

✓

FOR REFERENCE ONLY

**MULTI-DIMENSIONAL INFORMATION  
REPRESENTATION - ENABLING THE VISUALLY  
IMPAIRED TO ACCESS GRAPHICAL USER  
INTERFACES AND DOCUMENTS**

**PHILIP DERWAS**

**17 JUL 2001**

**A thesis submitted in partial fulfilment of the requirements of The Nottingham  
Trent University for the degree of Doctor of Philosophy**

**January 2001**

40 0715251 7



ProQuest Number: 10183134

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest 10183134

Published by ProQuest LLC (2017). Copyright of the Dissertation is held by the Author.

All rights reserved.

This work is protected against unauthorized copying under Title 17, United States Code  
Microform Edition © ProQuest LLC.

ProQuest LLC.  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 – 1346

# Abstract

As graphical user interfaces (GUIs) replace text-based interfaces, the computer becomes increasingly accessible to sighted people. But for the visually impaired, this trend causes significant problems. For both types of interface, a screen reading system is required to capture screen information and present it non-visually. For graphical user interfaces, these functions are significantly more difficult than for their text based predecessors. This work addresses the problems and evaluates solutions.

A graphical information access system has been constructed using a continuous process of development and evaluation. The basis for this system is an Off Screen Model maintained using OCR and an Icon Matching recognition engine. On top of this, an interface has been created, enabling the user to access display information via a set of experimental interface features. These features have been designed and developed with continuous input from visually impaired computer users. Qualitative evaluation of their effectiveness has been undertaken throughout the course of the project, and results fed back into the development process.

Input and output techniques have been investigated, including use of the mouse as an additional input device, and non-speech sounds to supplement spoken output. The presentation of spatial information, to provide context as well as content, has been a fundamental aspect of the work. User response to this approach and the implemented methods is described. This work draws from the fields of human factors and psychology in the construction of a usable interface, and from research into non-visual display techniques and sound technology in the addition of access features. Comparisons with both research-based and commercial screen reading systems are drawn, and the benefits of the approach taken in representing graphical information described. Furthermore, the concepts and methods introduced have been applied to alternative sources of multi-dimensional information, notably, graphical documents and data tables.

User response suggests that presenting spatial information is a valid approach to take and that further work would be worthwhile. Well chosen sounds to convey information can be faster to impart and less distracting than speech in some situations. Maintaining a balance between speech and sound use is essential in providing an efficient interface, therefore guidelines for deciding how to represent a particular piece of information are developed throughout the work. The work has been evaluated using a number of features embodying the main themes of the project, thereby suggesting possible paths for future work to take. Results suggest that the work undertaken can provide the visually impaired user with a means of accessing multi-dimensional information.

# Contents

<b>ABSTRACT.....</b>	<b>2</b>
<b>CHAPTER 1 INTRODUCTION .....</b>	<b>6</b>
1.1 RESEARCH AIMS.....	6
1.2 MULTI-DIMENSIONAL INFORMATION AND THE VISUALLY IMPAIRED .....	7
1.3 PROJECT ORIGINS.....	8
1.4 TOOLS AND TECHNIQUES .....	9
1.4.1 Input and output devices .....	9
1.4.2 Screen capture and maintenance .....	10
1.5 PROJECT DESIGN .....	11
1.5.1 Approaches to information presentation.....	12
1.5.2 Interface structure.....	14
1.5.3 User-centred design.....	15
1.5.4 Design and evaluation methods .....	16
1.6 PROJECT EVALUATION .....	17
1.6.1 Aims of evaluation.....	17
1.6.2 Approaches to evaluation.....	17
1.6.3 Users.....	19
1.6.4 Internal evaluation.....	20
<b>CHAPTER 2 LITERATURE SURVEY .....</b>	<b>21</b>
2.1 RESEARCH FOR THE VISUALLY IMPAIRED.....	21
2.2 ACCESS SYSTEMS.....	23
2.2.1 Access to text-based computers.....	23
2.2.2 Access to graphical user interfaces.....	24
2.2.3 Non-visual access to documents.....	36
2.3 NON-VISUAL DISPLAY TECHNIQUES .....	39
2.3.1 Input techniques.....	39
2.3.2 Auditory output techniques .....	43
2.3.3 Alternative output techniques.....	52
2.3.4 Multi-modal interfaces.....	56
<b>CHAPTER 3 TALKING WINDOWS.....</b>	<b>57</b>
3.1 SYSTEM ARCHITECTURE .....	57
3.1.1 The Off Screen Model .....	57
3.1.2 Accessing features.....	58
3.2 BASIC FEATURES OF TALKING WINDOWS .....	59
3.2.1 Basic operation .....	59
3.2.2 Message format.....	60
3.2.3 Screen navigation.....	61
3.2.4 Using lists for selection.....	65
3.2.5 Maintaining the windows metaphor .....	67
3.2.6 Providing access to text .....	68
3.2.7 Status information.....	70
<b>CHAPTER 4 SPATIAL INFORMATION PRESENTATION USING SOUND MAPS .....</b>	<b>71</b>
4.1 SOUND MAP CONCEPT .....	71
4.1.1 Implementation .....	71
4.1.2 Types of sound map.....	71
4.1.3 Controlling sound maps.....	71
4.2 SOUND MAPS FOR DETERMINING POSITION.....	72
4.2.1 Edges sound map .....	72
4.2.2 Volume sound map.....	73
4.2.3 Piano sound map.....	74
4.2.4 Diagonal sound map.....	77
4.3 SOUND MAPS FOR LOCATING A POINT .....	78
4.3.1 Target finding sound map .....	79



4.3.2 Cross hairs sound map.....	80
4.3.3 Icon sound map.....	81
4.4 SOUND MAPS FOR SUPPORTING CURSOR MOVEMENT.....	82
4.4.1 Directional sound map.....	82
4.4.2 Grid sound map.....	82
4.5 COMBINING SOUND MAPS .....	83
4.5.1 Basic layout sound map .....	85
4.5.2 Piano-based layout sound map .....	86
4.5.3 Enhanced piano-based layout sound map.....	86
4.5.4 Comprehensive layout sound map .....	87
<b>CHAPTER 5 DIRECT PRESENTATION METHODS .....</b>	<b>90</b>
5.1 DESCRIPTION TECHNIQUES .....	90
5.1.1 Window state description .....	91
5.1.2 Desktop description.....	91
5.1.3 List windows.....	91
5.1.4 Individual windows .....	91
5.1.5 Cursor description .....	92
5.1.6 Complete screen description .....	92
5.1.7 Separation of two points .....	93
5.2 ATTRIBUTE PRESENTATION .....	94
5.2.1 Position of a point.....	94
5.2.2 Position of a region.....	96
5.2.3 Size of a region.....	98
<b>CHAPTER 6 DOCUMENT ACCESS.....</b>	<b>100</b>
6.1 INTRODUCTION TO THE DOCUMENT ACCESS SYSTEM.....	100
6.1.1 Aims of a document access system .....	100
6.1.2 System architecture .....	101
6.1.3 Approaches to providing features .....	104
6.2 SPEECH-BASED FEATURES OF THE DOCUMENT ACCESS SYSTEM .....	106
6.2.1 Giving an overview of document content .....	106
6.2.2 Working with regions.....	107
6.2.3 Presenting document attributes.....	108
6.3 SOUND-BASED FEATURES OF THE DOCUMENT ACCESS SYSTEM .....	108
6.3.1 Layout sound maps for documents.....	109
6.3.2 Octave sound map.....	114
6.4 SUPPORTING DRAG AND DROP.....	115
6.5 PRESENTATION OF GRAPHICS .....	117
6.5.1 Data tables.....	117
6.5.2 Supporting table presentation features .....	122
<b>CHAPTER 7 RESEARCH METHODOLOGY .....</b>	<b>124</b>
7.1 REVIEW OF TECHNOLOGY .....	125
7.2 CONSULTATION WITH VISUALLY IMPAIRED USERS .....	125
7.3 DESIGN PROTOTYPES .....	125
7.4 EVALUATION STUDIES .....	125
7.5 ANALYSIS OF STUDIES .....	125
7.6 REFINEMENT .....	126
7.7 FINAL VERSION .....	126
<b>CHAPTER 8 EVALUATION STUDY ONE .....</b>	<b>127</b>
8.1 METHOD .....	127
8.2 RESULTS .....	128
8.2.1 Results by user .....	128
8.2.2 Results by subject area.....	137
8.3 REVIEW .....	139
<b>CHAPTER 9 EVALUATION STUDY TWO .....</b>	<b>142</b>
9.1 METHOD .....	142
9.1.1 Selecting features for evaluation.....	142
9.1.2 Features selected for evaluation .....	142
9.1.3 Testing procedures.....	143

9.2 RESULTS .....	145
9.2.1 Scores.....	145
9.2.2 Reaction .....	145
9.3 REVIEW .....	149
9.3.1 Review of features .....	150
9.3.2 Review of scores given .....	154
<b>CHAPTER 10 EVALUATION STUDY THREE.....</b>	<b>156</b>
10.1 METHOD .....	156
10.1.1 Aims .....	156
10.1.2 Procedure.....	156
10.2 RESULTS .....	159
10.3 REVIEW .....	164
<b>CHAPTER 11 DISCUSSION.....</b>	<b>170</b>
<b>CHAPTER 12 FUTURE WORK.....</b>	<b>173</b>
<b>CHAPTER 13 CONCLUSION .....</b>	<b>175</b>
<b>CHAPTER 14 REFERENCES .....</b>	<b>178</b>

# Chapter 1 Introduction

Imagine using a computer with the monitor switched off.

You want to send an email, or print a letter, or even write a thesis. How can you do this when you can't see the screen?

Now consider career opportunities available to a visually impaired person.

There are an increasing number of careers in which computing is central. A considerable advantage of the computer is its ability to be used without mobility being a major issue.

This research involves investigating the ways in which visually impaired people can access computers and other sources of information.

This chapter describes that research; its aims, origins and design, and how the involvement of visually impaired users is fundamental to the project.

## 1.1 Research aims

The purpose of the research is to enable and improve access to multi-dimensional information by the visually impaired. To achieve this, the following project aims have been established.

- Identify sources of multi-dimensional information and the particular aspects which cause difficulties for the visually impaired
- Design and develop methods for providing an interface between visually impaired users and sources of multi-dimensional information
- Test and evaluate solutions
- Discuss the results achieved, the methods used, and the scope for future research
- Involve visually impaired users at each stage of the project

## **1.2 Multi-dimensional information and the visually impaired**

A typical interface between human and computer relies heavily on the vision of the human user, with output from the computer presenting a number of problems to visually impaired users.

The graphical user interface (GUI) has increased problems with computer access for the visually impaired. The GUI provides an easy to use, intuitive interface by creating a highly visual system, involving the direct manipulation technique of point and click operations using a mouse type device - a hand-eye co-ordination function. Whilst the GUI has undoubtedly made computers more easily accessible to sighted users, it has created problems for the visually impaired. The sequential nature of text-based systems allowed straightforward access for blind and partially sighted users by using a screen reading program to control the channelling of display output to a suitable output device such as a speech synthesiser or refreshable Braille pad. With a GUI, there are many more problems to consider due to the nature of the particular interface.

A GUI is typically constructed from interface objects such as icons, buttons and windows, themselves graphical symbols. Such visual objects must be presented in a form comprehensible to the visually impaired user. Furthermore, spatial relationships between screen objects often represent significant information easily detected by the sighted user but not by the visually impaired user. Another feature of a typical GUI is a multi-tasking environment which enables the user to switch between applications. This means that input is directed to the currently active application. A visually impaired user must know the recipient of any input given.

The interface between the human and computer must be accessible to the visually impaired user, and so must other sources of multi-dimensional information. For example, an increasing number of texts are being put on-line in the form of electronic books (Portugal 94, Arundale 94). Features of such books include hypertext linking for indexing, for example. The growth of the internet provides another source of information which must be accessible to the visually impaired.

The scope of this work is not limited to one particular source of multi-dimensional information, but rather in investigating methods and techniques applicable to a range of sources, including, for example, graphical documents. Text-only documents can be

considered as one dimensional; the text is sequential, and allows relatively simple non-visual presentation, by means of speech synthesis, for example. Documents containing graphical elements, such as graphs, pictures, tables, logos, and different font attributes, contain significant additional information which may be of importance to a visually impaired person faced with such a document.

Currently, sighted assistance is necessary for many tasks which the visually impaired user needs to perform. This may include activities such as checking the layout and content of a document, or the positioning of entries on a form. An aim of this work is to provide methods with which the visually impaired user can perform this kind of task independently. This will allow the checking of the visually impaired user's own work and that of others. By determining the content and layout of a document, an application or even an entire interface, it is anticipated that the user would be able to use this knowledge to create and manipulate documents and applications of similar structure.

Computers offer many benefits not only to sighted people but also to the visually impaired. With an appropriate interface, the ability to be able to work in a single location, without mobility problems, provided by the computer, opens a potentially large number of employment and leisure opportunities.

Therefore, for each new development in computer interface design, there has followed considerable research effort in making the computer accessible. The result of this is a few commercially available products, and many experimental research systems.

Typically the visually impaired user will be able to use the computer by using a 'screen reader' program. This provides a means of controlling input and output to and from the computer. A description of a number of screen reading systems is given in the following chapter.

## **1.3 Project origins**

This section describes the origins of the project and the basis from which the current research developed.

TIDE (Technology Initiative for Disabled and Elderly people) is a European Community R&D initiative in the field of rehabilitation technology. Under this initiative, Nottingham Trent University participated in the VISA project (project no. 135), as described in [Harness 93b].

Software developed during this project serves as a basis for current research. This software, described in [Harness 93a], uses Icon Recognition techniques and Optical Character Recognition (OCR) to construct and maintain an Off Screen Model (OSM). At any moment in time, attributes of any identifiable screen object can be determined by interrogating the OSM.

[Sherkat 93] reports that software performance has been evaluated for Microsoft Windows 3.1, and is ready for generalising for other WIMP (Windows, Icons, Menus, Pointing Devices) interfaces. It is also suggested that an improved user interface is required.

So, a basic screen reading system is available. Output is achieved using a standard soundcard with speech synthesiser. Further details are described in Chapter Three. The requirement is to construct an interface through which the visually impaired user can interact with the source of information.

## **1.4 Tools and techniques**

This section outlines possible input and output channels and those employed in this project. It then describes further the techniques available for capturing and maintaining screen information.

### **1.4.1 Input and output devices**

Consider a typical personal computer available today. Input devices include keyboard and mouse, and possibly voice input. Output devices include the display screen and probably a soundcard. It would be useful, for reasons of cost and portability, if no additional hardware were necessary in order for a visually impaired person to use that machine. Therefore, this work attempts to make use of widely available, standard hardware in providing input and output, in particular, determining the extent to which such standard hardware can provide effective interaction.

The keyboard does not present any significant problems to a visually impaired person, with both tactile key labelling and auditory confirmation of keys pressed available to support its use. The mouse, on the other hand, is not such a natural choice of input device.

For typical usage by sighted persons, it relies on hand-eye co-ordination. The movement of the hand produces a related movement of the cursor on the screen. Some

often suggested criticisms of the mouse include the need to move the hand to and from the keyboard to use it, its use of a port which may be required for further adaptive equipment, and motor problems of the user. Whilst these factors, linked with the conceptual problem, are not inconsiderable, they are not unique to visually impaired users.

Given that there is a demand for mouse support [Pitt 95], it appears that if appropriate auditory feedback can be given, and concepts carefully explained, then the mouse should not be rejected without some investigation. A further aim of this work is to determine whether the mouse can be used to enhance interaction between visually impaired user and computer.

Output will be based on a standard soundcard; the display screen will only be of use if the user has some vision. The work is primarily aimed at totally blind users but with the expectation that the methods developed will be applicable to the partially sighted and even have implications for fully sighted users.

A soundcard provides two types of auditory display. A speech synthesiser enables text to be spoken using a variety of voices and speech attributes including rate and volume. This is the standard output technique used in commercial screen reading systems; one aim of this work is to see how to use the speech synthesiser efficiently.

A less commonly used technique is the use of non-speech auditory display. Non-speech sounds are still under-used in computer interfaces [Petrie 98] despite substantial research. This work seeks to establish when and how sounds should be used in the interface.

There are other input and output devices available, many of which are more suited to the particular needs of the visually impaired. These are described in the following chapter.

### **1.4.2 Screen capture and maintenance**

An Off Screen Model (OSM) is a database which contains a representation of the display screen. When a window is opened on the screen, for example, the screen change is detected, causing the OSM to be updated to reflect this change. The OSM is an essential component in a screen access system. There are two main approaches to constructing and maintaining an Off Screen Model.

One is to intercept calls made to the graphics drivers by the application. The drawback of this method is the need for a dedicated solution for every target interface system [Harness 93b]. The other is to perform screen analysis. This involves analysing the information contained in the graphical bitmap destined for the VDU display. The analysis software is then not specific to any particular WIMP (Windows, Icons, Menus, Pointing Devices) product [Pugh 92].

Such a system would provide a higher level of access according to the criteria laid down in [Boyd 90], but question marks remain over its performance. In [Thatcher 93], Thatcher suggests that whilst character or document recognition are feasible approaches, they are not likely to be as effective for a changing screen as for a static screen.

However, in [Harness 93a], Harness reports that target speeds have been achieved and demonstrates that the approach is viable. A system has been developed which uses optical character recognition (OCR) and icon recognition algorithms to maintain an OSM. The screen is constantly monitored for change, and when one occurs, the area containing the change is automatically analysed and the OSM updated. This can be compared to the OSM introduced by Berkeley Systems Incorporated and described in [Thatcher 93].

To summarise, an OSM typically contains information such as object co-ordinates, type, colour, label, etc, and is updated either by monitoring specific internal workings or the displayed bitmap. Effectively, an OSM can be considered as an object database, the contents of which must be presented to the user in an appropriate fashion.

## **1.5 Project design**

The Talking Windows System comprises a recognition-based OSM and an interface between visually impaired user and an information source displayed on screen. Windows 3.1 is the primary source used for system development. Talking Windows is to become a test-bed for evaluating information access solutions.

In this section, significant aspects of the design of the project are outlined, including ways of approaching the presentation of graphical information, interface structure, user involvement and evaluation.



### **1.5.1 Approaches to information presentation**

There are two distinct approaches to the design of an interface between visually impaired user and computer. One is to construct this interface specifically for the visually impaired so that, whilst it relies on mappings between it and the visual interface, it is highly abstracted from it. In this way the visually impaired user is using the interface in a different way to that in which a sighted user interacts with a typical GUI. There are people who advocate this approach, as described in the following chapter.

The second approach is to take the interface between sighted user and computer and adapt this to be usable by non-sighted users. This involves retaining terminology and concepts but mapping visual information to non-visual display. This approach is favoured in this work, for reasons described below.

The former option is advantageous in that an interface specifically designed for visually impaired users will arguably be easier to use and possibly more productive. Its disadvantage, applicability only to blind users, renders the second method a more favourable option.

By interacting in a similar way to sighted users, communication will be easier, and thus assist teamwork in a mixed sighted/non-sighted work environment. Furthermore, a system related to the visual interface will assist partially sighted users. Auditory information will complement the visual information available to them.

It is also suggested that creating better interfaces for the blind will lead to better interfaces in general. Adding appropriate auditory information to a visual interface could result in a richer interface for sighted people.

A main focus of the work is on presenting auditory information with a view to enabling manipulation of the corresponding visual information. Therefore, if information can be presented in such a way as to provide layout and relational details, the visually impaired person may be able to build an image of the structure of that information. Such knowledge will enable the user to recreate and manipulate information of a similar nature.

For example, if a visually impaired person uses a screen reader to study an application of some kind, the spatial and locational information picked up during this process may enable the person to create a similar application 'look' for sighted persons. Similarly, a

document read with such a system may be checked by the visually impaired without sighted assistance, and visual graphical documents may be produced independently.

The approach taken in this work is to map visual information to non-visual display whilst maintaining the concepts and terminology encapsulated by the visual information. A description of this mapping process is given below.

It is not sufficient just to announce which objects are present on a screen. The user also needs to know where the objects are and how they are related to each other.

For example, a sighted user may read a menu item, but is also aware of the menu to which it belongs, the window to which the menu belongs, and possibly the application owning the window. Furthermore, the menu item may contain more information than simply its label. There may be an indication of key required to activate it, a short cut key combination, or shading indicating whether it is enabled or not. The label is the main object being viewed, but there is substantial information being processed by the user without significant conscious effort. The nature of visual information is such that whilst concentrating on a specific object, its context and location are acquired with a minimum of effort. Perhaps a glance to the title bar above the menu gives necessary additional information, or maybe simply knowing its relative position gives the user sufficient data to manipulate the menu item.

Now consider the same screen from the point of view of a blind person, assuming that speech and sound are available to announce/describe the menu item. The menu item label can be spoken, but how can the rest of the information be presented? This is a key part of this work. How can contextual information be presented non-visually? For auditory information, time is a factor. Using sight, all the information described above is assimilated in an instant. With the temporal nature of auditory display, it is necessary to determine what information should be available and how it can be acquired efficiently.

For the menu item example, one approach would be to use speech for the direct information involved. The label of the item may be spoken - "Menu item File Open", for example. Some contextual information is available by recollection of previous information presented. Other information can be incorporated into the spoken message. Voice attributes can be used to indicate whether the item is enabled or not. A preceding sound clip may indicate the type of window to which the item belongs. Information

which cannot be incorporated in such ways may need to be available using additional methods.

It is important that in attempting to provide comprehensive information, unnecessary redundancy is avoided. It takes time to deliver auditory messages, after which an assimilation time is required. This involves the user deciding whether or not the meaning of the message has been understood. In a visual environment it is possible to study the display until assimilation of information is complete. The transitory nature of auditory display precludes such behaviour. Only relevant information should therefore be available.

The balance between simplicity of presentation and quantity of information must be correct for the effective usage of an interface. Consideration of this lies at the heart of the work. An additional consideration is that of control. How can the user control what information is being presented, its quantity and its format? This question is addressed throughout the work.

### **1.5.2 Interface structure**

In order to investigate means of enabling access to graphical information, the interface between user and information source must be a flexible structure capable of coping with a number of demands. It must allow implementation of a wide variety of experimental information presentation features, provide sufficient functionality to allow use by visually impaired users, and allow evaluation of the developed methods. Most importantly, the initial design of the system should allow for expansion, and the inclusion of additional features as they become necessary.

The term 'feature' is used to describe an interface mechanism. This could be a low-level function, allowing the user to perform a small but vital task such as determining font size. Alternatively it could be a more substantial component of the system, enabling, for example, screen navigation using a particular set of guiding sounds. Such features will be used as a means of evaluating higher level information presentation concepts.

The work can be seen as a feature-based approach. Features are designed, implemented and developed in response to user requirements. Involving users both in the ongoing design of the system and in its evaluation is fundamental to the work.

### 1.5.3 User-centred design

When considering how a visually impaired person might interact effectively with graphical information, possible methods and techniques appear from a number of sources, including both visually impaired and sighted people. Those from visually impaired sources are often invaluable. In order to ensure a high level of user input in the design and development of features, it was necessary from early on in the work to be in touch with visually impaired people.

One way was to meet and converse with a number of visually impaired users regularly throughout the work. As will become clear, this is a vital means not only of ensuring user input in the design stage but also in assessing and evaluating methods and concepts employed.

A second, significant source of user contact was achieved using the internet. There are a number of news groups and mailing lists covering a variety of blindness-related topics. By monitoring these on a daily basis, in particular those related to general GUI access issues and dedicated commercial screen reader discussion groups, the needs and priorities of visually impaired computer users could be observed. As a result, users play a fundamental role in the development of the access system.

A further consideration involved structuring work in such a way as to allow evaluation by visually impaired users. An obvious difficulty with assessing the effectiveness of methods employed is time. A description of the work, its aims, the nature of the testing process is necessary for all potential evaluators of the system. Familiarisation time is inevitable, and motivation may be a factor if due consideration is not given to this early in the work.

For example, by becoming too abstract in terms of methods used, or by diverting too far from screen reader concepts and basic functionality, the blind user can feel that spending time is neither relevant nor of immediate advantage. This led to a desire to provide a usable interface, with experimental ideas adding to standard features. This approach has the added advantage of allowing routine screen reader features to be enhanced, with the use of sound and speech attribute changes, for example.

A potential problem with the evaluation of much work of this nature is the difficulty in comparing results with similar systems. Firstly it is hard to measure interface usability,

and secondly, finding comparable systems is not always possible given the experimental nature of many of the implemented ideas.

Perhaps the most important outcome of evaluation of this work is in the guidance of future development of this and other systems. By opting for an approach of continuous assessment and refinement of ideas and methods, the direction of ongoing work can be determined.

### **1.5.4 Design and evaluation methods**

A number of recent studies emphasise the need to involve the end user in the design and evaluation of user interfaces. This section highlights some of this research.

In [Petrie97b], Petrie describes the user-centred design and evaluation of assistive and adaptive technology. This includes the use of 'Wizard of Oz' studies in which the functionality of complex systems is simulated by a human.

[Salminen 98] describes the importance of the participation of disabled and elderly users in the design and development of assistive technology, and methods for its evaluation with such users.

In [Hale 97], Hale develops a model of user-centred software assessment in an educational context. The model comprises three components; user profiling, technical screening and a qualitative analysis of end user interaction with the software.

[Petrie 96] describes methodologies for the evaluation of navigational aids for blind persons. Petrie suggests that evaluation should form an integral part of the design process, and that user preferences are vital to the success of a design. The use of user-centred design in the development of the MoBIC (Mobility of Blind People Interacting with Computers) travel aid is described in [Petrie 97a]. [Strothotte 98] describes the evaluation of an orientation and navigation aid for visually impaired travellers using a battery of measures including performance, workload and stress ratings.

Usability studies, determination of user needs and requirements, and design and evaluation methodologies for disabled users have been undertaken for a variety of user interfaces including a mobility aid for frail and elderly visually impaired persons [O'Neill 99], screenphones [Johnson 98], automatic teller machines [van Schaik 97] and hypermedia systems [Morley 98a]. [Morley 98b] describes the problems associated with using Braille or audio-taped books and how digital talking books can improve access to

study, work or leisure material. The design and evaluation of the prototype Daisy Playback software with visually impaired users is described.

## **1.6 Project evaluation**

This section describes the evaluation undertaken during the course of the project, including its aims, design, the users involved and methods employed.

### **1.6.1 Aims of evaluation**

The primary aim is to establish the extent to which the project has enabled visually impaired users to access multi-dimensional information. Evaluation also guides the development of the project. User reaction gives an indication of the viability of certain methods used, and helps to determine the potential of ideas. The intended users of the system under development have a significant influence in the addition of new features or consideration of fresh approaches to tackling the problems at hand.

The Talking Windows system has a number of similarities with some of the screen reading software available commercially. A further aim of evaluation was to compare Talking Windows with commercial products. In what ways does Talking Windows enhance user interaction with a graphical user interface? Could the methods used by Talking Windows be merged with commercial products to improve access to information?

### **1.6.2 Approaches to evaluation**

This section describes how the evaluation process was designed to provide the required form of results.

Evaluation cannot yield a simple yes or no answer to the question of whether the work has satisfied its aims. Neither can it be completed as a one-off exercise. Rather, it is an ongoing process, one which continuously feeds results back into development. A further benefit of this approach is that it enables the refinement of the evaluation process itself.

When planning evaluation, the two fundamental considerations are the users required and the system being used as a test-bed. These two factors combine to provide a set of results. At the outset, the availability of users and their characteristics, particularly in terms of level of visual impairment and computer experience, cannot be predicted with certainty. To a lesser degree, the development of the system is subject to external factors.

Whilst the overall objectives of the system are known, detail varies over the course of the project, particularly once initial evaluation has resulted in user feedback.

These factors necessitate a flexible approach to evaluation. With this in mind, the first phase of evaluation with visually impaired subjects was designed as an informal, preliminary investigation with two goals. Firstly, to assess the users available - ages, experience, and tolerance to their use of the Talking Windows system. Secondly, to establish how the system can be set up to allow effective usage by subjects. The outcome of this study would affect the design of subsequent, more in-depth evaluation studies.

Following this preliminary investigation, it was decided that it may be possible to attempt a quantitative analysis of user reaction to the Talking Windows system. This idea was formed on the basis that user reactions to individual aspects of the system could be scored provided that a set of rules can be formulated and adhered to. The advantage of performing such a study would be that scores could be used to guide development and, most importantly, allow comparison with commercial products of a similar nature and with future versions of Talking Windows.

This study was designed using conclusions drawn from the preliminary investigation. The number of users available and time required with each could now be predicted with more accuracy with the benefit of experience. Individual features of the Talking Windows system which represented the key concepts involved and which elicited a response, positive or otherwise, from users were identified for use in the study.

The third study focuses on acquiring a set of data suitable for qualitative analysis. From this data, it was anticipated that more meaningful conclusions could be drawn.

The second study provided further indications of user tolerance to evaluation of the system. It is important when planning a study to know how long you can spend with a user in one session, for example. This in turn leads to practical considerations influencing the design of evaluation sessions. It would be useful to have unlimited access to users, but in reality, there are constraints, in particular from the point of view of users, who often had to volunteer their time between classes.

A second conclusion drawn from the attempt at quantitative analysis was that whilst you can identify features which represent the concepts involved in Talking Windows, you are essentially testing a range of low level interface mechanisms. This was addressed in the third study by switching to a concept-based approach. As an example, studying an

individual sound map can provide important feedback on the implementation of that sound map and allows its effectiveness to be improved. But by stepping back from the implementation, the concept of sound maps in general, and their contribution to enabling access to multi-dimensional information, can be judged.

### **1.6.3 Users**

It is not essential to have subjects with no sight at all. Whilst much of the system is largely aimed at totally blind users, the use of speech and sound could assist partially sighted or even fully sighted users. By insisting on totally blind users as subjects you limit the users available unnecessarily.

Another consideration is that of the computer experience of users. When evaluation commenced in 1994, some blind people interviewed had had little or no experience with computers of any kind. It was also found that there was a significant time period between the launch of a new operating system or piece of software and the availability of a non-visual access product. As a result, many users were familiar with text-based interfaces but not with GUIs such as Microsoft Windows. For practical reasons it was decided not to restrict subjects to those with experience of using GUIs. This hinders in enabling comparison between Talking Windows and commercial screen reading products, but the more important aim is to abstract the concepts from the implementation.

For example, the use of sound maps to enable conceptualisation of position of a pointer relative to its surroundings could be applied to a number of information sources rather than just a particular GUI and should be presented as such. Experience of users with the GUI is therefore largely irrelevant in user selection.

However, when a user is familiar with using a computer, especially a GUI, it is reasonable to expect that more of that user's concentration will be on the system being presented whereas an inexperienced user may be more concerned with operating the keyboard or mouse. Similarly, an experienced user may recognise the need for a particular feature more quickly than an inexperienced user. This should be considered when drawing conclusions from the results obtained, and prior to that in presenting a feature and its purpose to the user.

Age of user is not a significant factor in choosing subjects, but for the purposes of background information, the approximate age of each subject was recorded.



Similarly, the musical background of each user was discussed and noted. As musical concepts are used in the work, and provide scope for future development of features, some knowledge of a users musical background and whether this links to the success of their interaction with Talking Windows was sought. Some interesting observations were made in this area as described in the following chapters.

The three main studies were carried out at the Royal National College for the Blind, Hereford (RNCB), each over a number of days, using both students and staff. As a result, a range of experiences and ages was available. Much of the contact was made via email, which suggests that users are acquainted with computer use.

#### **1.6.4 Internal evaluation**

Before exposing the prototype to visually impaired users for evaluation, it is subjected to a process of testing and refinement in an attempt to prevent valuable time with visually impaired users from being wasted on poorly implemented or ineffective features.

This involves sighted users experiencing features with and without the display visible. This removes unwanted effects on the system and provides the first opportunity to see the potential of the feature. Some refinement then occurs until, to a sighted user, the feature is reasonably effective. Care must be taken to ensure that sighted users are involved only in the refinement of features rather than in the design of the interface. As reported in [Petrie 98], findings from sighted participants cannot be used as a substitute for data from blind users, even when those sighted users are unable to see the screen.

Once the feature appears to be of use, it is exposed to wider scrutiny by means of presentations and demonstrations to both sighted and non-sighted persons. This further indicates the potential of various features, and suggests possible enhancements. Such feedback is instantly used to modify features where appropriate. The result of this process is a system incorporating features which have been tuned over time until as useful as can be established prior to exposure to visually impaired users in a more formal study.

# Chapter 2 Literature Survey

This chapter describes work undertaken in areas related to the current research, and how the current work compares with similar projects. In the first section, general blindness-related issues are discussed, providing a setting for the current work. This is followed by a review of existing access systems. The final section describes the input and output techniques available in providing non-visual access.

## 2.1 Research for the visually impaired

According to statistics published in 1991 in an RNIB survey of the needs of blind and partially sighted adults in Britain [Bruce 91], there are almost one million partially sighted adults in Great Britain, approximately four times the number actually registered. Of these, it is estimated that only 20 percent are totally blind. Visual impairment is a term that covers a range of conditions described in [Chapman 78] as “at one end the situation of total blindness, followed by mere perception of light, through to perception of objects in the near and distant environment, finally embracing the level of impaired sight which demands a minimum adaptation in the presentation of ... materials”.

For the registered visually impaired in Great Britain, prevalence rates increase from 3 per 1000 amongst 16-59 year olds, through 23 per 1000 amongst 60-74 year olds, to 152 per 1000 amongst those aged 75 or more [Hartley 94].

The 1992 report on the needs of blind and partially sighted children [Walker 92] estimates that there are between 100,000 and 250,000 children in Great Britain with significant visual impairments, with as many as 80 percent reported to have had their sight problem from birth. More than half of the visually impaired children mentioned have additional learning or physical disabilities which may preclude use of some computers or media.

Further research has been undertaken into the nature and causes of visual disability [Chapman 78], the status [Lowenfeld 75] and psychology [Lowenfeld 80] of people with visual disabilities, and the auditory skills of blind persons [Muchnik 91].

The need for the visually impaired to be able to access graphical information, in particular GUIs, is widely mentioned. In [Edwards 89a], Edwards points out that people

with disabilities in general have the same need to access computer-based equipment as anyone else, in work, education and leisure.

The need to provide accessible computers in the workplace has achieved legal status in the USA with two new laws relating to adaptive technology [Schwerdtfeger 91].

In [Scadden 84], Scadden points out that access to text-based computer systems had allowed the attainment of a degree of equality in job opportunities in areas using such technology, for example, word processing, secretarial work, or programming. With the spreading of visually complex interfaces, Scadden warns that unless alternative approaches are available, much of this newly acquired equality would be lost.

In [Hjelmquist 93], a study of workplaces employing one or more blind employees is described. This investigation found, amongst other things, that whilst information technology enabled otherwise impossible social functions, the use of graphical systems was a genuine concern in the workplace.

In [Gill 93], Gill describes evidence of the growing popularity and importance of GUIs, and how many large employers are adopting Microsoft's Windows as their software standard. As a blind employee needs to perform at a speed not too dissimilar to that of equivalent sighted employees, adaptive equipment must be available at an affordable price, with reasonable speed, and a modest level of training necessary. Gill also stresses that blind persons must be able to access the same computer systems and applications as their sighted colleagues, to allow full integration in a professional environment.

By the early nineties, over 12 million copies of Windows had been sold [Petrie 93b], with sales increasing rapidly since. Sales of Windows applications had, by this time, overtaken MS-DOS applications [Fountain 92].

In [Mansur 85], a blind user's requirements of a graphical system are described. A blind student must be able to accomplish many of the same data analysis and interpretation tasks that sighted persons perform when observing a collection of data. Currently, sighted assistance is required to help interpret the data, but this is not always practicable.

In [Jansson 85], work on mobility is described. This involves developing non-visual travel aids to allow safe and effective locomotion by the visually impaired. This may be

relevant in the metaphor used in allowing screen navigation by a mouse pointer as opposed to physical navigation. Where Gaver [Gaver 89] and Pitt [Pitt 95] advocate the metaphor of the cursor as a microphone, in some cases the display screen can be considered as a room, or environment, and the cursor a person in that space. For example, the Audio Rooms metaphor described in [Mynatt 93].

Further study of navigational aids for the visually impaired is described in [Petrie97b].

## **2.2 Access systems**

This section describes methods used to provide non-visual access to sources of information. Firstly, the historical aspect - access to text based computers. This is followed by a section detailing all aspects of graphical user interface access from the underlying characteristics of a GUI to specific commercial solutions to the problem. Finally, an alternative source of graphical information, the electronic document, is described.

### **2.2.1 Access to text-based computers**

Text based computers use a text display buffer (Boyd 90) to store information which is to be displayed on the computer screen. The screen of a text based computer has a number of 'cells' or 'positions' where a character may be placed. Typically there might be 25 rows, each able to hold 80 characters. In this case there would be 2000 positions on a screen in which a character can be displayed, as, perhaps, an 8x7 matrix of pixels. The text display buffer stores, in memory, two numbers for each of the 2000 possible positions. The first number represents the character and is typically its ASCII representation. The second number stores a code representing the attributes (colour and boldness, for example) of the character.

This type of system allows simple displaying of a character on a screen. Changing the appropriate number in memory is all that is required. The buffer can also be used to find out what characters are present on the screen at any given time. Accessing the text display buffer is a straightforward task. This is mainly because the information available can be mapped directly to speech or Braille; both sighted and non-sighted users are fed information a line at a time, therefore a means of controlling the information flow is all that is necessary. Cursor keys may be designated to select a row to be read from, and characters, words, columns and lines as the unit of reading. Options to read a particular line, find a certain word, or request text attributes may also be given.

There are a number of commercial screen reader packages available for accessing text based interfaces, for example, Tynytalk [Bohlman 92].

## **2.2.2 Access to graphical user interfaces**

### **2.2.2.1 Characteristics of a GUI**

Since the mid-eighties, there has been a marked increase in the use of graphical user interfaces, their purpose being to increase the bandwidth of communication between sighted user and computer [Mynatt 92a]. A fundamental difference between graphics-based and text-based computers is the screen rendering method. In graphical user interfaces, all screen-based information is composed from pixels, and pixel-based systems do not use a text display buffer. To display a character on the screen, the pixels forming the shape of the character in memory are set by the program. The computer then uses this pixel information to create an image on the screen. There are differing amounts of pixels on a screen depending on its resolution - a screen of 640x480 pixels, for example, contains 307,200 pixels. This allows for characters of any size or shape, graphical images, and screen displays that look like the printed information they represent. The use of pixels produces screen contents which are difficult to present to the user. [Mynatt 92a] reports defining characteristics of GUIs, including the use of pictorial representations of applications and files, the breaking of the screen into visually segmented areas, and the use of some form of pointing device to move the on-screen cursor. The visual interface that results contains a wealth of information, and this must be presented to the visually impaired user.

The graphical user interface is highly visual, and thus not apparently good for the visually impaired. The intention to provide a more intuitive interface, and thus allow wider use of computers, led to the use of graphical information. Is this necessary? In [Petre 92], it is suggested that the use of graphics does not necessarily lead to improvement of the interface and may introduce its own problems. Some writers assume that graphical representations are universally superior to text [Shu 88], which is not good news for the visually impaired or adaptive technology designers given that graphics are notoriously difficult to access. The conclusion drawn in [Petre 92] is that any text and graphics should complement each other, and so, inevitably, graphical systems may be expected to be prevalent for the foreseeable future.

### 2.2.2.2 Potential of GUIs for blind users

Graphical user interfaces can, however, offer significant advantages to the visually impaired user, something overlooked by some potential users and adaptive technology designers. In [Mynatt 93], three common characteristics of GUIs are given; the screen is divided into regions, called windows, which group related information together; an on-screen cursor is used to select and manipulate display items and is controlled by a physical pointing device such as a mouse; and small pictographs, called icons, represent objects which may be manipulated (direct manipulation) by the user.

In addition to direct manipulation, GUIs provide several benefits:

- they allow the user to work with different pieces of information at one time, and by laying out windows carefully, the user can access all needed information quickly.
- multi-tasking is generally supported, therefore the user can keep processes running but attend just to the required window.
- graphical images used in GUIs lend themselves to easy implementation of interface metaphors.

These benefits can provide similar power to visually impaired users if a suitable access system is provided [Mynatt 93]. For example, the ability to multi-task would be a significant advantage for a visually impaired user just as it is for a sighted user. The only requirement is that the control is available.

