particular, older people and children have specific needs when it comes to interactive technology.

Older people

The proportion of older people in the population is growing steadily. Contrary to popular stereotyping, there is no evidence that older people are averse to using new technologies, so this group represents a major and growing market for interactive applications. People are living longer, have more leisure time and disposable income, and older people have increased independence. These factors have all led to an increase in older users.

But the requirements of the older population may differ significantly from other population groups, and will vary considerably within the population group. The proportion of disabilities increases with age: more than half of people over 65 have some kind of disability. Just as in younger people with disabilities, technology can provide support for failing vision, hearing, speech and mobility. New communication tools, such as email and instant messaging, can provide social interaction in cases where lack of mobility or speech difficulties reduce face-to-face possibilities. Mobile technologies can be used to provide memory aids where there is age-related memory loss.

Some older users, while not averse to using technology, may lack familiarity with it and fear learning. They may find the terminology used in manuals and training books difficult to follow and alien (words like 'monitor' and 'boot', for example, may have a completely different meaning to an older person than a young person). Interests and concerns may also be different from younger users.

:s. The
ıments
nunic-
Some
ıphical
;peech.

ıutism,
ıts and
›rovide
is own
· is still
› in the

cognit-
way we
en have

trary to
ing new
eractive
income,
d to an

m other
up. The
65 have
logy can
munica-
ction in
ibilities.
-related

rity with
training
example,
person).

Once again, basic universal design principles are important here. Access to information must make use of redundancy and support the use of access technologies. Designs must be clear and simple and forgiving of errors. In addition, thought needs to be given to sympathetic and relevant training aimed at the user's current knowledge and skills.

In spite of the potential benefits of interactive technology to older people, very little attention has been paid to this area until recently. Researchers are now beginning to address issues such as how technology can best support older people, what the key design issues are, and how older people can be effectively included in the design process [46], and this area is likely to grow in importance in the future.

Children

Like older people, children have distinct needs when it comes to technology, and again, as a population, they are diverse. The requirements of a three year old will be quite different from those of a 12 year old, as will be the methods that can be used to uncover them. Children are, however, different from adults, and have their own goals and likes and dislikes. It is therefore important to involve them in the design of interactive systems that are for their use, though this in itself can be challenging as they may not share the designer's vocabulary or be able to verbalize what they think. Design approaches have therefore been developed specifically to include children actively as members of the design team. Alison Druin's Cooperative Inquiry approach [110] is based on contextual inquiry and participatory design, which we will consider in more detail in Chapter 13. Children are included in an *intergenerational design team* that focusses on understanding and analyzing context. Team members, including children, use a range of sketching and note-taking techniques to record their observations. Paper prototyping, using art tools familiar to children, enables both adults and children to participate in building and refining prototype designs on an equal footing. The approach has been used effectively to develop a range of new technologies for children.

As well as their likes and dislikes, children's abilities will also be different from those of adults. Younger children may have difficulty using a keyboard for instance, and may not have well-developed hand–eye coordination. Pen-based interfaces can be a useful alternative input device [300]. Again, universal design principles guide us in designing interfaces that children can use. Interfaces that allow multiple modes of input, including touch or handwriting, may be easier for children than keyboard and mouse. Redundant displays, where information is presented through text, graphics and sound will also enhance their experience.

10.4.3 Designing for cultural differences

The final area of diversity we will consider is cultural difference. Cultural difference is often used synonymously with national differences but this is too simplistic. Whilst there are clearly important national cultural differences, such as those we saw in Chapter 5, other factors such as age, gender, race, sexuality, class, religion and political persuasion, may all influence an individual's response to a system. This is particularly the case when considering websites where often the explicit intention is to design for a particular culture or subculture.

Clearly, while all of these contribute to a person's cultural identity, they will not all always be relevant in the design of a given system. However, we can draw out some key factors that we need to consider carefully if we are to practice universal design. These include language, cultural symbols, gestures and use of color.

We encountered the problem of localization of software in Chapter 5. While toolkits, with different language resource databases, facilitate the translation of menu items, error messages and other text into the local language, this does not fully deal with the language issue. Layouts and designs may reflect a language read from left to right and top to bottom, which will be unworkable with languages that do not follow this pattern.

Similarly, symbols have different meanings in different cultures. As we saw in Chapter 5, ticks ✓ and crosses ✗ represent positive and negative respectively in some cultures, and are interchangeable in others. The rainbow is a symbol of covenant with God in Judeo–Christian religions, of diversity in the gay community and of hope and peace in the cooperative movement. We cannot assume that everyone will interpret symbols in the same way and should ensure that alternative meanings of symbols will not create problems or confusion for the user. The study of the meaning of symbols is known as *semiotics* and is a worthwhile diversion for the student of universal design.

