Chapter 2

Human Computer Interaction

"Technology is a wonderful servant but a terrible master, and yet we are grooming it for that ultimate promotion"

[Martin Moore-Ede *The 24-hour Society*, 1993]

Overview

This chapter summarises the traditional approaches to the human-computer interface. It includes a literature survey of the last decade of specialist journals and analyses the proportion of articles dealing with issues of real-time control. It concludes by highlighting those areas which need to be examined afresh in the light of the increasing prevalence of interactive computer operation.

2.1 Interaction with Computers

Viewed at their simplest level, computers are devices which take input from the real world, perform a processing task based on the input and produce a useful output. The basic problem many computer designers face is that of bridging the gap between the communication methods of humans and computers (see Figure 2.1).

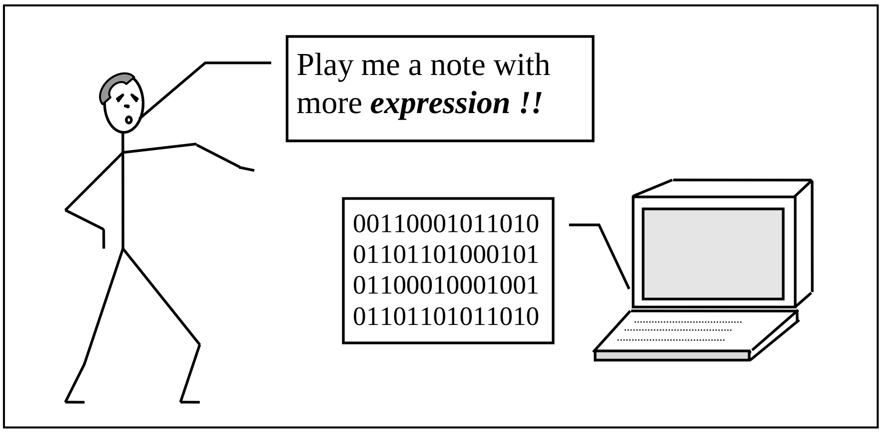

Figure 2.1: The gap between musicians and computers

People have goals and feelings and they communicate using verbal language and physical gestures. Computers work at the level of binary code, blindly following a list of primitive instructions. The designers of computer systems have the formidable task of mediating between these vastly different worlds.

It should, perhaps, surprise us that so many interfaces (designed by humans) seem to exist purely for the convenience of the computer. The very existence of the advertising slogan 'User Friendly' (to refer to a product which actually bothers to take the user's needs into consideration) is indicative of the sorry state of many computer interfaces.

2.2 Definition of HCI

The field of study known as Human Computer Interaction (HCI) emerged as a subject area in its own right during the 1980s. The Association for Computing Machinery Special Interest Group on Computer-Human Interaction recently defined it as :

> *"... a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them."*
>
> [ACM SIGCHI 1992].

A comparison of this definition with one just a decade earlier demonstrates the rapid change of emphasis in the subject area. Tom Moran described the 'Man-Machine Interface (MMI)', as it was then known, as

> *"those aspects of the system that the user comes into contact with."*
>
> [Moran 1981].

This rather implies a 'fait accompli' where users just have to put up with whatever the engineers have designed.

2.3 Models of HCI

Throughout HCI's brief history academic researchers have attempted to gain a better understanding of the complex issues involved at the meeting point of computers and people. To aid this understanding, they have developed a range of 'models' which attempt to describe the styles of interaction between machine and human. In this section the main model types are outlined.

2.3.1 The early days of computing

The engineering tasks associated with the first two decades of computing history (1950s and 60s) were primarily directed towards getting computers to work and then keeping them operational. Computers tended to be vast devices which consumed large amounts of energy and constantly needed maintenance. Users were assumed to be engineering operators who knew the machine operation intimately. Data was entered in a variety of formats - from binary codes stored on punched card or paper tape to the setting of control registers by spin-wheels. All computer operation was by 'batch' process; users submitted their data and were informed of the result at a later date. There was no need to study the 'interface' between machine and operator because it was assumed that users would be happy to code their programs and data into whatever form the computer could process.

2.3.2 Personal computers

The development of cheap personal computers for home and office use in the early 1980s forced manufacturers to consider the 'Man-Machine Interface' (MMI) as a major part of the design phase of each new product. As the MMI was the first point of contact with a potential user, concepts such as 'ease of use' and 'speed of response' became a focus for commercial rivalry. Whilst manufacturers designed hardware and software primarily to sell products, certain groups were more interested in the long-term implications of different user interfaces.

> *"Academic researchers . . . were concerned about how the use of computers might enrich the work and personal lives of people. In particular, they focused on the capabilities and limitations of human users, that is, understanding the 'people side' of the interaction with computer systems"*
>
> [Preece, 1994].

2.3.3 The Human as Information Processor

Between the mid-1970s and mid-1980s this commercial rivalry led to an increasing need to monitor and improve the information throughput of systems involving computers and people. During this time a notion was developed which was to prove extremely influential even today; that of modelling a human being as a processor of information provided by the computer.

Card, Moran and Newell (1983) characterise the interaction between humans and computers as a 'recognise-act' cycle. Human beings are seen as the information bottleneck, taking measurable times to read, interpret and respond to symbols displayed on the computer screen (see Figure 2.2).

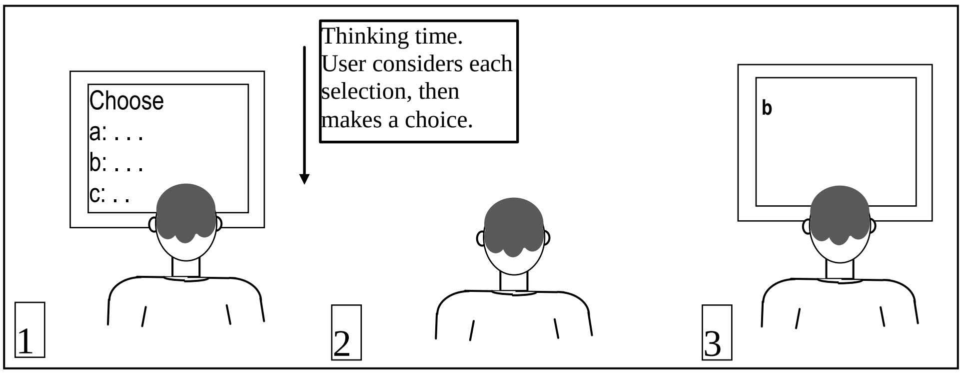

Figure 2.2: The 'recognise-act' cycle

Ironically, a person is modelled in the same sort of analytical way as one might model a robot: a reactive tool following a simple set of instructions.