Mynatt describes the computer access situation by stating that “the technical problem is capturing the information that is being sent to a bit-mapped display, the design problem is creating a non-visual interface which provides the same power as a GUI [Mynatt 93]”. Mynatt then suggests that the goal of a metaphor-based adaptive interface should be to provide easy access to the basic operating system functionality of a computer, while maintaining a level of abstraction which makes the system easy to use. The aim of the metaphor is, as with the visual metaphor, to make the system intuitive for new users, and, to provide common means of functionality across the entire system.

An example of the potential benefit of providing access to computer systems for the visually impaired is given in [Burger 92]. A single CD-ROM disc can store the equivalent of approximately one kilometre of library shelf space worth of embossed

Braille pages. Given the additional search and retrieval techniques available electronically, the rewards are rich.

Similarly, the computer may provide a richer interface for Braille reading. In [Hartley 94], Hartley suggests that to imagine the lack of contextual information available when reading Braille, cut a small hole in a piece of paper, and read a document through that aperture. Computer systems have the potential to present a Braille document with additional auditory locational cues. The same 'keyhole' example may also illustrate the narrow bandwidth of many auditory or tactile techniques. Most speech output devices are only capable of presenting small pieces of information at any one time, giving a limited view of screen content.

The graphical user interface is a source of concern for the visually impaired, but also offers potential benefits. In [Boyd 90], the authors describe the nature of GUIs and their differences from traditional interfaces. They then suggest that rather than considering existing access as an either/or proposition, it is more beneficial to determine to what degree new systems provide blind people with access. In particular, what benefits of the GUI are shared by sighted and non-sighted users? Four levels of access to the graphical user interface are proposed, ranging from level one, where access is restricted to one application and one operating system, to level four, where access applies across all applications, all operating systems, and allows modification to fit individual needs. Three main stages of development, a customising stage, a single sensory stage, and a multi-sensory stage, are also identified, coinciding with the four levels of access.

### **2.2.2.3 Approaches to providing GUI Access**

Two distinct approaches to providing GUI access have been identified, as follows:

- a) design a completely new non-visual interface based on a non-visual metaphor
- b) maintain the visual metaphor and appearance, but provide non-visual representations for visual entities

The former approach involves constructing an interface between the computer and the visually impaired user. The latter involves constructing a non-visual interface between the GUI and the user.

In [Edwards 89a], Edwards suggests that custom-designing a system especially for blind users should ensure that it will be nearly ideally matched to their needs. The design

will make the most of the abilities of the target group. However, Edwards also indicates that adapting the visual system has advantages of its own. In particular, developing large amounts of software for a small potential market is uneconomical. It is also psychologically important for disabled individuals to be aware that their needs can be met with a minimum of special accommodations. All members of an organisation should be using the same software so that data can be easily shared. The ability to use the same hardware and software as sighted users will inevitably improve employment prospects [Edwards 89a].

In [Mynatt 93], the use of adapted, visual interfaces is not encouraged, due to their inherent visual nature, resulting in inappropriate interaction. The retrofitting of the graphical desktop metaphor with audio information neglects the cognitive benefits of the GUI, it is claimed.

It is suggested that it is no longer satisfactory to develop assistive technology for each new system only as it becomes available [Scadden 86], and that this has led to the frustrating situation where assistive technology is not available for some months after the new product has been launched, causing potential for job loss in the interim [Petrie 93b]. However, it is inevitable that there will be occasions when adaptation of visual systems will be necessary as not all systems will be designed bearing in mind the requirements of visually impaired users.

In support of the adaptation of the visual interface is the desire of partially sighted persons to make use of their residual vision in any interface [Edwards 89a]. A visual interface equipped with additional non-visual output, which alone would suffice, would allow use by all persons, including fully sighted, partially sighted and totally blind persons.

In [Hjelmquist 93], the integration of visually impaired employees into a predominantly sighted work place situation is described. The importance of team work is necessary in many instances, thus the ability of the blind person to use the same systems as their sighted colleagues is essential.

In [McKiel 92], McKiel places emphasis on building an environment that allows the sighted and non-sighted user to share the same experience of the graphical aspects of the interface.



#### **2.2.2.4 Training visually impaired people to use GUIs**

The GUI introduces concepts and terminology with which the visually impaired user must become familiar. Traditional sources of training material such as manuals and on-line help programs may be either inaccessible or unsuited to the needs of this group of users.

[Morley 95] provides an introduction to Windows concepts for visually disabled users with the aid of tactile diagrams. The training of blind people in the use of graphical user interfaces is described in [Weber 94].

#### **2.2.2.5 Requirements of a GUI access system**

When considering the design of a system to allow access to graphical interfaces, what features are required? [Theberge] includes a list of spoken features of a program, Winkline, which provides basic speech output for MS Windows. Keys, icons, menus, controls, buttons are spoken as encountered, together with text and highlighted text. Features required of a word processing access system include the control of speech, regions in which contents are monitored, and event markers. Further features are described in [Boyd 90] and [Sherkat 95].

When considering the interface between the Off Screen Model and visually impaired user, two areas must be considered; what information to present, and how to present it. In [Laws 95], Laws lists the information needed; what windows are open, what programs are running, what is the foreground window, what is the program name and title, what type of control is currently selected, and what is the text associated with that control.

#### **2.2.2.6 Capturing GUI screen information**

Information displayed on the GUI screen must be captured by screen reading software to enable presentation to the user. This can be done either by intercepting information before it reaches the screen, or by waiting for the information to be displayed and then employing recognition techniques.

An example of the former method is described in detail in [Mynatt 92b]. The GUI being studied is the X Windows system, and the system designed to provide a mapping between the graphical interface and an auditory interface is called The Mercator System. Captured information is stored in a database known generally as an Off Screen Model (OSM). Various commercial screen reading systems also use interception techniques.

Screen Reader/2 for OS/2 [Laws 95] from IBM intercepts display driver calls before text and icons become pixels. Additional screen information is obtained through graphics engine and Applications Program Interface calls. This is used to construct, and thereafter update, the OSM database.

Interception methods are also used in screen readers such as outSpoken, Jaws and Window Bridge. Screen information capture is done in real time and is accurate. However, the practise of intercepting display driver, graphics engine and API calls is generally specific to a single platform rather than common to all. Therefore, for each new platform arriving on the market, there is a substantial delay before an access system can be written. It also depends on how many hooks into the operating system are provided. This can be illustrated by noting the period of time between Windows 95 being released and the first screen reader becoming available. Note though that with the use of standard protocols and access packs this delay will be reduced.

The second method for determining screen content is that of employing recognition techniques on the bitmap displayed on the screen. This has been studied in [Harness 93a, Harness 93b, Pugh 92, Sherkat 93]. Despite concerns over the validity of this approach in terms of speed [Thatcher 93], it has been reported [Pugh 92] that the approach is viable.

Optical character recognition (OCR) and icon recognition are used to construct an OSM database for the screen, containing co-ordinates and attribute information for each object displayed on the screen. Whenever a change occurs on the screen, it is automatically detected by the system, and the area that has changed is analysed to update the OSM. Therefore there exists, at all times, an up-to-date object database which may be used to present information to the visually impaired user.

Whichever of the two approaches is taken, the next stage in providing access to computers for the visually impaired is to create an interface between the user and the off screen model. This involves providing means for the user to determine what is currently on the screen and how to interact with it.

The current work uses the OSM created by using the bitmap recognition techniques investigated in [Harness 93a, Harness 93b, Pugh 92, Sherkat 93]. This OSM contains a label, type, colour and co-ordinates for each object present on the screen at any given time.

Generally, all screen readers for graphical user interfaces use some form of OSM [Thatcher 93]. Minimum content would be text, its position, colour and font, and window handle. Different screen readers will contain more and different information depending partly on the level of interception, and partly on what screen information can be recognised or deduced.

### **2.2.2.7 Control of presentation**

A screen reading program must provide ways for the interact with the Off Screen Model. The current work concentrates on input from the keyboard and mouse. Commercial screen reading systems tend to neglect the mouse and concentrate on keyboard entry of commands. This involves using key combinations to issue a command to the screen reader, but the combinations used vary considerably. For example, outSpoken uses the number pad section of the keyboard, which avoids key clashes with application hotkeys, but doesn't allow for more than a certain number of features. Other systems use unused key combinations from the main keyboard, or macros, or provide a means of using key combinations in use by an application but temporarily disabled.

It is maybe due to the variety of interfaces available that, from consultation with visually impaired computer users and observation of relevant newsgroups, there appears to be no single screen reader favoured over the others at the present time. Some are very powerful in terms of functionality but with a steep learning curve, whilst others have a simple interface but do not provide a means of customising the system for individual applications. Most features, however, are common to most of the available screen readers.

Control of the presentation of information is an important aspect of a screen reading system. In [Griffith 90], Griffith describes the inevitable problems caused in this area. Namely, the cognitive demands on the user in learning commands for both the application and adaptation, and the likely need to toggle keys between the two systems. If the user loses track of the particular step in a procedure, mistakes may occur.

In reviewing the hardware-based Frank Audiodata system [Frank 93], Edwards [Edwards 89a] suggests that whilst software-based solutions are generally less expensive, their reliance on the keyboard makes learning the system more difficult. If the keyboard is used to specify the portion of the screen to be spoken, to control the manner in which it

is spoken, and to input commands and data into the application, the keyboard will inevitably become overloaded.

In the following review of some current screen reading systems, the means by which this problem is handled is described, where possible.

### **2.2.2.8 Research-based GUI access systems**

#### **(a) The Mercator System [Mynatt 94]**

The Mercator System provides a mapping, transparent to applications, between X-based GUIs and auditory interfaces, for the purpose of providing the visually impaired with access to GUIs. The system maintains an OSM using a framework established to monitor, model and translate interfaces transparently to the application [Mynatt 92b]. Two approaches to collecting information are used; client server monitoring, and the Editres protocol. The former uses the architecture of X to 'tap' the client-server connection, eavesdropping on the conversation between the application and X [Scheifler 87]. The Editres protocol [Peterson 91] defines a method for querying a running application and retrieving from it information about the hierarchy of widgets which make up the application, information which client-server monitoring alone cannot provide. The OSM therefore contains types and attributes of interface objects, and a hierarchy providing some relationship information.

The system uses a compromise between mimicking the existing visual interface and adopting a new auditory interface by performing translation at the level of interface components such as menus, dialog boxes and buttons. It uses auditory output initially, due to its proven usability [Buxton 91] and the inability of some blind people to use tactile stimuli [HumanWare 90].

Visual interface objects are mapped to Auditory Interface Components (AICs) which convey two pieces of information, type and attributes, using everyday sounds to form auditory icons [Gaver 89].

Navigation must support scanning of screen contents, and operations on interface objects. The mouse is not used due to the small granularity of pixels involved, therefore, AICs are mapped onto a tree structure which may be traversed using the numeric keypad to make either small steps or larger jumps. This method only uses a few keys, so the problems described earlier do not apply.

Results suggested that although the navigation strategy was favourable, the sounds used for auditory icons were often unclear, and that more navigation controls were necessary. Future directions included the use of spatialised sound, peripheral information, and tactile output [Mynatt 94].

It appears that this approach does not allow the user the benefits of determining the physical layout of interface objects, thus, for example, precluding the development of visual interfaces by visually impaired designers, for example. Whether it is always possible to map interface objects into a tree-like structure, particularly without losing information, is also questioned. For some operating systems it is doubtful whether semantic information concerning interface objects is attainable, certainly not without resorting to tapping into internal workings, thus precluding the more universal OCR based techniques.

### **(b) Systems 3 [Vanderheiden 90]**

Systems 3 is a non-visual computer interface designed to allow access to standard graphics-based computers and software by blind persons. Wherever possible, the system bypasses the visual metaphor, presenting the information in the most effective form. It also provides tools to facilitate direct interpretation of the visual image where that is necessary. The aim is to allow non-sighted users to work at a rate comparable with that of their sighted peers, with the exception of graphics intensive applications.

The proposed system incorporates speech input and output, a full page virtual tactile display with single pixel resolution, and a Braille display. Whilst two of these methods are optional, the system appears costly, in particular the custom designed virtual tactile tablet. Whilst details on proposed information capture and specific presentation strategies are not detailed, the use of voice recognition to allow verbal input of commands should reduce strain on the keyboard.

### **(c) Audio-enabled GUIs [McKiel 92]**

In this system, the user controls a pointer using a mouse or similar device, receiving auditory-tactile feedback as opposed to the usual visual-tactile feedback. As the pointer moves over objects on the screen, sounds are generated to convey their identity. Sounds used include musical tones, filtered noise, recorded sound effects, synthesised speech and stereo effects to enhance the user's sense of position and size of displayed objects.

This system uses speech to announce the entering of a new window by the pointer, and increased volume to indicate the currently active window. Metaphorically intuitive sounds, such as a diminishing volume to indicate a minimise button, are used where possible.

Background sounds are used to indicate position on the screen, something also used in the current research. Stereo balance depicts horizontal movement, whilst vertical motion is accompanied by stepwise changes in filtering of the ongoing background sounds.

In this system, information was captured using discrete graphics calls to create a simulated window-type GUI, but ongoing work includes implementing the system on OS/2 Presentation Manager.

### **2.2.2.9 Commercial screen reading systems**

#### **(a) outSpoken [Schwerdtfeger 91]**

outSpoken was first developed in 1989 by Berkeley Systems (BSI) for the Macintosh GUI. It is a screen reader that communicates with blind users, via speech, as they move the mouse pointer around the GUI and come into contact with text and graphical objects on the screen. BSI constructed a database, the OSM, that the screen reader software is able to access. The OSM is the conceptual basis for GUI screen readers that vendors are developing today. It must manage information resources, namely text, off screen bitmaps, icons and cursors, including position and attributes.

How does outSpoken construct and maintain its OSM? The display reconstruction software is driven from low level GUI graphics functions. To build an OSM, you must hook low-level graphics commands to call the patch code that constructs the model. This is a system dependent approach.

The precise operation of outSpoken is given in the manual accompanying the demonstration version. The navigation system uses the number pad section of the keyboard, in combinations involving the control and shift key. Keys also simulate mouse actions, by allowing movement of the pointer, which acts as the screen review cursor, and single and double clicks. Further key combinations move the cursor onto various objects, causing their names to be spoken. The user may jump from one object to the next in a given direction, thus efficiently navigating the interface objects, but not picking up any spatial information. Using the number pad generally avoids clashes with running applications and with operating system key combinations, but to operate the screen

reader you need to move your hands away from the home keys. Also, the small number of key combinations available does not appear to allow for much expansion in terms of number of features available. In its favour, outSpoken is generally considered easy to learn, and able to cope with many applications without notice [Clark 96]. However, it is often said to be restricted by its inability to be customised to individual users' requirements.

### **(b) Jaws for Windows (JFW) [Oppermann 94]**

Jaws for Windows is a Windows-based screen reader developed by Henter-Joyce Inc. JFW uses the whole keyboard, not solely the number pad. In order to avoid keyboard clashes with Windows and other applications, the Insert key (Number Pad Key 0) is re-designated to perform a function similar to that of the shift or control key. By using it in combination with other keys on both the number pad and the main section of the keyboard, key combinations are almost certainly unique to JFW. The use of this key with number pad keys allows one handed selection of combinations in many cases, with the right thumb pressing the Insert button, and the fingers the other key. The right little finger can press the Insert key when the second key is on the main part of the keyboard. Alternative key combinations are used for laptops.

JFW uses the concept of two cursors, which is addressed by the current work but with the distinction not emphasised. The PC cursor moves highlights, dotted lines and carets, and is independent of JFW. The JAWS cursor moves the mouse pointer and is mainly used for reading functions. The user selects a current cursor, and functions act according to that.

The amount of descriptive information given to the user can be varied using a 3-level 'verbosity' setting. Inexperienced users would typically select the most descriptive setting, whilst experts would require less description.

JFW also provides a macro system, which allows two things. Firstly, a number of functions may be combined to form a single command. For example, instead of using 30 keystrokes to perform a sequence of functions, one key can be defined to carry out each function in order in one go, therefore saving time and effort. The second benefit is that of being able to redefine which key combinations perform which function. This allows the individual needs of each user to be met. So, for example, if a user is used to performing a particular task with a certain combination, and the default JFW key layout is different,

the JFW key can be reassigned, allowing a smooth transition between screen readers, for example.

It appears from the comments of a number of visually impaired computer users that the general feeling is that whilst JFW may be more difficult to learn than some alternative screen reading systems, the macro facility in particular provides a powerful and flexible system.

### **(c) Window Bridge**

Slimware Window Bridge (SWB) is a memory resident program (TSR) that communicates with the blind computer user through synthetic speech or Braille output of the screen contents. Both DOS and Windows (3.1, 3.11, 95) can be accessed using SWB, with support for over 40 speech synthesisers and 25 Braille displays provided. To allow the user to determine how the Windows screen actually looks, a Graphic Mode is incorporated. This allows the user to literally 'feel' the location of window borders, the location of icons, text, and the relative distance between objects on the screen as the Braille display is scrolled.

SWB uses a keyboard interface allowing the user to control the movement of the insertion pointer, highlight focus or mouse pointer with just one finger. The former two are moved using the standard Windows arrow keys, with full speech and Braille output. The movement of the mouse pointer is controlled by the numeric keypad, with screen information read using a single command. The mouse is simulated using a key to mimic each button, and two keys to navigate the mouse through the screen an icon at a time. The Window Bridge Control Panel is used to provide access to all the SWB features.

SWB can be configured by using customisation facilities to create files which enhance how either DOS or Windows applications are presented. The keyboard is flexible in its layout; without relying on macros, fast, on-line customisation is possible.

### **(d) Other commercial screen reading systems**

There are a number of other commercial screen reading systems not described here. At the CSUN conference held at Los Angeles, CA, in March 1996, a number of screen reader manufacturers took part in a 'Duelling Windows' session.

The "duel" involved seven different Windows screen reader software products performing identical tasks in the Windows 3.x environment. Each machine used was



identical, being an IBM Pentium with 16MB of RAM. The software on each hard drive was identical, with the exception of the Windows screen reader which was installed by each vendor prior to the duel. During the presentation, a large screen projector displayed for the audience what was on each vendor's screen. Also, the persons doing the presenting could neither see their monitor, which had been placed on the floor in front of their table facing the audience, nor the large screen. Each question was asked by one member of the panel of each vendor, one after the other, and the panellists rotated the asking of questions among themselves. Additionally, the order of which vendor went first was rotated with each question, so that no one vendor was able to benefit from always hearing the other vendors' attempts to answer. The results of this duel are described in detail in [Clark 96], and serve as a comparison between the various available systems.

### **2.2.3 Non-visual access to documents**

#### **2.2.3.1 Introduction to document access**

Graphical user interfaces allow screens to display information that looks very similar to the printed document it represents. This is due to the very high resolution of the screen and the use of pixels rather than a text display buffer. This capability enables documents to be placed on-line, for example, Project Gutenberg [Arundale 94]. Similarly documents can be created and manipulated using word processing packages which incorporate the facility to add graphics to the document. Such packages differ from their text based ancestors as they allow the creation of documents which take on any format which the user wishes, including many fonts, colours, graphical images, graphs, tables, etc.

The growth of electronic publishing requires that access to on-line graphical documents be investigated. Problems are often caused by the difficulty in assessing structure of documents. Much work has been done on looking at the language in which the document has been written and using this as the basis of the presentation work. For example, mark up languages such as SGML contain information such as the plain text together with attribute information. Publishers are being encouraged to use this in the creation of documents to allow uniform access by a standard document reader. Similarly, documents written using the Adobe system can be accessed using the Acrobat Viewer, which translates the coded document into a structured text.

These and other such efforts appear to transform a graphical document into a sequential form which can be read directly to the visually impaired reader. However, this leaves the person unaware of the rich visual structure of the document, and what it looks

like to a sighted person. This makes it impossible to create a visual document in a suitable format. Therefore, techniques designed to present spatial and layout information must be developed in an integrated document reading system.

### **2.2.3.2 Document access solutions**

In [Raman 92], Raman describes a system for presenting (La)Tex documents using auditory information. The source of information to be presented - document content and structure - is the (La)Tex source file. It is impractical to simply speak the source, as it contains streams of layout characters. Therefore, the content is to be picked out and read, but also the layout information is to be used in the conversion of visual formatting to audio formatting.

Audio cues have the advantage of presenting structural information quickly, allowing the spoken part to be solely the content of the document. The main content described in this paper is mathematical equations. A problem in the audio presentation of this type of information is the sequential nature of speech. When a sighted person looks at an equation, they are able to see the main structure, and continually refer back to the equation until its content has been assimilated. Requiring a non-sighted person to take in a complete complex expression is not suitable. Control and replay of auditory information is necessary. The ability to 'browse' or 'glance at' an algebraic expression is described in [Stevens 94].

In [Portugal 94], Portugal describes an experiment to investigate the use of auditory signals in the presentation of document structure information. This work involves using non-speech sound cues to indicate where in a document the reader is at any time. That is, sounds indicate the chapter and sub-section of a piece of spoken text. Results indicated that sound cues alone were not sufficient to provide the same level of navigational information as the visual or combined audio-visual cues.

The issue of capturing document structural information is widely researched. Either the document itself contains the required structure and content information, or the document image can be analysed using recognition techniques. The goal of the current work is to present the information only.

Documents written in plain ASCII text are easily presented using either Braille or speech. Other documents may not be so straightforward. World Wide Web documents are often written in HTML format (Hypertext Mark-up Language), where paragraphs,

sections, links to other documents, and so on, are coded within the document source file. The code, however, is still ASCII, and so again, speech or Braille can be used to read HTML documents.

Web pages can be made more accessible to people with visual impairments, for example, by coding an image description together with the image. A World Wide Web Consortium project has developed Web Content Accessibility Guidelines to assist web page authors in making their pages accessible to people with disabilities. An evaluation of these guidelines is described in [Colwell 99].

Whilst the World Wide Web provides a wealth of information and entertainment possibilities there is concern about the difficulty of access by people with disabilities. In [Laux 96], issues are discussed. This includes the current problems and barriers, the lack of tools for automating user interface design for people with disabilities, and who has responsibility for enabling access, the user or the web designer.

A user interface design tool is described in [Akoumianakis 95]. This provides tools which enable the needs of different user groups, for example, the visually impaired, to be included in the design of user interfaces, enabling access to be part of user interface rather than something which is added on as an afterthought.

Document publishers want to define how documents will look to the reader, something that HTML does not allow - here the final appearance depends on the software used to read it. Therefore, other more universal formats are used, for example, Adobe's Portable Document Format (PDF). This requires a piece of reading software to view the document, which in this form is inaccessible to the blind. To alleviate this problem, Adobe have developed a 'helper application' which will convert PDF format to a form accessible by screen readers [McQuarrie 95]. A further document format, SGML (Standard Generalised Mark-up Language), of which HTML is a subset, also fits in this category.

Document image analysis is an area which attracts a great deal of research interest. An overview is given in [Rahgozar 95]. Document image analysis involves putting a document into a form that can be queried [Haralick 94]. Therefore the kind of document, its logical structure, its areas of text and non-text, the read order of text blocks, and information such as author, title, and so on, need to be determined. A further description of document image analysis is given in [O'Gorman 93].

### **2.2.3.3 Requirements for a graphical document access system**

The second area of work is that of presenting graphics based documents. For this it is necessary to make certain assumptions. Two stages are involved. The first is the use of document analysis to construct a database of contents of a document. This includes co-ordinates, labels, style and other attributes of text, graphical images, and blank spaces on the document. The second stage is the presentation of this information in a suitable manner. Current work concerns the second stage only. It is therefore assumed that certain information is available to work with, namely a database of document objects. Literature suggests that this is reasonable.

## **2.3 Non-visual display techniques**

### **2.3.1 Input techniques**

Conventional input devices such as the keyboard and mouse may not be the most suitable input devices for the visually impaired.

In [Griffith 90], Griffith describes the human factors issues surrounding the use of special purpose input-output tools by blind or visually impaired users. He addresses the seven classes of human factors problems [Aaronson 87] that affect the use of such devices. These classes bear some relevance to the choice and usage of appropriate input-output devices in the current work.

Certain i/o products may be incompatible with certain computers and interfere with existing devices. In the current work it has been observed that the mouse port is often used by either an external speech synthesiser or other assistive hardware. Some i/o devices may have limited perceptibility - unclear or muffled speech, distorted sound playback, etc. Over-stressed memory capacity is an important human factors issue to consider, as the reduced bandwidth of information transfer through non-visual channels requires more information to be remembered if the user is to function effectively. Disruption of motor tasks refers to the need to remove fingers from the home keys frequently in order to issue commands. It has been observed in the current work that some blind users reject the idea of using the mouse based on this argument. The result of the range of human factors shortcomings is summarised by the consequent problem of slowed information processing [Griffith 90].

### **2.3.1.1 Voice input**

Voice input is a possible means of input which doesn't require any vision. In [Vanderheiden 90], Vanderheiden describes the use of voice input in a non-visual interface. This would involve the use of a voice recognition system. Much work has been done in the field of voice recognition, both on techniques [Kuah 94, Higgins 94], and applications [Elmaghraby 89, Gordon 90, Sola 90]. Problems with voice input include use in environments in which constant talking is undesirable, the need for training, difficulties with accents/speech clarity, or where the individual is unable to speak [Vanderheiden 90].

### **2.3.1.2 Keyboard input**

The keyboard is expected to be a suitable input device given that many visually impaired persons are able to touch type [Edwards 89a]. There are several problems, however, which must be overcome. In his study of human factors issues in computer access for the visually impaired, Griffith [Griffith 90] noted that whilst each of the research participants had been taught how to touch type, speeds were quite low.

There is also evidence to suggest that use of the keyboard in access systems increases the cognitive demands on the user [Griffith 89]. This is due to the visually impaired user needing to know the commands for controlling both the application program and the adaptive system.

In [Frank 93], Frank mentions that it is difficult to imagine the problems of navigating a computer screen using only a small number of keystrokes, especially using the same keyboard as for the application program.

In [Dix 93], the effectiveness of textual communication is questioned, and the use of predictive algorithms to reduce the amount of typing required considered. It is also suggested that certain users, for example those with learning difficulties such as dyslexia, may have problems entering textual information via the keyboard.

The designers of commercial screen readers for the visually impaired have used a variety of ways of overcoming problems in accommodating both application keystrokes and screen reader commands whilst not using additional keypads or custom built hardware.

### **2.3.1.3 Mouse input**

The mouse is an input device, but it is very closely linked with the cursor that is part of the output displayed on the screen [Edwards 89a]. The user may only use the mouse for input if the corresponding movements of the cursor displayed on the screen can be tracked. The mouse is an example of a pointing device that acts in a relative fashion, in that there is no correlation between the position of the mouse on its pad and the position of the cursor on the screen.

In [Pitt 95], Pitt suggests two reasons why the mouse should be used in an adaptive system. Firstly, there is a demand from blind people to be able to use the mouse, and secondly, there are certain actions that are simply better performed with a mouse than in any other way.

Pitt also investigates whether the mouse can be made usable by blind persons. An adaptation was developed which created an auditory space based on the metaphor of the cursor as a microphone. Subjects used non-speech sound cues to direct the mouse pointer towards eight on-screen targets without vision. Sounds were presented both monaurally, such that loudness represented distance from the target, but without directional information, or in stereo. Here the sound would come from the left or right speaker depending on the position of the cursor relative to the target. Results suggested that whilst speeds in locating targets were slow in comparison to sighted users, they were good enough to suggest that it would be possible to use the mouse as one component of a non-visual interface.

There are other reasons for not using the mouse as an input device other than its inherent visual nature. Users with physical disabilities or motor impairments may lack the hand movement and control necessary to move the mouse with the precision generally required [Dix 93]. However, by making suitable adaptations to the system, and providing keyboard alternatives to mouse actions, the mouse should provide an extra input device of benefit to some blind users.

### **2.3.1.4 Custom-built input**

Some systems designed to allow the visually impaired access to computers use custom-made input devices not standard on commercial machines. It would appear that the Eyegaze system, which tracks eye movements to control the cursor, or a keyboard

driver which can be attached to the user's head [Dix 93], would be unfeasible. The Reactive keyboard [Greenberg 93], a predictive system, could be of some use.

In [Vanderheiden 90], Vanderheiden suggests the use of a 'Virtual Tactile Tablet' as an input and output device simultaneously. This consists of a standard graphics tablet with a special tactile mouse. There are additional buttons on this tablet which can have their function dynamically changed to meet the specific needs of the particular user.

In [Frank 93], Frank describes the TASO (Tactile Acoustic Screen Orientation) principle and Audio Screen. Here, the keyboard contains two sliding keys, one vertically oriented and to the left of the computer keyboard, and the other running horizontally below the space bar. These sliders give direct access to portions of the computer display screen.

In [Gill 93], the GUIB project terminal is described. This integrates specialised input and output devices, providing the ability to input and display each type of information in the most suitable format. A pressure sensitive touchpad is used in place of the mouse with the finger directly positioning the review pointer. Furthermore, extra buttons provide additional means of inputting information. Routing keys move the cursor to the chosen location, whilst two buttons substitute for the mouse buttons. Output from this system is described later.

There are alternatives to the mouse as a pointing device. Relative devices include trackerballs and spring loaded joy sticks, but these are less prevalent than the mouse, and are of similar nature in terms of use without vision. Absolute devices such as graphics tablets or floating joysticks could be more easily used by the visually impaired, but as with custom-designed devices, their cost could be prohibitive. For example, it would be an extra cost to a potential employer.

With absolute devices, the visual feedback may be augmented by kinaesthetic feedback from both the hand and arm movements and tactile cues on the surface on which the device is moved [Edwards 89a].

Although the mouse is optimal for sighted users [Card 78], it need not be the most suitable for blind users.

A comparative evaluation of input devices for the visually impaired computer user is given in [Petrie 93a].

## **2.3.2 Auditory output techniques**

Auditory output includes the use of both synthesised speech and non-speech sounds. Computer access systems for the visually impaired have tended to concentrate mainly on speech output, but with the increasing availability of inexpensive soundcards, the use of non-speech audio in interfaces for both visually impaired and sighted persons has been studied in depth.

### **2.3.2.1 Hearing and perception**

Hearing is a powerful perceptual process. Often we are not even aware of the constant stream of auditory information we are processing [Mynatt 93]. Due to our perceptual abilities, hearing is naturally multi-tasking. For example, we can monitor multiple conversations at the same time, a phenomenon known as the 'cocktail party effect' [Cherry 53]. Comparing this with vision, in which fast context-switching is used by the single-focus eye to assimilate complex information, it is reasonable to assume that sound can capture the multi-tasking properties of computer systems as well as or better than graphical output [Mynatt 93].

Whilst investigating the use of auditory output in computer systems, the hearing abilities and perception of sound of humans are inevitably discussed. A popular misconception is that the senses are naturally balanced, and therefore if a person is lacking in one sense, they will make up for it with the others. There is, however, no evidence to support this [Edwards 89a]. In fact, "any higher efficiency of the blind in interpreting the sensory data perceived must be the result of attention, practise, adaptation and increased use of the remaining faculties" [Lowenfeld 75].

Much work has been done to prove or disprove the above theory, in particular the case of the blind person and a possible improvement in hearing ability. For a review of such efforts, see [Muchnik 91]. Here, a comparison of the central auditory skills between blind and sighted subjects is made. The skills evaluated were the localisation function, the temporal auditory resolution ability of the blind adult, and the ability of the blind person to discriminate speech material in noise. In each of the three experiments, the blind subjects achieved significantly better results than the sighted subjects, suggesting a certain superiority of the blind individual with regard to central auditory function.

The human ear has several characteristics that may be exploited, in particular the ability to remember pitches or sequences of pitches. Most listeners can differentiate



between two pitches only slightly varying in frequency [Mansur 85]. In [Saslow 67], Saslow demonstrated that pitch discrimination was still very good for two test pitches near 120Hz, separated by just 0.3Hz. It has also been claimed [Yeung 80] that some twenty characteristics of sound can be identified.

Furthermore, sound has a distinct advantage over tactile methods in its greater bandwidth as a channel. A typical Braille user may read 50-100 words per minute whereas speech may be comprehended at speeds from 200 to as high as 400 words per minute [Mansur 85].

The frequency range of hearing, 20-20,000Hz in the very young, is not generally fully utilised, with the 300-3000Hz range of speech usually the limit. Reasons for this include hardware and storage limitations and the less aesthetically pleasing sounds at the extremes of the range of hearing [Mansur 85].

In [Kaufman 79], Kaufman observes that the average listener is most sensitive to sounds between 1000 and 4000Hz. Kaufman also points out that if a human is exposed to a loud sound (though not loud enough to create damage), the ear will be restored to almost normal sensitivity after a fraction of a second. When using sound in a computer system, it is essential to take into consideration the hearing of the user and the consequent appropriateness of the chosen sounds.

### **2.3.2.2 Reasons for using auditory display**

Voice is a primary means of auditory communication, just as text is a natural first choice for visual presentations [Blattner 89]. Voice and text have similar properties as a means of communicating information; both are sequential, highly dependent on a natural or programming language, and generally require the full attention of the user for comprehension. It is therefore natural that speech should be explored in computer interfaces before non-speech sounds.

To the visually impaired individual, sound provides the main source of information about their environment [Mansur 85]. Communications with other people, musical enjoyment, and access to printed material on tape are examples of the importance of speech and sound in the everyday life of visually impaired individuals. It is therefore natural to utilise this in computer systems for the visually impaired.

In [Dix 93], further arguments for the use of speech and non-speech sounds are put forward. The ability of the human to differentiate a wide range of sounds and to react faster to auditory stimuli than to visual cues requires exploitation in interface design.

The serial nature of speech and hence the time needed to extract the meaning of a sentence is an incentive to investigate the use of non-speech sounds in some cases. The short assimilation time and universality of non-speech sounds are also favourable. Sounds have no language barriers unlike speech. It is also noted in [Dix 93] that non-speech sounds have to be learnt.

In [Raman 92], Raman suggests the use of non-speech audio cues, such as short chords, beeps, etc, to augment speech by conveying non-textual content.

In [Gaver 89], Gaver suggests reasons for using sounds in interfaces in a role complementary to vision. These include the fact that sound uses an extra modality, it is present and familiar in everyday life, and it complements vision.

### **2.3.2.3 Use of speech in the interface**

Requirements of a speech-based system are described in [Mynatt 93]. Speech should be understandable, and support a large vocabulary. Variable speed is necessary for scanning information at a much faster rate than a natural speaking pace, and different voices and inflections should be provided - a variety of voices can differentiate types of messages in the same way that colours and fonts can add meaning to textual information. The vocabulary should be context sensitive - a character string should be presented according to its type, for example, a phone number as individual digits as opposed to the verbose number [Mynatt 93].

As many people have difficulty in comprehending synthesised speech, training is necessary. [Griffith 90] describes a training program designed to develop facility in understanding synthesised speech.

In [Gerhard 88], Gerhard describes how a speech output unit must completely replace text output by performing all screen functions. It must pronounce every word, requiring an unlimited vocabulary, temporarily store screen text contents to enable checking of previously written text, and be able to jump to any place in the text and announce that position. This is mostly applicable to text-based rather than graphical interfaces.

Gerhard also observes that speech must be used carefully to avoid frustrating the user. A novice user would often require the name or function of each key announced upon striking, whereas to an experienced user this could cause irritation. This type of situation requires the ability to customise the system, or to have some form of intelligence built in.

Speech synthesis devices have the advantage of being low cost products addressing the needs of a large future market.

Although speech can be used to simultaneously present information on multiple applications, it is doubtful whether this is practical. Speech requires directed attention, and, like text, is serial in nature, requiring processing from beginning to end. Therefore, non-speech sounds may be a preferable means of monitoring background applications [Mynatt 93].

Edwards [Edwards 89a] reports that speech is insufficient for modern interfaces using complex displays. He also suggests that although some speech synthesisers allow the increase of speech rate, speech communication is generally slow, with improved interfaces requiring new technology to allow accelerated speech and pitch correction.

A further disadvantage of speech-based technology is poor representation of material and lack of adequate control over information flow [Stevens 94].

In [Smither 93], the demands synthetic speech place on short term memory are investigated. The performance of old and young adults on an ordinary short term memory task were compared. Results suggested that the comprehension of synthetic speech imposes increased resource demands on short term memory compared to human speech. An interesting comparison here is the observation that performance at reading text is unaffected by whether the text is displayed on paper or on a computer screen [Hartley 88].

When using speech, the demands placed on the listener must not be neglected. This suggests the presentation of spoken information in short, digestible chunks. The size of these will depend on the individual user, but experimentation will determine the approximate number of words or syllables suitable.

An application of the use of speech in the presentation of information is given in [Hartley 88]. This makes use of the similarities between text and speech mentioned above. The three main sources of information for the visually impaired, large print,

Braille, and speech via spoken audio-cassettes, may be enhanced by careful design of text [Hartley 94].

### **2.3.2.4 Classification of non-speech sound**

Non-speech sounds can be used in a variety of applications for both sighted and non-sighted persons. Sounds used in computer systems can be categorised as either of two types. Representational sounds use digitised samples of naturally occurring or 'everyday' sounds, for example, a door closing. Abstract sounds are constructed from single or groups of pitches, producing a rhythmic and tonal pattern.

#### **(a) Representational sounds**

Gaver [Gaver 86, Sumikawa 85] studied auditory icons as caricatures of naturally occurring sounds from the surrounding environment. For example, sounds include bumps, scrapes, or files hitting mailboxes. There are three different types of representational sounds; symbolic mappings rely on social convention for meaning, such as applause for approval; nomic representations are physical such as the sound of a closing metal cabinet for closing a file; and metaphorical mappings are similarities, such as falling pitch for falling objects [Blattner 89]. Gaver noted that auditory icons only need capture the essential features of an object, not a realistic representation.

An example of the use of representational sounds in an interface is The SonicFinder [Gaver 89]. In this interface, information is conveyed using auditory icons as well as standard graphical feedback. An example interaction in SonicFinder involves a file. Selecting the file makes the sound of an object being tapped. The type of file is indicated by the material of the object, and the size of the file by the size of the object struck. An important application of this work is in modifying graphical interfaces for the visually impaired [Edwards 89a] and more generally, using such interfaces over the telephone.

In [Gaver 89], Gaver mentions navigation when the screen is not visible. The problem here should probably not be of one knowing location in Euclidean space, but of knowing how to get to objects or positions of interest. Auditory cues can be useful in this task, for example, 'auditory landmarks' [Jenkins 85].

The GUIB project [Gill 93] used sound to explore screen objects. Non-speech sounds were chosen for the presentation of interaction objects, and these sounds were natural sounds metaphorically related to the screen objects. For example, the sound presentation

for mouse tracking was steps walking on different natural surfaces, depending on the condition of the screen at the position of the mouse pointer.

## **(b) Abstract sounds**

In [Blattner 89], a study of non-representational sounds is described. Earcons are audio messages used in the interface to provide information and feedback about computer entities to the user. Audio patterns may be used to design 'modules' for earcons, which are then assembled into larger groupings called 'families'. The modules are single pitches or rhythmicised sequences of pitches called 'motives'. Motives create a short, distinctive audio pattern often characterised by the simplicity of its rhythm and pitch design [Christ 66]. This brevity and distinctive manner makes motives very powerful tools for composing earcons [Blattner 89].

In [Brewster 95d], the problem of 'lack of design guidelines' for designers of auditory interfaces [Arons 94] is addressed. This follows much research into the creation and application of earcons [Blattner 89, Brewster 95d]. Earcons in these studies also make use of abstract notes and are thus defined as abstract, synthetic tones that can be used in structured combinations to create sound messages to represent parts of an interface [Brewster 95d]. There are occasions when a natural sound cannot be found to represent an event or interface object, in which case abstract earcons will be required. In this case the interface will be using combined earcons.

For an earcon to be useful, listeners must be able to learn and remember it, and associate it with a meaning [Blattner 89]. For representational sounds, this should not be difficult according to [Gaver 86]; natural sounds are related to events in a principled, systematic way, and people learn this mapping from early childhood in their interactions with the world.

For abstract earcons, learning and memorising the earcon and its association with an entity is related to the human ability to remember distinct musical patterns or notes [Gaver 89].

### **2.3.2.5 Applications of non-speech sounds**

In [Mansur 85], the use of sound to present graphs to blind people is described. Computer-generated sound patterns of two-dimensional line graphs are used to give the blind a means of understanding line graphs in the holistic manner used by those with sight. The sound is continuous, but varies in pitch to represent motion in the x-direction.

It was found that individuals were able to learn rapidly to interpret sounds in a meaningful way.

Gaver [Gaver 89] uses representational sounds in an auditory interface designed to complement vision.

In [Edwards 89a], Edwards describes Soundtrack, a word processor with an auditory interface. It is an attempt to adapt a mouse-based interface into an auditory form. Soundtrack incorporates auditory menus as a means of specifying commands. The layout of the component auditory interface is based on a grid. As the user moves the mouse over the grid, a tone indicates quickly the type of object under the mouse pointer, and a press of the mouse button will reveal the name of the object. To move down a level, the mouse button is double clicked within the required object window, which then becomes subdivided into a set of component objects. These behave as the upper level window.