Another area where diversity can cause misunderstanding is in the use of gesture. Recently, one of the authors was teaching a new class of international students and was disconcerted to see one sitting in the front row, smiling and shaking his head. After the lecture this student came and asked a question. Every time the author asked the student if he understood the explanation, he shook his head, so further explanation ensued, much to the frustration of the student! It was only after a few minutes that it became clear: the student was from India and his gestural convention was to shake his head in agreement, the opposite of the European interpretation of the gesture. Use of gesture is quite common in video and animation and care must be taken with differences such as this. As interactions begin to incorporate gesture in virtual reality and avatars, issues such as this will become even more significant.

Finally, colors are often used in interfaces to reflect 'universal' conventions, such as red for danger and green for go. But how universal are these conventions? In fact, red and green mean many different things in different countries. As well as danger, red represents life (India), happiness (China) and royalty (France). Green is a symbol of fertility (Egypt) and youth (China) as well as safety (Anglo-American). It is difficult to assume any universal interpretation of color but the intended significance of particular colors can be supported and clarified through redundancy – providing the same information in another form as well.

on and
This is
ation is

will not
aw out
niversal

While
f menu
lly deal
om left
do not

saw in
in some
ovenant
and of
one will
nings of
e mean-
ndent of

gesture.
tudents
king his
e author
further
after a
ral con-
n inter-
tion and
rporate
n more

ns, such
? In fact,
danger,
een is a
erican).
ntended
undancy

10.5 SUMMARY

Universal design is about designing systems that are accessible by all users in all circumstances, taking account of human diversity in disabilities, age and culture. Universal design helps everyone – for example, designing a system so that it can be used by someone who is deaf or hard of hearing will benefit other people working in noisy environments or without audio facilities. Designing to be accessible to screen-reading systems will make websites better for mobile users and older browsers.

Multi-modal systems provide access to system information and functionality through a range of different input and output channels, exploiting redundancy. Such systems will enable users with sensory, physical or cognitive impairments to make use of the channels that they can use most effectively. But all users benefit from multi-modal systems that utilize more of our senses in an involving interactive experience.

For any design choice we should ask ourselves whether our decision is excluding someone and whether there are any potential confusions or misunderstandings in our choice.

EXERCISES

10.1 Is multi-modality always a good thing? Justify your answer.

10.2 What are (i) auditory icons and (ii) earcons? How can they be used to benefit both visually impaired and sighted users?

10.3 Research your country's legislation relating to accessibility of technology for disabled people. What are the implications of this to your future career in computing?

10.4 Take your university website or another site of your choice and assess it for accessibility using Bobby. How would you recommend improving the site?

10.5 How could systems be made more accessible to older users?

10.6 Interview either (i) a person you know over 65 or (ii) a child you know under 16 about their experience, attitude and expectations of computers. What factors would you take into account if you were designing a website aimed at this person?

10.7 Use the screen reader simulation available at www.webaim.org/simulations/screenreader to experience something of what it is like to access the web using a screen reader. Can you find the answers to the test questions on the site?

RECOMMENDED READING

M. F. Story, J. L. Mueller and R. L. Mace, *The Universal Design File: Designing for People of All Ages and Abilities*, The Center for Universal Design, NC State University, USA, 1998. Available at: www.design.ncsu.edu/cud/pubs/center/books/ud_file/toc3b14.htm (last accessed March 2003).

A full discussion of universal design principles with examples to illustrate.

A. D. N. Edwards, editor, *Extra-ordinary Human–Computer Interaction*, Cambridge University Press, 1993.

A collection of papers representing research on interfaces for users with disabilities. The first of its kind.

The Web Accessibility Initiative: www.w3.org/WAI/

The World Wide Web Consortium's own project to make the web universally accessible. Contains extensive advice and guidelines.

I. Pitt and A. Edwards, *Design of Speech-based Devices: A Practical Guide*, Springer, 2002.

Covers the use of speech in the interface, including both the practical and theoretical issues.

J. Yunker, *Beyond Borders: Web Globalization Strategies*, New Riders Publishing, 2002.

A practical guide to internationalization of websites.

D. J. Moore and J. Taylor, Interactive multimedia systems for people with autism, *Journal of Educational Media*, Vol. 25, No. 3, pp. 169–77, 2001.

A useful review of the actual and potential uses of interactive media in the education of people with autism.