> *"Perceptual processes activate working memory which in turn activate motor processes resulting in action"*
>
> [Monk, 1995]

In this model the computer is totally in charge of the interaction protocol. Users can make decisions but based only on the options presented on the screen. The time taken to make these decisions and to take appropriate action is the metric by which human performance is measured and subsequent improvements made in the interface design.

It is important to note that the phrase 'interactive computer systems' is often applied to those systems based around this model. The word 'interactive' simply implies an exchange of information between a machine and its user, and has no bearing on how dynamic or continuous that exchange is.

2.3.4 Scripts for human tasks

Card et al (1983) also outlined a method of analysing the tasks which a human wants to perform with a computer system. It is known as *GOMS* which is an acronym for Goals, Operators, Methods and Selection rules. This method is based on the fundamental assumption that users have a series of tasks to do in order to achieve their goals, and that these tasks can be broken down into a hierarchical series of simpler tasks resulting eventually in a sequential list of actions to be taken. An ‘action’ is defined as a task which needs no further problem solving (e.g. “press the Return Key”).

The result of the consistent application of the GOMS task analysis is a series of ‘scripts’ which define exactly what to do in order to achieve every possible combination of user goals on a particular system. This method does indeed allow careful analysis of ‘closed’ systems (those which define a finite set of user responses) but is often inappropriate when applied to more open ended or responsive systems.

2.3.5 Mental Models

The development of larger and more complex systems (where comprehensive task analysis would be impractical) prompted researchers to try to understand more about what was going on inside a user's mind when using a computer.

The ‘schemata’ model [Schank, 1977] assumes that users develop internal representations of the world in the form of networks of related knowledge that can be applied to different tasks. When a ‘schema’ is applied consistently to a particular task, the user develops a ‘script’ (for example “how to edit a document on a word-processor”). This is very similar to the GOMS hierarchical analysis mentioned above, except that the assumption here is that users unconsciously develop their own internal scripts for each computer task they have to do (in contrast to a GOMS analytical *written* script drawn up by an HCI researcher).

2.3.6 Structural v Functional

Users are thought to develop two distinct types of mental models about a computer system; ‘structural’ and ‘functional’.

> *"A structural model is one where it is assumed that the user has internalized the structure of how the device or system works in memory, while a functional model assumes that the user has internalized procedural knowledge about how to use the device or system"*
>
> [Preece 1994]

In other words the key issue is whether the user is required to know how the system operates internally, or simply to know how they use it. It is proposed that the better a device's design, the less the user is concerned with the mechanics of its operation. Examples of household systems which require users only to have functional models are the telephone and television, whereas the timer programming function of a video recorder typically requires more of a structural knowledge and hence is more difficult to operate.

2.3.7 Automatic v Controlled

A further distinction is made as to whether the computer operator uses 'automatic' or 'controlled' processes [Shiffrin, 1977]. An automatic process is defined as an action that has been learnt by the user so that effecting the action in response to a certain stimulus takes little or no cognitive effort. The most obvious example is a car driver slamming a foot on the brake pedal in response an object unexpectedly veering into the path of the car. By contrast, a controlled process is one which demands a certain amount of conscious control and cognitive effort (e.g. reading through a list of menu options and selecting the most appropriate one).

Many computer interfaces are designed to raise the profile of controlled processes at the expense of automatic ones. This has serious implications for the control of interactive dynamic systems and will be returned to later in this thesis.

2.3.8 Humans as actors

A completely different type of mental model rebels against the idea of modelling humans as automata. Bannon [1991] coins the term human *actors* (meaning "one who initiates actions") so that

> *"emphasis is placed on the person as an autonomous agent that has the capacity to regulate and control his or her behaviour, rather than being a simple passive element in a human-machine system."*

This model offers a much more realistic interpretation of the ways in which humans frequently operate in an increasingly computerised world. It is too easy for engineers to design systems which make all the decisions, pausing every so often to ask the human operator for a carefully structured morsel of input data.

2.4 Interface Metaphors

Researchers have responded to the different HCI models by developing different ways for information to be transferred between people and computer systems. The assumptions made about the user in the design of software and hardware systems have given rise to a number of different styles of computer interface. Some of the most influential of these interface metaphors are explored here.

2.4.1 Form-filling

Perhaps the simplest style of interaction consists of the user being required to answer questions or fill in numbers in a fixed format rather like filling out a form [Shneiderman, 1992]. There is no free dialogue, so this can hardly be described as a flexible interface style. However, it is one which favours those applications where the user is simply the provider of information sets, for example as used in travel-agent booking systems (see Figure 2.3).

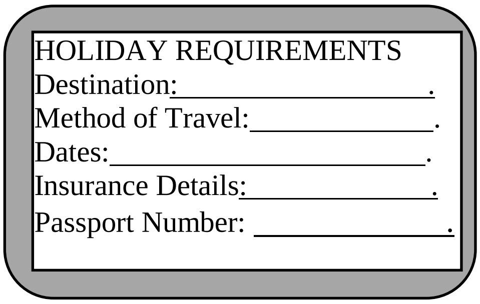

Figure 2.3: An example form-filling interface in a travel agent's office

2.4.2 Command Line

One of the most prevalent interface metaphors throughout the 1970s and early 1980s was the 'command line'. Here the computer gives a 'prompt' to the user (for example an on-screen C:> in MS-DOS) and awaits the user's instructions. These instructions are usually in the form of a set of tightly syntactically defined letter commands (for example *mv file1 file2* which under the UNIX system will rename 'file1' to 'file2'). An example command line environment is shown in Figure 2.4.

```
host% mv start.txt chapter1.txt
host% ls *.txt
chapter1.txt chapter2.txt
host% rm chapter3.txt
chapter3.txt: No such file or directory
host% rm chapter2.txt
host%
host%
```

Figure 2.4: An example command-line environment

The advantages of a command-line environment are ease of programming, no restriction on the choice of the next command, and a fast way of issuing instructions for those who have become accustomed to the system by using it for a long time. The disadvantage is the apparently cryptic and unforgiving set of keyword commands and associated syntax which a user has to know before being able to use the system effectively.