In [Lunney 81], an investigation into the use of non-speech sounds in communicating experimental data to blind chemistry students is described. Infrared spectra, which provide a means of identifying the components of a chemical sample, are encoded, and data translated into an auditory form based on the 96 notes in an eight-octave chromatic scale. The resultant auditory spectrum can be inspected by either listening to the components in sequence, or played together as a chord.

In [Raman 92], Raman describes the use of sound in conveying document structure information. (La)Tex is used to typeset documents. The typesetting source captures a quantity of document structure which is then used to typeset the document. This can also be used to convey structure in audio renderings of the document. This work involves methods for auralising mathematical expressions. Since prosodic cues are unattainable using current speech technology, non-speech audio cues are used instead. The sound is synchronised with spoken output. The ability to recognise document structure from the electronic source allows browsing of an audio document provided adequate control is available.

In [Mynatt 93], the concept of Audio Rooms is introduced. This is part of The Mercator Project [Mynatt 92a, 92b]. Audio Rooms is based on a rooms metaphor aimed at providing an intuitive non-visual interface. The metaphor resulted from evaluations of the work habits of people using GUI-based systems, in particular the tendency to group windows based on functionality. The implementation, Audio Rooms, provides the same

cognitive organisational benefits as the rooms metaphor, plus the additional benefits of its familiarity to blind users with mobility skills, and its richness.

A further application of non-speech sounds is to provide an auditory 'glance' at an algebraic expression, described in [Stevens 94]. For sighted persons, an overview of a document is an important part of planning the reading process. Similarly, a glance at an algebraic expression reveals its format, main components, and purpose. Such a glance is not possible for a blind person, who must generally listen to the entire expression being spoken. However, in this work, prosodic cues used in the speaking of an expression were combined with earcon design theory to produce a high level view of expression structure.

The use of non-speech sounds to guide an unsighted person using a pointing device to up to eight on-screen targets is described in [Pitt 95]. An auditory space is created based on the metaphor of the cursor as a microphone. Therefore, the target object makes the sound, and the loudness and direction of the sound perceived by the user moving the mouse pointer (microphone) depends on the position of the pointer relative to the object.

Auditory conditions tested included monaural sound, where loudness only represents distance from the target; stereo sound, where the direction from which the sound is heard indicates position of cursor relative to target; and volume/distance effects, where the change in volume varies either linearly with distance from target, or non-linearly such that movements near the target produce greater volume changes. Results suggest that the mouse should not be dismissed as an input device for blind users, and could be seen as an additional component in a non-visual interface.

In [Brewster 95a], the problem of slipping off a button without noticing is described, and non-speech audio in sonically-enhanced buttons is used to solve this problem. In [Brewster 95b], non-speech audio is used to sonically enhance a set of interface widgets. Each widget is analysed to find usability problems, and sound added to overcome these. And in [Brewster 95c], a structured method for integrating non-speech sound into graphical interfaces is described. The method analyses interactions in terms of event, status and mode information, and then categorises this information in terms of feedback needed to present it. This is then combined with guidelines for creating sounds [Brewster 95d] to generate the auditory feedback required.

In [Petrie 98], Petrie describes the selection of non-speech sounds to represent events and objects in a non-visual interface. Where possible, auditory icons are used, providing

a link between sound and the event or object which it represents. For example, the sound of scissors snipping to signify a cutting event. When such a relationship is not possible, an earcon is used, for example, a piano motif. Studies of the use of these sounds are also described.

### **2.3.2.6 Multi-dimensional sound**

Some designers of computer systems for the visually impaired use the localisation capability of humans to allow a richer form of auditory feedback.

#### **(a) Use of two-dimensional sound**

In [Strothotte 93], a new device for two-dimensional sound output to enable users to locate the position of a screen object acoustically is described. This is an aspect of blind peoples' access to graphical user interfaces addressed in the GUIB project [Gill 93]. Since the display screen is a two-dimensional output device and conventional mono or stereo sound only one-dimensional, sound generally presents little or no spatial information. By constructing the 2D Sound-Screen, the two dimensional locational information of objects displayed on screen can be represented by using non-speech audio. The hardware involves a five channel loudspeaker array system named Sound-Screen. Directional acoustic cues are produced by setting the intensity of the single loudspeakers using a digitally controlled amplifier matrix (DCA).

In quiet environments, or noisy ones, presenting sound through speakers may be inappropriate. For this reason, headphones producing similar sound qualities are being developed [Strothotte 93].

#### **(b) Three-dimensional sound**

As well as the listener hearing sound as if it originates from a point on a two-dimensional plane, it is possible to add in the third dimension. The sound then appears to come from a point somewhere in space around the listener. This makes use of the localisation capability of the human. Applications for three-dimensional sound include directing the visual attention of airline pilots [Doll 86, 87], maintaining the orientation and awareness of surroundings of pilots [Calhoun 87, 88], and virtual reality [Pope 93, Burgess 93].

Whilst the use of two-dimensional sound systems described in [Strothotte 93, Gill 93], are ideally suited to the two-dimensional nature of the display screen, the use of three-



dimensional sound to present information to a visually impaired person is tempting. For example, the concept of foreground and background windows, in other words the three-dimensional nature of the windows metaphor, would be ideally suited to sound that would appear to come from in front of or behind the listener. Problems include the technical requirements of an accurate three-dimensional sound system [Begault 91, Wenzel 92, Loomis 90, Doll 92]. Furthermore, in some environments, headphones are needed for sound output, and localisation of sounds through headphones is a complex problem [Begault 93].

### **2.3.3 Alternative output techniques**

#### **2.3.3.1 Tactile display**

There are several alternatives to visual output from computers. These generally involve haptic or auditory devices.

At any given instant in time, only a part of the information displayed on the screen is available using non-visual channels. Therefore in most situations the efficiency of people using non-visual display techniques is much lower than for visual information presentation [Burger 92]. For example, the user often has to reconstruct semantic links between pieces of information at different locations, plus there is the lack of contextual or layout information.

Furthermore, listening and feeling requires more assimilation time than when reading with eyesight [Gerhard 88]. This is a major hurdle to overcome in the non-visual display of the information contained in graphical user interfaces.

Much work has been done on various output techniques. This tends to concentrate on either finding the most appropriate form of non-visual output for a particular information type, or investigating the viability of presenting information using a particular output device.

An example of the former approach is work described in [Vanderheiden 89]. Here, visually displayed information is divided into eleven types. These include running text, text attributes, spatial relationships between text items, picking from a list, interrupts/alerts, directing, image icons, stereotypic images, pictorial images, animated images and three-dimensional images. Means of presenting information of each type are discussed - this includes the use of Braille output, speech synthesis, tactile graphics and non-speech sounds.

### **(a) Tactile graphics**

Much research has been undertaken in this area, which includes the use of refreshable Braille. The electromechanical transfer of tactile information is described in [Jarmul] with the objective of providing a portable system which would aid the blind in obtaining sensory data about their immediate environment. The significant interface parameters involved in the transfer of sensory data are also discussed.

Haptic methods have been found to have some distinct drawbacks. In [Mansur 85], a study of the use of sound to display graphs is described. Here, it is mentioned that while haptic methods such as tactile graphs are of proven usefulness, they can be expensive, bulky and require the user have someone create them for them. They are also slow and require a large amount of learning time before effective use is possible.

### **(b) Braille**

The use of Braille to display output information has some advantages and disadvantages. In general, assimilation speeds when listening are slower than when feeling [Gerhard 88]. Therefore, the use of haptic devices, in particular Braille displays, would be expected to increase effectiveness of information transfer. This would be especially so if the design of the output was investigated [Hartley 94] and could be achieved with current Braille display technology.

Refreshable Braille displays allow the translation of any printable ASCII character into a matrix of 8 dots. Often, they are based on a piezoelectric crystal which changes form when a voltage is applied [Burger 92]. The most common types of device can display up to 20, 40 or 80 characters on one or two lines.

There are, however, sound reasons for not relying solely on Braille output from computers. One is that only a small proportion of the blind population actually use Braille [Edwards 89b], with only about 10% of blind or visually impaired persons proficient [Bowe 84]. There are a number of reasons for this surprisingly small percentage. Many people lose their sight late in life, and are unable or unwilling to learn Braille. The increased number of elderly blind people [Edwards 89a], [Hartley 94] may be significant. The lack of sensitivity in the fingertips is often cited as a reason for not reading Braille. Similarly, it is reported that a significant portion of people who are blind also suffer from diabetes which causes a reduction in their sensitivity to tactile stimuli [HumanWare 90]. Motor problems would also have an affect. Finally, the advent of low

cost speech synthesis and sound reproduction technology, offering an inexpensive alternative, may influence the decision of recently unsighted persons as to whether to invest the time and effort needed to learn Braille.

Braille is also insufficient for some classes of information. For plain text, it can be ideal, and has an advantage over synthesised speech in that it is less transitory with information remaining on the display longer than speech. However, with adequate control features, speech can mostly replicate Braille output [Edwards 89a].

Another drawback is the cost of refreshable Braille displays. This could be prohibitive to many potential users, both private persons and potential employees. In the USA, two new laws have been introduced to assist the development of adaptive technology for the disabled. An amendment to Public Law 506 (29 U.S.C 794d) states that a company can only sell a product to the US government if that device is accessible to persons with disabilities. The Americans with Disabilities Act (ADA) states, in part, that if a company has more than 25 employees, it must provide reasonable accommodation (including technology) in the form of job adaptations for the handicapped [Schwerdtfeger 91]. Adaptive technology which requires expensive hardware would not tend to be favoured over solutions where such devices were not compulsory. Two possible reasons for the apparent high cost of refreshable Braille are the small potential market and the technology involved.

Note, however, that Braille may be more useful to deaf-blind persons than auditory feedback.

It may be assumed, therefore, that while Braille has undoubted benefits, it should only be a component of an adaptive system, and not the sole output mechanism.

### **(c) Non-Braille**

Non-Braille tactile devices have been studied as means of representing graphical information either at the interface level or within the application. For example, a research team at Nagoya University in Japan is investigating the use of a 16x16 matrix of pins which can produce three dimensional tactile displays [Gill 93].

The Optacon (Optical To Tactile Converter) [Sensory Systems, Telesensory] is a device which scans text and repeats the shape of the letters scanned in a small pad of vibrating pins, sensed by the fingertip. It can be used to scan text printed on paper, or

with a special camera it will read text on a CRT screen [Edwards 89b]. By transmitting the regions of varying brightness of an image to the grid of vibrating rods, it is possible to track the shape of an x-y curve, for example [Mansur 85]. The Optacon was commercially available until recently, when the manufacturers, Telesensory Systems [Telesensory], announced that from the end of 1996, it would be discontinued, due to OCR technology and low cost speech synthesisers becoming the preferred choice of most blind people.

However, the use of tactile information is being investigated still, from several viewpoints; what classes of information are suited to tactile representation [Vanderheiden 89], what is the most suitable device in terms of technology, cost and ease of use, how information can be presented in tactile form, and how tactile devices may be used in conjunction with audio output.

As with Braille output, the main disadvantages of non-Braille tactile devices are cost, reliability, portability, suitability and the number of potential users within the visually impaired population.

### **2.3.3.2 Large print**

For people with some vision, large print or screen enlargement can be used. This has the advantage of making use of the residual vision of the user, something that is welcomed by partially sighted people [Edwards 89a].

Software programs can be written in order to use the CRT display as a 'magnification' system. Various portions of an initial screen can be blown up in order to make them more visible [Burger 92].

Font sizes for use by the partially sighted are described in [Hartley 94]. The Royal National Institute for the Blind considers 10 point type too small for many sighted users, and recommend 14 point as the minimum size for material intended for the blind and partially sighted.

In studying optimal character size for partially sighted students, Schweitzer [Schweitzer 84] found that by only slightly increasing font size, the percentage recognition of the characters increased sharply.

Note though that magnification is not a universal remedy for partially sighted people. For those with a reduced field of vision, such as tunnel vision, magnification can make

reading more difficult because it implies there being less text within their field of vision [Edwards 89a].

### **2.3.3.3 Custom-built output techniques**

Some systems are designed to use custom-built input and/or output devices. As described earlier, the Virtual Tactile Tablet [Vanderheiden 90] combines the input abilities of a mouse with direct tactile output using a small tactile array of simulators mounted directly above the 'virtual ball' of the mouse. As the mouse moves the user would feel on their fingertip a raised, vibrating representation of the image on the screen at that point.

### **2.3.4 Multi-modal interfaces**

Multi-modal systems have been developed to take advantage of the multi-sensory nature of humans. Utilising one or more sense, or mode of communication, these systems may improve interaction [Dix 93]. In the case of systems for the visually impaired, a multi-modal interface would make use of the auditory and tactile channels as well as providing facilities such as screen enlargement to make use of any residual vision.

Some information found in graphical user interfaces or documents is redundant, for example, icons typically have a graphical symbol above a text label. Either one may be viewed to glean the meaning, together one of them is not strictly necessary. However, the presentation of redundant information using alternative channels may increase a user's performance [Dix 93]. For example, in a multi-modal system for the visually impaired, when an icon is encountered, its name may be spoken, but also a non-speech sound representing the icon and a tactile representation could be given as background information.

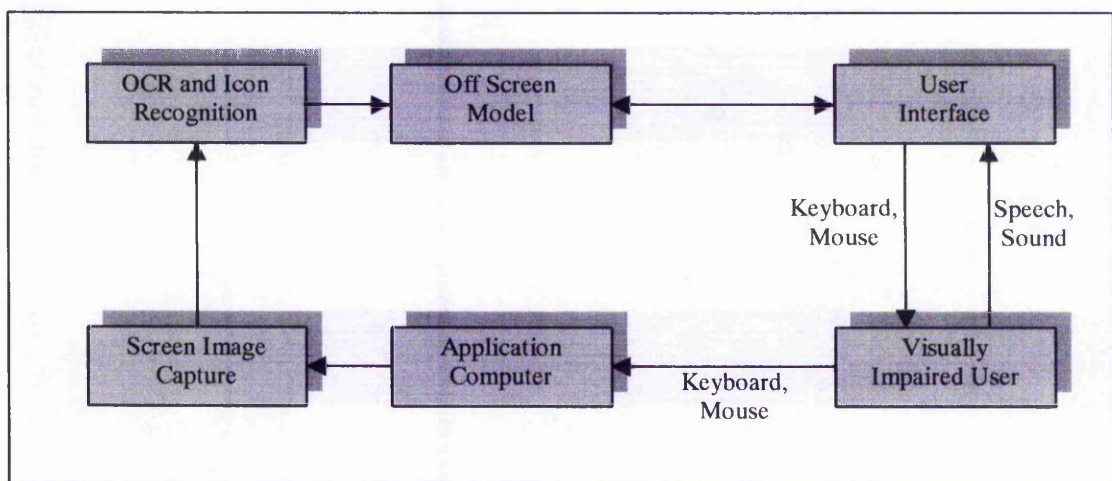
In [Burger 92], multi-modal interfaces are described. A system can be said to be multi-modal if it possesses hardware that can capture and display messages on various channels with a response time compatible with the user's expectations, if it is able to select the most suitable modality for output; and if it is able to decode multi-modal input expressions - which implies it is capable of linking events through temporal relationships [Coutaz 91].

# Chapter 3 Talking Windows

This chapter includes a description of the Talking Windows system architecture including the Off Screen Model used, and the basic features added.

## 3.1 System architecture

The Talking Windows system enables the visually impaired user to access a GUI. Its components are shown in the diagram.



**Figure 3.1 Components of the Talking Windows system**

The main element is the interface between the Off Screen Model and the visually impaired user. This interface consists of a number of features which enable the user to access screen information.

### 3.1.1 The Off Screen Model

The Off Screen Model (OSM) is a database of information about each recognisable screen object. The OSM used by the Talking Windows system is constructed using Optical Character Recognition and Icon Recognition as described in [Pugh 92]. This involves analysing the screen bitmap, looking for objects. An object could be a button, an icon or a character, for example. Objects found are matched against a database of known objects.

For the purposes of this work, eight basic types of object are used; buttons, title bars, icons, text, scroll bar controls, dialog box controls, resize corners and graphics. A ninth object type, the window, is created by using these basic types.

For each screen object, type, label and co-ordinates are stored. Where an object exists on the screen but can't be identified, nothing is stored. It is possible to add objects to the known object store as encountered, for example, a newly created icon.

The OSM must be kept up to date. The screen is monitored constantly, so that when a change occurs, for example, a new window opened, it is noticed, and the OSM updated. The time that it takes to update the OSM depends on several factors. Firstly, the proportion of the screen which has changed. Only this area needs to be considered. Secondly, speed depends upon the size of the known object store. The speed of the computer is also a factor. If the time taken is more than a couple of seconds, users will experience frustration.

### **3.1.2 Accessing features**

Features make use of the OSM. For example, a feature could give the user a count of the number of windows present on the screen at a given time. How does the user access the feature? The simple answer is by pressing a key on the keyboard. However, there are complications.

There are many features available, particularly because the system is an experimental one. There are more features than different keys on the keyboard. It is possible to use a simple key combination, using, for example, the control or alt keys. However, because Talking Windows is a 'background' application, working at the same time as, say, a word processing application, there is potential for clashes.

During the earlier stages of the project this problem was addressed, perhaps with undue emphasis. Existing screen readers were studied and user opinion sought in order to find a suitable solution. The advantage of the commercial products is that they tend to have fewer features and so require fewer key combinations. Solutions included a customisable keyboard layout and designating a new key rather than control, shift or alt to form combinations. Users did not tend to favour any particular method, with each having advantages and disadvantages.



The solution used was to concentrate on the number pad together with either control or alt key. Although it was said that moving hands off the main part of the keyboard was not ideal, the method did not result in clashes with the other applications used, which was an advantage.

It was necessary to devise a system of 'modes' in order to enable such a larger number of features to be accessed in this way. This meant that you had to press two or three key combinations to access a feature. To assist the user, an optional key confirmation system was added. On first pressing of a key combination, a description of the feature was announced. On pressing the combination again, the feature was activated. A tutorial system was also added to help learn the location of available features.

The system of 'modes' allowed related features to be grouped logically to assist learning. In Sound Map Mode, for example, you could switch on or off all the available sound maps. When new functionality was added, a new 'mode' was often necessary. The system could handle the increasing number of features using this method.

Further options were considered for this purpose. A dialog box belonging to Talking Windows would accept commands and provide an on-line help system. An alternative would be a menu approach. The ideal solution could be voice input, however, this was not pursued due to the availability of the technology at the time.

## **3.2 Basic features of Talking Windows**

### **3.2.1 Basic operation**

The system does not attempt to create a non-visual metaphor as is the case with some systems, for example, The Mercator Project [Mynatt 92a, 92b]. Instead, the visual interface is maintained, and the user able to interact with the contents of the screen in much the same way as the sighted user. The layout of the screen is preserved, but visual feedback is replaced by non-visual feedback.

In general, when the cursor is found to be 'on' an object on the screen (this is determined by constant monitoring of cursor position and comparison with the co-ordinates of objects in the OSM), the name of the object, and its type, are spoken in the form of a message. The following section describes the format of messages given.



### 3.2.2 Message format

The general format of messages throughout the system is a short non-speech sound followed by a spoken string, as shown:

**[non speech sound] [spoken text]**

The non speech sound is used for two purposes -

- to inform the user that there is a spoken message to follow,
- to inform the user of the type of the message.

When choosing the non-speech sound, it may be tempting to use several different sounds depending on the message. However, from communications with visually impaired users it appears that large numbers of sounds can cause problems. The learning of each new sound and associating it with its intended meaning will take longer the more sounds there are. Therefore, care should be taken to ensure that a 'new' sound is only introduced when it is expected to enhance the message, for example, when an indication of message type is important.

For this reason, a general sound is used in order to inform the user of the forthcoming message, but when message type is not specific. There are three general purpose sounds currently used. The first indicates that a message follows, but it also signifies that everything is in order - a function has been carried out. Therefore it takes the form of a pleasant-sounding, high-pitched single tone.

The other two sounds correspond to events that occur throughout the system (independent of the message type), but indicating either an error, or a count of zero. The error sound consists of three high-pitched tones played rapidly, sounding a warning. The sound indicating a count of zero is a dull, low-pitched tone, similar to a thudding sound.

By using the preceding non-speech sound as a preparatory cue, once the user becomes familiar with what the sound represents, the type of message will be known before the spoken part commences, thus aiding assimilation of the message. Currently, whilst the sound generally represents a type rather than a complex entity, a simple sound should suffice. Everyday, or representational, sounds would be suitable if such could be found with a link to the meaning the sound is portraying.

### **3.2.3 Screen navigation**

This section describes ways of propelling the cursor about the screen. As previously described, the mouse is one way of doing this. The main problem experienced at this stage is that whilst navigating blank areas of the screen, there is no feedback as to where the cursor is in relation to either screen objects or the screen itself. This problem will be addressed later in this chapter.

#### **3.2.3.1 Cursor movement using the keyboard**

An alternative to the mouse is to use key combinations to move the cursor horizontally and vertically. In some instances, the ability to move in a straight line is an advantage, for example, when navigating a menu. In order to address the problem of lack of speed of cursor movement, two features have been implemented. The first involves increasing the rate at which the key combinations move the cursor. The technique used for this feature is to jump the cursor a number of pixels in the direction intended each time the key is pressed.

The number of pixels used must be such that objects are not missed by too large a step, and frustration caused by the slow tracking speed of too small a step avoided. Experimentation shows that a range of suitable step sizes can be used, varying according to direction - the step size can generally be reduced for vertical movement due to the fewer pixels in that direction. For each direction, two step sizes are chosen, one from each end of the range. For general cursor movement, the larger step size will be used; for more precise movements, the smaller step size would be preferable.

#### **3.2.3.2 Cursor positioning**

The second feature intended to increase speed of cursor navigation is that of cursor positioning. Here, the cursor can be sent directly to any of many screen locations, irrespective of the current layout of objects. For example, the cursor can be moved to any screen corner, the centre of the screen, the edge of the screen in any direction, or to the nearest edge or corner. These features carry more significance once feedback is provided as to the whereabouts of the cursor at any time. This is described later.

#### **3.2.3.3 Bookmarks**

If the user spends time and effort locating an object or screen position of interest, and then moves the cursor away, it is inefficient to then have to repeat the process the next

time that particular position is needed. A number of features have been developed, ranging in complexity, all allowing positions to be saved for later return. In one case, co-ordinates and labels are saved to file for repeated use. An example of the use of this would be to locate each button and menu item of a word processing application, say. Each time that application is used these could be navigated to directly provided the application re-opened in the same position. This kind of functionality could be extended to give a navigation history which could be traversed as required.

### **3.2.3.4 Window component navigation**

An aspect of GUIs which can be of benefit to both sighted and visually impaired users is the use of standard controls, allowing applications to have a similar appearance. For both, learning new applications is easier - for example, once a menu system has been learnt for one application, a similar menu system used by a new application will be understood. For visually impaired users, the fact that things happen in the same places, for example, window buttons being either side of the title bar, helps with navigation.

A window area in Microsoft Windows 3.1 generally consists of a title bar, a close button, a minimise button, a maximise or restore button, a number of menu items below the title bar, and often a horizontal or vertical scroll bar, or both. Furthermore, the window, provided it is not obscured or maximised, will have four resize corners.

A feature has been implemented to allow the cursor to be moved quickly to any of these components in a selected window, provided they exist. The basic object types which form the OSM can be analysed to determine which components belong to which window. For example, a minimise, maximise and resize button with the same vertical co-ordinate are likely to belong to the same window. The position of the title bar will confirm this and provide the window title. The co-ordinates of the window can be calculated using the data available, and text appearing just below the title bar can be identified as menu items. There are exceptions, but for the purposes of this work, these calculations were sufficient. The user now has more information available and can navigate screen objects in a logical manner.

Scroll bar controls can be identified, and navigated to. The speed of screen analysis prevented a closer examination of methods to support the use of scroll bars. Note however, the work done in sonically enhancing scroll bars by [Brewster 94].

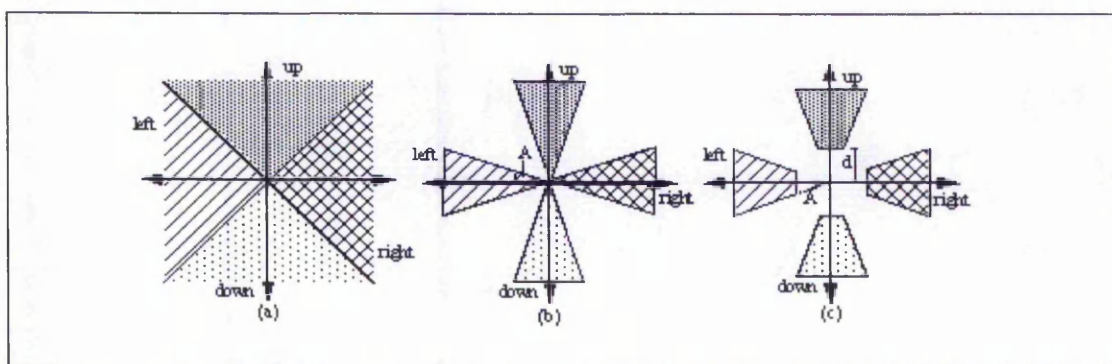
### 3.2.3.5 Object selection

There may be occasions when the user would require access to objects by object type. For example, select the type, say icons, then navigate to each in turn. This is a crude feature, likely to be inferior to alternative navigation methods, however it does provide an example of the use of lists, the manipulation of which is described later in this chapter.

### 3.2.3.6 Object hopping

The aim of this feature was to enable the user to navigate to the next object in a given direction. This could assist in giving information on the spatial relationships between objects. The obvious method for deciding which direction to choose is the keyboard, however, an attempt to use the mouse was made. You can compare mouse positions captured at frequent intervals and identify the movement as being either up, down, left or right.

Figure 3.2 illustrates this. The origin in each case represents the first captured mouse position. The second position captured determines the direction of movement chosen. The first diagram is the crudest method and suffers because it implies that any movement produces a direction. Brief testing of this method showed it to be poor. The second method is preferable, but also has disadvantages. Small movements of the mouse can register inaccurate directions. The third method produces a very usable method. Some tuning of angle 'A' and distance 'd' was made to produce a good result, but user preferences may vary and could be set.

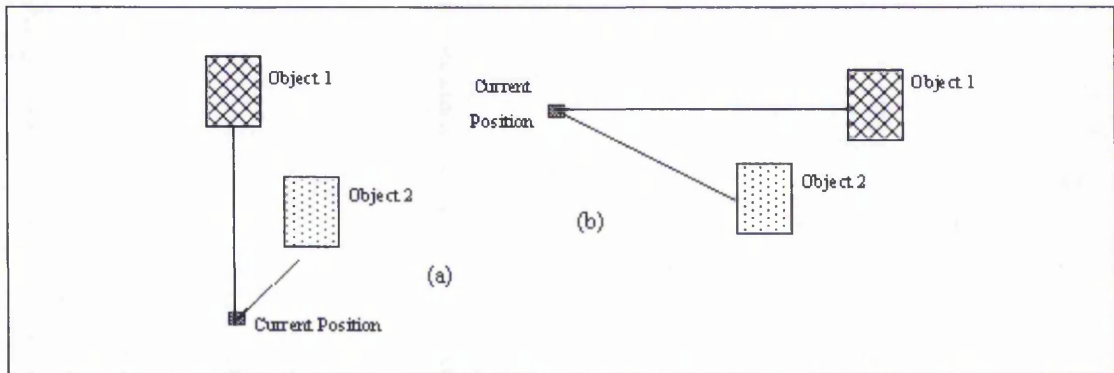


**Figure 3.2 Determining direction of movement of the cursor**

The next task is to determine which object is the 'best' object in the chosen direction. This is not as straightforward it appears. Figure 3.3 shows two situations where the choice of which object to navigate to may be difficult. In part (a), Object 1 could be used on the basis that it is directly above rather than above and to the side. However, Object 2

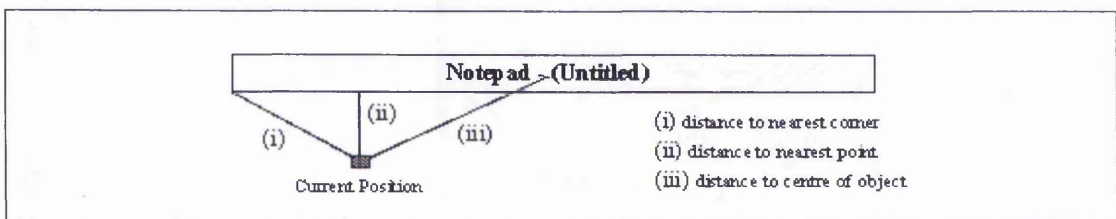


is closer. A similar example is shown in part (b), but the decision may be different in this case.



**Figure 3.3 Choosing the 'best' object to move to next**

You could justify moving to Object 2 in part (a) but Object 1 in part (b) because Object 1 in part (b) could be text. Choice may depend on direction and even object type. The current implementation would indeed make this decision and some experimentation showed this to be acceptable. An additional complication is that measurement of distance to an object can also depend on object type as illustrated in Figure 3.4.



**Figure 3.4 The possible measurements of distance between a point and a rectangular region**

The algorithm used attempts to take into consideration these factors and in many cases navigates to the most appropriate object in the chosen direction. However, there are cases when it doesn't, and this detracts from the feature. The feature, whilst it raised some interesting questions, has not been developed further because of the potential for confusion outweighing the potential benefits.

### 3.2.3.7 Navigation shortcuts

It is possible to navigate quickly to key objects. For example, the minimise button of the active window. This request was made by a user and added to the system. The number of key objects which you can define must be restricted due to the need for a new key combination for each. In general, window commands such as minimise, maximise

and close can be accessed using standard keystrokes and therefore Talking Windows does not need to duplicate this functionality.

An additional feature added was for when a user had located an object and then either moved the cursor slightly or forgotten the current object. Pressing a key combination would move the cursor to the nearest object and announce its type and label. Whilst this is a simple feature it does, on occasions, prove useful.

A search facility was also requested to allow the user to enter an object label and, if it could be found in the OSM, navigated to.

### **3.2.4 Using lists for selection**

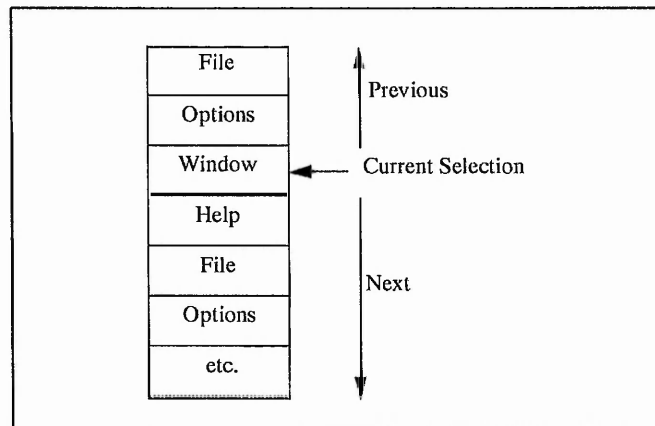
There are many occasions when information can be presented to the user in the form of a list, for example, menu items, running applications, etc. The user must then be able to determine what items are present in the list, and select one for manipulation.

Having implemented several features incorporating list selection, it has been observed that implementation of the list depends on the precise nature of the particular feature. There are two stages involved in the use of a list. One is the selection of one item from that list, the other is the manipulation of that item. For some features, the two stages can be combined, whilst for others, they must remain separate.

For example, consider the feature in which the user selects an application from a list. The user determines the contents of the list by selecting each item in turn. When the required application is found, it is run. In this situation, selection is separate from the manipulation of the item because the selection process is, in effect, moving a pointer through the list, revealing each item in turn. It is obvious that in this example, it is inappropriate to run each application whilst selecting the required one.

Features where both stages can be combined often include those involving cursor positioning. For example, the user may be presented with the menu items of a particular window in list form. Manipulation of an item will involve placing the cursor on the menu selection for the subsequent use of the visually impaired user. In this case, such manipulation can be combined with selection. The user will know once the required item has been announced that the cursor is then on that object.

Figure 3.5 shows a representation of a list of menu items, and how three functions are used in the selection of an item. In this case, manipulation involves moving the mouse pointer to the menu item itself, therefore this can be combined with selection.



**Figure 3.5 A representation of a list of menu items**

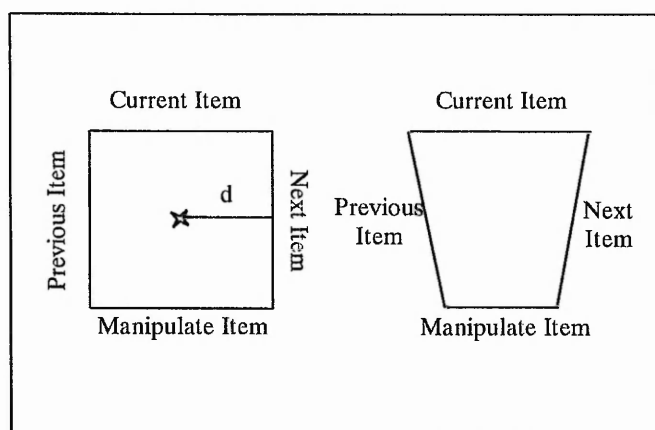
#### **3.2.4.1 Keyboard-based list selection**

For lists where combination of selection and manipulation is not possible, four basic functions are necessary, hence four key combinations - announcement of the current item (the item currently pointed at), selection of previous and following items, and manipulation of the current item. For combined selection/manipulation, only two combinations are necessary, for previous and next items. This can in theory be reduced to just one combination if, firstly, the number of items is small, and secondly, the list is cyclical. That is, once the end of the list has been reached, the process starts again from the first item, and vice versa. A list length of less than five items, with interruptible speech, ideally, would typically allow the use of one key. The feedback given at each selection/manipulation will depend on the feature. Further support would include key combinations assigned to functions to announce the number of list items or the complete list. In this case, a number could be associated with each item; the user would then be able to direct the system to simply select/manipulate that item.

#### **3.2.4.2 Mouse-based list selection**

The number of key combinations required led to consideration of alternative list manipulation methods. Inevitably this involved the use of the mouse. Could the mouse be used to select list items? If the user is able to determine the approximate distance and direction of mouse movement, then it can provide a number of 'switches'. For example, movement upwards would constitute one distinct action, and downwards another, etc.

Two implementations exist, both based on this principle, namely Direction Selection and Boxed Selection. In the former, if the cursor is moved to the left, the system selects the previous list item, to the right, next item, and upwards the current item. Movement downwards performs the required manipulation. In order to prevent unwanted manipulation, the conditions for a downward movement to be registered are more stringent. Boxed selection provides a different conceptual basis to the user. Metaphorically this equates to a person standing in a room and touching one of the four walls to select a particular function. Once again, the action required to manipulate the selected item can be limited, as illustrated in Figure 3.6. If the box is square, 'd' is adjusted to allow easy, non-random selection. The user can be forced to select the 'manipulate' direction three times consecutively if inadvertent manipulation is to be avoided.



**Figure 3.6 Selecting a list item with the mouse**

### **3.2.5 Maintaining the windows metaphor**

As has been mentioned previously, the OSM has been expanded to store information concerning windows present on the screen. Where there are overlapping windows the current methods used would need to be developed further, but in general enough information is available for the purposes required.

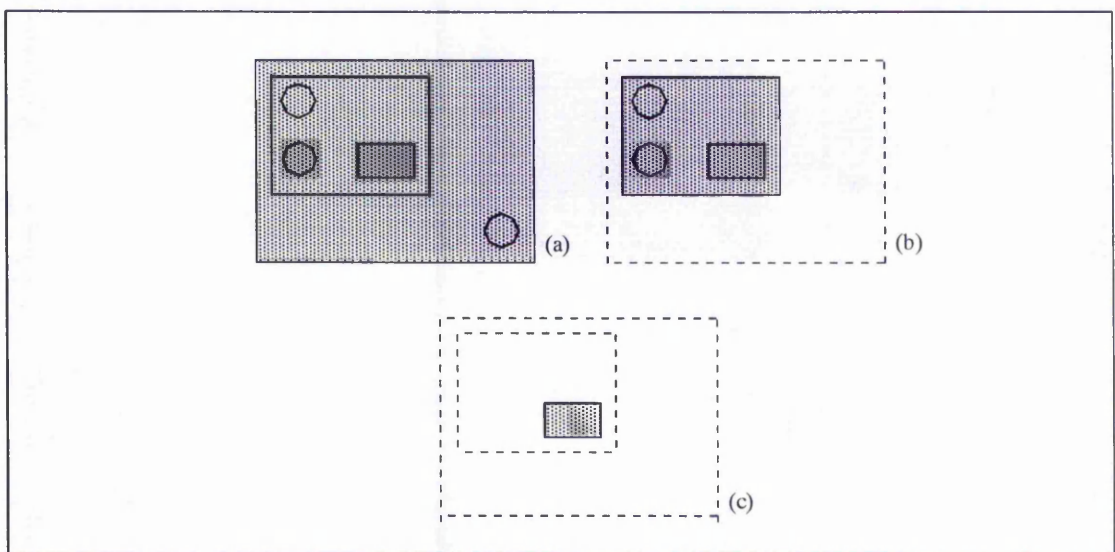
It is important to be able to present information relating to the windows present, as these are the fundamental logical unit for the GUI studied. The approach taken is to present the screen as a hierarchy. At the highest level, the screen consists of a number of windows, which may or may not be displayed as icons. The next level is the window itself. This itself has content, which may include child windows or icons. A window can



be broken into its constituent parts, which may include controls such as menus and toolbars, text regions and graphics.

This segment of work aimed at allowing the user to navigate this hierarchy. The first step is to present a list of high level windows. This equates to a list of running applications. The user selects a window, for example, from a list. That list would commence with the active window. This window can then be explored in further detail. For example, what is its title, and does it have child windows? If it does, these can be traversed.

Once a window of interest has been located, the cursor can be constrained to movement within this window. This reduces the complexity of the screen. Figure 3.7 illustrates this concept, with shaded areas indicating the scope of the cursor as the user hones in on an area of interest.



**Figure 3.7 Breaking the screen into smaller, more manageable areas**

A variety of features exist to allow the user to determine the content and location of these regions, and manipulate them. These are described later.

### **3.2.6 Providing access to text**

It is essential that a visually impaired person is able to access text. This includes the reading and editing of text, and the ability to determine text attributes and layout. Therefore, it is necessary to provide appropriate features on the Talking Windows system.

The initial recognition-based OSM was unsuitable for some of the required features. All text was recognised sufficiently using OCR techniques, however, it was stored in such a way as to make the accessing of units of text smaller than a 'line' (i.e. word, character) difficult. It is important that the user has sufficient control over text, otherwise effective manipulation will not be possible.

A number of facilities for reviewing text are available. The user may choose the unit of text, which may be, for example, a paragraph, sentence, line, word or character. The point at which reading should commence can be specified, for example, start of the current text area, start of the current line or caret point. A direction is also applicable, so you can choose to read the previous line or word, for example.

For editing text, a number of features are available. Being able to ascertain that the text entry point is in the correct location is important, and the reviewing features described above provide this information. Characters may be spoken as they are typed if required, although often the user will switch off this facility. The text review features will again be used after typing to verify that the edit has been made as required. A useful addition to the range of features available would be support for cut and paste text editing.

The user must be able to interrogate the system for text layout information, including line, paragraph and column counts, justification status and position of objects. This type of information can be incorporated into the editing and reviewing process, for example, by sounding a note whenever the text wraps onto a new line. The speed limitations of the screen analysis method prevented a more detailed study in this area.

Text attributes such as font colour and position must be available to the user, either in response to a direct request or by incorporating this information into the text review functions. For example, bold text could be announced in a louder voice. The first word of each paragraph could be preceded by a longer pause than usual. This kind of information can be considered as 'background' information, which is available to the user, but does not interfere directly with the primary objective of presenting text content.

On occasions, text may contain embedded graphics. When this occurs, the user has the option of investigating that graphic in detail using alternative features.

### **3.2.7 Status information**

As auditory information is transitory, there are a number of pieces of information which the user may need reminding of. For example, the user may want to know the current window title, or the currently playing sound map. There are a number of flags which may be set to determine how information is presented. For example, unit of text may be set to paragraph, line, sentence, etc. In order to enable the user to keep track, this kind of information must be available when requested. A sighted user may typically look at the status bar for this.