2.4.3 Windows, Icons, Menus, Pointers (WIMP)

Perhaps the most influential of all current interaction environments is the 'WIMP' interface. The origins of graphical windowing systems go back to the 1960s, although the ideas were developed in greater detail by researchers at the Xerox corporation during the pioneering 'Dynabook' project [Ryan, 1991] in the 1970s. Their goal of producing a book-sized personal communication system was not achieved at the time, but their ideas of a user interface that could be used by anyone were to have a dramatic effect on all subsequent interface developments. The same project gave birth to the 'Star' system which was later used as the model for the tremendously successful Apple Macintosh series of personal computers launched in the mid-1980s.

The fundamental goal behind such designs is to give the user a meaningful working metaphor, for example an office or 'desktop' representation (with filing cabinets, waste-paper baskets and in-trays) as opposed to the arbitrary code words of command-line interfaces. To achieve this goal, the user is expected to point at 'objects' on the screen to access their functionality. This is traditionally done using a mouse pointer, but there are a variety of other devices available such as tracker-balls, joysticks, graphics tablets and 'concept keyboards', each of which has its own ergonomic advantages and disadvantages.

Data of different types are arranged in on-screen 'windows' (see Figure 2.5). These mimic a set of paper pages on a desk, and can be moved around with the pointing device to get the best arrangement for the user.

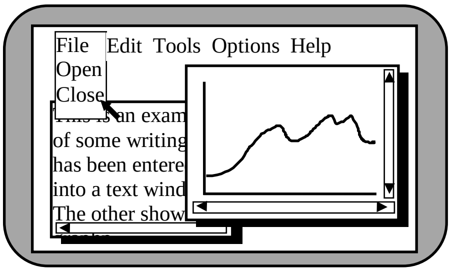

Figure 2.5: An example 'WIMP' interface

Not all computer operations will fit into the graphical metaphor of a desktop, so the concept of the 'menu' emerged:

> *"A menu is a set of options displayed on the screen where the selection and execution of one (or more) of the options results in a change in the state of the interface . . . Unlike command-driven systems, menus have the advantage that users do not have to remember the item they want, they only need to recognise it"*
>
> [Preece 1994].

It is this 'search and recognise' requirement of menuing systems (which have become a widespread interface metaphor in the computing world) which needs to be seriously questioned. This issue will be revisited towards the end of this chapter.

2.4.4 Direct Manipulation

Ben Shneiderman coined the term 'Direct Manipulation' in the 1980s to describe those computer systems which seemed to actively engage the user's attention more fully than others. He noticed that these interfaces captivated their human users by presenting them with a series of on-screen objects which could be moved, edited, explored and executed by the actions of pointing, clicking, pressing and dragging. The objects responded with immediate and continuous feedback in terms of visual display or sound. Video games are cited as commercially successful examples of this genre [Shneiderman 1983].

Shneiderman stated that direct manipulation interfaces can be identified by the following three characteristics:

- Continuous representation of objects.
- Physical actions instead of complex syntax.
- Rapidly reversible operations which have an immediately obvious impact on the object in question.

The advantage of direct manipulation systems is that beginners can explore the functionality by simply ‘trying out’ the interface, as most actions are reversible. Advanced users find that their most common tasks are executed increasingly as ‘automatic processes’ (section 2.3.7) rather than requiring much thought-power.

The early ‘WIMP’ interface designs were based on these same fundamental ideas in the quest to find an interface which would suit both new and experienced users. Unfortunately direct manipulation interfaces are not easy to program. They require a large processing overhead and it is often difficult for programmers to think of suitable metaphors for representing complex actions. The result has been that WIMP interfaces have become far from direct. As their complexity has increased, the user's operation has become focused on menu selection. Much HCI work is centred on discovering the optimum structure of hierarchical menu systems! A significant part of the HCI literature (see section 2.5.2.1) nowadays seems to simply equate ‘the use of a mouse’ with ‘Direct Manipulation’, irrespective of the *type* of interaction that occurs. With more and more real-time interactive systems being required, it is perhaps the right time to investigate other methods of interaction.

2.4.5 Computer Supported Cooperative Work (CSCW)

A developing focus of HCI researchers relates to the management of people-to-people communication using computer systems as intermediaries (see section 2.5.3.4). This is hardly surprising due to the academic and media focus on the worldwide communications network (notably the World Wide Web). For example, work is being done at the University of York to manage the complexities of Video Conferencing systems [Monk 1995].

This focus is likely to get stronger as the amount of information being exchanged around the world increases and more people become connected to the network. One possible result of this however could be the implicit acceptance of the WIMP interface as the pinnacle of achievement in human-computer interaction. It is the purpose of this thesis to point the way to new areas of interaction which may be far more appropriate to dynamic systems.

2.4.6 Virtual Reality (VR)

Another major growth area is the field of Virtual Environments (see section 2.5.3.16) which aim to allow users to experience 'Virtual Reality'. This means that users are given the means of interacting with the computer in ways that they would interact with objects in the real world. It is often associated with 'immersive' interfaces, where the user puts on a helmet or goggles (for stereo audio and visual output) and gloves (as input devices). An example headset is shown in Figure 2.6.

Figure 2.6: A virtual reality headset in action

Preece et al define the subject area as follows :

> *"Virtual environments and virtual realities typically offer a sense of direct physical presence, sensory cues in three dimensions, and a natural form of interaction (for example via natural gestures)"*
>
> [Preece 1994].

There is a disagreement in the academic world as to whether VR is an important new interface development or is merely a temporary distraction.

> *"Virtual reality is a lively new direction for those who seek the immersion experience, where they block out the real world by having goggles on their heads"*
>
> Ben Shneiderman [cited Preece, 1994].

> *"Human beings are used to operating in a three-dimensional spatial environment . . . I am convinced that integration of the real world with a virtual world offers the best compromise with today's technology"*
>
> Roy Kalawsky, British Aerospace Virtual Environment Laboratory [cited Preece, 1994].

> *"Virtual Reality is actually making us move further inside the world of the computer, rather than make computers access further into our world"*
>
> David Creasey, University of York [personal communication].

Whatever the future for VR systems, the fundamental questions of how users should interact with the system still remain. Some virtual environments e.g. [Bromwich, 1995] ironically require users to point their virtual hand at virtual menus in order to navigate around the system!

2.5 Review of HCI literature

Each of the following journals was reviewed for the ten-year period 1990-99, giving a total of 1374 published papers:

- International Journal of Human Computer Interaction
- Human Computer Interaction
- Interacting with Computers
- Behaviour and Information Technology
- International Journal of Human-Computer Studies (formerly Man-Machine Studies).

2.5.1 Categories of HCI Literature

Every paper was placed into one of 23 categories according to its main subject. These categories were created and refined as the survey progressed. Many papers addressed several issues, but each was placed in only one category according to its central focus.

Figure 2.7 shows the relative numbers of papers in each category (as a percentage of the total number of papers reviewed). The categories shaded black are those regarded as potentially relevant to this study. In the discussions which follow, these relevant areas are examined in detail.

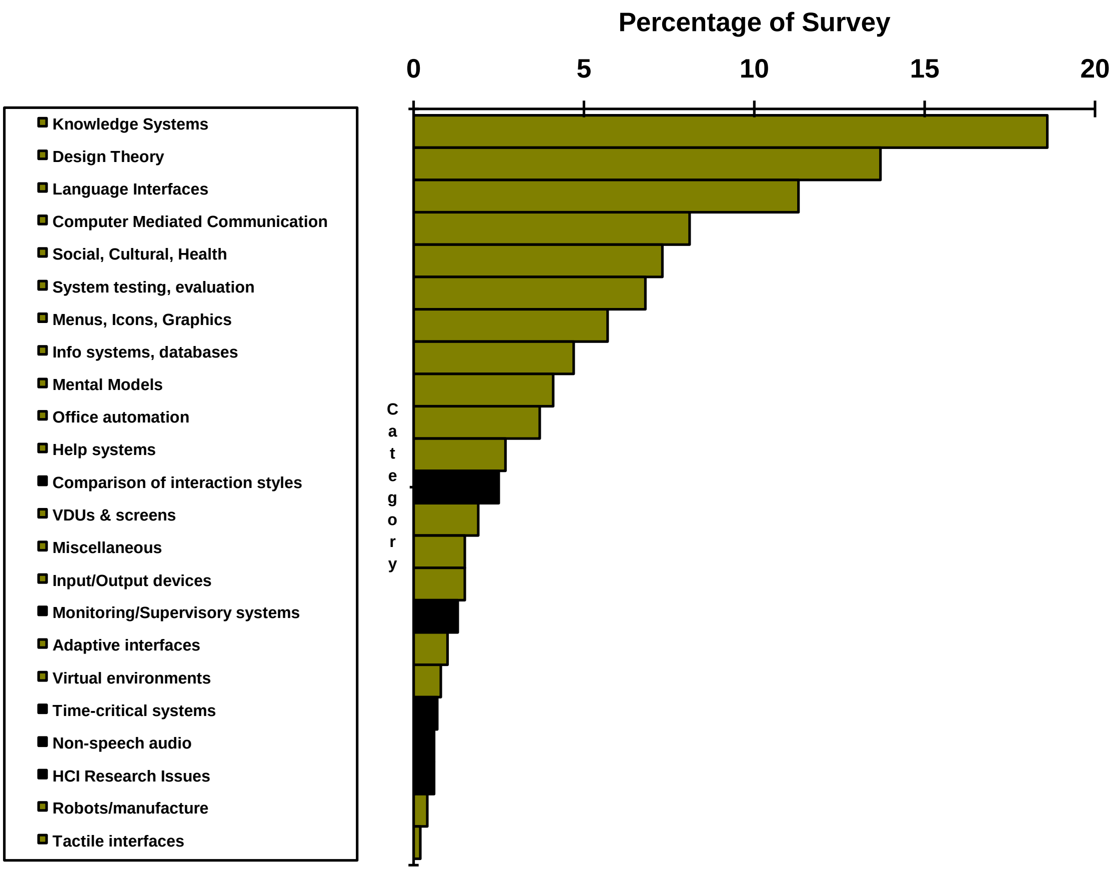

Figure 2.7: Percentage breakdown of HCI categories from survey

2.5.2 Consideration of Relevant Areas

The survey has highlighted 23 different categories in the HCI literature, of which the following five were considered as potentially relevant to this thesis:

- Comparison of Interaction Styles
- Monitoring & Supervisory Systems
- Time-Critical Systems
- Non-speech audio
- HCI Research Issues

Each of these categories is now expanded and the key papers summarised. The relative proportion of papers in the survey is given in brackets after the title.

2.5.2.1 Comparison of Interaction Styles (2.5%)

A popular scenario for testing involves users trying to achieve a problem-solving task by both ‘command/language’ and ‘Direct Manipulation’ interfaces, then comparing the results, e.g. [Jorg, 1993], [Benbasat, 1993] & [Bekker, 1995]. In these tests Direct Manipulation tends to give better results. However, Karl & Shneiderman show that when speech is used as an *extra* input channel (e.g. to directly activate menus, rather than by using the mouse) then the overall control bandwidth is increased [Karl, 1993].

Cohen [1993] advocates the use of audio cues in windowing systems to help users navigate complex data areas (also see section 2.5.2.4; Non-speech audio).

Gestural interpretation is discussed in *Parallel Use of Hand Gestures* [Bordegoni, 1994]. The different ways in which two hands can operate are identified. Parallel gestures can be *synergistic* (the meanings of each gesture is combined into a new meaning) or *concurrent* (two individual gestures happening at the same time).

Sellen & Buxton [1992] describe how a foot-pedal is used as an extra input for changing the ‘mode’ of text input from ‘typing’ to ‘editing’ without having to take the hands from the QWERTY keyboard.

Reflection and Goal Management [Trudel, 1995] discusses the ways in which users learn how to use button-type devices. They compare the following of instructions with what they call ‘exploratory learning’. This is taken to mean that users identify their own *tasks* and order them sequentially to meet their *goals*.

Schar [1996] shows that command-based interfaces tend to trigger an 'explicit' learning mode which is rational, selective and involves focusing the conscious attention. Direct manipulation interfaces, on the other hand, tend to induce 'implicit' learning, where trial-and-error leads to the user understanding the relationship between variables. These issues are discussed more in Chapter 4 of this thesis.

2.5.2.2 Monitoring and Supervisory systems (1.3%)

Although the HCI issues in controlling complex machinery would seem to be a highly relevant area of study for this project, it transpires that most of the papers discuss knowledge-based support systems, which help the human operator in a complex environment by providing information. An example of this is the *Human Operator Support System* [Sussen, 1994] which aims to help human operators to diagnose faults in complex systems. It is pointed out that the stress of real situations could make this sort of system difficult to use, because stressful humans do not absorb new knowledge well!

A few papers discuss this issue further:

In *Trust, Self-Confidence & Automation* [Lee, 1994] there is a discussion regarding the way that humans in an automated pasteurisation plant tended to choose either ‘fully manual control’ *or* ‘fully automatic’. Operators fail to successfully switch back to manual control once they rely on the automation.