# **Chapter 4 Spatial Information Presentation using Sound Maps**

## **4.1 Sound map concept**

Sound maps provide the visually impaired user with auditory support for a number of functions, in particular, display screen navigation. The sound heard by the user is a reflection of the position of the cursor at that time. Sound maps are ideally operated with the mouse, although keyboard alternatives are, of course, available. In fact, sound maps can be considered to support the use of the mouse in performing other tasks.

### **4.1.1 Implementation**

Cursor position is monitored frequently, and a sound played depending upon this position. The sound used depends on the particular sound map. For the sound heard to accurately reflect cursor movement, cursor position must be checked frequently, typically one hundred times per second, and the sound sample used must be short in duration. The result is a warbling effect.

### **4.1.2 Types of sound map**

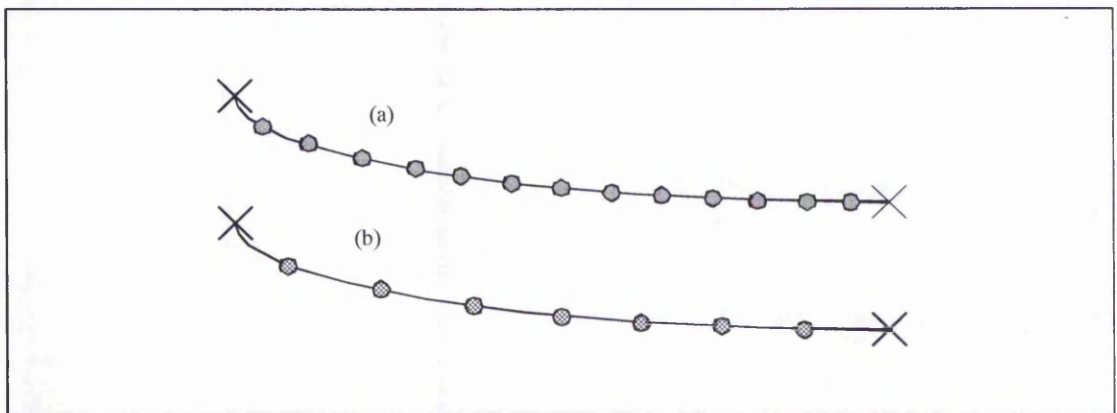
A number of sound maps have been implemented, each either performing a slightly different task, or providing an alternative for the same task. There are several 'basic' sound maps which are based on one concept and perform one task. Two or more such sound maps can be combined to form a 'complex' sound map, allowing a greater amount of information to be received by the user.

### **4.1.3 Controlling sound maps**

Sound maps can of course be switched on and off as required by the user. However, in order to reduce the possibility of irritation caused by the sounds heard, sound maps can be switched on and off automatically. When a sound map is on, but the cursor has not moved for a number of seconds, the sound heard ceases. It starts again once the cursor moves. A period corresponding to approximately five seconds mouse inactivity was found to give a suitable effect.

There is an option of playing a single sound to completion or allowing sounds to be interrupted by each subsequent sound. The latter method produces a warbling effect which is more pleasant to listen to than the bolder effect of complete sounds. The main advantage though, is the way in which this enables the sound to reflect the current cursor position, as illustrated in Figure 4.1. Each point indicates a sound played. In part (b), a sound is played to completion, whereas in part (a), sounds are interruptible.

Note that the precise warbling effect is very much dependent on the particular sound card used and the sounds played, but in general the warbling gives a very good effect, and is an accurate reflection of cursor position and travel.



**Figure 4.1 The path of cursor movement - sound heard accurately represents position**

## **4.2 Sound maps for determining position**

### **4.2.1 Edges sound map**

When the cursor hits an edge of the screen, or a region, the user should be informed, including which particular edge. This can be done using both speech and sound. Using speech can take slightly longer, although the user does not have to make the connection between the sound heard and its meaning. An advantage of using non-speech sounds is that the ability to present screen edge information will be necessary in complex sound maps when non-speech sounds will provide background information concerning screen layout.

The cursor position is monitored as described earlier, and whenever the co-ordinates are on an edge of the region specified as the domain of the sound map, a short non-speech sound will be played. A different sound may be used for each edge, although it is hoped that if the user simply knows that the cursor is on an edge, which particular edge



will be known from previous and following movement of the cursor, typically by use of the mouse. One sound for all four edges would reduce the number of sounds needing to be found, and learnt by the user. A further sound may be needed for when the cursor is in a corner of the region.

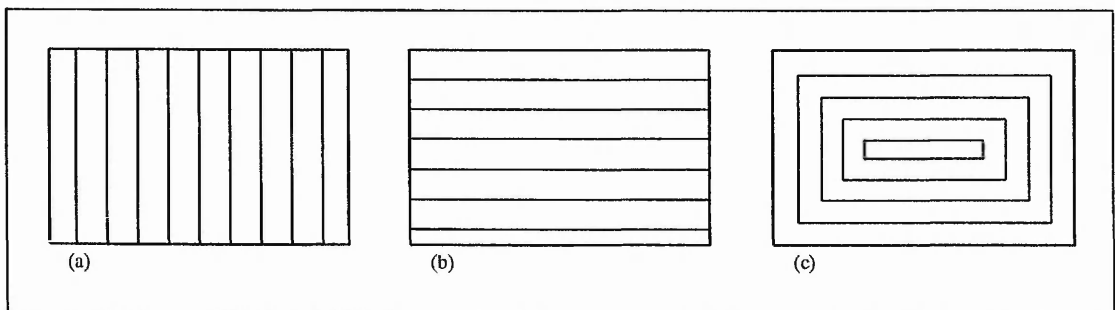
For the initial implementation, four sounds are used. A high-pitched sound indicates the top edge, whilst a low note indicates the bottom edge. It was hoped to continue this link between sound and edge by forcing the left edge sound to be heard only through the left speaker or headphone, and the right edge sound through the right. The soundcard used during development of this sound map could not achieve this result and so two different, distinct sounds were used instead.

If the cursor lands on an edge, the sound is heard continuously until the cursor moves off the edge. If the cursor 'bounces' off an edge, a sound is always played however short the period of time during which the cursor is on the edge. This is due to the frequency with which the cursor position is checked.

Alone, this sound map is not particularly useful, however it will be used in conjunction with other sound maps, at which stage its use will be clearer. It is implemented as a single sound map for testing and evaluation purposes.

#### 4.2.2 Volume sound map

This sound map uses the change in volume of a sound as the cursor moves in order to present the position of either the cursor itself, or objects hit by it. It is necessary to define which parts of the screen will cause each available volume of sound to be played. This is known as the 'Screen Map'. Some possible screen maps are shown in Figure 4.2.



**Figure 4.2 Screen Maps**

A different sound will be played in each zone shown. For the volume sound map, one sound is played at a number of volumes. These volumes may be so close to each other

that the volume changes smoothly as the cursor moves across zones. Alternatively, each adjacent volume should be distinct from the last. Evaluation with users led to the latter approach being favoured on the grounds that you can count zones as you move and therefore determine size of movement.

This limits the number of zones available. Volume can range between silence and irritation, and there are only a small number of distinct volumes between these limits. The choice of sound has little influence on the range, although quality of reproduction can reduce the gap between two adjacent volumes.

Experimentation has shown that around five zones can be used without using a loud volume. Increasing the volume as the cursor moves up the screen provides a logical link between sound change and position, as high volume represents a position high up the screen. The overall effect of the sound map is not particularly pleasant for the listener, however, the concept is very simple to describe to users and can serve as an introduction to the sound map principle, and therefore, development of this feature has continued.

#### **4.2.3 Piano sound map**

The volume sound map described above is limited by the small number of distinct sounds available. In an attempt to find a set of sounds which are both distinct and related to each other, the use of the musical scale was investigated. An obvious starting point was the range of notes found on a piano scale. A short burst of each of fifty consecutive piano notes was sampled into a sound file for later playback by the Talking Windows system. This allows the screen map to use up to fifty sounds. Initially a screen map similar to the volume sound map was implemented. The differences with the piano sound map are the use of change of sound rather than the change of volume of sound, and the increased number of sounds available.

Depending on the direction chosen, the sound heard changes from high to low as the cursor moves either from right to left or from top to bottom. Both of these directions have been chosen deliberately. A high note at the top of the screen is easily memorable, whilst a high note on the right of the screen and a low note on the left is the similar to the layout of notes on a piano.

Note that the difference between two consecutive notes is sufficient for the user to be able to distinguish between them, allowing the user to count the number of notes/zones between two points.

Implementation of the piano sound map in either horizontal or vertical form raises a number of important questions. This is mainly due to the need to be able to apply the map to any sized region. The problems are concerned with the number and width of zones used; a description of how problems arise follows.

Ideally, three conditions would be met:

- i) zone width is constant irrespective of the size of region concerned,
- ii) consecutive notes must be used, no note must be excluded,
- iii) first and last notes of the scale correspond to the first and last zones.

It is impossible to satisfy all three conditions, therefore, the least important must be relaxed. In order to decide which is the least important, the benefits of satisfying each will be described:

i) constant zone-width: if the user knows that the zone-width is equivalent to a fixed number of pixels, then accurate calculations may be made as to the separation of points, the distance of movement, etc. By insisting on this zone-width being constant whatever the size of region involved, only one zone-width needs to be remembered throughout.

ii) consecutive notes: if the user has sufficient musical ability, the precise note can be named. If no notes are missing, then the distance between two notes heard may be calculated based on the number of notes and the zone width. This is similar to the above condition, except that no counting is necessary, an advantage because counting zones requires more concentration.

iii) end points: if the user hears a note that can be distinguished and compared with either the first or last note in the range, then the distance from the edge of the region may be calculated. If one end zone corresponds to the first or last note, then all comparisons can be made between the current point/zone/sound and that edge. If neither edge is fixed, then no position can be estimated relative to the edges, unless the cursor has been moved to the edge and the corresponding sound noted.

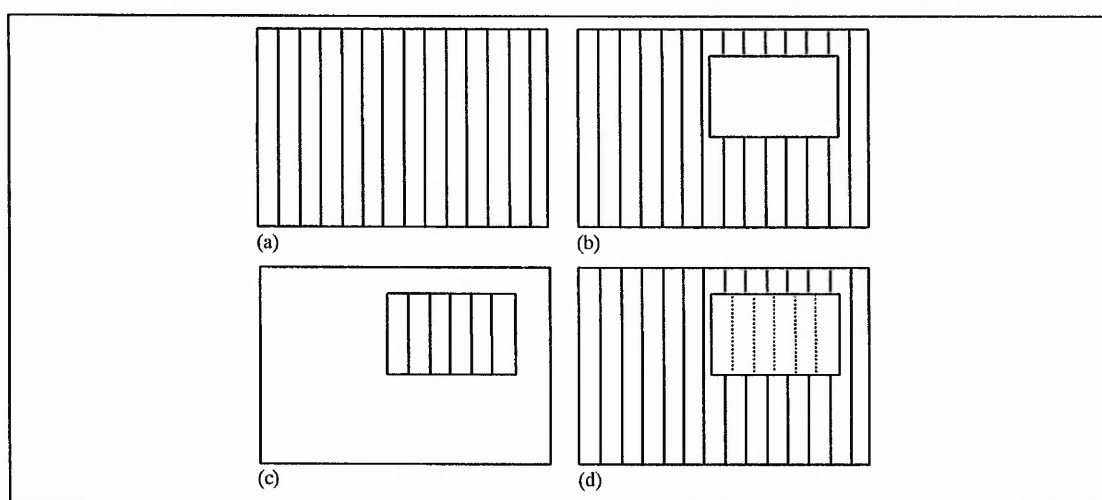
Initial implementations missed out some notes, which, to the non-musician was hardly noticeable. However, on evaluating with blind users, this was found to be an unsuitable solution. Some users could detect the absence of notes and found it to be a distraction. An alternative method was therefore sought. This involved relaxing the fixed zone size



condition, with some zones being larger than others. However, this was implemented in such a way that in most cases, zone size would only vary by one or two pixels. The work was carried out on a 640 by 480 screen; higher resolutions would improve the solution.

This solution therefore allows counting of zones encountered between points and an estimate of distance between the points. Zone size varies depending on the size of the region in which the sound map is playing, but an announcement of zone size can be made available.

Figure 4.3 illustrates the use of the piano sound map with screen regions. Part (a) shows the sound map used for the entire screen. In part (b), no sound is heard when the cursor is in the window region. This enables some determination of position to be estimated. Alternatively, part(c) illustrates the sound map playing only when the cursor is in the region. This may occur when the user has selected a region as described in the previous chapter, and has perhaps constrained the cursor to movement within that region. Finally, part (4) illustrates how the region can be distinguished from its surroundings. Either a different sound map can be used within the region, or, the same sound map can be used, but at a different volume. Therefore when a slightly higher volume is heard, the user will know that the the cursor is within a window region. Position of the cursor can be identified from the piano note heard, and type of region by the volume. This idea is developed with the concept of complex sound maps.



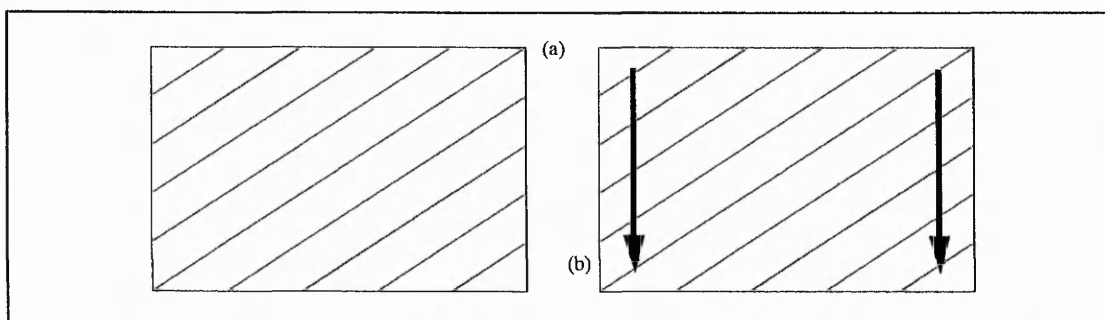
**Figure 4.3 The piano sound map**

#### 4.2.4 Diagonal sound map

The piano and volume sound maps previously described are one-dimensional - without switching between horizontal and vertical versions, position can only be estimated in one direction at a time. There are a couple of things which can be done to change this. One is to alter the sounds used in another way. For example, the piano sound map could be modified such that whilst the note played changes, say, from left to right, the volume of each note could increase as the cursor moves up the screen. Therefore, horizontal position could be determined by note change, and vertical position by volume change. Similarly, change in pitch in one direction could be added to the volume sound map.

A second solution involves allowing very quick switching between horizontal and vertical orientation. The way in which orientation is changed depends on how the user is propelling the cursor about the screen. If, as intended, the mouse is being used, then a click on the right mouse button will change the orientation. Another press will change it back. This is mainly for use with the piano sound map, due to its greater popularity at the early evaluation stage. Allowing quick orientation switching provides the ability to evaluate the piano sound map in terms of which direction is preferred, and the suitability of the switch.

There is a further option for providing position in more than one dimension. This is implemented as the Diagonal Sound Map. This uses a different screen map. Rather than horizontal or vertical zones, diagonal zones are used, as shown in Figure 4.4a.



**Figure 4.4 Screen map for diagonal sound map**

Here, the sound changes as the cursor moves from top to bottom in a straight line, and it also changes from left to right. Initially, change of volume was employed, but once again, due to the popularity of the piano sound map, the piano scale concept was also implemented. The implementation is slightly more complicated than for horizontal and

vertical sound maps, due to the need to determine which diagonal zone the cursor is in at any time.

Firstly, zone width is calculated based on the length of the diagonal from corner to corner. This length is divided into 50. Note that whilst the effectiveness of this feature is in doubt, the consideration given to the number and size of zones in the piano sound map was not applied. Now, for each cursor position gained from the monitoring process, its diagonal distance from the top left hand corner is calculated. By using this distance, and the zone-size, the zone in which the point occurs can be determined, and the appropriate sound played.

Whilst the sound does change when moving in either direction, the effectiveness of this may be reduced by the differing ranges of sounds heard for each horizontal or vertical sweep. For example, consider two movements from top to bottom of the screen; the first down the left edge, the second down the right (see Figure 4.4b). For the first, the sound heard at the top of the screen will be the highest note on the scale. The sound heard at the bottom will be about midway through the scale. For the second movement, the sound heard at the bottom will be the lowest note on the scale. This will be the same for horizontal movements also. This will make estimating position difficult without acquiring context, that is, moving the cursor in order to find surrounding sounds and nearby edges. It was therefore expected that this map would require more experience and perhaps more cognitive effort in order to be effective.

The sound maps described in this section are intended to give an impression of cursor, and hence object, position during navigation. Later, their use in combination with other sound maps will be described. There are two more categories of basic sound map; target finding, and cursor support.

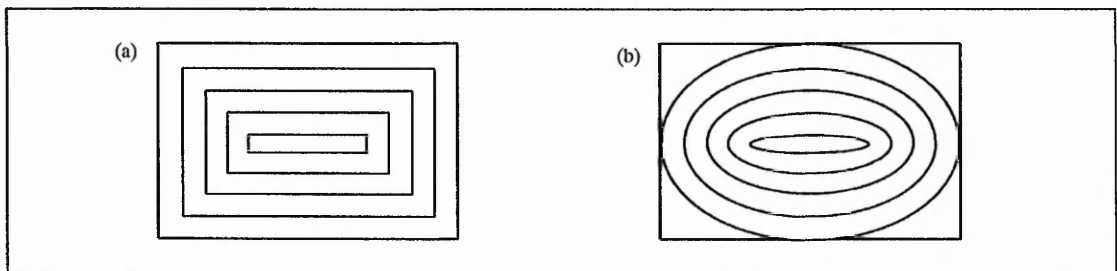
### **4.3 Sound maps for locating a point**

The aim is to use sound to guide the user towards screen objects or locations. Again, a screen map is created to define which sounds to play where, and this is used to provide a target finding sound map.

### 4.3.1 Target finding sound map

The aim of this sound map is to guide the user to a point on the screen, in this instance the central pixel. Applications of such a feature include guiding the user towards alert messages or status bars.

For this type of sound map, changes in sound are used to guide the user towards something, therefore the screen map must reflect this. Initially, change in volume was used, as in the volume sound map. It has been found to be more pleasant if the user aims towards quieter sound, rather than away from it. This is due to the irritating nature of high volume warbling sounds as described earlier. In order to guide the user towards the centre, then, loud sounds will be played near the edges of the screen, and volume will decrease as the cursor moves towards the centre. This leaves two obvious screen maps, as shown in Figure 4.5.



**Figure 4.5 Screen maps based upon rectangular and circular zones**

There is also the question of whether to keep zone widths constant or to have narrower bands towards the centre for more accuracy.

The choice of shape of band is not expected to affect the effectiveness of the feature, therefore rectangles were tried first. With the initial choice of changing volumes of sounds rather than the sounds themselves, the number of bands is limited according to the range of volumes used.

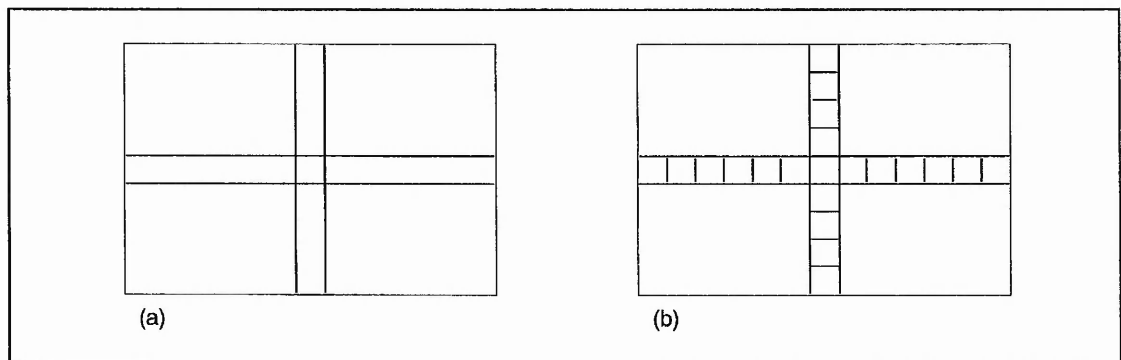
To determine the sound played, the cursor position is checked, and its distance and direction from the centre determines its zone. When the cursor is within a few pixels of the centre, a very different sound is used to indicate that the target has been reached.

In order to allow sound maps to operate in the background, volume may be reduced compared to that of speech. This is not so appropriate for either the volume or the target finding sound map, where the lower volumes must be heard. This, together with the

previously described limitations of volume changes, suggests an alternative approach, such as using piano notes to direct the user to the target.

### 4.3.2 Cross hairs sound map

The aim of this feature is once again to enable the user to locate a target, in this case the centre of the screen. However, a different screen map is used. Here, the target is at the intersection of a horizontal and a vertical strip of sound, as shown in Figure 4.6a.



**Figure 4.6 Screen maps for the cross hairs sound map**

Here, the only sounds played as part of the sound map are heard when the cursor is on the same horizontal or vertical level as the target position. The aim is for the user to move the cursor until a sound is heard. By using the mouse this would be a quick process. The user then knows that the target is directly above, below, left or right of the current position. The actual sound heard will determine which of these is the case, and the user will then simply move on that level in the appropriate direction until the target is reached. Keys would be the best method once sound has been located.

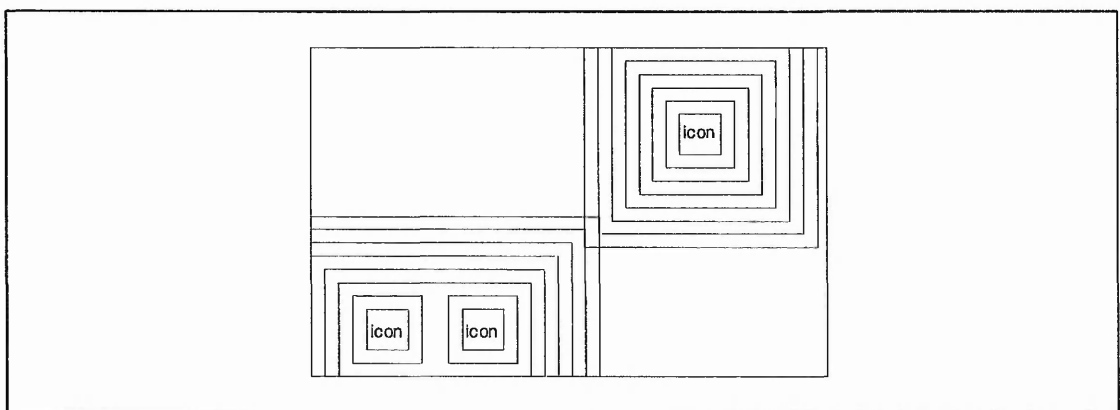
There are two ways in which the choice of sounds can determine the direction in which the cursor should move to reach the target. For one method, four sounds are needed, one for each direction, therefore it is sensible to use the four sounds used to represent screen edges as described for the Edges Sound Map. So, for example, the high pitched sound used to indicate top of screen would now mean either move up to hit target or cursor is directly above target. The choice would depend on which makes more sense to the user.

The alternative method would be to use a sound change for each of the four sections (Figure 4.6b). For example, above the target, volume could decrease as the cursor moves downwards towards the target. Similarly, the notes from the piano scale could be used.

### 4.3.3 Icon sound map

This is intended as a background sound map giving an impression of where certain objects are located on the screen, and allowing the user to move towards them. The implemented version uses the icon object type; the user is guided towards any icons on the screen.

This uses the same principle as the target finding sound map, with a change in the sound heard directing the cursor. Initially, change in volume has been used, with five distinct volumes allowing five zones around each icon. The same sound is used for each icon; the aim is to guide towards an icon, not any particular one. The situation becomes interesting when two or more icons are close together, causing the zones around them to interfere with each other as shown in Figure 4.7.



**Figure 4.7 Screen map for the icon sound map**

An advantage of the use of a limited number of zones of a small width is that there is plenty of screen area where no sound related to the sound map is heard. This enables the user to only hear a sound when the cursor is fairly close to an icon. However, often icons are close together so interference of zones will occur. The method of implementation resolves any problem that may occur.

Firstly, the cursor position is checked. Then the distance between the cursor and each object is measured (since icons are generally stored as square objects, distance to centre is acceptable). The shortest of these distances is taken, and based on the chosen zone size, the zone of the cursor relative to that icon is calculated. A sound or no sound is played accordingly, and a new cursor position checked, etc. The user simply moves the cursor until a sound is heard. The user then knows of the nearby presence of an icon, and can follow the changing sounds until the icon is located.

## **4.4 Sound maps for supporting cursor movement**

The final group of basic sound maps provides the user with information as to the distance and direction of cursor movement.

### **4.4.1 Directional sound map**

Very few blind people use a mouse, mainly due to the visual nature of feedback and the lack of any non-visual support. This feature addresses these issues by providing information as to direction of movement. An inexperienced mouse user could use this feature to ascertain that the cursor is moving in the expected direction.

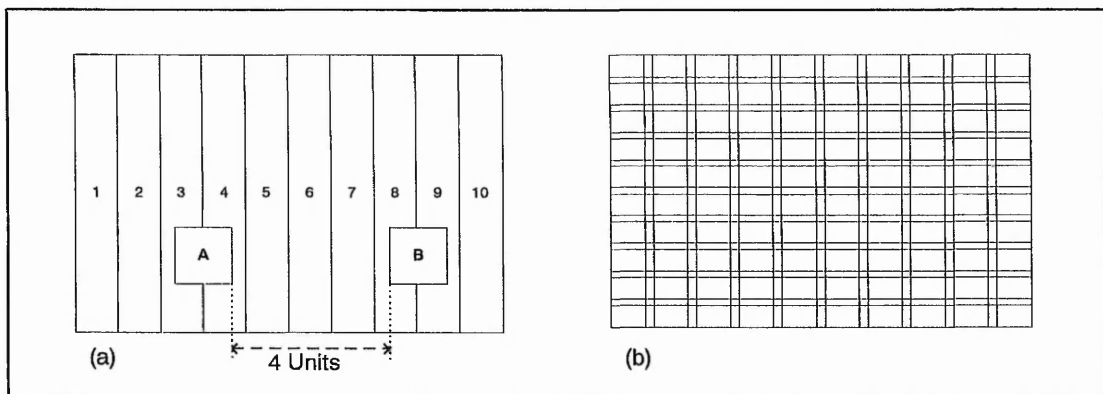
The feature works by monitoring the direction of cursor movement and playing a sound which reflects this direction. Therefore, four sounds are necessary, one for each direction to be reported. Once again, the same sounds as those used to represent screen edges may be used. Therefore if the high pitched sound is heard, the cursor is moving upwards, for example. Some further conditions are necessary. If the cursor is moving diagonally, no sound will be played. If the cursor is not moving, again no sound will be played. If only strictly horizontal or vertical movement were to count, coherent sound would rarely be heard when using the mouse to move the cursor. Therefore, movements near the horizontal or vertical will register as a legal direction.

This is implemented by comparing the position of the cursor with its position a fraction of a second later. The angle between the two can be calculated and if it is sufficiently close to the horizontal or vertical, the appropriate sound is played. Experimentation has led to a satisfactory loosening of the horizontal and vertical condition.

### **4.4.2 Grid sound map**

The aim of the grid sound map is to provide an unobtrusive means of determining the distance of movement of the cursor, and to some extent, the direction. Therefore, if the user is performing a general navigation task, hitting and moving between screen objects, the distance between objects may be estimated using the grid sound map playing in the background.

The grid sound map is based on counting units between objects or points on the movement path of the cursor as illustrated in Figure 4.8a. The screen map for the horizontal grid sound map shown in Figure 4.8b indicates how this can be achieved.



**Figure 4.8 The basis for the grid sound map, and its screen map**

Each bar represents a strip of screen where a sound is played. When the cursor is moved over a strip, a non-speech sound is played. The user can then count the number of strips passed over between two points. By knowing the distance between strips, the number of pixels between objects may be estimated.

There are two measurements to be decided, width of strip, and distance between strips. Experimentation with these quantities yields a suitable result. A width of seven pixels, and strips forty-three pixels apart (to give a total of fifty pixels) is a good starting point.

The sounds used are distinct from other sounds likely to be heard when this sound map is in use. One sound indicates a horizontal strip, the other a vertical strip. Intersections between strips can be treated as horizontal strips without adverse effect. Note that as a side-effect, the user can move the cursor in roughly a straight line, and feel the corresponding tactile effect of the hand and mouse that caused this motion, by keeping the cursor within a channel between two strips, and making adjustments when one of the boundaries is struck.

The basic sound maps described in this section are sometimes, alone, insufficient for presenting the volume of information required by the user. However, combinations of two or more may provide a richer non-visual environment. The following section describes such combinations, together with additional features aimed at allowing the user to discover more complex information.

## 4.5 Combining sound maps

An example of a combination of basic sound maps would be the addition of edge sounds to the piano sound map.



Now, the user is able to estimate position on the screen, and know which edge the cursor has hit. The aim of complex sound maps is to combine the better parts of the basic sound maps into more effective features.

One of the main aims of the work overall is to present locational and layout information to the visually impaired user. If the user is able to understand the layout of a visual screen, not only should its manipulation be simplified, but also interaction with sighted users enhanced. Complex sound maps are one way of allowing the visually impaired user to acquire such information.

Several layout sound maps were implemented in order to determine whether they could assist the visually impaired user in comprehending the layout of a typical screen. Such a screen could consist of a number of windows and icons on a desktop. A layout sound map uses different sounds or sound maps to denote the various objects and regions present. The position of objects or regions can be determined in relation to both the screen and other objects. Also, the size of a region may be discovered using a suitable sound map.

A screen has a border, as does each window region. Sounds can be played whenever the cursor is on the border of the screen or of a window. Each border can be treated as a strip of screen a few pixels wide, so that, as described earlier, when the cursor moves over the strip at a reasonable speed, a sound will still be heard.

Each window region, and the entire screen, has four corners, and on occasions it may be necessary to distinguish corners from edges. For example, consider the Edges Sound Map described earlier. Whenever the cursor is at a corner of the screen, no separate sound is used to indicate this, the user only knows that the corner has been reached when one edge sound changes to another.

For some sound maps, icons may be treated as a region of the screen. There is a reason for this. Consider a graphical document. This will include regions of text, and graphical symbols such as logos or diagrams. The visually impaired user will need to be able to determine the positions of text regions, the positions of graphical regions, and the positions of regions containing mixtures of text and graphics. Furthermore, the user will need to know the relationships between the regions, in other words, the layout of the document.

There are many similarities between GUI screens and graphical documents, and therefore the same methods for presenting layout information for a GUI screen can be used for a graphical document. For documents, a distinction between text and graphical regions may be made, due to the need to access the text being a basic goal. Mixed text and graphic regions can be considered as graphical regions for now. In order to simulate the situation using a GUI screen, windows can be used to represent text regions and icons to represent graphic regions. Whilst the layout sound maps on the Talking Windows system are applicable to the GUI screen, they could equally apply to a scanned-in graphical document.

The desktop or background is a further component of a screen, comparable with the blank areas of a document. Icons and windows sit on the desktop in a similar manner to text regions/graphics on a sheet of paper. Overall, the comparisons between GUI and graphics-based documents can be summarised as shown in Table 4.1.

<b>Graphical Document</b>	<b>Graphical User Interface</b>
Graphical Region	Icon
Text Region	Window
Background	Desktop
Paper Edges	Screen Edges
Paper Corners	Screen Corners

**Table 4.1 Comparing information sources**

Each GUI component can be represented using a basic sound map or a non-speech sound in order to build up a layout sound map. Four sound maps are described, with each being an adaptation of the last. Each has its merit though, with the first only having a few sounds to be learnt, whilst the last requires supporting keyboard-based functions mainly due to the increased number of sounds used. Note that the increased functionality of these features compared to the basic sound maps means that they can be used as a main feature, rather than in the background whilst the user is engaged in other tasks.

#### **4.5.1 Basic layout sound map (Figure 4.9a)**

For this sound map, no distinction is made between window and icon regions. On the edges of the screen, the edge sounds described earlier are played, using the edge sound map. The same sounds are also played on the borders of a window. The user can distinguish between the edge of the screen and the edge of a window by the fact that the cursor can move across a window border, but only hit a screen edge. When the cursor is

on the desktop area, a warbling sound is played to reflect this. The sound has been carefully chosen to sound pleasant and not irritating, as it will be encountered often.

Within a window, no sound will be played, providing a clearer distinction between non-window and window areas. Objects both within and outside windows will be announced as struck as usual. This sound map therefore only requires the learning of one new sound on top of the four now familiar edge sounds.

Whilst this sound map allows the user to distinguish between window areas and their contents and the desktop, the amount of locational information given is limited. A small enhancement is made in the next layout sound map.

#### **4.5.2 Piano-based layout sound map (Figure 4.9b)**

The difference between this and the previous layout sound map is in the treatment of the desktop area. Instead of a single sound, the piano sound map is applied. Therefore, whenever the cursor is on the desktop, a piano note will be heard, giving an indication of cursor position. There is still a clear distinction between desktop and non-desktop, but by using the piano sound map, further locational information can be obtained.

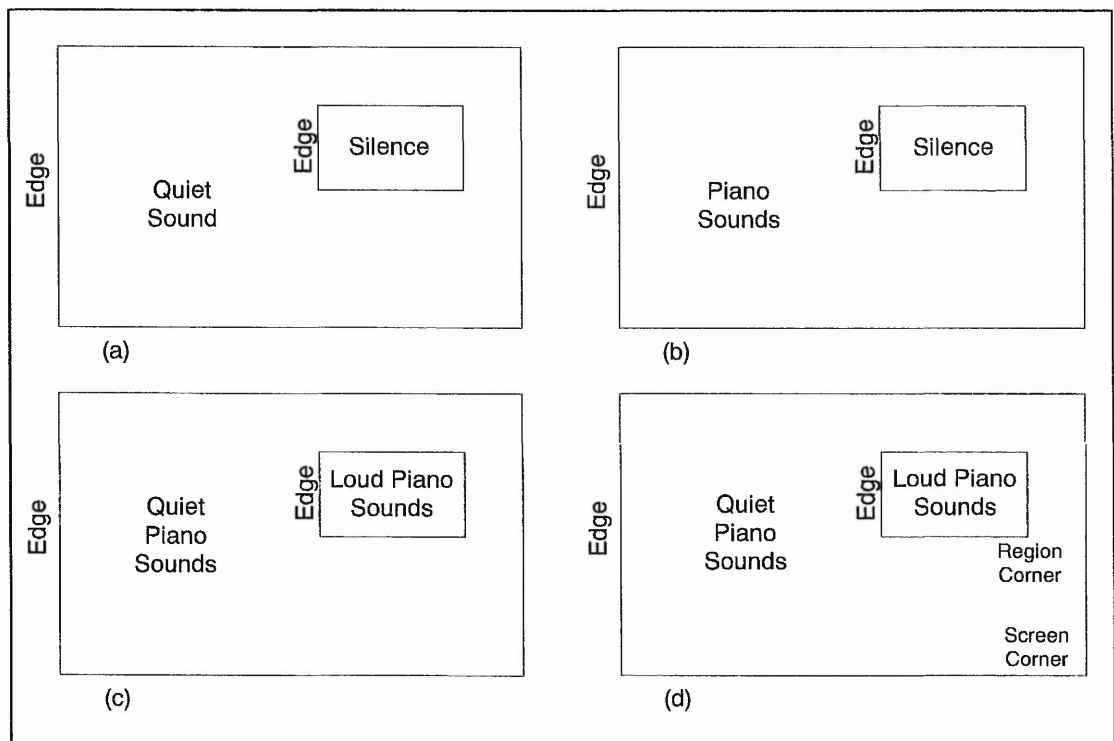
As the user moves the cursor from left to right, position may be estimated given the notes heard. When the cursor then hits the left border of the window, the user will know the horizontal position of the left edge of the window. Similarly, by determining the position of the right border, the horizontal position of the right edge may be found. These then give the width of the region. Although more efficient methods for this may be provided, in a situation where several windows are present, the ability to browse around using this kind of feature may ultimately prove more effective.

#### **4.5.3 Enhanced piano-based layout sound map (Figure 4.9c)**

The difference between this and the previous layout sound map is in its treatment of window areas. In the previous sound map, once the cursor was within a window, there was no feedback as to position of the cursor or of objects within that window. Ideally a layout sound map could provide this sort of information. One possibility is to apply the piano sound map within windows rather than on the desktop, however, this then loses the benefits that positional feedback on the desktop provides. What is needed is a cursor position sound map for both desktop and window areas, however, they must be distinct. The piano sound map is used for both instances, but within the region notes are played at a higher volume.

The application of the two piano sound maps can be achieved in two ways. One is to apply one sound map to the entire area, and when the cursor is in a window, play the relevant notes at the different volume. The second method is slightly more complicated, but may have its benefits. Once again, the desktop sound map will be played in such a way as to cover the entire screen with the exception of window areas. Now, rather than treat the window regions as subsets of this played at a different volume, the window region map will be a distinct piano sound map, with the sounds at the edges of each region being the first and last piano notes of the piano scale used.

This allows determination of cursor and object position within a window relative to that window. However, it does not allow estimation of the size of the window from within. The second method allows position of the cursor, and objects within windows, to be determined relative to the entire screen, and also allows the user to navigate within a region and gauge its size based on the number of zones within and the fact that zone size will be the same for the entire screen, both desktop and window regions.



**Figure 4.9 Illustrations of four layout sound maps**

#### **4.5.4 Comprehensive layout sound map (Figure 4.9d)**

This sound map can provide more information than previous layout sound maps by adding further sounds to extra screen components. In order to provide the user with

support for this added complexity, and to increase functionality, a range of keyboard-based functions are supplied.

The most significant difference between this and other layout sound maps is the introduction of two types of region, in preparation for the application of this kind of feature to graphical documents. As mentioned previously, windows will correspond to the text regions of a graphical document, and icons to graphical or mixed regions. This requires a means of dealing with the extra component. A further difference is in the treatment of edges. Now, edges of the window areas will be distinguishable from edges of icon areas, which in turn requires that screen edges are different still. The final difference is to add the facility to distinguish between a corner and an edge.

One consideration which may be made whilst designing this feature is that the user may have learnt each of the three preceding layout screen maps before this one, becoming acquainted with a further aspect for each one. It is therefore desirable to maintain a consistency throughout. For this reason, the desktop area continues to be represented by the piano sound map. The two remaining region types should use two distinct sounds or sound maps, preferably related in some way. The implemented feature uses two variations of the volume sound map, which, given the typically small regions involved, is not likely to be too irritating.

Rather than change the volume, as was used to distinguish between two piano sound maps, a change of sound is used. Therefore both use change of volume to indicate position, but each with its own sound. This introduces the burden of learning a new sound, however support is available for this as will be described.

For each edge (left, top, right, bottom), three sounds are needed for each type of edge (screen edge, text/window region edge, graphic/icon region edge). This is achieved using variations of the existing edge sounds, for example, changing volume and note length. To prevent further new sounds, corners are identified using a single sound, with context used to determine which corner is current.

A number of features support this sound map. An overview of the contents of the screen in terms of the components used by the layout sound map is given. This includes a count of the number of regions present, including the number of text (window) and graphic (icon) regions. The message should be terse, yet clear. The format used is a general purpose non-speech sound, followed by a message of type 'a plus b regions'

where 'a' is the number of text regions, 'b' graphics. This minimises the number of words needed, and implies also the total number of regions on the screen. The user only needs remember which region type is given first.

The user may also select each region in turn. Selecting a region involves announcing its type and an associated number, and placing the cursor in its centre. The user may cycle through the list of regions in either direction until the required region is found. Once selected the user may move to each corner of the selected region and request its distance from the corresponding screen corner. This will add to the non-verbal presentation of distance given by the sound map.

Similarly, each edge of the currently selected region may be chosen, and the distance between it and the corresponding screen edge announced. This can be used to confirm estimates made using the sound map. Once the user is familiar with the sound map, this feature may be discarded. This would be more in line with the aim of providing a fast, non-visual, non-verbal impression of screen layout.

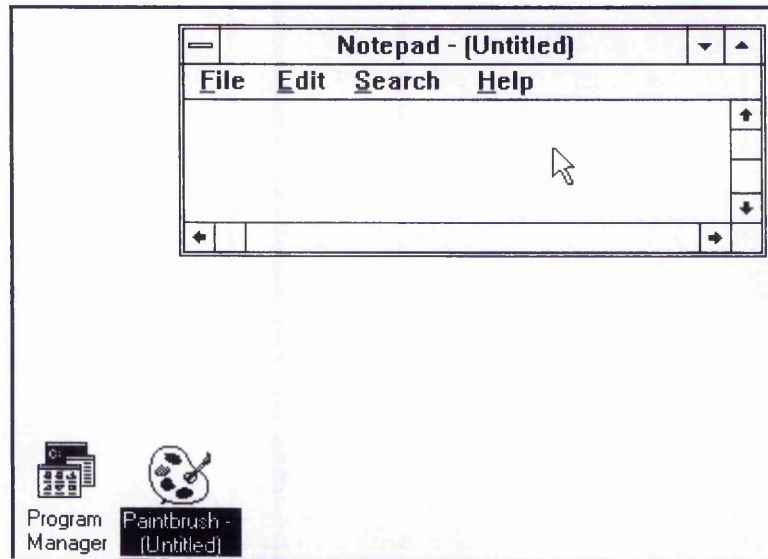
A final supporting feature, which also fits in this category, is intended to help the user who may be unsure what the currently playing sound represents. The component represented by the currently playing sound can be announced. This should initially help the user to learn the sounds used in this sound map. As an addition to this feature, the user may also request the position of the cursor when that sound is heard. Depending on the component, either the co-ordinates of the point, or a distance to some point will be announced.



# Chapter 5 Direct Presentation Methods

This chapter describes those features of Talking Windows which primarily use speech to present direct information to the user.

## 5.1 Description techniques



**Figure 5.1 An example screen requiring description**

A description of the screen shown in Figure 5.1 would be as follows:

“There is one window, title ‘notepad’, and two icons, program manager and paintbrush, on the desktop. The window is situated in the top right screen corner, occupying one third of the screen area. The two icons are in the bottom left corner. The window has four menu items, File, Edit, Options and Help.”

A set of features which describe the screen to the user have been implemented on Talking Windows. Descriptions should be concise, consistent and controllable. A balance between terse and verbose messages must be struck and messages must be of similar structure for any given screen. Furthermore, the user must be able to control the verbal information being given by means of selecting specific types of description, or by interruption of lengthy messages.

### **5.1.1 Window state description**

For any screen, the number and type of open windows present can be described. The Off Screen Model contains the type of each window on the current screen, therefore this feature simply involves making an announcement based on this data.

### **5.1.2 Desktop description**

A distinction is made between icons with which a name has been associated, and those which are not recognised (that is, icons which have not been added to the list of known objects). The former are referred to as icons, the latter as graphics. The desktop description announces the number of icons present on the desktop area. Any icon whose co-ordinates do not fall within those of any window present is said to be a desktop icon. Similarly, the number of desktop graphics is announced.

For each icon, its label and position are announced as part of this description. Position is given, in this case, as the percentage across and down the screen of the central point of the icon. As graphics do not have a label, they are associated with a number, and positions given as before.

An example desktop description would be as follows:

“There is one icon and one graphic. Icons include Program Manager at three percent across and ninety-five percent down. Graphics include graphic number one at twenty percent across and ninety-five percent down.”

### **5.1.3 List windows**

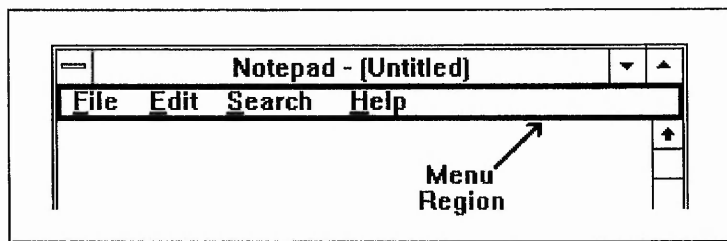
Windows open on the screen are listed using their title bar text as titles. For example, “window notepad”, “window file manager”, “active window control panel”. If there are no windows open, then this is announced.

### **5.1.4 Individual windows**

Any window on the screen can be selected and described in detail, including title, position, area, menu, parent window, child windows, child icons and child graphics. Position currently gives the percentage across and down of the central part of the window. Further position features are described later in this chapter. Window area includes the horizontal and vertical percentages of the window compared with the screen, and the surface area of the window as a percentage of the total screen area. Menu items



are counted and listed as part of the description. These are not stored as such in the object store, therefore the text lines which are in fact menu items must be deduced. Figure 5.2 indicates how this is done:



**Figure 5.2 Extracting menu items from text line store**

Text lines whose co-ordinates lie within the marked rectangle can be considered to be menu items. This has proved to be accurate for most windows. In narrow windows, menu items may occupy more than one line, in which case this method will not retrieve all menu items. In this case, the underlining of characters could help identify with identification.

If the currently selected window falls wholly within the co-ordinates of another window, it can be assumed to be a child of that window. This possibility is announced. Similarly, any windows whose co-ordinates are within those of the currently selected window can be described as potential child windows.

Icons and graphics whose co-ordinates are within those of the selected window are described in the same way as those on the desktop area.

### **5.1.5 Cursor description**

The position of the cursor can be described, either in terms of pixels across and down the screen, or percentage of screen across and down. The relative position can be given - is the cursor over desktop area, a window, an icon, a graphic, etc? The choice depends on which the user can assimilate the easiest. The option can be set prior to running the Talking Windows system.

### **5.1.6 Complete screen description**

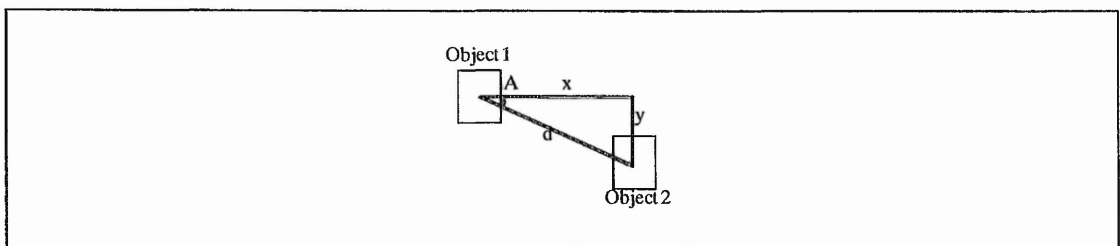
The above descriptions have been implemented as individual features, and as a single, comprehensive screen description function. This single feature can be used in a passive manner, with the user simply listening to the whole description. Due to the time that this

takes, and the difficulty in assimilating such a large quantity of information, it is more likely that an active form of manipulating this information will be required.

Firstly, if the user wishes to abandon the description, an interrupt function enables this. Once the user has heard sufficient information, the description can be stopped. A second requirement is the ability to pause the description, for example, to assimilate information before continuing.

### 5.1.7 Separation of two points

A further description which can be given is that of the relationship between two points. Consider a screen containing two icons. How can the user be informed of the position of one relative to the other? The position of an object will often be given relative to a point on the screen, usually a fixed point, typically the top left corner. This allows the user to gain an impression of where objects are on the screen, and thereafter use that information in tasks such as re-locating that object, adding further objects, etc. A feature which uses similar methods to give position of an object relative to other positions will allow similar tasks to be performed, and add to the quantity of information available to the user for determining screen layout.



**Figure 5.3 Separation Of Two Points**

Consider the two objects depicted in Figure 5.3. To describe the relationship between the two, it is necessary to either give the distance 'd' and an angle, for example, 'A', or, to give distances 'x' and 'y'. The latter appears more straightforward, being a simple 'across and down' concept. Furthermore, two distances, as opposed to one distance and one angle, should be more easily assimilated.

There are two steps which the user must take in preparation for this feature. The first is to 'select' the first object, the second to 'select' the second. Their separation can then be given. There are two formats for the message, terse, and descriptive. An example of the descriptive message would be as follows:

“Point two is fifty pixels right of, and twenty pixels below, point one”.

The equivalent terse message would be like this:

“fifty twenty”.

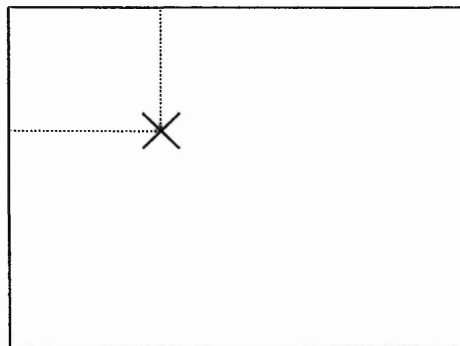
Positive values imply ‘right of’ and ‘below’. The use of positive and negative is relative to the top left corner, and allows a very short message. This would only be of value to experienced users who would know what this undescriptive message meant.

## 5.2 Attribute presentation

This section of work describes the presentation of position, size and shape of objects, and the position of a point.

### 5.2.1 Position of a point

There a number of ways of presenting position of a point on the Talking Windows system. The first involves describing how far across and down the screen the point is, as shown in Figure 5.4. The most precise unit used is the pixel. There are instances when such precision may be useful, but more meaningful units are available. Real distances may be given, in either inches or centimetres. An alternative is to calculate a distance as a percentage of the appropriate screen dimension. The user is able to specify which unit is preferred.



**Figure 5.4 Position of a point**

A second method is to give a ‘rough’ position. This gives the general area of the screen in which the point is. The screen is divided into nine zones, as shown in Figure 5.5, and the zone in which the point is announced.

Top Left	Centre Top	Top Right
Centre Left	Centre	Centre Right
Bottom Left	Centre Bottom	Bottom Right

**Figure 5.5 Position of a point - descriptive version**

A third method is an adaptation of this. The aim is to give more precision and to reduce the time taken to speak the message given. The same 3x3 grid is used, but each zone is associated with a number, these being based upon the number pad arrangement, as seen in Figure 5.6a.

(a)

7	8	9
4	5	6
1	2	3

(b)

7	8	9
7 8 9 4 5 6 1 2 3	5	6
1	2	3

**Figure 5.6 Position of a point based on a grid**

Now the number is announced compared with a short description in the previous method. To increase accuracy, a second number can be given. This represents the position of the point within the zone implied by the first number, based on the same layout. So, for example, all that need be heard in the situation illustrated in Figure 5.6b is 'four, six' (the figure in bold print indicates current position).

The final method available simply gives the position of the point relative to screen objects. A typical application of this is in describing cursor position. The point in question may be on an icon, a graphic, a window, or the desktop area. This is determined and announced.

## 5.2.2 Position of a region

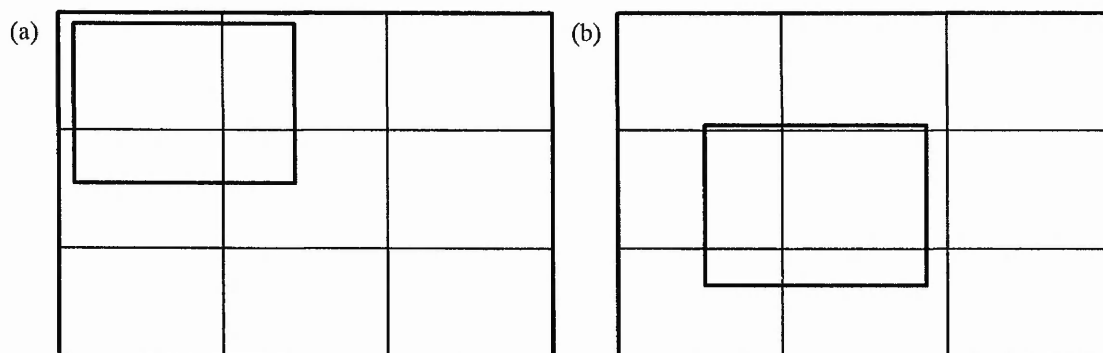
This section describes methods available for determining position of a region. Whilst the position of any point of a region can be given using the above methods, this only gives a reasonable impression of region position for small regions.

### 5.2.2.1 Edges of a region

The distance of each edge of the region from its corresponding screen edge can be given using pixels, inches, centimetres, etc. This involves four separate measurements, and so assimilation becomes difficult, and the time taken to present information is longer than ideal. However, the ability to pick each individual measurement could be useful at times.

### 5.2.2.2 Rough position

This is similar to the rough position of a point, however, for a region, the zone in which the region appears is not always clear, as Figure 5.7 illustrates.



**Figure 5.7 Complications in describing region position**

When a human looks at a window on a screen, say, you can describe its position in a descriptive manner, for example, in Figure 5.7a, you would say that the region is in the top left corner of the screen. It is not always so easy to describe other situations, for example, Figure 5.7b, and especially difficult to establish rules by which the system can make such a decision.

A method has been implemented which gives such a description. It is based upon the positions of the top left and bottom right corners of the region. The screen is divided into 30 zones as shown in Figure 5.8. The zones in which the two corners appear are then calculated.

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30

**Figure 5.8 Zones used to determine ‘rough position’ of a region**

For each possible combination of two zones, the most apt description of the position of a region with those corners is recorded in a table. There are 315 such combinations and 14 descriptions, for example, “top right”, “upper centre”, “almost all”. By using fewer than 30 zones, the increased zone-size could reduce accuracy of description. For a larger number of zones, the number of combinations would rise dramatically, requiring time-consuming consideration of the best fitting description.

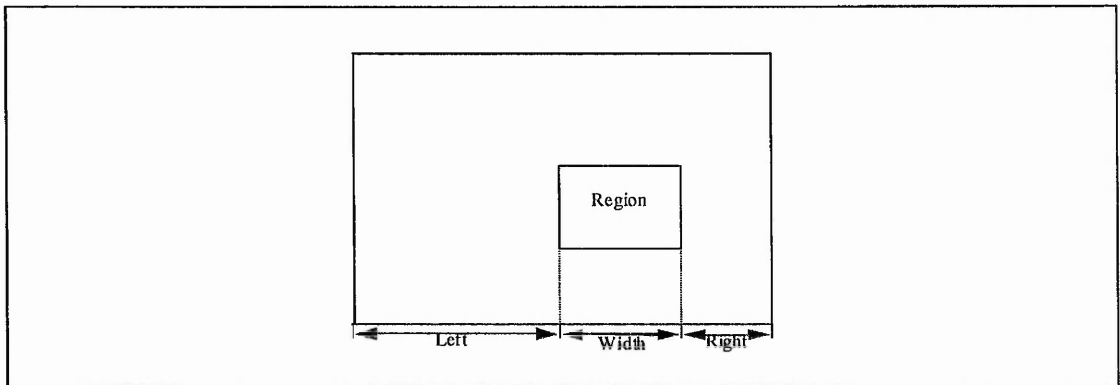
Of the 315 cases, most have an obvious position, for example, corners in zones 1 and 9 suggest that the region is in the top left corner of the screen. The small number of difficult combinations have been changed according to results of experimentation with regions of various size and position.

The user therefore hears a rough description of the position of the object on the screen.

### **5.2.2.3 Three section position**

This feature not only provides an indication of position of a region, but also gives its size. The method can be applied both horizontally and vertically. As an illustration, consider the horizontal case.

A region on the screen divides the screen into three sections - the gap between the left edge of the screen and the left edge of the region, the width of the region, and the gap between the right edge of the region and the right edge of the screen, as shown in Figure 5.9.



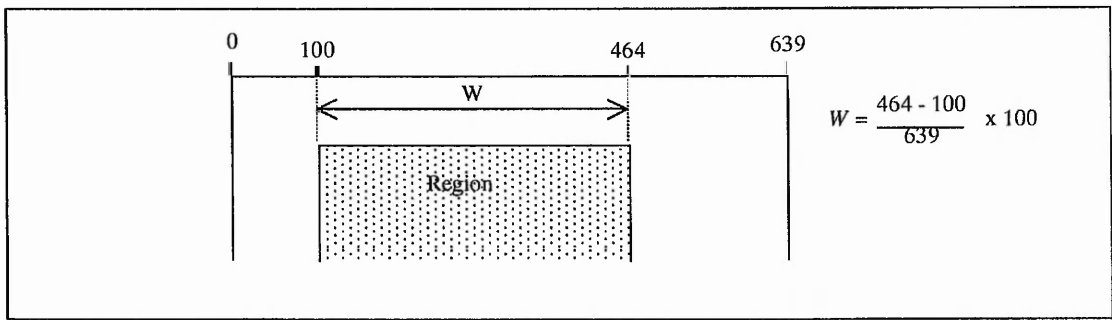
**Figure 5.9 Three section positioning - the three quantities used**

These three measurements can be presented to the user in one message. For instance, each could be given as a pixel count. This, however, is not the most suitable form. Firstly, it takes time to announce three large numbers. Then there is the problem of assimilating such large numbers, particularly as relating the three numbers to each other is a fundamental part of the feature. Using percentages is another possibility. The user would then hear a message such as ‘ten, twenty seven, sixty three’, for example.

As announcing three percentages also takes some time, a new method was developed. This uses ratios. Each measurement is converted into sixteenths, such that the above percentage example would be given as ‘two, four, ten’. The user can easily relate the three numbers to each other and to the screen, and by using small numbers speed of presentation is increased.

### **5.2.3 Size of a region**

If the user is able to work with pixels, precise dimensions may be given. Alternatively, the familiarity of inches or centimetres can provide more meaningful measurements. Both of these methods provide absolute dimensions, and as such are meaningless without knowledge of the dimensions of the screen. Therefore, dimensions can be given as percentages of screen dimensions. For example, in Figure 5.10, width of region (W) is 57 percent of screen width, based upon the equation given.



**Figure 5.10 Size of a region**

Area may also be given as a percentage of total screen area, however, it may be more meaningful to convert the percentage to a common fraction, for example, all areas would be given in terms of eighths of the screen. To a sighted user the concept of area is practically irrelevant, however, to a blind user, this could help indicate the nature of the region being dealt with.



# **Chapter 6 Document Access**

## **6.1 Introduction to the Document Access System**

### **6.1.1 Aims of a document access system**

#### **6.1.1.1 Document content and user requirements**

The second source of multi-dimensional information to be considered in this work is the graphical document. This includes documents incorporating different types of graphics - pictures, graphs, tables, etc, - and text with various font sizes, styles and colours. Such documents cause two major problems to the visually impaired. Firstly, the content of the document is not laid out in a linear way - the order of presentation of information is less clear than for a basic document where the sole content is text, readable from top to bottom of the page. Secondly, layout, text attributes and graphics are used to give the document impact. Large, coloured text stands out from the page, italic text adds emphasis to a word or phrase, and graphical representations add aesthetic value to raw data.

It is necessary to provide a visually impaired person with a means of acquiring such document information using non-visual display. The user must be able to not only determine content of a document, but also to both discover layout of information and detect the presence of text attributes. Comprehension of graphical components is also necessary. The knowledge of this will ultimately aid the visually impaired user in manipulation of such documents

#### **6.1.1.2 Why present document layout?**

The need to present document content is obvious; the need to present document layout is less clear. As mentioned, manipulation of documents (for example, creating, editing, analysing) is expected to be more easily achieved if the user is aware of how the document is structured. Also, sharing documents with sighted people should be more straightforward. An example can be given to illustrate this.

In monitoring an internet email list dedicated to issues regarding computer use by the blind, some unusual message formats were observed. Two authors in particular wrote in such a distinctive style that it was assumed, correctly as it later transpired, that they were

totally blind. The reason for this assumption was that the messages were not easy for a sighted person to read. One format involved using no more than approximately 20 characters per line so in general only 2-3 words per line - giving long, thin messages. The other format involved large spaces between words, equivalent to perhaps 5-10 character widths. At least one author was queried over this, and indicated that the adaptive software used to create the messages was the reason. The person, having been unaware of the appearance of these messages, adjusted a setting to produce a more orthodox format. Had these people had some means of checking the layout of the documents they were creating, and compared this with layout of messages from sighted users, then they would not have been sending messages in an unorthodox format. This example is simply an illustration - in other circumstances it may be essential that the blind person uses a format that doesn't frustrate a significant proportion of recipients. By enabling a visually impaired person to determine the layout of a document, that person is more suitably equipped to communicate with sighted people.

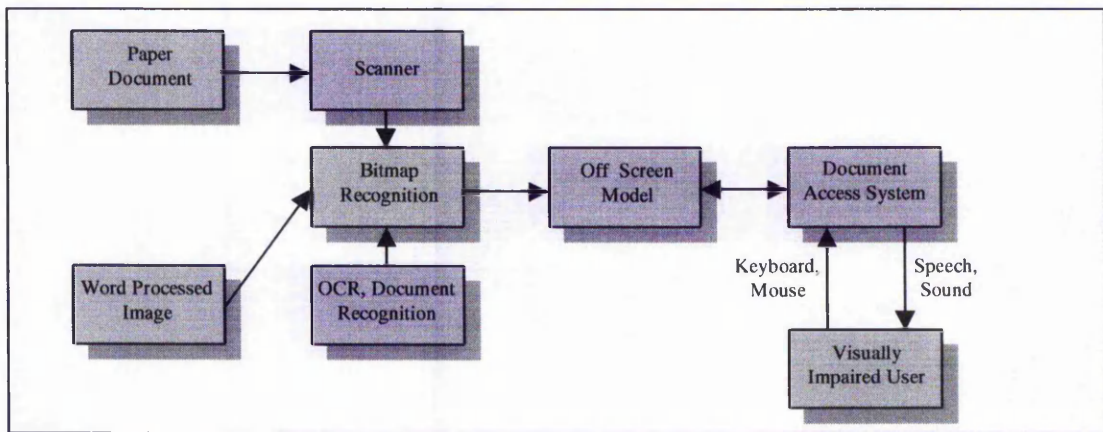
### **6.1.1.3 The need for a new information presentation system**

In order to allow unaided use by visually impaired persons, a system for the presentation of document information must be easy to use and fast enough to prevent user frustration. The architecture of the Talking Windows system is suitable for presenting GUI screen information, but with the screen analysis method it employs, and the large number of features it accommodates, it was not, at the time, ideal for adaptation to a system capable of presenting document information. Therefore a new system has been developed, the Document Access System. This is intended to be easy to configure and use, to cut out time-consuming and irrelevant functions of the Talking Windows system, and to allow the addition of features to present a variety of possible document components. The following section describes the architecture of the new system.

## **6.1.2 System architecture**

### **6.1.2.1 System overview**

The starting point of a document access process is the visually impaired person acquiring a document. This could be a letter received in the post, or a magazine cover, for example. Alternatively it could be a word processed document. The steps that are then taken on the way to the user being able to access this document are illustrated in Figure 6.1.



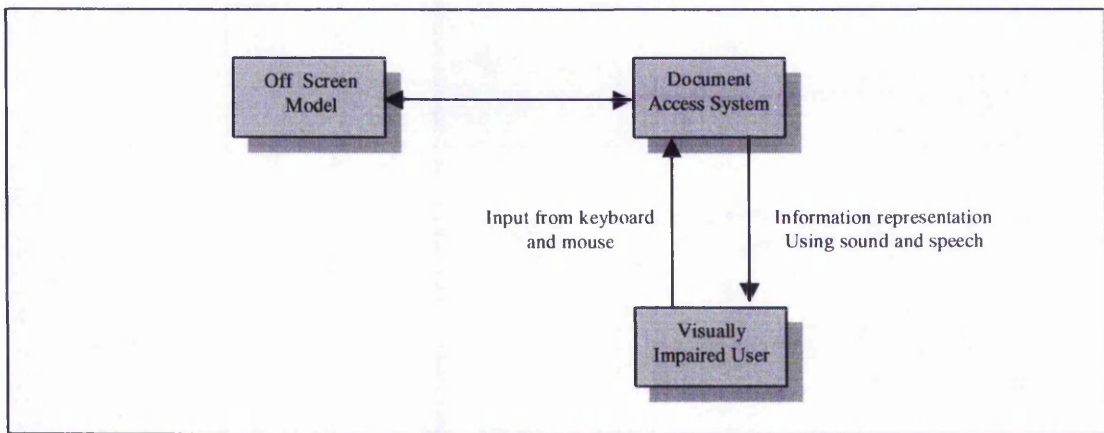
**Figure 6.1 Document Access System overview**

Based on this diagram, a possible scenario is as follows. A blind person opens a letter, and uses a scanner to form an image of the document. A series of recognition processes are then initiated. Document analysis and recognition will be used to recognise structure and content - what text is present, what graphical regions are identifiable, etc. This process combines to form an Off Screen Model (OSM), a database containing document information in a format accessible by the user interface. The user interface responds to commands issued by the user, typically using keyboard and mouse, and presents information using non-speech sounds and synthesised speech. The information held in the OSM is therefore being converted into a format useful to the user; the user controls what is presented, and when.

To complete the process, the user may manipulate the document provided it is of appropriate source. For example, the word processor could be used to add text, move a diagram or fill in boxes on a form, based upon the user's knowledge of the existing document. The recognition engine will be used to update the OSM, and the user can subsequently verify the changes using the interface.

### **6.1.2.2 The Document Access System overview**

From the diagram above (Figure 6.1), we can identify the parts relevant to the objectives of this work. The emphasis is on the representation of spatial information, therefore we need not be overly concerned with earlier steps, such as the recognition processes, and the construction of the OSM. Instead, the step between OSM and user is the focus of attention. Figure 6.2 illustrates the process.



**Figure 6.2 The Document Access System**

### **6.1.2.3 The Off Screen Model**

This section gives a description of the Off Screen Model, and how it will be constructed. The main concern of this work is to develop and evaluate a variety of presentation ideas, and this ultimately has priority over a complete product. It would be unproductive to incorporate sufficient recognition procedures into the system to cater for any given document. A more efficient approach is to take a small number of sample documents which cover all the required types of document component, and manually construct an OSM for each, based upon what can be reasonably expected to be recognisable by existing methods. So for example, we can look at the OCR and icon recognition techniques used in the Talking Windows system [Sherkat 93] and from that we know that we can take any given document and safely assume that printed text can be located and recognised, and stored alongside position and font information. A similar method can be used to identify text, graphic and mixed text and graphic regions. We can therefore simulate an Off Screen Model for any documents which we wish to use as samples.

Initially, this store will hold co-ordinates for each identifiable region of the document. For each region, the co-ordinates of each recognisable object within the region will be stored. This region hierarchy is continued down to the base components of the document. This may be either a unit of text, an icon tag, or some other means of storing a graphical construct. This basic OSM can be adapted during system development to accommodate different types of object being considered.

#### **6.1.2.4 Input and output**

As with Talking Windows, features will be accessed using key combinations, but with the system having fewer features and working as a foreground application, the key layout is more easily constructed. Output will once again be restricted to speech and non-speech sounds.

#### **6.1.3 Approaches to providing features**

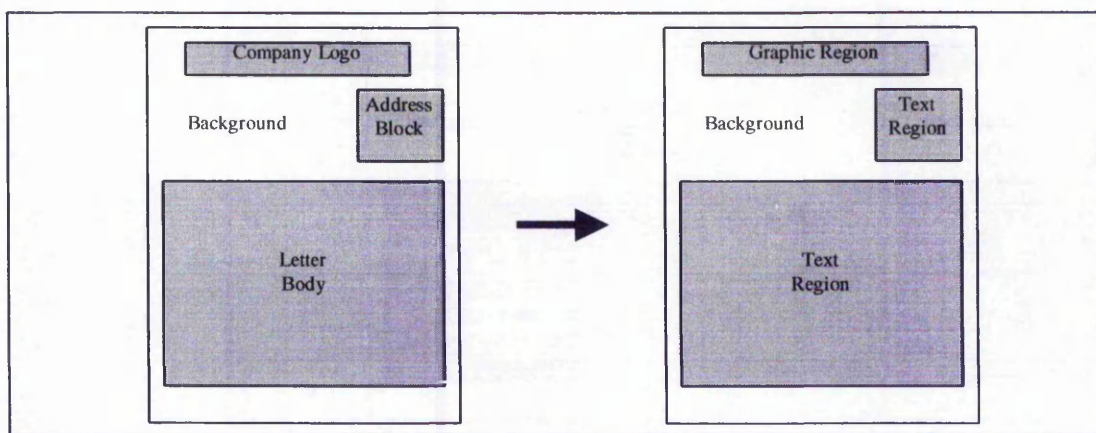
##### **6.1.3.1 A hierarchical view of a document**

A document can contain an enormous amount of information. For a sighted person, it is possible to scan over a document, note key words, diagrams, colour, etc, and focus on only those parts of interest. As an example, you don't need to read each article in a newspaper before finding one of interest. Scanning headlines and perhaps photographs is sufficient. However, when speech is used rather than vision, its sequential nature typically requires that you do have to listen to a large quantity of information before hearing what you are interested in.

Therefore it is necessary to control what is presented, and important to give a logical structure to presentation. In earlier work involving GUI screens, the initial screen could be broken down into parts - Windows provided a logical sub-section. The user could then concentrate on individual windows, thus having less information to cope with. Furthermore, windows could be further sub-divided, allowing the entire screen to be 'viewed', but in a structured manner. The same approach is possible with documents.

A document can contain text, and areas of non-text which we can refer to as graphics. There is also a further type of component, which can be called 'mixed', where text and graphics are interwoven - a bar chart is an example of this. Finally, instances of these types usually reside on a background area, for example, white paper. In general then, we can identify four region types including the background. Figure 6.3 shows an example - a business letter is divided into regions.





**Figure 6.3 Dividing a document into regions**

Now that we have identified region types, we are able to break down almost any document into a number of smaller, more manageable chunks. The letter shown above (Figure 6.3) has three regions not including the background. Each may be further subdivided, for example, the letter body might consist of two paragraphs and the signature section. The visually impaired reader may only need to read the first paragraph and the signature. Or read the signature first - often you want to know who the letter is from before reading it, and this is easy enough for sighted people.

To summarise then, in order to create smaller chunks of information, essential when using auditory display, regions dictated by their type give a more logical breakdown than, say, decomposition by physical position. The features developed reflect this approach.

### **6.1.3.2 Categorising features**

By restricting display to the auditory channel, there is a choice to be made between the directness of speech and the speed of sound. Speech is more familiar, but sound may, on occasions, be more effective. Features available fall into two categories; those based on sound, and those which predominantly use speech.

### **Speech-based features and verbosity**

These involve the direct presentation of information. For example, descriptions of document content and layout, and announcement of position or other document component attributes. The user must pay full attention to the speech in this instance. Presentation may be slow depending on the number of words to be spoken, but there are two measures taken in this system to reduce the problem.

Firstly, speech rate is adjustable. The system chooses a speech rate (which can vary between perhaps 120 and 400 words per minute) according to the information being given. As an example, a passage of text can be spoken very quickly (400wpm is comprehensible to experienced speech synthesis users), whereas numeric data or warning messages may be spoken more slowly.

The second measure is the provision of a verbosity setting. To make the system informative enough for novice users and fast enough for the experienced, it is necessary to incorporate such a facility. The user selects from a menu choice of terse/average/verbose, and any spoken message required can be worded differently for each possible setting. For example,

"Twelve fifteen"	Terse
"It is twelve fifteen"	Average
"The time is twelve fifteen"	Verbose

## **Sound-based features**

The intention is to present information with a minimum of speech, due to the higher speed of playing bursts of sound instead. Sound is also ideal for background information, or non-essential information. For example, a sound map could be used in the background while speech-based features are giving direct information to the user. The sounds heard can add to the information being taken in. As an example, consider the process of filling a glass with water - watching is sufficient to know when to turn the tap off but the change in sound heard can assist if the eyes are distracted. Similarly if the disk drive stops its clicking and whirring you can tell that a file has finished being copied without seeing the screen dialog box. It makes sense to reinforce the main information source with alternatives - speech with sound in this work.

## **6.2 Speech-based features of the Document Access System**

### **6.2.1 Giving an overview of document content**

This section describes the two pieces of information given at the top level of document presentation. They inform the user of the identity of the current document, essential for the user to know to prevent the treatment of the wrong document.

### 6.2.1.1 Announcing document title

It is assumed that each document has a title. In cases where there is no clear title, for example, a letter, the first text line encountered scanning across and down the page is used. Whilst this may not be ideal, it is the presentation which is significant.

As in Talking Windows, a standard sound precedes the spoken message. The message itself varies according to the verbosity setting. The terse message simply gives the first three words whereas the verbose message would commence 'The document title is ...' followed by the whole title. In addition, the speech rate is lowered slightly to aid comprehension.

### 6.2.1.2 Summarising document content

An indication of document content reinforces the title information given and informs the user of what must be done to 'read' the whole document. This feature involves announcing the number of regions of each available type.

In Figure 6.4, there are four text-only regions, one graphic region and one mixed region, giving a total of six regions. This can be presented in a short and meaningful spoken message, for example, 'Six, four, one, one'. Words can be added according to verbosity setting. As an example, the average setting gives 'there are six regions, four, one, one'. The user, on hearing this, becomes aware of the 'scale' of the document.

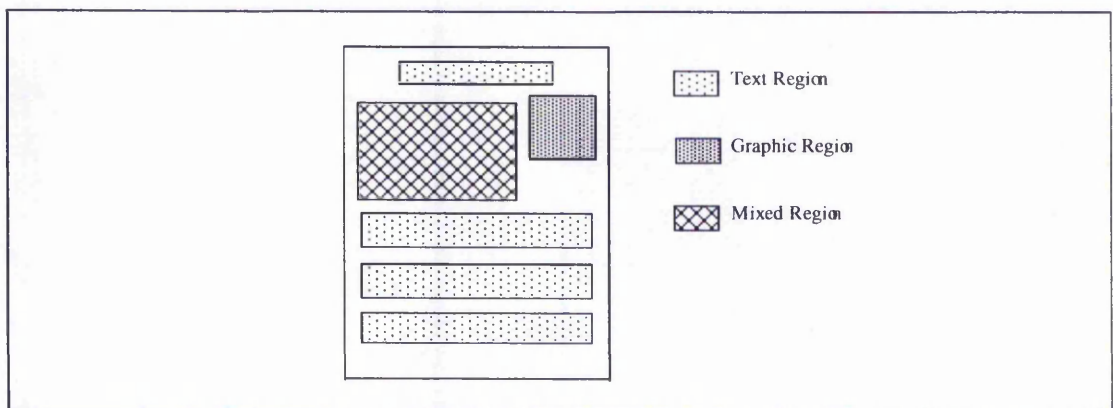


Figure 6.4 Announcing number of regions

### 6.2.2 Working with regions

The user can select an individual region, which then becomes the 'current region'. Features which previously applied to the entire document now apply to the current region. For example, the summary feature described in the previous section now gives a



count of each type of region within the current region. A text region may consist of a number of paragraphs, each of which may be seen as a single region.

To select a region, a list is created ordered by region position. The user moves forward or backward through the list, hearing titles according to the verbosity setting, until the required region is found.

A number of navigation features are provided, applying to either the whole document or the current region. These involve moving the cursor to the specified corner or edge, or the centre of the region. A sound indicates the position moved to, but this is reinforced by a spoken message, for example, 'left of region'.

### **6.2.3 Presenting document attributes**

Features giving position and size were covered in the Talking Windows system. The more successful are used in this system to support the more experimental features available.

Cursor position is given as a percentage of document width and height across and down. For region position, the 'rough' position feature is used, giving, for example, 'top left' of document. For more accuracy, the three section position feature is used, and this also gives an indication of region size.

## **6.3 Sound-based features of the Document Access System**

There are a number of features which use sound rather than speech as the main way of conveying information. Before detailing these the sound formats available are described, together with the speech attributes which can be used.

The wave file format is a useful source of sounds. Samples are easily created and manipulated, with length, volume, pitch, echo and stereo balance all being adjustable. The main disadvantage is the size of files in this format.

Midi is also used, and has two considerable benefits. The first is the musical capability that it offers, and the second is that no storage space is required. To give the flexibility required, a function was created allowing a note to be generated with instrument, note, note length and volume chosen at runtime. The limits are as follows.

- **Instrument** - 128 instruments are available, including musical instruments, novelty sounds and effects. So we can select a piano, a tenor saxophone or a helicopter noise, say.
- **Note** - there are approximately 100 audible notes, varying with instrument, so for the piano, we can use the full scale plus some notes higher or lower than the usual endpoints if necessary.
- **Note length** - a midi sound can be played for one millisecond or any value upwards of that figure. Applications of this are described later.
- **Volume** - Each midi sound can be played at any volume in the range. Again, applications are described later.

Together, midi and wave files provide a wide range of sounds and noises with which to experiment.

In addition, the rate and volume of speech can be altered, to change emphasis, for example. Different voices give scope for increasing the amount of available information. The message being spoken gives the user the main information, but the voice it is spoken in can give additional, important but not essential detail. As an example, if the cursor position is announced in a male voice, the user could infer that the cursor is also within a text region. If it is a female voice, then the point described is in a graphic region. Region type is not usually part of this feature, but can be given in this manner to save time in selecting a separate feature, provided the user is experienced enough to know the relevance of different voices.

### **6.3.1 Layout sound maps for documents**

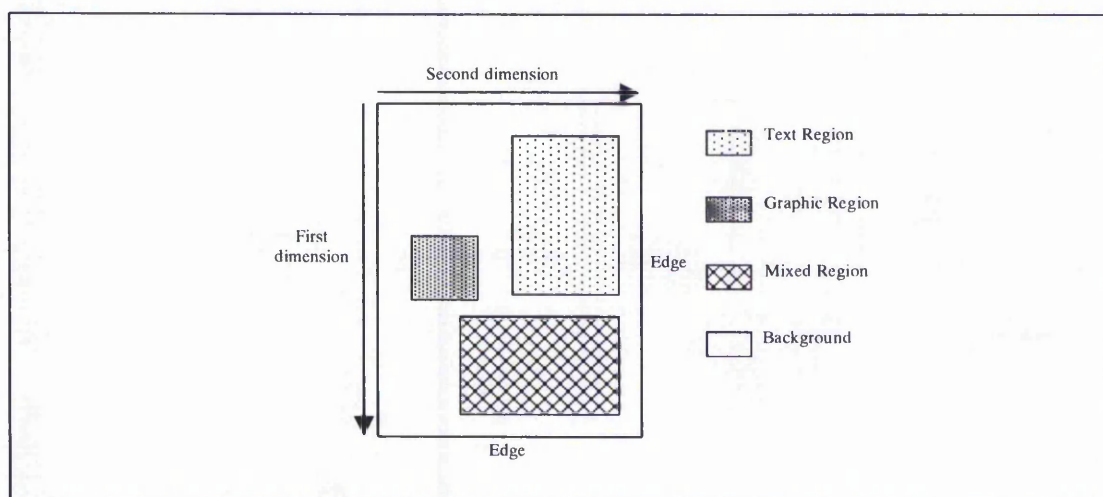
The Document Access system provides a useful platform for developing the sound maps introduced in Talking Windows. The availability of more controllable sound allows more comprehensive sound maps to be developed, with these being more pleasant to listen to.

The aim of these sound maps is to convey layout information. As well as indicating types of region as they are encountered, positions of regions and their interrelationships can be determined.

By using sound rather than speech, an increased quantity of information can be presented because of the time taken to impart spoken messages, but to assist the user, speech-based features are available for support. These indicate type of region under the cursor, and what the current sound represents.

Figure 6.5 shows the different region types, and the two dimensions along which sound may change. From the sounds heard the user should be able to determine position across and down the document, and type of region or edge under the cursor.

As in Talking Windows, four sounds are used to indicate edge type, with a high pitched note for the top edge, a low pitch for the bottom edge and the left and right edges identified by the speaker through which the sound is played. Also common to each sound map is the use of the musical scale to indicate position down the screen. As there are many variations using this scale (approximately fifty notes) compared to, say, volume changes, it is logical to use this scale for the larger dimension.



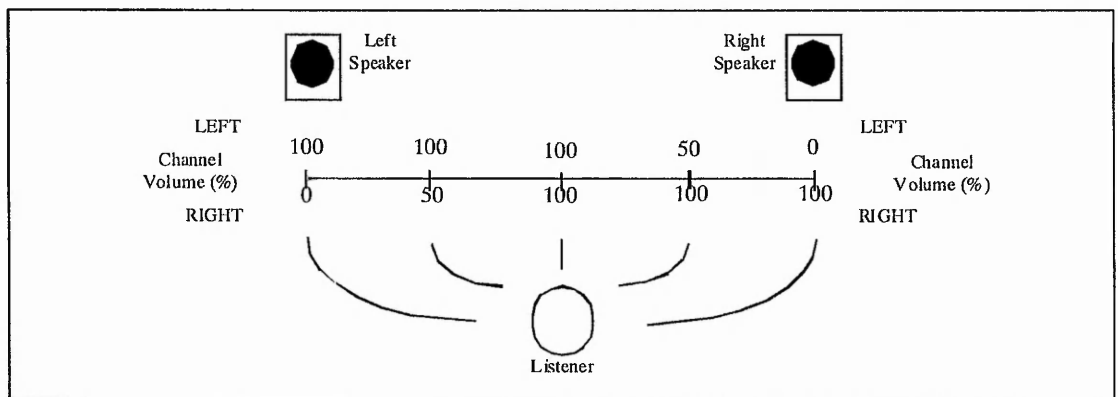
**Figure 6.5 Applying sounds to a document**

### **6.3.1.1 Stereo balance sound map (Figure 6.7)**

The second dimension is indicated by the apparent source of the sound being played relative to the two speakers or headphones. This is achieved by adjusting the volume of sound through both left and right channel. Figure 6.6 illustrates this.