> *"Designers should make special provision to allow users to operate under manual control - particularly in the early stages".*
>
> [Lee, 1994]

This theme is picked up in *Monitoring Behaviour & Automation* [Lin, 1993] where the term ‘out of the loop unfamiliarity’ is coined to describe what happens when humans become passive monitors. The system described is actually a supermarket computer system, but the idea is worth considering for all dynamically interactive control systems. Again Kontogiannis [1996] states, in relation to the control of nuclear power plants

> *"Training [people] to cope with emergencies becomes very important, especially with the increasing levels of automation which leaves little scope for practising these types of skill".*
>
> [Kontogiannis, 1996]

Singh [1997] warns of the dangers of user 'complacency' in aviation systems. He notes that over 500 incidents of "crew over-reliance on automated systems" are highlighted in NASA's aviation safety reports.

2.5.2.3 Time-Critical Systems (0.7%)

A good example of a time-critical system is an interactive computer game. Given the number of such systems present in the world it is astonishing that only two papers in this survey even refer to computer games!

Ben Shneiderman's original definition of Direct Manipulation was inspired by watching people play games with active and continuous involvement. The 'fun' side of such user engagement is discussed in *Usefulness, Fun and PCs* [Igbaria, 1994]. They deduce that while the purchase of technology by businesses is primarily driven by perceived 'usefulness', the acceptance of the technology by people actually working with it is often driven by perceived 'fun'.

Scown appeals to the HCI world to look beyond office-based tasks (see 2.5.3.10). In *HCI and Multi-Agent Real-Time Systems* [Scown, 1992] he suggests that flight and plant control are examples of real-time systems which are currently being overlooked. His argument is that since most HCI tests are carried out on single-user office-based tasks such as word-processors there is a wealth of parameters which are *not* being measured. These are identified as:

- multi-person interaction in a complex system,
- real-time issues (particularly those of a 'control' nature),
- continuous changes in parameters,
- lack of state replicability (i.e. complex real-time systems cannot simply return to the last 'well-known state', instead the system needs 'steering' through a complex parameter space).
- the cost of making a mistake (not all control systems have an 'undo' button!).

Other time-critical systems include air-traffic control. This is a highly complex, safety-critical multi-person job and several papers looks at specific tasks, e.g. *The Role of Flight Progress Strips in Air Traffic Control* [Edwards, 1995].

2.5.2.4 Non-speech audio (0.6%)

Several authors consider the use of 'earcons' (non-speech sounds) as an effective way of giving interactive feedback to the user. Brewster [1995] shows that humans are able to process different sounds in parallel, while Jacko [1996] studies the age that children begin to respond to auditory cues in software.

Rauterberg [1998] explains how sonic alarms are designed to be distinct. In doing so he explains the essential difference between audio and vision.

> *"The eye is a directed sense and focuses attention, the ear is an all-round sense . . and guides the visual attention . . Humans cannot close their ears in the same way that they close their eyes . . therefore auditory devices are generally preferred to visual signals as warning indicators".*
>
> [Rauterberg 1998]

Vertegaal [1996] describes an experiment to compare mouse, joystick and PowerGlove for controlling a sound's timbre. He states that:

> *"though it is clear that a direct correlation between gesture and sound reduces cognitive processing load and enhances performance . . . this impairs the use of the system as a generic sound synthesis control".*
>
> [Vertegaal 1996]

Chapter 3 gives more details about the human-computer interface in music systems and Vertegaal's work is covered there too. Vertegaal also describes that:

> *"certain parameters of a task are perceived as being integrally related to one another . . . while others are separably related . . . Consequently users manipulate certain parameters simultaneously (such as the x and y position of a graphical object).*
>
> [ibid]

He concludes that the mouse and joystick give better performance at the timbre exercise than the glove, but that most of his subjects were familiar with the mouse. He also notes, regarding the mouse and the joystick, that he would not *"expect the musician to use these as a musical instrument".*

2.5.2.5 HCI Research Issues (0.6%)

Not surprisingly perhaps, the topics covered in this category tend to be those which are *already* receiving the greatest coverage in the literature. Particular focus appears to be given to CSCW and CMC which are summarised in category 2.5.3.4; 'Computer Mediated Communication'.

The following authors talk about wider issues:

Keeler and Denning [1991] challenge the accepted notion of HCI communication theory. They emphasise the importance of what they call ‘the user’s engagement with objects’ i.e. the task is so engaging that the interface becomes ‘transparent’. They contrast this feeling of ‘involvement with objects’ with the conventional notion that the human communicates with an intermediary system (the computer) which then goes and does the work, thus giving the user a ‘3rd-party’ experience.

It is interesting to note that DeGreene [1991] uses the terms ‘analystic’ and ‘holistic’ (see Chapter 4) to describe respectively the formal approach of mainstream HCI design and the more dynamic ‘ergonomics’ approach. He implies that much of HCI design is too reductionist and analytical and that much more needs to be done to include the users (who are the real experts) as they are involved in intimate day-to-day operation.

Jacob et al give a good HCI research overview in *Interaction Styles and I/O devices* [Jacob, 1993]. They identify a series of research directions which include developing new input devices and pursuing further study of different interaction styles. Their paper suggests the following categories requiring further work:

- The ‘next’ interaction style. This is not fully defined but is based on moving away from the ‘command-issuing’ paradigm and concentrating more on ‘sensing’ the user’s body. This will result in less verbal information transfer and more real-time control.
- Interface Devices: 3-D pointing and manipulation, gesture abstraction, simultaneous two-handed input, stereo image displays, virtual input devices, speech, eye-input technology and directional audio.
- Ergonomic considerations: adaptations for disabled users, computer sensing of operator position in the room, better graphics resolution over a much larger area, force and tactile feedback, environmental controls, etc.
- Methods for data manipulation and visualisation: non-speech audio for data ‘display’, touchable 3-D displays, techniques for representing high-dimensional data, and even smell as a way of portraying data!

The authors of *Approaches to Interface Design* [Wallace, 1993] focus on ‘ease of use’ and refer again to Ted Nelson’s often cited comment:

“Any system which cannot be well taught to a layman in ten minutes, by a tutor in the presence of a responding set-up, is too complicated”.

[Nelson, 1987]

However, musical instruments have existed for thousands of years, and it is a rare experience to find one that takes less than ten minutes to teach!

2.5.3 Other categories of HCI literature

The categories of literature that were not deemed as directly relevant to this thesis are now described in more detail for completeness. Those readers who wish to follow the flow of argument in this thesis should proceed directly to section 2.5.4 where the literature survey is summarised. What follows here is an explanation of each category of paper, along with some example titles and a summary of their content. Where a paper or topic appears (at first glance) to be relevant, it is given more space in order to explain why it is not so significant to this thesis.

Each of the categories is now described, in decreasing order of percentage coverage. The actual percentage is shown in brackets after the title.

2.5.3.1 Knowledge-Based Systems & Theory (18.6%)

Consisting of nearly one fifth of the HCI literature since 1990, this area covers all aspects of the gathering, storage, processing and retrieval of knowledge. Subtopics include examples of ‘Knowledge-Based Systems ‘(KBS), ‘Decision Support Systems’, ‘Knowledge Theory’, ‘Expert Systems’ and ‘Artificial Intelligence’ (AI). Typical paper titles are *Measuring the Value of Knowledge* [Reich, 1995] and *Fuzzy Sets Based Knowledge Systems* [Santamarina, 1991].

A few papers concern the use of expert systems in safety-critical situations. Lin et al [1998] acknowledge that;

"The management of emergency situations often requires human operators to make prompt and accurate decisions under stringent time conditions."

[Lin, 1998]

They explain that people opt for the most easily accessible menu options (i.e. they do not take the time to search for the most appropriate option). However, rather than considering whether menus are the most appropriate interface, their focus is on making an expert system that comes up with a better set of options.

2.5.3.2 Design Theory & Software Engineering (13.7%)

One of the main focuses of HCI is in developing theories and applications which help designers of computer systems. The three areas covered by this category are that of general design theory, software design (software engineering) and programming. Typical papers in this section include *Cognitive Activities in Design* [Lee, 1994] and *Programming Strategy* [Davies, 1993].

Very few papers address the needs of interactive control system design. Johnson & Harrison [Johnson, 1992] argue for a theory of temporal logic to enable scientific proof of real-time system design. Colgan et al [1995] describe a method of allowing the human designer to visually interact with an auto-design program in a similar manner in which a pilot monitors an autopilot.

Faconti [1996] presents a framework for analysing inputs that come from more than one device.

> *"User interfaces of many application systems have begun to include multiple devices which can be used together to input single expressions. Such interfaces . . . are commonly labelled* ***multimodal*** *because they use different types of communication channels to acquire information".*
>
> [Faconti 1996]

The type of multimodal input featured is gesture-enhanced speech such as the statement "Put that there" where the user points to 'that' and 'there'. This is not directly related to this thesis, but it is interesting to note that designers are beginning to think in terms of multimodal inputs.

2.5.3.3 Language Interfaces: Text, Speech, Hypertext/media (11.3%)

This popular category covers the interaction between humans and computers using language - either typed, clicked on (with a mouse) or spoken. Papers range from *Automatic Speech Recognition in Practice* [Jones, 1992] and *Errors in Natural Language Dialogue* [Veronis, 1991] to the subtleties of text editing, for example

Blinking Cursor [Coll, 1993] (which compares word-processors with and without blinking cursors).

The increasing use of the World Wide Web has dramatically increased the studies into how hypertext can be designed and enhanced. Example papers in this area are *Spatial Metaphors and Disorientation in Hypertext Browsing* [Kim, 1995] which considers the user of spatial cues to help people remain oriented when browsing. Shneiderman [1997] discusses how we cope with this new flood of information that comes with the World Wide Web.

2.5.3.4 Computer Mediated Communication (8.1%)

The topics of 'Computer Mediated Communication' (CMC) and 'Computer Supported Co-operative Work' (CSCW) are gaining increasing importance now that people are working in distributed locations, yet still requiring human-human collaboration and conversation. Generally CMC regards the computer as a communication device e.g. *Text-Based On-Line Conferencing* [McCarthy, 1993] and *Live Audio-visual Communication Systems* [Colbert, 1995].

CSCW implies that several people are working in a computer-supported environment. Examples of this are *Managing Design Ideas with a Shared Tool* [Lu, 1993] and *Interpersonal Communication and HCI* [Greatbatch, 1993] which complains of the limitations of "single-person, goal-oriented designs" that do not work in busy multi-person environments such as hospitals. Wambach [1995] studies the effect of email on an organisation's structure.

2.5.3.5 Social, Cultural and Health implications of computers (7.3%)

This increasingly popular topic relates to how computers are being accepted (or rejected) in society and what effects they are having on human health. Topics range from *Information Society and IT* [Kobayashi, 1991] to *VDT and Excretion* [Tanaka, 1992] which analyses the chemical content of users according to how much time they spend in front of a computer monitor! Emphasis is given to anxiety and stress in the computer-oriented workplace e.g. *Stress, Control and Computer-Based Work* [Wastell, 1996] and [Henderson, 1995].

2.5.3.6 System Testing & Evaluation (6.8%)

This topic covers all aspects of how software is tested and evaluated, with particular regard to the Human-Computer Interface, the environment in which the testing is to be done and various theories for analysing the results. Example papers include *Evaluating User-Computer Interaction* [Sweeney, 1993], and *Effects of Running Fewer Subjects* [Cordes, 1993].

Lin [1997] notes that *"well-designed computer software should be easy to learn"*, whether by rote learning or understanding or explanation. Macleod's *The MUSIC performance measurement method* [1997] sounds very promising for this study, but actually describes a system for comparing banking tasks, MUSIC being an acronym for Measuring the Usability of Systems In Context!

2.5.3.7 Menus/Icons/Graphics (5.7%)

The graphical interface is still the main method for the display and editing of information. Much discussion takes place on the different types of graphics e.g. *Classifying Graphical Information* [Lohse, 1991] and how best to structure menus e.g. *Context in Hierarchical Menus* [Field, 1990] and *Pull-Down v Traditional Menu Types* [Cancy, 1996].

2.5.3.8 Information Systems & Databases (4.7%)

This category concerns ‘information points’ i.e. systems whose role in life is to allow people to have access to information. It would not be too far-fetched to include this topic as a subsection of category 2.5.3.1 (Knowledge-Based Systems) which would then constitute nearly a quarter of the entire HCI literature. A typical paper in this category is *Deriving Requirements for a Hospital Information System* [Symon, 1992]. To distinguish between categories 2.5.3.1 and 2.5.3.8, this paper would have been classed in category 1 if it actually helped doctors to make medical decisions. As it stands, it simply provides an information point for patients and administrators.