As the mouse pointer moves from left to right of the document, the source of the sound appears to change from left to right speaker in distinct steps. Experimentation revealed that only about five settings can be used to give distinct changes. This allows the user to estimate that the mouse pointer is in one of five zones across the document.

The instrument used to play the notes can indicate type of region (text, graphic, mixed, background). This requires selection of four instruments from the 128 commonly available midi instruments. Ideally a link between region type and instrument would be used to help the user memorise and identify sounds heard, but finding such a link is not easy, particularly when some instruments have to be discarded due to their inability to play notes at either end of the scale at a consistent volume or sufficiently distinct from adjacent notes. A second option would be to allow the user to choose preferred instruments.

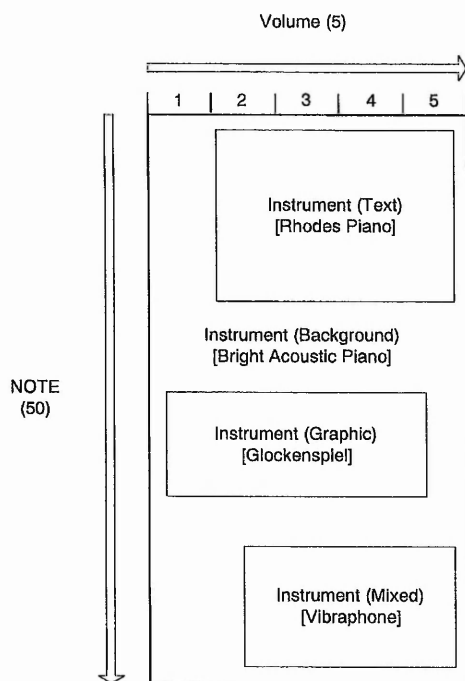


**Figure 6.6 Using the source of sound to indicate position**

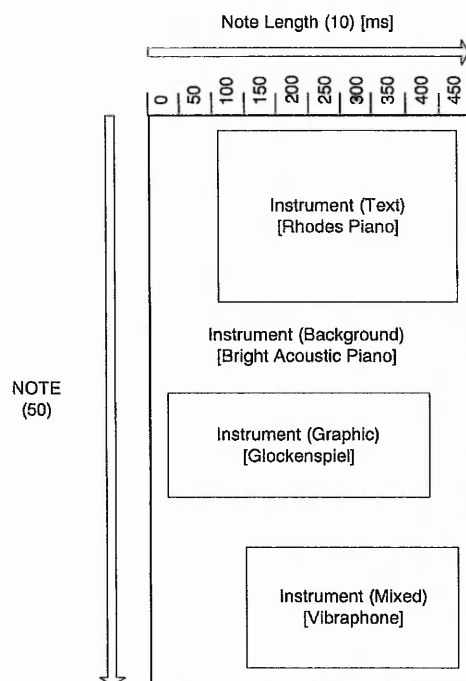
The choice used for the initial evaluation with this sound map relies on a link between the initial letters of region type and instrument, giving text/trombone, graphic/glockenspiel, mixed/music box and background/banjo. This choice attempts to keep instrument sounds distinct, which is more important than the link.

### **6.3.1.2 Volume changing sound map (Figure 6.7)**

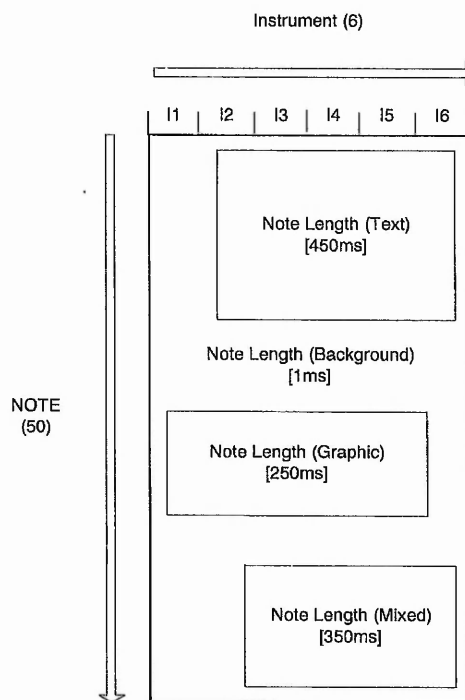
The sole difference between this and the Stereo Balance sound map is with its treatment of the second dimension. Rather than alter the stereo balance of the sound, its volume is changed according to position across the screen. Determining an appropriate range of distinct volumes is difficult, as it was for Talking Windows, but with the improved clarity and adjustability of sounds available for this work, it is worth attempting because the concept is very simple for the user to grasp. As with stereo balance, five zones are possible across the document.



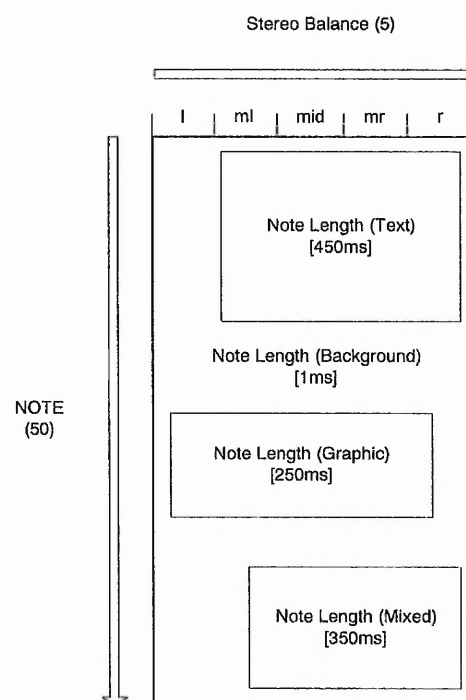
**Volume Sound Map**



**Note Length Sound Map**



**Instrument Sound Map**



**Stereo Balance Sound Map**

**Figure 6.7 Summarising the layout sound maps**

### **6.3.1.3 Note length changing sound map (Figure 6.7)**

There is only one difference between this and the previous sound maps, a new sound variation used to indicate position along the second dimension. To describe this, recollection of the mechanics of a sound map is necessary. The cursor position is monitored many times a second to enable an accurate indication of position. A sound is played for each cursor position observed. This results in the warbling sound heard. The length of the notes being played affects how the warbling sounds - if a note length of 1 millisecond is played repeatedly the sound is rapid and intense. If this note length is increased to, say, 200 milliseconds, then the warbling is more relaxed and calm. Note that this note length is still sufficiently small to ensure that the position is accurately reflected by the sounds heard.

There are advantages to using note length compared with the two previous methods of indicating position along the second dimension. Volume changes are susceptible to problems caused by the listener's surroundings, as, to an extent, are stereo effects. Also, more zones across the document can be established - experimenting with various note lengths has indicated that increasing note length by 50 milliseconds makes the new warbling sound audibly distinct from the previous one. Furthermore, note lengths of up to at least 500 milliseconds still give a pleasant warbling sound reflecting cursor position accurately. This gives us roughly 10 zones across the document, with note lengths 1, 50, 100, 150, ... , 450ms. Now position in this dimension can be estimated with more precision - by counting zones passed through or, if possible, gauging current note length, which may be possible with experience.

### **6.3.1.4 Instrument changing sound map (Figure 6.7)**

This is slightly different to the previous three sound maps, each of which used instrument to represent region type and just differed in the treatment of the second dimension.

Here, the instrument heard will indicate the position across the document, which requires a different way of representing region type. The reason for using instruments for the second dimension is to test whether their more distinct sound will give the user a better indication of horizontal location. Carefully chosen instruments will sound more different to each other than volume changes, varying stereo balance or note length differences.

Since the width of the document will typically be small, not many zones are needed - between five and ten have been used by previous sound maps. For experimentation with this sound map, six zones are used, so six distinct instruments are required. Again there is the problem of choosing appropriate instruments. The user should be able to identify an instrument, and then know where it comes in the list of six instruments. This should be easily achieved by ordering instruments alphabetically. Six recognisable instruments to start off with are accordion, glockenspiel, piano, saxophone, timpani and violin. These all have the necessary range of notes where volume or distinctiveness do not diminish at ends of the scale. Therefore if the user hears piano notes, since piano is third out of six instruments in the list, the mouse pointer is just left of centre.

Note length is used to indicate region type - a note length of 1ms (rapid warbling) indicates background, increasing for text, graphic and mixed regions. A further consideration when choosing instruments is that one note played at an arbitrary large note length (greater than one second, say) must have duration greater than the maximum note length needed. If this isn't the case (for example, a drum note that lasts only 50ms) then the sound doesn't always play correctly. Experimentation determines suitable instruments.

### **6.3.2 Octave sound map**

The aim of the octave sound map is to give the user the approximate position of the mouse pointer in both dimensions using fewer sounds or sound changes than previously. With the piano sound map proving popular with blind users, the octave sound map is an attempt to capitalise on that by using piano notes to give position relative to both dimensions at once.

Rather than divide the document area into the 50 horizontal zones which the piano scale can cover, the musical concept of octaves can be used. Figure 6.8 shows a document outline with an indication of how octaves can be used to indicate mouse pointer position.

The higher octaves are at the top of the document, decreasing downwards. For musicians, this should be more useful, but for non-musicians it should still be usable. The context of the note heard will be important as for previous sound maps.



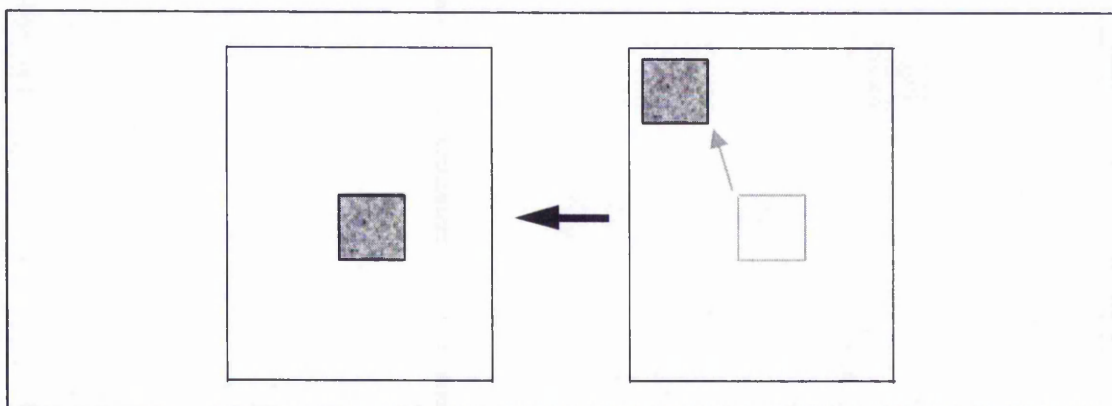
First Octave ...	c4	d4	e4	f4	g4	a4	b4
Second Octave ...	c3	d3	e3	f3	g3	a3	b3
Third Octave ...	c2	d2	e2	f2	g2	a2	b2
Fourth Octave ...	c1	d1	e1	f1	g1	a1	b1
Fifth Octave ...	c	d	e	f	g	a	b
Sixth Octave ...	C1	D1	E1	F1	G1	A1	B1
Seventh Octave ...	C2	D2	E2	F2	G2	A2	B2

**Figure 6.8 Dividing a document area into octaves**

## 6.4 Supporting Drag and Drop

The Document Access system allows the user to navigate to regions and determine their content. To allow regions to be moved, support for drag and drop was developed.

A simple test case is illustrated in Figure 6.9. The aim is to enable the blind user to move the region from the centre of the document to a specified location, for example, the top left corner.



**Figure 6.9 Dragging a region into position**

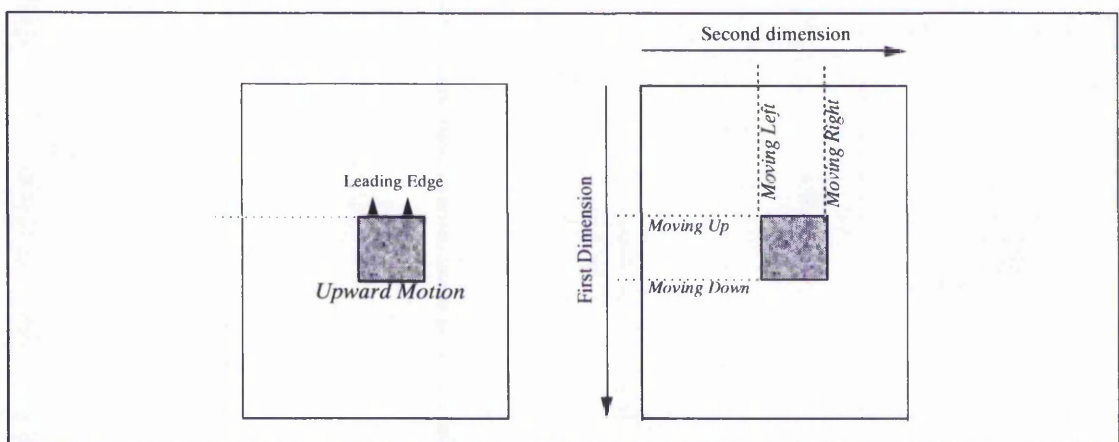
If such a task can be achieved effectively, then more complex situations can be attempted.



A visually impaired user will need to know where the region being moved is at any time. It would be possible to describe this position using a spoken message, for example. This may include co-ordinates of each edge, or the distance of each edge from the corresponding document edge. The time that it takes to say all this is a disadvantage, especially as it would be constantly out of date as the region moved.

A partial solution is to reduce the amount of speech necessary by just giving the position of the 'leading edge'. If the region is moving downwards, towards the bottom edge of the document, then the distance between bottom edge of region and bottom edge of document can be given and updated as the region moves. This still suffers from a speed problem, which may be solved by use of a more advanced speech synthesiser, but would not then be such a universal solution. Instead, the use of sound maps has been investigated.

As the user moves the region, if the region is moving upwards, then it can be assumed that the user is trying to get the top edge of the region into its desired location relative to the above limit, in the test case, this being the top edge of the document area. Similarly for the other three directions. This can be used as a basis for using a sound map. Rather than the current mouse pointer position determine what sound is played, the leading edge of the region will be used. Figure 6.10 illustrates this.



**Figure 6.10 Use of sound to manoeuvre a region into position**

If the current direction of movements is upwards, the top edge of the region is the leading edge. The position of this edge relative to the document top edge can be estimated using the sound variation used for the vertical sound map dimension. If the user then moves the region left, then the sound variation used for the horizontal sound map dimension indicates position of the left region edge.

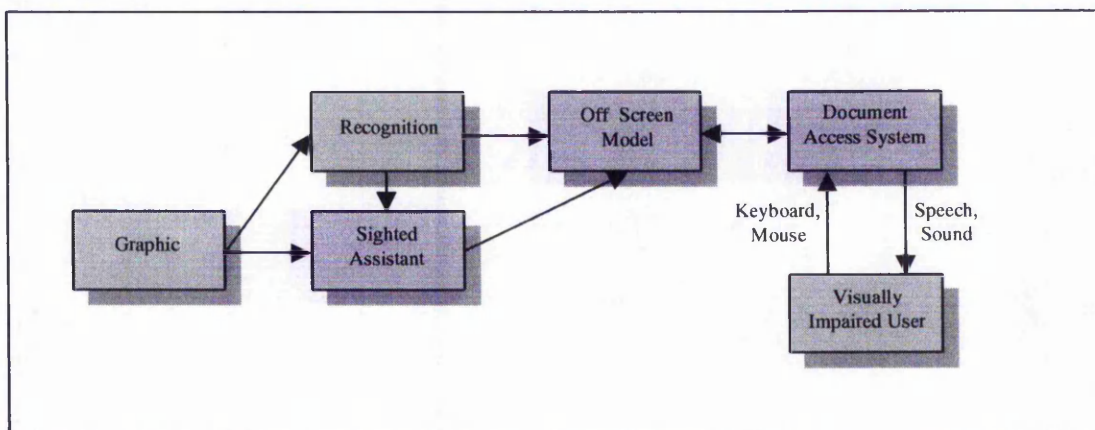
The sound maps described previously use a sound variation for horizontal and vertical dimensions, for example, note for vertical, volume for horizontal, or note for vertical and note length for horizontal. It is possible to just use one variation for the dragging feature, as only one dimension is given at a time. The piano sound map is a suitable choice.

## **6.5 Presentation of graphics**

### **6.5.1 Data tables**

A document may contain pie charts, bar charts, line graphs, pictures and other forms of graphical information. It is essential that a visually impaired person has some means of interacting with these. For pictures and animations, a spoken description can be given as they are encountered, if available. For pie charts and graphs, the underlying data must be accessible. Generally, a table of data is the basis for creation of a graph or pie chart, etc. What these representations do is add emphasis to trends or patterns in the data, or visual appeal to a set of figures. The current work aims to provide the visually impaired user with a non-visual means of interacting with a typical table of data.

In order to provide a link between a graphical structure appearing on a document and a set of features allowing manipulation of a table, the Off Screen Model must be considered. If recognition techniques are used to determine document content, then this must include a means of recognising a graphical structure and storing it in the OSM. The Document Access System can then access the OSM in order to provide the required presentation features. It is expected that most graphical structures would be stored in the OSM in table format. After all, this is generally what they originated from. Converting from a pie chart image to an OSM table format requires good recognition techniques or the assistance of a sighted person. This is the approach taken by [Kurze 95] in his study of access to graphics by the visually impaired. One sighted person can prepare a graphic once for the use of many visually impaired people many times. A further reference to presenting graphical information is the work on Sound Graphs [Mansur 1985]. Figure 6.11 illustrates the steps involved in graphics representation.



**Figure 6.11 A system enabling access to graphical information**

In order to allow evaluation with presentation methods, the main focus of the work, it is assumed that the OSM contains a representation of a table of data. This can be used to display the table as part of a document, and features developed to provide non-visual access to this table.

### 6.5.1.1 Presenting a data table

Figure 6.12 shows a sample table, reconstructed from data stored in the Off Screen Model. This shows us what must be presented to the visually impaired user. A sighted person will typically scan rapidly from one place to another, observing row and column labels, titles, tracing across a row or up and down a column, picking out values, looking for patterns, re-checking labels and so on.

	1974	1975	1976	1977	1978	1979	1980	1981
Switzerland	87	65	41	92	10	8	21	34
Belgium	78	80	76	54	19	39	33	2
Austria	7	0	0	18	31	67	39	1
Sweden	28	30	48	17	39	82	81	62
Finland	57	68	11	94	46	100	74	74

**Figure 6.12 A sample table**

The basic means by which a visually impaired person can access a table is by requesting all text in the table area to be read, row by row. A more interactive method is necessary, whereby the data is navigated efficiently and with flexibility. The visually impaired user should be able to perform the same tasks as the sighted person but in a non-visual manner. So, for example, the current row or column label could be spoken, or

the complete row or column entries given. The aim is to allow the visually impaired person to get information from the table with a speed close to that of the sighted person.

### **6.5.1.2 Navigating a table**

A sighted person navigates a table by movement of the eyeballs, and picks up information by reading the entry at which the eye is pointed. A blind person will have an entry spoken, but first needs a way of pointing at which entry to speak. Getting to this 'current entry' involves navigation around the table. The tools available include the keyboard and the mouse. The keyboard is well suited to horizontal or vertical movement, whereas the mouse is ideal for fast movement in any direction about the table. A sighted person doesn't read every entry, but skips to certain rows or columns and entries within these. Therefore, three types of navigation facility are required.

- horizontal and vertical steps with the cursor keys
- fast unconstrained movement with the mouse
- 'jumps' to various table locations

The use of these techniques is seen in the following section which describes the announcement of entries to the user.

### **6.5.1.3 Use of speech and sound attributes**

If we consider that by some navigation means the user has selected a current table entry, the announcement of that entry basically requires a spoken message. A value may be spoken, and then the user moves to the next entry and hears its value. However, there is scope for providing more information than just the value. We can use a brief non-speech sound to precede the spoken value, and the attributes of the speech and sound to give more feedback to the user.

A sighted person looking at a table will read a value, but also, perhaps subconsciously, absorb other information, such as position in the table. In the context of a table, this suggestion is difficult to illustrate. However, a similar thing occurs when, having read part of a book, the reader wishes to refer back to a previous mention of say, a character or place. Often the part of the page on which such a mention occurred can be recalled, which aids the search. The reader has subconsciously noted position of certain words or phrases whilst reading. The result of this is that if a visually impaired person is only



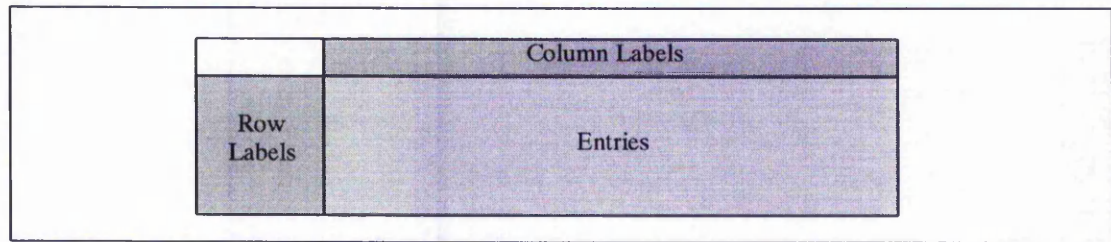
spoken the value, already some information available to a sighted person is lacking. The remedy is to include information of this kind in the announcement of an entry value by means of speech and sound attributes. Further uses of these will be described shortly.

The presentation of position of an entry as well as its value will alleviate the problem outlined above, in that where a sighted person observes surrounding features such as table borders or screen edges to take in this positional information (consciously or not), some part of the auditory message given to the visually impaired user will replicate this. Furthermore, by indicating position, at least two other advantages are to be had. Firstly in distinguishing between two neighbouring entries of identical value and secondly in knowing where to look to find entries found previously.

In order to experiment with different methods of presentation of a value, the concept of a sound pattern has been established. A sound pattern defines which speech and sound attributes are to be used to represent what information. Three such patterns have been used, and are described as follows.

**(a) Basic sound pattern**

A typical table will consist of different types of element, as shown in Figure 6.13.



**Figure 6.13 Types of element in a table**

By attempting to indicate the type of an element when announcing its value, the user is given extra information which may assist in navigation. For example, having heard that the value just spoken was a column label, the user is able to move from side to side to determine other column labels, or move downwards to reveal the values of entries in that column. Without such knowledge, distinction between labels and entries may not be as easy.

To indicate element type, the instrument used to play the sound preceding the speaking of element value varies. Four instruments are necessary - the top left corner is intended to be a 'reference' point where the user can go back to re-start navigation. No

message need be spoken, but a sound can be played, with an instrument that clearly indicates it not being a true element. Using a novelty instrument rather than a musical one is an option, the ‘bird-tweet’ sound currently providing a distinct sound.

As for the other sounds required, three distinct musical instruments are necessary, a possibility being to select one from the brass section, one from strings and one from wind instruments. Alternatively, sounds could be selected by the user - in cases where a clear link between sound and the entity it represents can be found, it could be beneficial to insist on this sound, but where choice is more or less arbitrary, allowing the user to decide may ultimately improve interaction.

### (b) Comprehensive sound pattern

More can be done to improve the layout information being presented. As well as the sound giving element type, it can also give position within the table. This can be achieved by using the other variable sound attributes in a manner similar to that used by sound maps. Once again, instrument will determine element type. The note played can be used to indicate vertical position - in the first row the note Middle C can be used as a starting point, stepping one note down the scale as row number increases, thus mimicking previous sound map ideas. As note length proved useful for the second sound map dimension, it can be used for the horizontal variation here. In column one, a short note length is used, 1 millisecond. This is increased by the smallest distinguishable step, say 50ms as column number increases.

As a final measure, to provide distinction between adjacent element values, two voices are used to speak values, with no two adjacent elements being spoken in the same voice. Figure 6.14 illustrates a table with the sound/speech attributes marked for each entry.

< Increasing Note

	Increasing Note Length >				
	<i>Bird Tweet, C 1ms, male</i>	<i>Trumpet, C 50ms, female</i>	<i>Trumpet, C 100ms, male</i>	<i>Trumpet, C 150ms, female</i>	<i>Trumpet, C 200ms, male</i>
<i>Clarinet, D 1ms, female</i>	Piano, D 50ms, male	Piano, D 100ms, female	Piano, D 150ms, male	Piano, D 200ms, female	
<i>Clarinet, E 1ms, male</i>	Piano, E 50ms, female	Piano, E 100ms, male	Piano, E 150ms, female	Piano, E 200ms, male	
<i>Clarinet, F 1ms, female</i>	Piano, F 50ms, male	Piano, F 100ms, female	Piano, F 150ms, male	Piano, F 200ms, female	
<i>Clarinet, G 1ms, male</i>	Piano, G 50ms, female	Piano, G 100ms, male	Piano, G 150ms, female	Piano, G 200ms, male	
<i>Clarinet, A 1ms, female</i>	Piano, A 50ms, male	Piano, A 100ms, female	Piano, A 150ms, male	Piano, A 200ms, female	

**Figure 6.14 Using sound and speech attributes in a table**

### **(c) Sound-only pattern**

In experimenting with table navigation, time is wasted if the user wants to go from one element to another distant element. In general the delay waiting for speech to complete when that speech is not required will cause frustration. By using the previous pattern without speech, movement is very much quicker, whilst sounds are still available to guide the user to near the required element, at which point speech can be restored. Alternative methods of skipping to distant elements are possible, but this method also allows evaluation of the use of sound attributes to present positional information and therefore improve the user's understanding of the table.

## **6.5.2 Supporting table presentation features**

The navigation and element presentation methods described above can be supported by additional, more direct features, made possible by identifying data stored in the OSM.

Firstly, trends in a single row or column may be detected by hearing all the values in that line. This could be achieved by locating the row or column and using the above methods. However, a more efficient way is to use the co-ordinates stored in the OSM to determine which elements are in the same row or column as the current element and speak these in order, directly. When the user has located an entry, the following functions are available.

- announce row/column label of current element
- read entire row/column containing current element

Another necessity is that the user be informed of more general information regarding the table, including

- a title/label, according to what the OSM stores
- dimensions, that is, number of rows and columns
- row/column labels

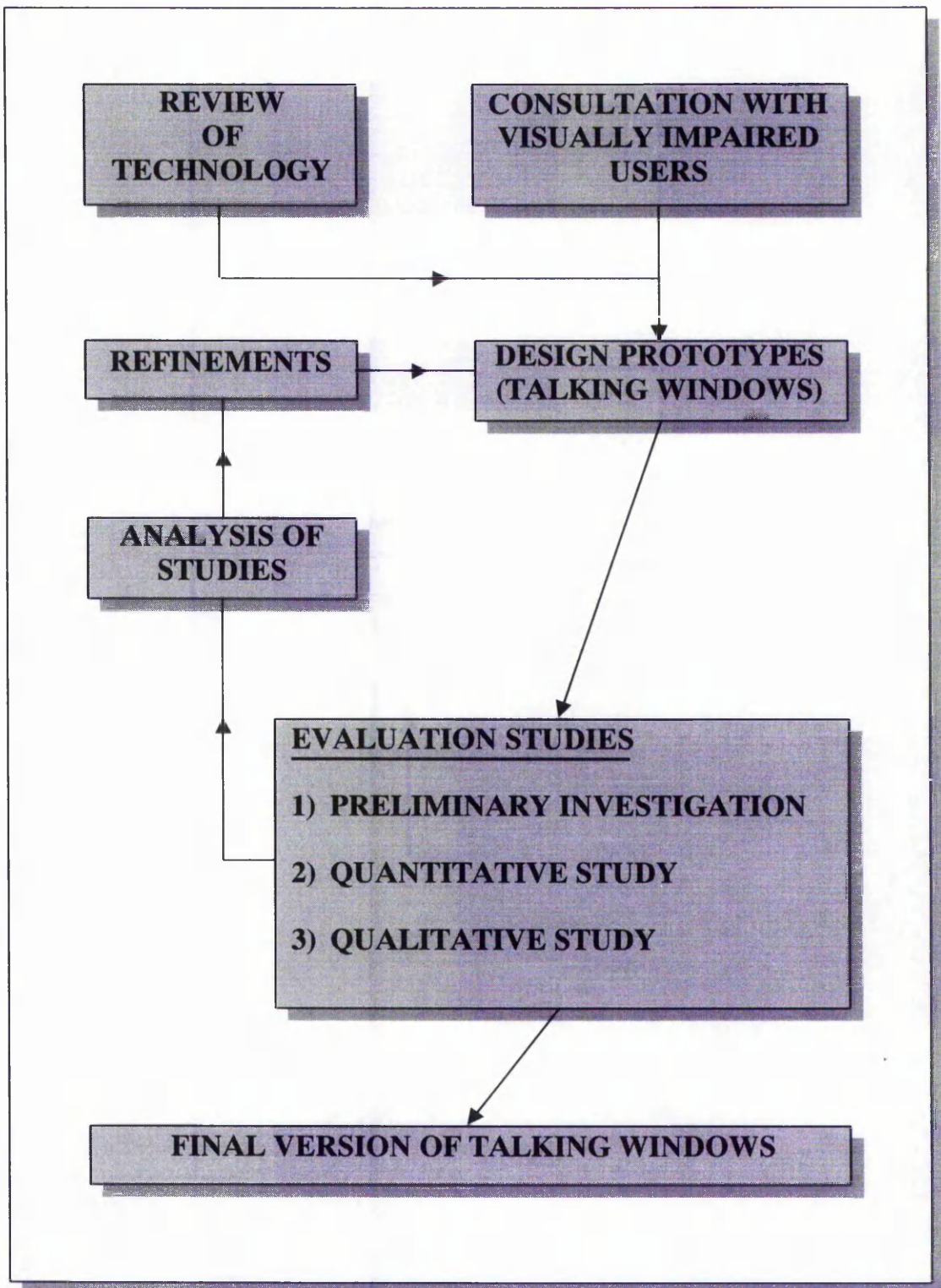
By providing a direct means of presenting number of rows and columns the user quickly knows what sort of table is being confronted. A 2 by 2 table will require less

manipulation than a 20 by 20 table. The above features use the basic message announcement with sound preceding speech. Verbosity setting is used to decide how detailed the spoken part will be.

In order to provide a means of evaluating the usability of these table features the Document Access System generates a random table. Number of rows and columns both range between 2 and 10, and their labels vary as well. Entries can be between 1 and 100 inclusive, altogether giving a wide range of possible table dimensions and values.



## Chapter 7 Research Methodology



## **7.1 Review of technology**

This involves investigating existing technology, including commercial screen reading products, methods of input and output, and research projects based in this area. This is described further in the literature chapter.

## **7.2 Consultation with visually impaired users**

Visually impaired users are involved at each stage of the project. Initially, at the analysis stage, feedback from such a source helps determine what is currently available, and what is required. Consultation helps to establish the needs and abilities of the target users of the system being created. This is described further in the introduction chapter.

## **7.3 Design prototypes**

The first two stages provide the knowledge required to design and construct a prototype of the Talking Windows system. Talking Windows contains a range of features, including those derived from existing systems, and those newly designed based upon user requirements. Further details of the Talking Windows system are described in the theory chapters.

## **7.4 Evaluation studies**

A prototype is evaluated with visually impaired users. The process involves deciding how to conduct the evaluation, visiting the subjects to carry out the study, and recording the required data as planned. Methods used for evaluation studies are described in the introduction chapter. Details of the studies undertaken are described in the following chapters.

## **7.5 Analysis of studies**

Data collected from each evaluation study is analysed. It is used to assess the level of success of the prototype in achieving its aims, and to guide development of the prototype. Analysis of the method of evaluation enables subsequent evaluations to be carried out more effectively. The following chapters contain the results and analysis of each study undertaken.

## **7.6 Refinement**

Each prototype is refined, based on the response from visually impaired users. Existing features are improved, and additional functionality added. The following chapters show how user reaction during evaluation studies leads to refinement of Talking Windows.

## **7.7 Final version**

The process continues, until the three evaluation studies have been completed. After the final study, the version of Talking Windows which results is the product of substantial user input and the continuing refinement process. Analysis from the final evaluation study suggests possible directions for future development. This is described in the future work chapter.

# Chapter 8 Evaluation Study One

## 8.1 Method

An informal investigation was carried out in the first year of the project, seeking to establish whether the approaches being pursued were worthwhile, and to determine what visually impaired users require from a screen reading system.

This initial investigation enables preparations to be made for more extensive and analytic evaluation studies to follow. It involves gauging the general response of blind users to the prototype Talking Windows system, and the fundamental concepts introduced including mouse use, non-speech audio cues and key layout, for example. Furthermore, the range of users available, appropriate times required with each, and the number of features which can be presented can be established.

Staff and students at the Royal National College for the Blind in Hereford took part in the sessions. Following consultation with the contact at RNCB, a request was made for users via email, with no stipulation of age or experience required given. Those who responded were allocated a time at which to be present. An email sent prior to the sessions together with word of mouth attracted further users resulting in occasions where there were several users present at the same time. Whilst this was not a disadvantage for this particular study, a note was made that this should be avoided on future occasions.

Users had hands-on experience of the software, together with demonstrations, and discussions of the various aspects of the system.

Of thirteen users, eight were male and five female. Age ranged from approximately 15 to 70. There were different levels of computer experience, and of visual impairment, from partially sighted through to total blindness.

All comments, suggestions and questions asked were recorded. This resulted in an account of the time spent with each user. From each such account, a number of conclusions were drawn. Furthermore, suitable conditions for future studies were established.

## 8.2 Results

The results of this study are first listed by user, as a set of case studies. Following this, the results are grouped by subject area.

### 8.2.1 Results by user

#### User A

- Partially Sighted – able to see the screen reasonably well, although blurred. Finds it difficult to locate the mouse pointer
- Computer technician, has a lot of experience of visually impaired computer users attempting to use various types of access software

The first exercise involved using the mouse to navigate the screen hearing object labels spoken when hit by the mouse pointer. As a technician, use of the mouse was a familiar operation. Being able to see the mouse pointer when it moves, the user had no difficulty in getting to the various objects on the screen. However, this user was able to offer some valuable information concerning blind people using the mouse.

When the mouse is moved, the pointer can move in any direction. He suggested that the screen could be set up in a kind of grid with the pointer moving only horizontally and vertically from one cell to another. This could be achieved with either the mouse or the cursor keys.

He also suggested that the use of the mouse depended very much on the individual. Some blind users will not use the mouse, whereas others are happy to at least try to use it. The main reasons for not using the mouse are as follows:

#### a) Time Consuming

Moving the hand off the keyboard to the mouse and then relocating the home keys takes time. Once the home keys have been found many users will not be keen on moving their hands off the keyboard. To some extent this also applies when the number pad is used by an application. Furthermore, this is not exclusive to visually impaired computer users.

#### b) Motor Problems

Blind people with motor disabilities will also experience difficulties with the mouse. Included in this section is the sensitivity of the mouse - moving the mouse a small distance can cause the pointer to move a large distance and without visual feedback this may cause problems.

#### c) Conceptual Problems

Similarly, the user may have problems relating the mouse, or the movement of the mouse, to its corresponding function of moving the pointer on the screen. The user pointed out that some blind users on first using the mouse would lift the mouse off the pad, or move it great distances in order to try and achieve something. This was witnessed in the following sessions.

#### d) Hardware Constraints

Apart from these factors, it is often the case that blind computer users disconnect the mouse to free a port for access hardware such as external speech synthesisers or Braille displays.

The user suggested that ideally the more frequently used commands would be found on the Qwerty keyboard, and less common features activated from the number pad. Mouse functions should be available on the keyboard as well, allowing the individual who refuses to use the mouse to access the feature nevertheless.

The next exercise involved using the Edge Sound Map. This plays a sound whenever the mouse pointer cursor hits the edge of the screen. However, the user pointed out that to a blind person, the computer is regarded as a friend, and that the more human the interface can be made to appear, the better it will be. Therefore speech on the edges would be preferable to sounds. In this case speech would be more informative. It was then suggested that the whereabouts of the cursor on that edge could be reported. The edges could be scaled in some way.

It was suggested to the user that speech is slow compared to audio signals in that the time that it takes to speak a string of text is large compared to the time that it takes to play a short burst of sound and that a balance must be achieved between a fast interface and an informative one. The user suggested that the experience of the user with the system would affect this balance.

## **User B**

- Totally blind
- Student at the college, and a music technician
- Computer experience consists of using Eureka (a speech only computer with a virtual screen), Hal5 for DOS, and limited use of Window Bridge

Arriving at the end of the first session, User B agreed that use of speech and sound would need careful consideration. He suggested that ideally the system would need to be tailored from an informative interface for new users to a fast interface for experienced users.

The first exercise involved moving the mouse around to navigate the screen. The user was concerned that having heard a message, he was unsure of whether the mouse pointer was still actually on that object. Whilst the system does have a 'magnetic' effect, whereby the mouse pointer jumps back to the centre of the object as its label is spoken, it was suggested that this may not be sufficient for the user to feel comfortable clicking or double clicking.

It was suggested that the actual message given when an object is hit may need altering. Currently it consists of a sound indicating the active state of the object, followed by the type and label of the object. A pause could be inserted between the type and name to give time to assimilate the information. The user then mentioned that this is something that could be tailored depending on the experience of the particular user, as a pause would increase the time taken to give the message.

The user was concerned that a uniform message is given for all objects. The messages should have the same format, incorporating a pause if necessary. A current inconsistency occurs when some icons are linked to the text below them, whilst others are not, for example, Windows group icons.

The next exercise was to use the Object Hopping facility, where the user can jump from object to object without encountering the 'white space' between objects. This needed to be more informative. Instead of the current audio signals informing of the direction of the next object, or that there are no more objects in that direction, speech should be used instead. Initially both sound and speech could be used, so that when the user requires more speed, the speech may be dropped, but the remaining sound will by

then have been associated with the particular function. This emphasised the need for more spoken messages.

The next exercise was with the use of the piano sound map. With this user being a musician, this opened up more possibilities than expected, and his response to this feature was particularly interesting.

The reduced volume of the piano sounds which was recently incorporated to allow the sound map to be used in the background without irritating the user was well received. Its low volume compared to speech and other sounds allowed it to be left playing without causing any noticeable distraction. A negative point was the one-dimensional nature of the feature. That is, the piano scale used can only change from top to bottom or from side to side. The vertical version was more popular with this user, and it was with this that he was able to try to determine the spatial relationship between objects. Having heard a particular note for one icon, he was able to tell that a nearby icon was not quite on the same level, because a slightly different note was heard. In fact they were both on the same level, but the cursor had been at the centre of the first icon but at the centre of the combined icon/text object for the second, causing a difference of one or two piano notes. However, if the cursor consistently went to the same point of an object, the possibility of using the piano sound map for determining object relationships is increased. This depends though on the musical ability of the user and the clarity of the piano sounds available. As this user is a musician, it is likely that his ability to distinguish between two similar notes would not be matched by everyone. However, he did suggest that "most blind people are by nature musicians", and that it would be worth developing this feature.

This also demonstrated that for musical users the piano map sound could be used to determine precise distances in general. If the zone in which a certain note plays were to always be the same size, then the distance of travel could be ascertained. However this wasn't the original intention of this particular sound map, and zone widths vary according to the size of window in which the map is playing.

The next feature, the window sound map, also uses piano sounds. A window was opened on the screen, and the sound map activated. The user spent several minutes with this sound map, trying to determine precisely where on the screen the window was located. This produced some useful comments. Firstly, the concept of there being a single window on the screen only. The desktop area was also considered to be a window. Then it was questioned whether a blind person would ever need to be able to determine the



location of a window. The user had some success with this map however; he was able to determine that there was an area below and to the right of the window, and that the window was near to the top and left edges of the screen. It may have proved to be possible to discover how far from each edge the window was by using this feature, but time prevented further progress. As it is also possible with Talking Windows to announce these distances, the need for this kind of map was questioned.

Similarly the next exercise, the use of the grid sound map to determine distances, proved too time consuming to be considered effective. It did, however, demonstrate that some feature is needed to give the distance between two objects.

### **User C**

- Partially sighted
- No experience of using Windows, or a mouse

The first exercise involved using the mouse, which despite having no previous experience, caused the user few problems. This user was more interested in how the system would assist in the task of editing text, rather than in going through the features available as planned beforehand. Unfortunately there was little to support text editing at the time due partly to the speed of the recognition process.

The user expressed a wish to be able to minimise or close the current window. When informed of the current methods for accessing objects on the screen, for example, finding a window and navigating to the close/resize buttons, it was observed that this was too slow, and that a 'hotkey' would be needed for such frequently required functions.