Databases are a special, more flexible, form of information point designed to hold and manipulate data at the user's requirements. *Retrieval of Images from Image Databases* [Whalen, 1995] explains how to look for trademarks which distinguish images from each other.

2.5.3.9 Mental Models (4.1%)

This topic covers those papers which discuss the psychology of what goes on inside the heads of humans. Much of this research is used to support the design of computer systems by 'modelling' the user's thoughts and actions at the design stage. Papers range from *A Descriptive Study of Mental Models* [Payne, 1991] to *Understanding Calendar Use* [Payne, 1993] which considers how humans represent time in their lives.

2.5.3.10 Office Automation & Business Applications (3.7%)

‘Office Automation’ (OA) is an accepted sub-branch of HCI, presumably because there is so much commercial pressure to improve and sell word-processors, accounting packages, spreadsheets and computer filing systems. Papers cover topics such as *Managerial Competence and New Technology* [Burnes, 1991] and *Learning to Use a Spreadsheet* [Kerr, 1994].

2.5.3.11 Instruction & Help Systems (2.7%)

Systems or software which offer the user information or instruction are included here. Studies of ‘Help’ systems such as *Advice Giving and Following* [Hill, 1993] are included alongside computer-aided instruction features such as *Animated Demonstrations* [Palmiter, 1993]. Software help agents are discussed, such as Smalltalk Gurus which help with programming problems [Alpert, 1995].

2.5.3.12 VDUs and screens (1.9%)

Many papers are still published on the physical nature of visual displays, for example *VRT Luminance* [Saito, 1991]. Picking [1997] discusses the advantages and disadvantages of reading a musical score on a computer screen instead of paper.

2.5.3.13 Input / Output Devices (1.5%)

This topic has a relatively small coverage in the HCI literature. At first glance it might appear to be a relevant area for this thesis. However, on closer inspection the papers are concerned with the physical design of the interfaces, rather than the issues of what parameters they control. In this category comparisons are made between different physical input/output devices and new ones are proposed. Example papers are *Typing Speed Using a Stylus* [Sonkoreff, 1995] and *A Vision-Based 3D Mouse* [Nesi, 1996] which uses a camera to track hand-position in three dimensions.

2.5.3.14 Miscellaneous (1.5%)

This category holds those references which do not talk about systems or humans or the interaction between them. Topics include the dissemination of information about HCI and curriculum development for HCI courses e.g. [Preece, 1991]. Other papers include *It's Not Really Theft* [Sealce, 1998] - a discussion about the reasons for software piracy!

2.5.3.15 Adaptive Interfaces (1.0%)

This area might sound relevant to this thesis, but it deals with interfaces that attempt to change in accordance with the user's current skill level. This is referred to (in Chapter 3) as 'moving the goalposts'. There are areas where adaptive interfaces are required, but the control of real-time systems should perhaps involve *stable* interfaces that do not change. Musical instruments and vehicles are examples of systems where the *human operator* is the adaptive part of the system, whereas the object under control stays constant.

The topic of Adaptive Interfaces considers how the user interface should change as users progress from novice to expert. The emphasis is on changing the interface as and when the user progresses (e.g. *Matching Interface to User Skill* [Trumbly, 1993]). Some consideration is given to the definitions of the terms 'novice' and 'expert', for example *Defining the Novice User* [Fisher, 1992]. Gong [1995] describes an interface which changes its menu options according to how the user scores.

2.5.3.16 Virtual Environments (0.8%)

This literature has appeared since 1995 and details some of the many aspects of Virtual Reality. Witmer [1996] describes buildings which are 'mocked up' in virtual form so that personnel can find their way around even though the building is not yet in existence. Paulos [1997] explains how we are hampered by the lack of a body in cyberspace, and suggests how we might use robots to re-establish our physical presence in a remote location!

Burden [1996] reviews the contemporary technology for VR and suggests that the feeling of immersion will increase if tactile and force feedback systems are used. He also emphasises the need for 3D audio. He concentrates on the number of different sensors that can be used, rather than the style of interaction.

2.5.3.17 Robots & Automated Manufacturing (0.4%)

This category concerns human interaction with mechanised processes (e.g. robots and computer integrated manufacture). This topic receives much greater coverage in non-HCI journals. An example paper is *Actions Representation in 4D Space* [Adorni, 1991] which discusses issues of moving parts and how to automatically control them over time.

2.5.3.18 Tactile Interfaces (0.2%)

This final category concerns systems that allow humans a sense of touch or force-feedback in the interface. It is not directly concerned with the issues of interface parameters and dynamic control, but it may play a part in future work which arises from this thesis.

Akamatsu [1996] considers the use of a mouse with and without force feedback. He records how 'error rate' is affected when such feedback is used in 'targeting' tasks. Gobel [1995] outlines a simliar task and concludes that standard 'tracking' tasks are worsened by force-feedback, but that 'ballistic' movements are improved.

2.5.4 Summary of HCI Literature

This survey shows that the predominant categories that are discussed under the umbrella term HCI are:

- Knowledge Based Systems and Data access.
- Design Theory & Software Engineering.
- Language-based interfaces & Hypermedia.
- Computer-Mediated Communication.

- Social, Cultural and Health implications of computers.
- User interface testing and modelling.
- Menus, icons & graphics.

Nearly three quarters of the literature concentrates on these topics. In contrast, less than 6% of the literature addresses topic areas which might be considered relevant to this study. On closer inspection, less than 1% of the HCI literature is calling for the community to re-think its priorities in terms of humans controlling complex real-time systems with a greater degree of engagement. This is not to say that the majority of HCI research is misdirected, but rather to point out that there is most definitely scope for further study in this area.

2.6 Suggested areas of change

Computer interfaces have been developed over the last three decades, but it is important to remind ourselves of just how recent a subject area this is. Even in a static subject, we would not expect to have discovered everything in such a time span. However, the technological advances towards the end of the twentieth century have kept the topic area moving and changing at an incredible pace. Some of the basic tenets of HCI will probably hold true throughout this change, whereas others will become obsolete, or at least less relevant. It is the purpose of this section to highlight those areas which may need to change or evolve.

2.6.1 Dynamic Systems

The phrase 'Dynamic Systems' is used in this context to describe those systems where a user needs to be in constant control of a complex device. A car is a good example, as it cannot drive itself for very long because it requires constant multiparametric input from its driver in order to make it move and to keep it going along the correct path. Musical instruments are also good examples of dynamic systems, hence their use as exemplars throughout this thesis. An acoustic instrument, such as a clarinet, does not make sound unless a human physically activates it. Even then, it needs constant control and fine balancing to keep the note sounding. The player is continuously and subconsciously listening to the sound produced and taking appropriate subtle or gross action to maintain the required note. In addition the player experiences kinaesthetic feedback at all times, and takes corrective action based on this.

Other dynamic systems include:

- Industrial control plants, which may dictate the smooth running, or otherwise, of chemical, nuclear or other industrial installations.
- Aircraft, which are increasingly being flown by computers.
- Air traffic control systems.
- Countless 'embedded' computer systems in household and industrial devices.

Computers are increasingly prevalent in dynamic systems due to the increase in computing power coupled with a desire to link devices into a worldwide network and to minimise human effort (for financial reasons and because the systems are increasingly complex). This makes it all the more important to examine the role of human-computer interfaces in this area.

When computers are used to control dynamic systems, the human operator is often relegated to the role of a passive observer. Leonard Lee, in *The Day the Phones Stopped* describes a catalogue of worrying stories which seem to confirm the belief that computers should not be in charge.

He cites one pilot's comment on the new technology in a plane which only just survived a 30,000 feet plummet towards the Pacific Ocean :

> *"The crew's role is reduced to one of monitoring the performance of boringly reliable systems. Humans make bad monitors, however. To perform well they need to be giving commands or control inputs and then feeling the result"*

[Lee, 1992].

In *The 24 Hour Society* Martin Moore-Ede describes a similar array of disasters and near-misses (including the nuclear meltdown scenario at Three Mile Island) which have subsequently been proved to be human-machine communication problems. A recurring theme is that of human operators suddenly struggling to regain control of a system which can no longer be handled automatically by the software. Moore-Ede's emphasis is on the unrealistic demands designers place on operators' attentiveness. Operators are given the impression that everything is under control until the emergency situation when they are expected to come `back on line' instantly and know exactly what to do.

> *"A paradigm shift in thinking . . from machine-centred priorities toward human-centred priorities, is needed. Rather than abandoning the advantages of technology, this instead means making sure that the technology enhances rather than degrades human performance."*
>
> [Moore-Ede, 1993].

If people are to be in charge, then the interface must be designed to enable this to happen. As has already been noted, it is too easy for designers to opt for the simplest human-machine interface. Not only are complex, responsive interaction tools difficult to design and program, but there seems to be pressure from the HCI community to produce 'easy-to-use' interfaces.

2.6.2 Ease of Use

> *"Usability, a key concept in HCI, is concerned with making systems easy to learn and easy to use. Poorly designed computer systems can be extremely annoying to users"*
>
> [Preece, 1994]

It is difficult to criticise the above comment. Most people know what it is like to be annoyed by a poorly designed computer interface. Most people would expect an ideal computer system to be easy to use and easy to learn. However, nobody expects learning to drive a car to be easy. Equally no musician would ever say that a violin or a clarinet was easy to use. Yet the subtlety of control and real-time interaction which can be shown by car drivers and instrumental performers is often astounding.

By making the assumption that interfaces should be 'easy' we are in danger of undervaluing the adaptation capabilities of human operators and thus limiting the potential human-computer interaction to a lowest common denominator of

'easy to use' commands. We may not be utilising all of the interactive capabilities of human operators.

Another statement which at first seems to be common sense, but on closer examination becomes rather more questionable, is:

> *"People should NOT have to change radically to fit in with the system, the system should be designed to match their requirements"*
>
> [Preece 1994].

People are actually very good at adapting to new situations and at learning by practice and repetition. Users of traditional dynamic systems, such as musical instruments, put themselves through hours (and years!) of practice in order to achieve control intimacy with their instrument.

It was mentioned earlier that a clarinet player must constantly control many aspects of the instrument simultaneously and creatively in order to play a note. This is a difficult task which takes a great many hours of practice. Interestingly when an electronic device is made where the playing interface is made easy, objections are immediately raised by the listeners who complain that the sound is 'flat', 'boring', 'lifeless' and 'nothing like the real thing'.

A musician who has to work hard to produce a good sound is likely to be rewarded with appreciation from an audience. In contrast 'easy-play' instruments tend to produce 'cheap-and-easy' musical results.

2.6.3 Information and Menus

Computers have traditionally been used to handle and process information (or data). There is an implication here which needs to be challenged; that the purpose of a computer interface is the management of information transfer.

A car driver isn't usually thought of as 'giving information to the car', or a violinist of transferring data to the violin, but rather, in both cases, of directly controlling the dynamic system in question. We should perhaps be careful, when we are designing computer systems for the control of dynamic systems, to use the model of direct control, rather than that of information gathering. It is a subtle shift of thinking, but without it the computer interface is likely to default to a WIMP interface with point-and-click menu options, without questioning if this is the correct paradigm for this situation.

It has been shown that users of menuing systems do not usually learn the menus, but instead rely on continuous traversal and cognitive interpretation.

> *"Menu selections are made by semantic matching"*
>
> [Monk 1995]

This means that users are constantly interpreting menu systems; keeping them at the level of 'controlled' processes (section 2.3.7) and not being able to promote these to 'automatic' processes. In fact automatic processes are almost frowned upon since . . .

> *"in times of stress people revert to the automatic processes which were previously learnt"*
>
> [Preece, 1994].

Preece is describing here how users who have learnt commands are at a disadvantage when learning a new system, since they easily revert to the commands that they have learnt. Perhaps this is more of an indictment of a situation which expects users to keep having to learn new systems, but it is used here as an argument against allowing the user to develop automatic processes.

There is a lesson to be learned here. The catalogues of disasters show that people struggle with user-interface systems in times of stress and often make serious errors of judgement. These user interfaces often rely on menuing systems. The operation of menuing systems relies on the use of controlled processes. The definition of a controlled process is one that is not learned, but requires cognitive interpretation, whereas automatic processes are those most likely to be reverted to when under stress. And yet the conclusion arrived at is that automatic processes should not be encouraged! Perhaps an alternative conclusion is that people should not be expected to navigate through menus in times of stress, but instead utilise rich gestural interfaces that are controlled by automatic motor functions, supported by practice and experience of use.

2.7 Summary

The increasing emphasis on information transfer and display in today's climate of world communication is likely to compound this problem by diverting HCI research towards this area. The aim of this thesis is to redress the balance in favour of highly interactive human control of complex dynamic systems. In order to gain insight into this area, Chapter 3 now focuses on the use of computers in the performance of music; a situation which demands extremely flexible and controllable dynamic audio interfaces, usually referred to as musical instruments.