As a result of this session, it was clear that a more window-oriented approach was required, including a window summary after each screen analysis, a main key to select the current window and a priority of getting to the active window, reading it and closing it.

### **User D**

- Totally blind
- No experience of using Windows, or a mouse

The first exercise was to use the mouse, which caused considerable difficulty. The user was initially seen to lift the mouse away from the table. Having been guided to the mouse mat, the user was able to move the mouse pointer around the screen, but in a random manner, and expressed doubts about using a mouse in the future. This experience resulted in a reluctance to continue with further exercises.

### **User E**

- Totally blind
- Young student

Having moved the mouse pointer so that an object was encountered and its label spoken, it was observed that the speech was unclear. When the message was repeated the user was able to understand the message, and suggested that unfamiliarity with the speech synthesiser was the problem, the voice used being quite different to that which the user was accustomed to.

The first exercise involved using the mouse to select items from a list, with cursor movement used to navigate the list. This user was sufficiently comfortable with the mouse to be able to attempt to use the two versions of the feature available, however, the mouse was not moved with the accuracy required to select items as intended. The user pointed out that he was unsure which direction the mouse was moving in at any time.

This user had more success with the edge sound map, becoming more and more competent with the mouse. However, in the next exercise, the user did not have sufficient accuracy with the mouse to be able to locate the centre of the screen with the target finding sound map. The mouse pointer moved randomly across the screen, and after some time the centre had not been located.

### **User F**

- Partially sighted (very limited vision – able to see the mouse, but not the mouse pointer on the screen)
- Young student

The first exercise was to use the mouse to select from a list of items. The user was good with the mouse, being capable of making small movements. This enabled the user to cycle forwards and backwards through the list of items, hearing each item name

spoken in turn. By this time it was apparent that there was no advantage in having the two slightly different versions of this feature.

The edge sound map was used successfully, with the user navigating to each edge in turn, and able to announce which edge the mouse pointer had struck when the sound was heard. On questioning the user, it was suggested that this was not due to the particular sound heard, but as a result of hand/mouse movement.

The target finding sound map was not successful, despite the user spending a couple of minutes concentrating on the sounds heard and attempting to use them to guide movement.

## **User G**

- Totally blind
- Young student

The first exercise was to use the mouse to make selections from a list of items. Although both versions of this feature were attempted, there was no indication of either being better. The user was able to cycle through the list without the impression that movement was random.

This user was able to locate the centre of the screen using the target finding sound map after some time trying. At the time, the loudest sounds played indicated proximity to the target. It was observed that the user was able to keep the mouse pointer in the loudest zone, but then the location of the central zone, indicated by a distinct sound, was achieved seemingly at random, which the user later admitted. Mouse use was very accurate, with the user taking care not to move the mouse too fast. This resulted in some frustration as the sound map didn't provide sufficient feedback for the final part of the target finding process to be achieved.

## **User H**

- Partially sighted
- Staff member

When using the mouse to make selections from a list of items, this user suggested that the mouse was too sensitive for it to be used in this way, although when attempting one

of the two versions, it was observed that the user did have good control of the mouse pointer.

When using the target finding sound map, it was pointed out that the edge sounds were an unnecessary distraction – you would only need sounds for the zones surrounding the target. The sounds used to guide the user were said to be too similar to each other, in other words, the difference in volume between two zones was too small. Furthermore, the sounds were irritating, particularly as the louder sounds were found nearer to the target.

## **User I**

- Totally blind
- Staff member

The first comment from this user was that the sounds heard were of a fairly good quality.

The sounds used by the edge sound map were enjoyable, and the user was able to remember one sound and identify it later in the session when used in the window sound map.

The target finding sound map sounds were not as popular though. These were said to be uncomfortable. This user was able to move the mouse pointer fairly accurately, but was distracted from the target finding task by the poor sounds heard.

## **User J**

- Partially Sighted
- Young Student

This user was cautious with the mouse, and expressed surprise at being asked to use it. The sounds heard appeared to amuse the user, particularly when using the edge sound map.

When using the piano sound map, the user liked the sounds heard, and took some time moving the cursor around the screen, hearing the resulting sounds. The user had sufficient vision to be able to see the screen and some of the objects, but text could not be

read. It was suggested that it was useful to have sound to guide movement, but that enlarging the mouse pointer could be more useful.

### **User K**

- Totally Blind
- Young Student

This user had some difficulty in understanding some of the messages heard when an object was hit by the mouse pointer, appearing to be confused when more than one or two words were spoken in a message. Having had no previous experience with the mouse, this also caused problems, with the mouse pointer moving very quickly, and the mouse itself occasionally raised from the mat. The user seemed slightly bemused at the concept of the system and suggested that she would need some training in order to be able to use it.

### **User L**

- Totally Blind
- College Lecturer

Initially, this user had difficulty moving the mouse pointer with the mouse, and when informed that keyboard alternatives were available learnt which keys to use and for a while preferred this method. However, after a couple of minutes using the keyboard, the user made another attempt at using the mouse, and decided that this was his preferred choice.

The user concentrated on learning the sounds heard whilst using the edge sound map, and was able to tell the particular edge encountered using the sound rather than from the movement of hand on mouse.

### **User M**

- Totally Blind
- Highly computer literate, trained musician

Before starting to use the mouse, this user was fairly negative about whether it was appropriate to attempt to use the mouse in a screen reading system, saying that while he

himself would use it, many blind people would not, and that keyboard alternatives must be found for every mouse function.

He suggested that the centre of the screen is a good reference point - whenever the user requires, there is a fixed point on the screen to which the cursor can always be moved. If a fixed screen location is not suitable, then an omnipresent object may be used, such as the title bar of the active window or active icon.

The user mentioned that many blind people are, by nature, musicians, which makes features such as the piano sound map worth developing. He also wanted to see a different approach to the typical screen reader, maybe with a hierarchical or treelike system with submenus or 'floors/levels' being used. Finally he suggested that status information be readily available as it is easy to lose track of settings, etc.

## **8.2.2 Results by subject area**

### **8.2.2.1 Mouse use**

One user suggested dividing the screen into a grid with movement restricted to horizontal and vertical movement to adjacent cells. It was suggested that use of the mouse would depend on the individual, with some blind users unwilling to use the mouse at all, but others happy to at least attempt its use. Arguments offered against its use were varied. Firstly, it is time consuming moving the hand between keyboard and mouse. Secondly, users with motor disabilities will experience difficulties, with mouse sensitivity, for example. Thirdly, conceptual problems may arise, with users having difficulty relating mouse movement to its effect on the on screen cursor. Finally, hardware constraints may cause difficulties, with blind users often disconnecting the mouse to free a port for external speech synthesisers or Braille displays.

Of these, the third is the most significant, the first two not applying solely to visually impaired users, and the latter a surmountable technical problem. The third is addressed in the work by providing non-visual feedback and would be supported by user training.

It was, however, observed that a couple of users did experience difficulty with the mouse, moving it large distances or off the table. Sensitivity was an issue raised when using the mouse-based list selection features, rendering the variations present irrelevant. While this was partially due to implementation, it indicated the dangers of making assumptions concerning mouse use. However, it was encouraging that some users with no prior experience of using the mouse were able to use it with a sensitivity sufficient to

successfully navigate the sample lists by using the mouse as a switching device. Furthermore, having used both the keyboard and the mouse to propel the cursor, one user declared that he preferred the mouse as “it is much faster”.

#### **8.2.2.2 Accessing features**

Means of accessing the available features were discussed. Use of the Qwerty keyboard for frequently used features was suggested to reduce hand movement required. It was also requested that all mouse functions be accessible from the keyboard.

The need to use more than one key combination to access features wasn’t popular and didn’t help give a positive impression; this is unavoidable in such an experimental system but was later improved to help subsequent studies.

#### **8.2.2.3 Sound maps**

In response to presentation of sound maps, it was suggested that to a blind person “the computer is regarded as a friend”, and that the interface should be as “human” as possible. Therefore speech may be used in preference to sounds where possible. It was suggested that you need to strike a balance between a fast interface and an informative one, and that some customising according to user experience would help.

Use of piano notes was popular. The reduced volume enabled the sound map to be used as a background sound without apparent irritation. Its use by one user in particular was fascinating. As a musician, he could identify a particular note heard and later return to that note, therefore knowing that the cursor was at the same “level”. Missing notes were detectable, which caused some concern. The user attempted to discover the distance between icons by counting notes heard, and then requested the “width” of a zone to calculate a precise measurement. For the other sound maps demonstrated there was mixed reaction. Some success was observed, particularly with the target finding sound map, however, implementation problems, to be expected at this early stage of the project, meant that closer inspection was not feasible. An encouraging aspect was that the need for sound maps was almost universally unquestioned.

#### **8.2.2.4 Screen reader functionality**

One user was unsure of whether the cursor remained over an object once its label is spoken. It does, but it is important to realise that a user may prefer a more certain means

of verifying that the cursor is still where expected prior to attempting to perform an action on the object.

Message format was addressed with one user requesting a pause between sound and speech, whilst some users would have preferred a faster message. Consistency of message format was requested, with a user pointing out that some icons caused a message without a preceding sound. Whilst this was actually a limitation of the soundcard at the time, such inconsistencies must be dealt with to avoid confusion.

## **8.3 Review**

Each feature takes a certain amount of time to evaluate, including time needed to prepare the display, call the feature, explain its purpose, its usage and the test, plus the time taken by the user in becoming sufficiently accustomed to the concept, operating the feature and providing feedback. Whilst the time required varies according to the feature, an average time spent on each feature may be estimated. A figure of ten minutes per feature was calculated at this point.

The initial study also provided an indication of how much time each user could be expected to spend using the system in a single session. Each user had been prepared to spend thirty minutes in a preliminary session, with a thirty minute gap between appointments being available should extra time be required. The majority of users spent over forty-five minutes with the system, with some having to leave after between sixty and ninety minutes either to accommodate following users or to attend other appointments.

This suggests that for the following study, a minimum period of one hour should be requested per user, with the possibility of extra time if and when required.

The results from this study demonstrate that using the mouse as part of the system is feasible. The way in which it is used falls into two categories. Firstly, as an additional input device, used, for example, to select options from a list. By providing this functionality, fewer functions need to be accessed using the keyboard. This is beneficial provided that the user is able to move the mouse with sufficient control and speed. It was observed that this was the case for most users. However, it was stressed that keyboard alternatives would be required.



A more useful way to use the mouse is as a means of accessing screen content. To propel the cursor in order to hear auditory feedback you can either use the mouse or cursor keys. It was seen that users were, in general, capable of using the mouse with sufficient dexterity to make such an approach possible. It was also noticed that users liked the faster speed of the mouse to move the cursor.

The main conclusion drawn from this study regarding mouse use is therefore that whilst keyboard alternatives should be available, fears that visually impaired users would be unable to use a mouse are unjustified. The mouse may improve interaction due mainly to its speed, provided support is available.

The use of sound rather than speech for some parts of the system caused a mixed reaction. Some users could understand the advantages of sound in terms of speed of presentation, but it was argued that speech is a more friendly, human medium.

It was generally felt that sounds can be learned without difficulty, and can be pleasant as well as informative. It was observed that some users became impatient when a lengthy spoken message was given. It was felt that a faster voice should be available but that the user should be able to control this according to preference. A balance between speech and sound use must be found, with a degree of customisation available to allow for individual preferences. The use of sound to increase the amount of information available to the user was considered to be a good approach.

In general, where speech can be used effectively, use speech, but where speed is necessary, or where additional information can be given using sound, use sound.

It was observed that message format must be consistent otherwise confusion can arise. The requirements of users varied, with some preferring descriptive messages, but others preferring a short, fast format. This suggests that to cater for all users, it will be necessary to allow the user to choose a setting which determines how verbose a message is. The use of a sound preceding the spoken part of a message was observed to be helpful both in separating messages and in indicating type of message.

This study also addressed the presentation of spatial information, in the use of descriptions of object position and sound maps. Two encouraging observations were made. The first was that users were generally happy that these types of features were available and felt that this kind of information is necessary. Several suggestions for everyday uses were given. The second was the response to sound maps, in particular the

piano sound map. It was encouraging that users achieved success in identifying objects 'hit' and were able to determine position of objects with some accuracy.

There was a positive reaction to the use of piano notes, and some unexpected observations made. One was the way in which a user not only noticed a missing note, but could even name it. Another was the way a user identified a note heard when the cursor was over an object and could then return to the same 'level' subsequently. Whilst this precision was explained by both users being musicians and could not be expected of all users it raised a significant issue, that of the use of music in the interface. It was suggested that many visually impaired users are musicians, which is logical, and therefore there is potential in making further use as part of the interface.

To summarise, results suggest a continuation of the approach taken with regard to mouse and sound use and presentation of spatial information, and that the preferences and abilities of users be taken into consideration.

# **Chapter 9 Evaluation Study Two**

## **9.1 Method**

The aim of this study was to measure the response of users to each of a set of features and then use the numerical results to evaluate features and, subsequently, compare response to that from both commercial products and future versions of Talking Windows.

### **9.1.1 Selecting features for evaluation**

Based upon observations made in the first evaluation study, a set of between ten and twelve features was deemed to be required. These must include those features to which response is unpredictable and those which relate most strongly to the concepts involved in the system.

There is a question of whether to include those features available on commercial products. If you do, comparisons can be drawn, but you then leave less time for the features and concepts which set this work apart from those commercial products. It was decided to concentrate on Talking Windows, taking the most significant areas of development such as sound maps and spatial determination. There is inevitable overlap with commercial products, not on a feature by feature basis, but in aspects of features, for example, message format and non-speech sound use.

The following section lists each feature chosen together with a summary of its purpose, and the attributes which make it a suitable feature for testing.

### **9.1.2 Features selected for evaluation**

#### **9.1.2.1 Cursor Position**

The position of the cursor is announced, demonstrating message format and the use of a preceding non-speech sound. This allows for a discussion of the need for such a feature and if so the preferred unit of measurement, and issues such as speech attributes. Feedback will apply to a number of related features in Talking Windows.

### **9.1.2.2 Sound Maps (Volume, Piano, Diagonal, Target Finding and Layout)**

The user must perform tasks depending on the sound map. This allows a number of issues to be addressed including presentation of spatial information, use of non-speech sounds, determination of layout and the use of music. The chosen sound maps represented the range of sound maps available at the time of testing. There are a number of tasks which the user can perform, such as naming which screen edge has been encountered, identifying screen position, locating a point and determining type and layout of screen objects.

### **9.1.2.3 Separation Of Two Points**

Two items are positioned on the screen and their separation announced to the user. Amongst the issues here are message format, assimilation of a quantity of spoken data, spatial information and speech attributes.

### **9.1.2.4 Three Section Positioning**

The position and size of an object are described in a single message. Message format and assimilation of complex data are tested here.

### **9.1.2.5 Mouse List Selection**

The user manipulates a list of items using the mouse both to navigate the list and to activate the chosen item. This tests mouse use, the concepts of list manipulation and decision making and the use of non-speech sounds.

### **9.1.2.6 Mouse Navigation**

This is a test of how well the subject uses the mouse and whether the user is comfortable with the idea of using the mouse.

## **9.1.3 Testing procedures**

For each feature selected, a testing procedure was defined. This included the following information.

### **9.1.3.1 System set-up**

This refers to the contents of the screen and the state of various flags when the feature is selected. By defining this for each feature, tests can be repeated with precision for each

user and replicated in future studies. The set-up is chosen with care to ensure that effective testing is possible. Screen layout and flag setting are particularly important when evaluating sound maps. If you have a screen containing many objects, each time the cursor moves over an object a message may be given. This is undesirable in many cases as it distracts the user from the purpose of the test. Unnecessary messages can be filtered out to avoid distractions.

#### **9.1.3.2 Level of operation**

This defines the tasks which the user must perform and those which the evaluator performs. It is a distraction, for example, if the user is asked to enter the key combinations required to select a feature. This, and other actions not a direct part of the test, should be performed by the evaluator. In general, the evaluator performs tasks on the keyboard whereas mouse-based activity is done by the user.

#### **9.1.3.3 Recording feedback**

The information to be recorded is defined for each feature. This includes measurements which can be made, questions to be asked and observations to be made whilst the feature is in use. This allows the evaluator to record the same format of data for each user. The data recorded for each feature enables a set of results to be created, from which it is possible to analyse the effectiveness of the feature.

#### **9.1.3.4 Scoring features**

A score is to be allocated for each test. A range of zero to ten was thought to give a suitable amount of variation. The score should reflect the success of the user with the feature. It is therefore necessary to define, for each feature, success. This is not an easy definition and is largely dependent on the feature and the test devised. However, some common points exist.

User reaction to the concept is significant. If the user reacts positively then the feature is likely to have potential irrespective of its implementation at the time of testing. The effectiveness of the feature is also a factor. This can be judged by measurements, for example, time taken to accomplish a task, or by observation. A score could be affected if, for example, time constraints restrict the test. The experience of a user could determine the time required to become familiar with the concepts and the actions required.

Such factors make the creation of guidelines a rather imprecise business. In practise, guidelines can be followed for the first users, after which comparison with previous results guides the assignment of scores.

### 9.1.3.5 Evaluation format

As with the first evaluation study, staff and students from RNCB Hereford participated. Each subject was contacted by email and given a time slot of 90 minutes. Features were presented in the order given above. As well as assigning a score, all comments and observations were recorded.

## 9.2 Results

Results from this study are presented in two sections, covering scores assigned and user reaction.

### 9.2.1 Scores

Feature	User	1	2	3	4	5	6	7	8	Feature Average	Testing Order
Mouse list selection		8	8	7	8	9	6	8	9	7.9	10
Edge sound map		8	7	8	7	9	4	8	9	7.5	2
Piano sound map		7	9	8	5	8	5	8	8	7.3	4
Mouse navigation		4	8	8	6	9	5	8	9	7.1	11
Layout sound map		5	6	6	7	9	4	8	9	6.8	9
Cursor position		1	6	5	6	8	9	8	9	6.5	1
Three section positioning		4	9	4	8	8	5	6	7	6.4	8
Volume sound map		2	7	7	5	7	3	7	9	5.9	3
Separation of two points		2	8	5	6	5	7	6	7	5.8	7
Diagonal sound map		3	4	4	0	5	0	4	4	4.0	5
Target finding sound map		3	4	5	0	0	0	3	3	3.6	6
User Average		4.3	6.9	6.1	6.4	7.7	5.3	6.7	7.5		

The above table lists scores assigned for each user and feature. These are ordered by average feature score, highest first. An additional column shows the order in which the features were presented to each user.

### 9.2.2 Reaction

The response to each feature is summarised as follows.

### **9.2.2.1 Cursor Position**

To some users the concept was clear, to others there is no need for such a feature. A number of users suggested that they would prefer to know how much space remains, for example, how many characters or lines left. The message was thought to be too slow, in particular the pixel version where large numbers must be spoken. One user suggested giving the quantity before the value rather than vice versa.

A variety of responses were given to the choice of units available. Percentage was the most popular followed by inches then pixels. A lines and column approach was suggested by several users, to “make more physical sense”. Most could see a need for pixels “for accuracy” but comments also included that this is like giving “distance home in millimetres”. To another user the message was “double Dutch”. It was suggested, however, that choice of unit is “task dependent”.

### **9.2.2.2 Edges Sound Map**

Overall response to the concept was positive. Only one user had problems with its use, finding it initially “confusing”. It was said to be “very useful” to know when an edge has been hit and a use in spreadsheets was suggested, indicating when “too far left or too far right”.

Most users were able to identify edges successfully, and the sounds were easily differentiable to most users. One user suggested that there were “too many sounds”, whilst another found it hard to relate position of mouse on pad to movement of the cursor on screen.

### **9.2.2.3 Volume Sound Map**

Most users were quick to grasp the concept involved, with one user admitting to “quite liking” the idea and another declaring it “useful”. On the other hand, one user was “not impressed” and another “confused” by the number of sounds heard. The range of volumes used was thought to be too great, and possibly the repeat rate too high. This caused some irritation. However most of those users who continued with this feature were observed to be very accurate in describing the position of icons located.

### **9.2.2.4 Piano Sound Map**

In most cases, initial reaction was to the sounds, with these being described as “pleasant”, “more natural” and “more understandable”. As to the concept, this was



mostly understood, “liked” by one user and rated as “interesting” by another. One user who had not previously used a mouse had problems grasping its purpose.

Most users had success in gauging position although some took some time to get used to the idea. Three users detected missing notes, which, given that only a couple were not used from the entire piano scale, was interesting. It was suggested that you “shouldn’t miss notes”, should use the complete range, and that zones should be equally sized. This presents some difficulty in implementation as has been described.

Users were able to return to a position by remembering the note they had heard the first time. Several users expressed a preference for this over the Volume Sound Map, whilst one preferred the Volume Sound Map having been put off by the missing notes.

#### **9.2.2.5 Diagonal Sound Map**

This feature generated some negative comments. Whilst, on the whole, users could see the reasons for attempting this feature, the result was “not so useful”. In particular, it was found to be difficult to identify the range in any direction; the full scale only exists for one diagonal movement. One user had problems being unable to tell whether the cursor was moving in a straight line. These factors contributed to making accurate estimations of position difficult.

#### **9.2.2.6 Target Finding Sound Map**

Unfortunately, testing of this feature suffered due to poor implementation. The volume was too high causing irritation and confusion, with one user admitting to becoming “lost”. However, it was said to be a “worthwhile concept”. One user described how “the edge seems to catch up with you”. Another found insufficient changes in sound and absence of directional clues near the target to be a problem. The most successful user appeared to be following the sounds to get to the target but was likely to be simply avoiding the louder sounds.

#### **9.2.2.7 Separation of Two Points**

A suggested use for this feature would be in giving an idea of the separation of two pie charts in order to determine whether there is room for another in the space between. Most users were happy with the concept, and the content of the message given, although one initial opinion was of it being “not very useful”. Whilst some users were able to assimilate the message given in pixels and one expressed this unit as his preference,

general opinion was that this is far more detailed than necessary. Pixels were described as “accurate”, “picky” and “abstract”. Percentage was suggested as an “easier to understand” unit, or perhaps centimetres. It was requested, by three users, to have quantity spoken before value.

### **9.2.2.8 Three Section Positioning**

The need for this feature was not understood by most users, initially at least. One user was “not convinced” of its applications “except perhaps in aligning paragraphs”. Another, who admitted to being “not very mathematical” was unable to remember the numbers spoken for long enough to assimilate them and suggested a manual would be useful. This applied to the pixel version, which again caused some concern, one comment being that it “doesn’t make sense to use big numbers for a small screen”, and that “the message takes too long to speak”. The ratio version was far more popular, being described as “easy to understand” and “concise”. Users were able to understand what the numbers represented but relating the numbers to the object’s position was a problem in a couple of cases.

### **9.2.2.9 Layout Sound Map**

Although more time was used to test this, it was observed that it is not a feature which is easily learned. One user suggested that “a day of training” would be required to use it properly. However, improvement was noticed as users became more familiar with the sounds heard. By presenting this feature towards the end of the sessions, users had experience both of using the mouse and the basic sound maps combined in this feature.

The concept was generally understood, although one user questioned the need. One suggested a possible application; a user can create using Lotus but needs sighted assistance in checking what has been created. Another observed that it wouldn’t “speed up work” but that it will reduce the need for sighted assistance.

In using the feature, most users had considerable success with various aspects. Identification of regions as either text or graphic was generally good, as was distinguishing between regions and the background ‘desktop’. One user was able to navigate larger regions using the sounds heard, but admitted difficulties with the smaller regions. Firstly, there were thought to be “too many sounds” in a small area, requiring fine mouse control, and secondly, it was easy to “miss” smaller regions as the mouse can be moved for some time before a region is located. The need for assistance in locating

some regions was expressed. The same user suggested that a means of “cross-referencing” using horizontal and vertical orientations would be necessary to determine position.

It was suggested that the volume sound map is a “warning” and that this should be used when the cursor is on the background, with the piano sound map for regions rather than vice versa which is currently the case. The piano sounds were again found to be popular and useful in helping determine position on the desktop. One user liked the auto switch off facility stating that “noise would affect concentration”.

One user suggested that the feature is “difficult to analyse with” and found it “hard to form a mental concept” with. Another suggested using speech “unless there is a good case for using sound”. Another liked the sounds heard but expressed a preference for the vertical orientation.

#### **9.2.2.10 Mouse Based List Selection**

Every user was able to understand both the concept and need for this feature, and able to make selections from the list. Some users were so adept at controlling the mouse that navigating the list caused no problems. However, a couple of users who were not so controlled with the mouse navigated the list but with some apparently random movement.

The idea of running applications without finding icons was described as “appealing” whilst another user encouraged an increased use of lists within the system.

#### **9.2.2.11 Mouse Navigation**

At the end of each session the performance of the user with the mouse was recorded. Encouragingly, only one user was not ‘fluent’ by the end of the session; the rest either performed well throughout or had improved significantly with experience. The one user moved the mouse too quickly and haphazardly. Of the remainder, the majority were precise, able to make small movements, and aware of the direction in which the cursor moved in response to their hand movement.

### **9.3 Review**

Whilst the results from the first evaluation study indicate the feasibility of using various ‘tools’ in providing access to multi-dimensional information, this study focussed

on individual features in order to assess their effectiveness in the system. Each feature is discussed followed by a review of the average scores calculated.

### **9.3.1 Review of features**

#### **9.3.1.1 Cursor Position**

Results suggest that the need for this feature was mostly recognised, but that implementation could be improved. It was felt that pixels provide too much detail and that giving values as a percentage of screen dimensions would be preferable. Furthermore, giving quantity before value would make more sense to some users. This is certainly valid, although when more accustomed to the feature you would know what you were going to hear and would probably only need values. The need to provide a 'verbosity' setting to allow for user preferences was emphasised. In general assimilation of position was good.

#### **9.3.1.2 Edges Sound Map**

This feature was generally found to be useful, with users able to identify edges hit. However, it was observed that having a different sound to identify each edge was not really necessary as it was the hand and mouse movement that provided the means of identification. As there are so many occasions when sound is or could be used, the system may have so many different sounds that learning what each represents may be difficult. Keeping the total number of sounds used to a minimum should be an aim. Using one rather than four for this sound map is a good example of how to achieve this aim without detrimental effect.

#### **9.3.1.3 Volume Sound Map**

The concept of using changing sound to indicate position and direction of movement was new to the users, and took some time to get used to. However, once users had become familiar with the range of sounds heard, some success was observed in identification of position of screen objects. It was felt that volume changes are not ideal because there are only a small number of distinct volumes within an acceptable range. The use of volume should be retained, it was felt, but to reduce the above problems it could be restricted to use in small areas with a small 'strip of silence' inserted between volume changes to assist in detection of changes.

#### **9.3.1.4 Piano Sound Map**

This sound map was successful in enabling the user to determine position of both the cursor and screen objects encountered. A significant factor in this success was the use of piano sounds which were both pleasant and familiar. This reinforced opinions given in the earlier study and emphasises the need to develop the use of music further. Users again found missing notes a distraction, therefore this should be avoided. Users also counted the number of sounds heard when moving between objects which suggests a further use for sound maps.

#### **9.3.1.5 Diagonal Sound Map**

Whilst it was worthwhile experimenting with this feature the results suggest that it would not be useful pursuing its development. The disadvantages, namely the varying ranges and inability to determine direction of movement, outweigh the benefit of this sound map - the ability to use a single sound change to convey two dimensional information. It is unlikely that improved implementation and increased user training could compensate for the inherent problems.

#### **9.3.1.6 Target Finding Sound Map**

Results suggest a limited success for this feature, with the implementation making sufficient testing difficult. With better implementation, in particular, using changing piano notes rather than decreasing volume to guide the user towards the target, it is expected, based on user performance with the feature as is, that users would be able to locate an object. However, the need to be able to do this was not confirmed by this study, and the speed with which target finding could be achieved would not, it appears, give this method an advantage over a more conventional one.

#### **9.3.1.7 Separation of Two Points**

This is an interesting feature because it uses speech to convey information which could be picked up using sound, for example by using a sound map. It is also a feature for which users were able to suggest possible applications. By using speech, it is possible to give an accurate description of the relationship between the two points. However, the more descriptive the message given, the longer the message becomes, which in some circumstances is not ideal. To cater for individual preferences and abilities, this feature, and others which involve announcing measurements, would need some customisation. Firstly, the user would choose the measurement given. Pixels give a very accurate view

but are not easily assimilated and the large numbers involved increase message length. Percentages and possibly inches or centimetres were more popular, and are sufficiently accurate for most uses and users. The second form of customisation required would be the introduction of a verbosity setting. This would govern the description of quantities given, which can be reduced as the user gets to know what the quantities represent.

In developing this type of feature further, it is worth recalling the comments made by users concerning giving an overview first, with the ability to go into more detail later if required.

### **9.3.1.8 Three Section Positioning**

It was generally accepted that you need a means of determining the position and size of an object, but does this feature provide this capability satisfactorily? Once again there are advantages and disadvantages of the method. A lot of information can be given in a short message, which is useful provided the information can be assimilated. The study did not provide a definitive answer. It was observed that not all users were clear what the numbers represented and the three different units of measurements available added unnecessarily to the confusion. It would have been better to concentrate on one unit, and judge the concept rather than the choice of unit which had been addressed previously. Results were, however, positive enough to suggest further development and testing of this approach.

### **9.3.1.9 Layout Sound Map**

It was expected that this sound map would cause some difficulties due to the amount of unfamiliar concepts involved and therefore more time was spent with this than with the other features.

The encouraging aspects were the response to the concept, and the willingness of users to attempt to identify what the various sounds related to. Whether a feature such as this is useful was only questioned by a small percentage of users. The suggestion that you should use speech “unless there is a good cause for using sound” is valid, but to provide comparable functionality using speech would not be feasible for this feature. Sound is not used instead of speech here, but presents what can be considered as ‘background information’ whilst speech is available for ‘direct’ output.

The other ‘negative’ comments are of more concern, these being that the layout sound map is “hard to analyse with” and that when using it it is “hard to form a mental

concept". These are significant comments in that they relate to a fundamental part of the work. The user must be able to use the features available to construct some representation of the screen which will allow interaction. This includes determining what objects are present, how they relate to each other and how they are to be manipulated. These comments cast some doubt as to whether a sound map could enable this. There are two questions which obscure this issue. Would increased exposure and an element of user training have helped the users to build a "mental concept"? And could a better implementation have improved interaction? Bearing in mind the generally positive response to the basic sound maps and the suggestions made regarding both implementation and training, the conclusion is that further investigation is needed.

Firstly, implementation can be improved using a number of suggestions made by users, these relating both to the individual sound maps and the way in which they combine. It is expected that with these amendments, a layout sound map could be used as a 'background' feature which would be switched on and off easily, and work in conjunction with speech based features to provide a suitable means of accessing a range of screen information. The process of building such a concept is time consuming, requiring testing at each stage. This project has started this process, and shown there to be some benefit to the visually impaired user. Further training and testing would be needed to determine its success.

It was observed that this feature highlighted an advantage of using the mouse over using the keyboard to move the cursor. Users who preferred the keyboard moved slowly, and could 'miss' objects. You can move from side to side and not encounter any objects, then move down a little and repeat this, and have missed objects between the two lines. With the mouse, although movement may be less precise, it is very fast and therefore covers more of the screen in the same time period. Because there is feedback indicating position the movement is not random. Users tended to encounter objects more easily. The cursor keys can then be used for precise movement from that point if the user is unable to move the mouse accurately.

#### **9.3.1.10 List Selection with a Mouse**

The response to this feature confirmed that the mouse can be used for list selection, but the question of whether it should be used is debatable. Results showed that the users were mostly able to control the mouse with sufficient precision but fail to show any advantage over using keys to manipulate a list. If users could select more quickly with



the mouse, or have more control over the list, then you could show that this feature improved interaction. However, the only advantage exhibited was that it did not need additional key combinations to be reserved. As the number of features available increases, so does the difficulty of finding free key combinations. Whilst earlier in the project this was considered an important issue, it is now thought that alternative means of accessing features from the keyboard can be used, and that lack of key combinations is no longer a valid reason for using the mouse in this way.

### **9.3.1.11 Mouse Navigation**

Results suggest that users are able to use the mouse as long as feedback is given and training is available. However, the mouse should only be used where it offers an advantage over the keyboard, and where possible, keyboard alternatives should be available.

### **9.3.2 Review of scores given**

There are problems with assigning scores, as described later, however, the scores obtained do show some patterns.

The highest scoring features all use the mouse, and include three sound maps. Near the centre of the scoring range are those features involving a spoken message. This is perhaps contrary to expectation, but may be explained by the fact that the spoken message given in each case contained a quantity of information, unfamiliar in format, requiring assimilation. The sound maps, on the other hand, are more abstract, and don't rely on a precise assimilation of the sounds heard. There were a number of implementation issues raised concerning the spoken messages, including unit of measurement, order of presentation, voice attributes including speed, and customisation.

The ordering of the three spoken message features is as expected, with scores decreasing as the complexity of the message increases.

The lowest scoring features, the diagonal and target finding sound maps, are explained by flawed concept and poor implementation respectively.

Order of testing each feature does not appear to have influenced scoring significantly. Some users were observed to lose interest in sound maps once one or two had been encountered with little success. It is possible to pick out unusual scores in the table by looking for numbers which don't fit the general decrease in scores down each user

column. A low score near the top or high score near the bottom would suggest that a user had not followed the general trend, but this is not clearly apparent.

Of greater significance is the effect of the results from two or three users on the average scores for certain features. For example, the individual scores for the three section positioning feature show three high scores and three low scores. This is repeated for the volume sound map, but with different users scoring badly in each. This suggests that some features will not satisfy all users and that features which are of benefit to some users will not help others. Overall scores can be affected by these types of features, making the drawing of conclusions from the average scores a dangerous business.

# Chapter 10 Evaluation Study Three

## 10.1 Method

### 10.1.1 Aims

The second evaluation study demonstrated that user comments and observations made during the tests provide more useful information than a system of scoring features unless a more robust method of scoring can be established. A further observation was that the emphasis was too strongly upon evaluating the implementation of features when it is the underlying concepts that require response.

The aim of the third study was to acquire a set of data in the form of user comments and observations made.

As a background to this study, consider the following overview adapted from [Strauss 1990]. Qualitative research can be considered as any kind of research that produces findings not arrived at by means of statistical procedures or other means of quantification. There are three components. Firstly, data. This is sourced from interviews and observations. Secondly, analytic and interpretative procedures. These are used to arrive at theories or findings. The third component is a written or verbal report on findings.

### 10.1.2 Procedure

This study was able to benefit from the experiences of the previous evaluation studies, and from some significant developments made in the development of Talking Windows.

The evaluation process, from contacting users to conducting interviews has been refined. A simple but previously unused tactic was to contact users personally. Initial emails sent to each staff member and student at the college sought replies direct. Previously, the contact had arranged for users to attend. The advantage resulting from this was that each user could be contacted personally prior to the study. In this communication, the nature of the work, the format of the sessions and requirements of the user were discussed. This ensures that the user knows what to expect. Furthermore, it is useful to have communicated with users prior to the study rather than the session itself

being the first contact. In many cases the background/experience of the users was, as a result, known before the study.

In this prior communication, the experimental nature of the system was stressed. In previous studies, problems arose when users expected to be using something closer to a complete product. This was seen in a small number of cases to cause disillusionment, and led to the temptation to focus on the more 'standard' features available. This is avoidable if the users are kept informed.

As has been described, user comments are as useful, if not more so, than their actions. Therefore a means of encouraging comments was sought. The first strategy was to explain this to users prior to the study and at the start of the session. In previous studies, some users would pay close attention to the sounds heard but not respond automatically. To counter this, questions were asked whenever convenient, that is, without interrupting a feature. Whilst this sounds obvious, it is possible to concentrate on observing the users actions whilst neglecting to encourage spoken feedback. Time spent with users is limited, therefore it must be used efficiently.

Some aspects of earlier versions of Talking Windows made the testing process awkward. The first problem was the speed of refresh. If an action inadvertently caused the screen to be changed, it may take several seconds to re-scan the screen. Often the change would then need undoing causing further delay. As well as wasting valuable time, this can cause irritation. Accessing features had a similar effect, when you may need to press several key combinations to call a feature. It was felt, after earlier studies, that these issues should be addressed. This third study benefited from this, with mechanisms in place to minimise disruptions.

Some features had suffered previously from sound and speech limitations caused by the equipment used at the time. The use of a more adaptable sound card and speech synthesiser significantly enhanced the interface encountered by the users for the third study. Of particular relevance here was the ability to control with precision the volume of sound and speech.

A move away from individual features, and the particular GUI used, towards concepts was required. To achieve this, less emphasis was placed on Talking Windows, and more on using methods available to perform tasks. The approach used was to identify examples of what could be termed as multi-dimensional information. This included a

GUI screen, typified by a window with buttons, text and icons, a document, with paragraphs, text attributes and layout, and a table of data.

The developed system, Talking Windows, together with its document access component, has a number of methods available to the user for accessing and manipulating these types of information. The aim is to allow the user to choose the tools which they will use. This of course requires some guidance, in particular, prompting on the methods available. However, as a result you not only determine how useful the current methods are, but also what else is needed.

The format chosen was to present the user with a test screen showing a representation of the information source. The evaluator then led the user through the methods available, asking questions, observing reaction, and noting all comments made.

As an example, consider the case of a graphical document. A representation of a sample document was displayed on screen. The concept of considering any document as a set of related regions was described, followed by the available methods - title, region count, region overview, region navigation, content of regions, layout of regions. For the latter, a choice of sound maps are available.

The same strategy applies for a windows screen, which differs in that a document is often more static. Similar methods are available, for example, sound maps, descriptive mechanisms, attribute presentation techniques, navigational tools. Likewise for tables of data.

As an addition, the concept of drag and drop was introduced, with the user asked specifically to 'move' an item to the centre of the screen using the sounds heard to locate, pick up, drag and drop the item as required. As an example of manipulation of multi-dimensional information, it was interesting to gauge user response.

It was hoped to spend between one and two hours with each user, therefore, users were asked to come when they were free for at least one hour. Slots were set at approximately two-hour intervals. Whilst you can move quickly through the tests planned in order to cover all the required topics, it was felt that it would be more effective to adopt a more relaxed approach. If the user is providing useful feedback on a feature, this should not be hurried. It does not matter if this means that some tests have to be skipped as it did in the previous study. To ensure that each subject is covered, subsequent users would start on a different topic. This is achievable because the three

main subjects, document, GUI screen and table of data, are not required to be presented in order.

However, this was largely irrelevant as in most cases there was time to cover all the intended ground, with time for discussion to conclude the session.

## **10.2 Results**

Results obtained from the third study consist of a set of recorded observations, user comments and answers to questions posed during the sessions. In this section, these results are presented grouped not by user, but by subject.

### **10.2.1.1 Access to a graphical document**

The first feature encountered by users was an announcement of document title followed by the number and type of regions present. Reaction to this was positive from all users. One suggested that the region count gives an idea of “scale”, and that if you know the number of components you are expecting to encounter, you will be less likely to “miss” any. The example quoted was in word processing a document where you may be unaware of the existence of a graphical element. Breaking down the document into regions was “a good approach”, whilst the message was in “a good format” and “very useful in giving an idea of document content”. It was useful for “getting significant information quickly”, and serves as an “indicator or prompt” giving an essential overview “before delving into more detail”.

There were several comments on the implementation of these messages. One was that the message was “far too slow”, which is not surprising as the voice chosen from the range available was chosen for its clarity rather than speed. A user would have the choice of voice to suit his or her individual preferences. Other users mentioned the importance of the verbosity setting, commenting that “blind users tend to be very impatient” therefore “only the bare bones need presenting in speech”. It was said that “a slow rate of speech can be as tiresome as an overly verbose message”. One suggestion was that “warnings could be given in a different voice rather than slower”.

Navigation of regions was, according to one user, “too slow”, this being due to waiting for speech to complete. A “Go To” feature would be “useful”. Another user was content with the speed, and liked being able to hit a key to hear the title of the region

spoken, and spent some time cycling through the regions attempting to “visualise, no, mentalise” the document. In general this was considered to be “essential to have”.

The next stage of document presentation was that of giving size and position of regions. One user remarked that this can be used to “gauge the importance” of a region which implies that size is an indicator of importance. The feature helps in “giving the dimensions of the document”, at least, its component regions. One user suggested its use in giving the “height of bars in a graph, and perhaps their relation to each other”. However it was said to be “not an everyday feature”. The rough position feature, which gives a simple description such as “top, left”, was, on the other hand, good at providing a “visual picture” of the document and a “sense of what is there”. One user pointed out that this was “more meaningful” than the three section positioning feature, another commented that it was “informative”, and should be “given before the three section position”, which relates to the point made concerning hearing an overview before going into detail. A number of uses for this kind of features were suggested, including internet, desktop publishing and spreadsheets. One user described three section position as “not too bad” and potentially “very useful”, particularly when considering an application form with a logo contained. Combined with region count this could “assist in consideration of a form”. The same user described rough position as “very good - simple yet useful.”

### **10.2.1.2 Access to a graphical user interface**

Whilst the methods described above for accessing documents apply also to a GUI screen, so sound maps apply also to documents. Having used the document to illustrate the concept of breaking down a complex info source into more manageable components, the GUI screen was used to present sound maps in this study. The reason for this was to provide the most realistic scenario to the user.

Whilst using the Volume Sound Map to navigate the GUI screen, it was observed that some users used it more to identify where the mouse was travelling rather than how this movement related to screen location and contents. One user commented that he could “keep the mouse on a level”, and felt that this was “a useful thing to be able to do”. It was observed that users were able to identify slight changes in volume, with one remarking that only “very slight movements are necessary” to hear changing sounds, and that this gives “good accuracy”. Changing volume as the mouse moves vertically was felt more appropriate, to “ensure that you keep on a straight line”. One user said that he liked the way that by combining the sound map with the speaking of object names encountered



he was able to identify “where the object is”, but that it would be “more useful” if you could return to previously encountered objects.

The use of changing note length to indicate position suffered from implementation problems in that the instrument chosen was said to “fade out” therefore not giving as crisp a sound as required. Using a different instrument would “improve the feature”. Nevertheless, users were able to detect the changes in note length, and one was able to identify the type of region below the cursor “very easily” once more acquainted with the sound map. It was observed that users who had some difficulty using the mouse struggled to make headway with this particular sound map.

One user said that he “enjoyed” using the instrument sound map, but that it was a “distraction” in helping identify regions. Some of the instruments used were identifiable but “sound poor”. Another user observed that this sound map “makes it more difficult to identify regions both from each other and from the background”. However, he could ‘see’ how far a graphic region was from the edge of the document.

The stereo sound map proved more popular, with users more able to tell where the cursor was. One user commented that it made the cursor “appear more real”. The absence of changes in sound when moving vertically was said to be “helpful”, as this can be an “unnecessary distraction”. The same user suggested that “you know when you are moving downwards”, but not, he conceded, “the scale of this movement”. According to one user, the stereo balance sound map was “the best of the four”, and “very usable”.

One of the concerns regarding sound maps is that of irritation, a point which was raised with the users. The general reaction was that a simple on/off switch would be required. It was observed that users did not appear irritated as was the case on occasions in the previous study. Improvements in implementation and volume control helped here. One user suggested that he experienced “no irritation” at the constant warbling sounds heard, but felt that “it may cause irritation in an office environment”, mentioning the Volume sound map as the main potential culprit. Another user said that “when you get used to the sounds they are not irritating”. Another claimed no irritation but that it “depends on the preference of the user”. He felt that it “could be a nuisance to other people in the room if not used through headphones” but that “the use of music is a factor in reducing irritation”.

The use of music was universally popular, being described as “pleasant”, “exciting” and “effective”. One user stated that “music is a good way to present information, and, on occasions, better than speech”. Another welcomed use of music, saying “it is faster than speech in presenting some kinds of information”.

There were many general comments made regarding sound maps. One “liked the concepts” of the sound maps encountered, describing them as “a good feature”. Another claimed to be “excited” about using them further and their future development. His comment was that they can “guide you to where you need to go, can keep you moving in a straight line, and keep you on a line even when the cursor is within a region”. Another felt it was “more comfortable using the sound map than not”, as he didn’t need to concentrate on the screen which can be “very tiring” when you need to look at it very close and at an angle. He added that if the need for this is reduced the computer can be used for longer periods of time. Incidentally, this user prefers to use a black screen with white text rather than vice versa for this reason. Sound maps are “easy to use” and could be used “for long periods with no strain”. This user also noted a similarity with the use of tones to guide users in Hal4.

It was suggested that customising would be useful, allowing choice of sound attribute change according to user preference. One user suggested that sound maps could be used to support cutting and pasting activities, although it was not said how. Another user gave the less specific word processing as a potential use. Another user was confident, in response to a concern raised about the quantity of sound used, that “sounds can be learned fairly easily”.

### **10.2.1.3 Access to data tables**

The initial announcement of number of rows and columns was felt to be a good way to indicate the “size” of a table, and a “good overview”, giving a “useful impression of what there is to deal with”. One user expected the number of rows to be given before the number of columns. There is an argument for either case but this is a reminder of the need for careful consideration of even apparently simple features.

It was suggested that a small pause between row and column labels would help. In this case, speed of presentation is a low priority compared to announcing significant information clearly. Labels should not “run into each other”. Another user suggested that

it would “be useful to distinguish column labels from entries”. As a preceding sound was used for this purpose this suggests that the sounds chosen were not sufficiently distinct.

The notion of “current entry” attracted some ideas. One was to highlight it for partially sighted users. Another suggestion was to use a different voice to announce its value. A spreadsheet-like approach of labelling each cell with a letter/number combination, F1, G5, for example, was recommended.

It was observed whilst navigating the table that the sounds heard were inconsistent with a feature encountered previously. This related to the direction in which sounds changed, a small error but noticed instantly by the user, significantly.

Rather than use the preceding sounds to keep track of position in the table, one user preferred to count rows, but did suggest that with time these sounds would be learned and “assist in navigation”. Another user suggested that changes in sound and choice of sounds were “good”, and that alternative voices for speaking values “useful” and “assisted clarity”.

Having been through the summary, navigation and position functions, users were set a task of identifying the row with the highest value in a particular column, requiring the user to make use of these features.

Each user was able to navigate the column labels until the specified column was found, before navigating down that column to find the highest value and reading off the row label. Speed varied between users. Some took some time, but a couple were quicker than expected having easily grasped the methods required. The need for a Go To function was mentioned by more than one user, with mentions of cell labelling occurring again. It was also suggested that a facility to “announce the smallest or largest entry within a selected region” would be useful.

The need for clear and concise status information messages was emphasised here, with examples of line and position in a text processing situation, record and field in a database application, and spreadsheet cell given.

Reaction to the support added for drag and drop was limited by the implementation, which, being a recent addition lacked refinement. However one user was “very pleased” that this had been added, and was able to use it successfully. The ideas behind the feature were quickly grasped, and the user was able to drag the item to the bottom left corner of

the screen. A task of positioning the item in the centre of the screen was then set. The user achieved this very efficiently, relying totally on changes in sound heard. It was suggested that different instruments could be used to distinguish between horizontal and vertical movements, and that learning which instrument represented what movement “would not be a problem”. It was also suggested that as an alternative, you could use spoken directions. If so, the speed of presentation would require some thought.

Another user was enthusiastic about this concept as it was “useful” in supplementing his very limited vision. He is able to see objects when very close and at right angles to the screen provided they contrast sharply with their background. Therefore moving icons can be very difficult, sometimes impossible when only the outline is shown.

Some interesting comments were made in concluding discussions. One was the need to make the mouse pointer “more distinct” for assisting partially sighted users. Similarly, highlighted or underlined items could be announced. As these suggest important items or significant information, they may be included in an overview of content of an information source.

One user described how he is able to create charts in both DOS and Windows, but “can’t always see the format or layout” of the chart, and welcomed the possibility of some means of non-visual presentation. Similarly, descriptions of word-processing documents, graphics created in any application, and maps/routes would help. It was mentioned that there is very little internet suitable for MS-DOS and therefore a graphical approach would be “essential” for many blind people.

An observation was made concerning the evaluation process but also relates to system design. On more than one occasion a user would attempt to use a feature before its use or purpose was fully explained. This is a natural reaction but emphasises a need to strive for intuitive features, with potentially harmful effects being guarded against. So for example if a user presses a button that may initiate a lengthy procedure or set a significant switch, a warning should be considered.

To conclude, a final comment from one user was that the system is “extremely promising and could certainly be developed further”.

## **10.3 Review**

This section discusses the results from the final study undertaken.

### **10.3.1.1 Access to a graphical document**

The results suggest that a valid approach to enabling access to a graphical document has been taken. The approach was to consider a document as a set of related regions, and provide the methods needed to identify regions and their interrelationships, and manipulate these regions. Users liked the ability to get an overview before breaking down a complex document into more manageable 'chunks'. The announcement of title is an obvious starting point, but the description of number and type of regions present less so. This was popular with users for giving the initial overview, enabling them to have an understanding of what the document contains. The approach is reliant on a document recognition system providing the Off Screen Model, however, as described previously it is a reasonable assumption that the required information can be provided.

The methods available for navigating document regions were shown to be useful, but with some minor implementation concerns. It was felt that speed was essential, and that this was lacking. However, allowing users to set their preferred voice and speech rate, coupled with use of the verbosity setting, will greatly increase speed of navigation. Two additional features not available at the time were a 'go to' feature, and the ability to interrupt speech. If you do not have to wait for a message to complete before continuing, navigation will be much faster. The 'go to' feature could work in a number of ways, and would save a considerable amount of navigation.

The methods available for determining size and position of regions were once again found to be useful although a number of small implementation improvements were suggested. It was encouraging that users were able to give examples of where this could be useful in everyday applications. The main implementation concerns were once again relating to message format, precision, speed and customisation.

It is now possible to consider the extent to which the work enables access to graphical documents. With the methods available, the user is able to take a document representation and hear an overview of document content including a count of the number of text regions, graphics regions and regions containing both text and graphics. The user is then able to navigate to a region that is of interest. The options available at this point are numerous. If the region consists of text, this text can be read to the user. For this the user needs to control what is spoken and when. Facilities to repeat text, move forward or backwards through the text and change speech attributes are some examples of what can be provided.

It the user wishes to edit the text, there are two issues. The first is provision of non-visual support. This is available using either the Talking Windows Off Screen Model and screen-change detection or potentially a commercial text access product. Straightforward access to text is trivial. The more important issue is the effect of editing text on the composition of the document. By adding text to a region the size and position of the region may alter, as may neighbouring regions. A new Off Screen Model can be calculated using the recognition module and this used by the user to build a new impression of the document.

If the region contains graphics, the options open to the user are more limited. One constraint is governed by what the recognition module can produce for the graphic. If the graphic is a picture, a description of this picture would be required. If the graphic is a graph or a chart, it is assumed that the underlying table of data will be available for presentation to the user. This subject is described later in this chapter, as is the moving of regions.

To summarise then, the work on document access provides a number of useful features for viewing a graphical document, however, amending the document requires the addition of methods found either on Talking Windows or elsewhere.

### **10.3.1.2 Access to a GUI screen**

A GUI screen can also be seen as consisting of a set of regions. Typically the 'window' is at the first level, with each window potentially capable of being considered as a further set of regions. The differences are the less static nature of a GUI screen and the potential for overlapping regions. In practise, visually impaired users suggested that they would tend to work with maximised windows.

The methods found to be popular for documents in this study also apply to a GUI screen, namely, an overview of the number of windows, their types, positions and inter-relationships. In this study, the use of sound maps to add to the information available to the user was tested.

The unexpected benefit of some sound maps was their ability to inform the user of the direction in which the cursor was moving. This is in addition to indicating position of the cursor, which was shown to be reasonably successful. Whereas prior to this study there was only change in volume or piano note to guide the user, the addition of stereo effects

and extra instruments provides more options. It is now possible to build a sound map consisting of several of the 'basic' sound maps to give a range of information.

The two concerns about this approach are the users ability to learn and make use of the sounds heard and the irritation factor. The latter was found to be a minor concern. Sound maps have been set to switch off automatically after a short period of cursor inactivity, and on as soon as the cursor moves again. An on/off switch could be made more accessible, with a right mouse button click, for example. Using headphones would also help in an office environment.

The other concern, that of the ability to learn and use the sounds was not seen by the users as a significant problem. It was necessary for users to ask what a sound meant, but a mixture of supporting features and training would help with this.

A number of implementation improvements were suggested, such as using volume changes more as a warning than as a standard part of a sound map. The other sound changes used were found to be pleasant whereas the louder sounds can irritate.

It is now possible to evaluate the contribution which sound maps make in enabling access to graphical user interfaces and documents. The aim is a single sound map which will be used in conjunction with speech-based features to enable large amounts of information including spatial relationships, cursor movement and object position to be given. The process required to reach this stage is lengthy, requiring much experimentation and user testing, and some fine and not-so-fine tuning of the implementation. The problems encountered are those of soundcard limitation and doubts over whether there is a need for this kind of concept.

Much progress has been made towards the goal, with basic sound maps such as the piano and edges sound map, being testing in each study, and others introduced at various stages. These have been refined and combined into complex sound maps such as the layout sound map. Issues such as the potential irritation of continuous sounds have been addressed, and the ability of users to make use of the sounds heard confirmed.

Indications are that further work is necessary to refine implementation, but that there is potential for sound maps to play a significant part in an access system. To assess their value in detail, future studies will benefit from users having more time to become accustomed to the sounds heard. The use of music was shown to be popular and certainly worth pursuing. These conclusions are backed up by user comments as described earlier.



Sound maps, together with the methods described for enabling access to graphical documents, provide a range of features with which users can access a GUI screen. It is now possible to determine screen content and navigate to required items. These can then be manipulated as required. Text can be read, and, if required, amended, and windows opened and closed.

It was necessary to concentrate on individual features rather than asking users to accomplish particular tasks, such as, for example, opening a word processor and entering text. This was due to the time which it took to reconstruct the off screen model each time the screen changed. This is not to suggest that the method used for maintaining the off screen model was not valid. For practical reasons, an early version of the recognition software was used, and this, combined with a slow processor, combined to give a full screen recognition time measured in seconds running into double figures. Later versions on a fast machine would reduce this significantly. When testing with users, it is a considerable disadvantage when you close a window, say, to have to wait twenty seconds before you can do anything else. This had an influence on the way in which the system was tested.

### **10.3.1.3 Access to data tables**

Once again, the approach taken is based on breaking the problem down into smaller parts. So, at first, you have an overview of the table structure before going into further detail. This was seen as useful by users, with the initial overview letting them know what kind of table they were dealing with.

The methods available for navigating the table, finding a row or column, and then an entry, were subject to now familiar issues. Speed is important, as is message format, consistency and use of non-speech sounds. A number of refinements were suggested, which was expected since this functionality was only completed prior to the final study and so had not been encountered by users before. However, with the exception of a labelling mechanism and 'go to' feature, there were sufficient methods available to enable users to access values in the sample table. A significant feature was found to be the preceding of each spoken value with a non-speech sound indicating whether the value was a row label, a column label or an entry. Changing the sound to indicate position in the table was an interesting idea which requires more practise to be of use, with users mainly counting rows or columns to keep track.

To conclude, there are some minor amendments to be made, but provided a table of data can be obtained by the system, its manipulation by visually impaired users can be enabled without difficulty.

# Chapter 11 Discussion

This chapter reviews the evaluation process, the merits of the techniques used and improvements which could help future work. It also reviews the project as a whole.

The methods used to evaluate the success of the work were not without problems. The main flaw was the way in which scoring was attempted. Prior to the study it was felt that this was essential in order to allow both comparison and an analytic judgement of the success of the work.

The use of scoring for this type of system, however, was not successful. Whilst guidelines for giving scores were created, they vary according to the kind of feature and therefore comparison of feature scores is difficult. Assigning a single score for a feature is flawed because there are a number of factors used to give that score but to allow valid comparison these factors should be scored separately.

It is possible to perform a quantitative analysis of this type of work, but using a better method. Research in this area should be undertaken prior to an evaluation of this type.

There were further problems encountered when users were unwilling to use a particular feature. There are occasions when you present a feature and the user says "Ok, what's next?". Is it better to persevere with this feature or attempt one which the user may be interested in? It was also observed that scoring leads to concentration on individual features rather than consideration of those features as part of an overall system. It is the system which enables access to the information system, with users deciding which of the tools to use for the task at hand.

It was useful to have learned this early enough in the project to be able to focus the final study on obtaining qualitative data. The first evaluation study, whilst informal and not precisely planned beforehand, was found to be a particularly useful way of introducing the system and its concepts and learning about the evaluation process including user availability and requirements prior to more in-depth studies.

The first two studies were a significant help in planning the final study. There are, however, further steps which could be taken to improve the process. One, a simple point,

is to record each session to ensure that user comments and reactions are not missed. A video recording could be very useful.

Another improvement would be to give users more information prior to the session. Ideally this would involve use of the system beforehand. This requires independent use of the system and a greater time commitment from the user. But somehow it is important to increase the time which the user spends using the system. Reaction will be different in some cases when the user has had more time to become familiar with the concepts and methods used. As has been noted previously, finding users with the time to spend on an experimental system and co-ordinating the evaluation is not easy.

Initially, it was intended that potential users be sent copies of the Talking Windows prototype with installation instructions and telephone support. It was therefore necessary to design this requirement into the system. For example, the system should be capable of being used without sighted assistance, and contain basic screen reader functionality and not just the experimental features. A basic help facility was added, as was a key confirmation facility.

It became apparent, however, that this approach was flawed. At that time, maintaining the Off Screen Model was too time-consuming. Furthermore, because of the number of features available, it required a number of key combinations to access a particular feature. Users who received the system were put off both by the speed and the difficulty of using it alone. Key combinations required learning, and, once features were accessed, they were largely irrelevant given the long pause whenever the screen changed. With no incentive to use the system for everyday tasks, the approach was soon rejected.

The result was that a great deal of time had been spent on implementing a complete system and preparing it for solo use. This time could have been spent more profitably in other areas. This was not the only occasion when concentration on implementation was misguided. There was perhaps too much focus on being able to access a particular feature and in ensuring compatibility rather than developing the system purely as a test-bed.

It was thereafter decided to concentrate on providing access methods rather than building a complete system and this improved matters significantly. The Document Access System can only generate a mock Off Screen Model, but this provides a fast and effective vehicle for testing the concepts and methods developed. This in fact makes the system suitable for solo use by visually impaired users. Day to day tasks cannot be

accomplished but the system is user-friendly enough to provide an incentive for its use. Unfortunately, this was not developed soon enough for distribution prior to evaluation.

A further advantage which the Document Access System benefited from was the availability of a new soundcard and speech synthesiser. The original soundcard was rather limited being an old model even then. It is as well to keep up with technology in this area if possible. The new soundcard enabled improved sounds to be created and the more adaptable speech synthesiser allowed choice of voice and use of voice attributes which improves the interface significantly.

The work undertaken covers a great deal of ground. This may be seen as a disadvantage, in that it makes thorough evaluation of all the aspects impossible. On the other hand, there are a number of benefits. The project has gone a long way towards developing a coherent access system, and points towards a number of areas where there is potential for further development. Although there have been some ideas which have not been seen to be worth developing, and others where thorough evaluation has not been possible, they have been attempted and documented, and therefore contribute to the development of the information access system.

# Chapter 12 Future Work

This project has explored a number of areas which have potential for future development. The ways in which the work could be advanced are described in this chapter.

The use of music in the interface proved to be an exciting development with scope for future development. The ways in which the piano scale, in particular, could assist the visually impaired user suggest that its use should be continued. The sound maps developed and tested with users provide one example of how music can be used. Similar concepts used to allow determination of position within a table of data are another.

Initially, it was intended that widely available hardware would be used to provide an inexpensive solution. This aim can still be achieved, in particular with the use of midi sound. Midi also enables musical scales to be generated without storage demands or significant processor overheads. Quality of sound reproduction is currently high and at low cost.

The use of stereo effects has been investigated by many people. This project has demonstrated a basic application for stereo balance using inexpensive equipment, namely in providing an alternative means of maintaining spatial awareness on a horizontal plane. The ability to use stereo effects without additional overheads is potentially very useful. Increased usage where appropriate could significantly assist the blind or partially sighted user.

These are useful tools, particularly for the development of sound maps. The current work shows that there may be benefits in expanding the sound map concept. In particular, complex sound maps require a more thorough evaluation before their true worth can be assessed. Deciding what sounds to use and for what purpose is part of the problem. Evaluation may prove a more difficult task. To use a sound map to better effect, a period of learning is expected to be required, so that the user can identify what a particular sound represents instantly.

The concept of using sound to indicate position of an item in a table similarly requires experience in order to receive the intended benefits. However, the use of sound to convey background information could provide an effective means of acquiring information

which a sighted user takes for granted, but is currently unavailable to the visually impaired. The use of sound attributes, in particular, the changing of sounds in response to events, has scope for continued research.

Whilst widely available soundcards enable high quality, low cost configurable speech and sound to be used for output, the increasing availability of voice input technology offers potential benefits to the visually impaired computer user. A limited vocabulary is sufficient to allow access to both GUI and screen reader. Whilst research in this area would represent a digression from the current work, there is a temptation to aim for a single, coherent information access system in which alternative input mechanisms would play a significant role. The reward for integrating the required components would be a system capable of being used productively by visually impaired subjects, making evaluation more straightforward. The components required for this exist, including fast recognition software for constructing and maintaining an Off Screen Model, a text manipulation module, plus the functionality of the Talking Windows and Document Access systems. This includes routines for manipulating sounds and speech to give the required flexibility. To progress the work, which was based on Windows 3.1, some time may need to be spent on combining these components, and updating for later operating systems.

The question of evaluating work done must be addressed by future work. If the developed ideas cannot be used without sighted assistance, the portability of the system must be considered. Maintaining communications with users prior to evaluation proved successful in the current work, and will be essential in the future.

As has been described, the current work leans more towards 'reading' of information rather than manipulating it. Work on text access and drag and drop, for example, addresses this issue, but further work is required. To an extent, this type of study is linked to the capabilities of the Off Screen Model, however, this should not restrict the work attempted. How, for example, does a blind person cut and paste a line of text? Could improvements be made, by incorporating non-speech sound cues, for example?

Whilst there is scope for a number of developments, the user must remain central. What does the user need to be able to do, and what tools can be provided? How can access to multi-dimensional information by the visually impaired be improved?



# Chapter 13 Conclusion

This work set out to investigate various means of enabling the visually impaired to access multi-dimensional information. The source of such information was initially the graphical user interface, followed by the graphical document, with the expectation that successful ideas will eventually apply to alternative sources.

A system has been developed as a platform for implementation and evaluation of ideas. Optical character and icon recognition are used to capture display screen information; this is maintained in an Off Screen Model accessible by the information presentation system.

The use of widely available and therefore inexpensive input and output devices has been investigated. For the user to enter commands to the system, the keyboard has been seen to be suitable, provided a careful mapping of features to keys is made. Means of supporting this aspect of user interaction such as a key confirmation facility have been developed.

The mouse has been used throughout the system both as an additional input device, and to investigate whether, on occasions, it is a more appropriate means of browsing screen content than keyboard-based alternatives. The inherent visual nature of the point and click aspect of mouse use traditionally excludes its use by blind computer users, however, with appropriate non-visual support, results suggest that it is a useful component in a GUI screen access system.

The increased use of non-speech sound within the user interface has been investigated as a means of enhancing interaction by increasing the quantity of information available to the visually impaired user and reducing the time taken to impart it. Sound maps have been found to be an effective means of satisfying these requirements.

In particular, the use of the piano scale to accurately represent the changing position of the mouse pointer was found by the majority of blind subjects to be a pleasant and informative presentation means. Furthermore, this suggested an increased use of music within the interface, something to be considered in future work.

The use of speech combined with non-speech cues has been investigated by means of both message format throughout the system, and various descriptions of screen content and object attributes. Employing distinct, memorable tones to represent additional information reduces the quantity of speech required and therefore time needed to deliver the message. This led to a consideration of when it is appropriate to use non-speech, and when speech is more meaningful. The speed of imparting non-speech sound must be balanced against the cognitive demands involved in assimilating an abstract tone or sound effect. An investigation of the use of speech and sound revealed that non-speech is a vital part of the interface, but must involve appropriate choice of sounds, and verbal support when required.

A fundamental difference between this system and many commercial screen readers is the emphasis placed on presenting spatial information. By providing a number of means for the user to determine the position of screen objects and their inter-relationships, it was intended that the user's understanding of the GUI could be increased and therefore interaction enhanced. A positive response from blind users of the system suggests the validity of this approach.

A second system has been developed to test solutions to the problem of enabling access to a graphical document. The Document Access System assumes the availability of the equivalent of an Off Screen Model, and develops many of the successful concepts introduced in Talking Windows.

The concept of a document consisting of a set of spatially related regions of either textual or graphical content has been used, and tools developed to enable the visually impaired user to determine content of and relationships between regions.

The validity of this approach has been addressed, with users welcoming the benefits it offers, as well as the methods in place to enable interaction.

The use of non-speech sound to provide additional information has been extensive, with results suggesting that increasing the quantity of information available can improve interaction.

The importance of customisation has been emphasised by this work, with users requiring the ability to adjust the interface to suit their individual needs. This has been addressed with a number of features, including setting of speech attributes, message

format and on-off switches, receiving an enthusiastic response. Control is a vital aspect of the system.

An investigation into presentation of charts and diagrams resulted in a table access component being developed. Incorporating non-speech sounds, navigational aids and the presentation of spatial information, this approach enabled users to successfully locate values in a table. The issue of manipulation of data, both in data tables and alternative information sources, has been considered, with existing methods, such as a text access facility, required to enable this.

In conclusion, the project has addressed a number of aspects of what is a broad area of study, and in terms of enabling visually impaired users to 'read' multi-dimensional information, achieved some success. The techniques introduced in the project provide support for manipulation of such information, and indicate a number of areas where further research can be directed for the benefit of the visually impaired user.

# Chapter 14 References

- [**Aaronson 87**] Aaronson D and Gabias P : (1987) "Computer Use by the Visually Impaired", Behaviour Research Methods, Instrument, and Computers, 19, pp275-282.
- [**Akoumianakis 95**] Akoumianakis D, Stephanidis C, Petrie H and Morley S : (1995). "Supporting user interface adaptability during the design and development process", In G. Allen, J. Wilkinson and P. Wright (Eds.), Adjunct Proceedings of HCI '95: British Computer Society HCI Group Annual Conference. Huddersfield, U.K.
- [**Arons 94**] Arons B and Mynatt E : (1994) "The future of speech and audio in the interface", SIGHCI Bulletin, 26(4), pp44-48.
- [**Arundale 94**] Arundale J : (1994) "What can we learn from Gutenberg - publishing and the technological revolution", Multimedia Technologies and Future Applications, Chapter 24, pp1-3.
- [**Begault 91**] Begault D.R : (1991) "Challenges to the Successful Implementation of 3-D Sound", Journal of the Audio Engineering Society, Vol 39, No.11, pp864-870.
- [**Begault 93**] Begault D.R and Wenzel E.M : (1993) : "Headphone Localization of Speech", Human Factors, 35(2), pp361-376.
- [**Blattner 89**] Blattner M.M, Sumikawa D.A and Greenberg R.M : (1989) "Earcons and Icons: Their Structure and Common Design Principles", Human-Computer Interaction, Vol 4, pp11-44.
- [**Bohlman 92**] Bohlman E : (1992) "Tinytalk: a Powerful, Low-cost Screen Reader", OMS Development, IEEE.
- [**Bowe 84**] Bowe F.G : (1984) Personal Computers and Special Needs, Berkley: Sybex.
- [**Boyd 90**] Boyd L.M, Boyd W.L and Vanderheiden G.C : (1990) "The Graphical User Interface: Crisis, Danger, and Opportunity", Journal of Visual Impairment and Blindness, Vol 84, pp496-502.
- [**Brewster 94**] Brewster S.A, Wright P.C and Edwards A.D.N : (1994) "The design and evaluation of an auditory-enhanced scrollbar", in Adelson B, Dumais S and Olson J (Eds), Proceedings of CHI'94, Boston, Massachusetts: ACM Press, Addison-Wesley, pp173-179.
- [**Brewster 95a**] Brewster S.A, Wright P.C, Dix A.J and Edwards A.D.N : (1995) "The sonic enhancement of graphical buttons", In Nordby K, Helmersen P, Gilmore D and Arnesen S (Eds.), Proceedings of Interact'95, Lillehammer, Norway: Chapman and Hall, pp43-48.
- [**Brewster 95b**] Brewster S.A : (1995) "The development of a sonically enhanced widget set", In Proceedings of EWHCI'95 (Moscow, Russia) International Centre for Scientific and Technical Information, 1995, pp126-129.

**[Brewster 95c]** Brewster S.A, Wright P.C and Edwards A.D.N : (1995) "The application of a method for integrating non-speech audio into human-computer interfaces", (Technical Report No. YCS 253). University of York, UK, Department of Computer Science.

**[Brewster 95d]** Brewster S.A, Wright P.C and Edwards A.D.N : (1995) "Experimentally Derived Guidelines for the Creation of Earcons", In Adjunct Proceedings of HCI'95, Huddersfield, UK.

**[Bruce 91]** Bruce I, McKennell A and Walker E : (1991) Blind and Partially Sighted Adults in Britain: The RNIB Survey Vol 1, London: HMSO.

**[Burger 92]** Burger D and Suchard J : (1992) "Methods for Improving Computer Access for the Visually Impaired", Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol 14, Part 1-7, pp1535-1537.

**[Burgess 93]** Burgess D.A and Verlinden J.C : (1993) "A First Experience with Spatial Audio in a Virtual Environment", Graphics, Visualization and Usability Center, College of Computing, Georgia Institute of Technology, Atlanta GA, 30332-0280, Technical Report Number 93-31.

**[Buxton 91]** Buxton B, Gaver B and Bly S : (1991) "The Use of Non-Speech Audio at the Interface", Tutorial Presentation at CHI'91, April 1991.

**[Calhoun 87]** Calhoun G.L, Valencia G and Furness T.A : (1987) "Three-dimensional auditory cue simulation for crew station design/evaluation", in Proceedings of the Human Factors Society 31st Annual Meeting, pp1398-1402, Santa Monica, CA: Human Factors Society.

**[Calhoun 88]** Calhoun G.L, Janson W.P and Valencia G : (1988) "Effectiveness of three-dimensional auditory directional cues", in Proceedings of the Human Factors Society 32nd Annual Meeting, pp68-72, Santa Monica, CA: Human Factors Society.

**[Card 78]** Card S.K, English W.K and Burr B.J : (1978) "Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT", Ergonomics, 21, pp601-603.

**[Chapman 78]** Chapman E.K : (1978) "Visually handicapped children and young people" London: Routledge and Kegan Paul.

**[Cherry 53]** Cherry E.C : (1953) "Some experiments on the recognition of speech with one and two ears", Journal of the Acoustical Society of America, 22, pp61-62.

**[Christ 66]** Christ W, DeLone R, Kliewer V, Rowell L and Thomson W : (1966) Materials and structure of music I, Englewood Cliffs, NJ: Prentice-Hall.

**[Clark 96]** Clark D and Henter T : (1996) "Duelling Windows", CSUN Conference, Los Angeles, CA, March 1996.

**[Colwell 99]** Colwell C and Petrie H : (1999). "Evaluation of Guidelines for Designing Accessible Web Content", In C. Buhler & H. Knops (Eds.), Assistive Technology on the Threshold of the New Millennium. (AAATE 99). Amsterdam: IOS Press.

- [Coutaz 91]** Coutaz J : (1991) "Prospects in Software Design with Multimodal Interactive Systems", in Proceedings of SITEF Symposium on Cognitive Interactions, Toulouse, France, October 24-25, 1991, pp47-63.
- [Dix 93]** Dix A, Finlay J, Abowd G and Beale R : (1993) Human-Computer Interaction, p521, London: Prentice-Hall, ISBN 0-13-437211-5.
- [Doll 86]** Doll T.J : (1986) "Synthesis of auditory localization cues for cockpit applications", in Proceedings of the Human Factors Society 30th Annual Meeting, pp1172-1176, Santa Monica, CA: Human Factors Society.
- [Doll 87]** Doll T.J : (1986) "Development of three-dimensional audio signals", in Human Integration Technology: The cornerstone for enhancing human performance, pp21-32, Warrendale, PA: Society of Automotive Engineers.
- [Doll 92]** Doll T.J, Hanna T.E and Russotti J.S : (1992) "Masking in Three-Dimensional Auditory Displays", Human Factors, 34(3), pp255-265.
- [Edwards 89a]** Edwards A.D.N : (1989) "Soundtrack: An Auditory Interface for Blind Users", Human-Computer Interaction, 1989, Vol 4, pp45-66.
- [Edwards 89b]** Edwards A.D.N : (1989) "Object-Oriented programming to build adaptable human-computer interfaces", IEE Colloquium, 16th November 1989.
- [Elmaghraby 89]** Elmaghraby A.S : (1989) "Voice recognition applications for programming environments", SouthEastCon 89, Proceedings Vol 1-3, Energy and Information Technologies in the South East, Ch 279, pp655-699.
- [Frank 93]** Frank J : (1993) "Access to Computers with Graphic User-Interfaces for the Blind", Work with Display Units 92 - Selected Proceedings of the 3rd International Conference on Work with Display Units, Vol 94, pp199-203.
- [Fountain 92]** Fountain T : "Engineering with Windows 3", IEE Review, 38, 11, pp377-379.
- [Gaver 86]** Gaver W : (1986) "Auditory Icons: Using sound in computer interfaces", Human-Computer Interaction, 2, pp167-177.
- [Gaver 89]** Gaver W.W : (1989) "The SonicFinder: An Interface That Uses Auditory Icons", Human-Computer Interaction, 1989 Volume 4, pp67-94.
- [Gerhard 88]** Gerhard E : (1988) "Modern Aids: Computer Work Stations for the Partially Sighted and Blind" Retinitis Pigmentosa: Patients' Fight for Sight, Vol 63, Part 40, pp63-66.
- [Gill 93]** Gill J : (1993) "European Project on Access to Graphical User Interfaces by Blind People", GUIB Consortium, ISBN 1 85878 004 7.
- [Gordon 90]** Gordon D.F : (1990) "Voice recognition and systems activation for aircrew and weapon system interaction", Proceedings of the IEEE 1990 National Aerospace and Electronics Conference, Naecon 1990, Vols 1-3, Ch 229, pp744-748.

**[Greenberg 93]** Greenberg S, Darragh J, Maulsby D and Witten I.H : (1993) "Predictive interfaces: What will they think of next?", in A.D.N.Edwards, editor, Extraordinary Human-Computer Interaction, Cambridge University Press.

**[Griffith 89]** Griffith D, Winfree D and Doss H : (1989) "Cognitive requirements for computer usage by the blind and visually impaired", paper presented at the First Convention of the American Psychological Society, Alexandria, VA.

**[Griffith 90]** Griffith D : (1990) "Computer Access for Persons Who Are Blind or Visually Impaired: Human Factors Issues", Human Factors, 1990, 32(4), pp467-475.

**[Hale 97]** Hale G : (1997) "Putting the user first: Developing a user-centred model of software assessment in an educational context", Learning Resources Journal 13/2, June 1997.

**[Haralick 94]** Haralick R.M : (1994) "Document Image Understanding: Geometric and Logical Layout", 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Part 165, pp385-390.

**[Harness 93a]** Harness S.J, Pugh K, Sherkat N and Whitrow R.J : (1993) "Fast Icon and Character Recognition for Providing Universal Access to a WIMP Environment for the Blind and Partially Sighted", Studies in Health Technology and Informatics, IOS Press, Amsterdam 1993. Proceedings of First Tide Congress, 6-7 April 1993 Brussels.

**[Harness 93b]** Harness S.J, Pugh K, Sherkat N and Whitrow R.J : (1993) "Enabling the Use of Windows Environment by the Blind and Partially Sighted", IEE Colloquium on Computing and Control Vision, "Access for people with disability", Savoy Place London, 5 May 1993.

**[Hartley 88]** Hartley J : (1988) "Using principles of text design to improve the effectiveness of audiotape", British Journal of Educational technology, 19, 1, pp4-16.

**[Hartley 94]** Hartley J : (1994) "Text Design for the Visually Impaired: A British Perspective", Designing Instructional Text, 3rd edition, London: Kogan Page.

**[Higgins 94]** Higgins A, Bahler L, Porter J and Blais P : (1994) "Robust matching for voice recognition", Automatic Systems for the Identification and Inspection of Humans", Vol 2277, Ch 27, pp95-98.

**[Hjelmquist 93]** Hjelmquist E and Jansson B : (1993) "Work possibilities for visually impaired: The role of information technology", Advances in Human Factors/Ergonomics, Vol 19PA, Part 175, pp498-503.

**[HumanWare 90]** HumanWare, Artic Technologies, ADHOC, and The Reader Project : (1990) "Making good decisions on technology: Access solutions for blindness and low vision", Closing the Gap Conference, October 1990, Industry Experts Panel Discussion.

**[Jansson 85]** Jansson G : (1985) "Implications of Perceptual Theory for the Development of Non-Visual Travel Aids for the Visually Impaired", Electronic Spatial Sensing for the Blind - Contributions from Perception, Rehabilitation and Computer Vision, Vol 99, Part 28, pp403-419.

**[Jarmul]** Jarmul E, Nunziata E, Perez C, Lipetz L.E and Weed H.R : "Investigation of significant interface parameters involved in the electromechanical transfer of tactile

information", Biomedical Engineering Centre, The Ohio State University and The Ohio State University Hospital, Columbus, OH 43210.

**[Jenkins 85]** Jenkins J.J : (1985) "Acoustic information for objects, places, and events", in Warren W.H and Shaw R.E (Eds), *Persistence and change: Proceedings of the first international conference on event perception*, Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

**[Johnson 98]** Johnson V, Petrie H and Mercinelli M : (1998). "An investigation of the user needs for screen-based telephony for people with visual impairments", In I. Placencia-Porrero and E. Ballabio (Eds.), *Improving the quality of life for the European citizen: technology for inclusive design and equality*. Amsterdam: IOS Press.

**[Kurze 95]** Kurze M : (1995) "Giving Blind People Access to Graphics (Example: Business Graphics)", *Proc. Software-Ergonomie '95 Workshop "Nicht-visuelle graphische Benutzungsoberflächen"*, Darmstadt 22.02.95, 1995.

**[Kurze 96]** Kurze M and Holmes E : (1996) "3D Concepts by the Sighted, the Blind and from the Computer", *Proc. 5th International Conference on Computers Helping People with Special Needs (ICCHP'96)*, July 17-19, 1996, Linz, Austria.

**[Kaufman 79]** Kaufman L : (1979) "Perception: The World Transformed", New York: Oxford UP, ISBN 0198575831, pp254-256.

**[Kuah 94]** Kuah K, Bodruzzaman M and Zeinsabatto S : (1994) "A neural-network-based text independent voice recognition system", *SouthEastCon '94 - Creative Technology Transfer - A Global Affair, Proceedings of 1994 IEEE SouthEastCon 94*, Ch 109, pp131-135.

**[Laux 96]** Laux L.F, McNally P R, Paciello M G and Vanderheiden G C : (1996). "Designing the World Wide Web for people with disabilities: A user centered design approach", *Proceedings of ASSETS '96: The Second ACM conference on Assistive Technology*. New York: ACM Press. ISBN 0-89791-776-6.

**[Laws 95]** Laws C.K : (1995) "Achieving greater accessibility for blind users of OS/2", *Special Needs System Development*, IBM Corporation, Boca Raton, Fla. 33429.

**[Loomis 90]** Loomis J, Hebert C and Cicinelli J.G : (1990) "Active localization of virtual sounds", *Journal of the Acoustical Society of America*, 88(4), October 1990, pp1757-1764.

**[Lowenfeld 75]** Lowenfeld B : (1975) "The changing status of the blind: From separation to integration", Springfield, IL: Thomas.

**[Lowenfeld 80]** Lowenfeld B : (1980) "Psychological problems of children with severely impaired vision", in W.M.Cruickshank (Ed.), *Psychology of exceptional children and youth* (pp255-342).

**[Lunney 81]** Lunney D and Morrison R.C : (1981) "High technology laboratory aids for visually handicapped chemistry students", *Journal of Chemical Education*, 58(3), pp228-231.



- [Mansur 85]** Mansur D.L, Blattner M.M, Joy K.I : (1985) "Sound Graphs: A Numerical Data Analysis Method for the Blind", Journal of Medical Systems, Vol 9, No.3, pp163-174.
- [McKiel 92]** McKiel Jr, F : (1992) "Audio-Enabled Graphical User Interface for the Blind or Visually Impaired", IBM Westlake Programming Laboratory, 1992 IEEE.
- [McQuarrie 95]** McQuarrie E : (1995) "The Accessibility of PDF(TM) and Adobe(TM) Acrobat(TM) Viewers for the Visually Disabled", Adobe Systems Incorporated, 345 Park Avenue, San Jose, California, 95110-2704 USA.
- [Morley 95]** Morley S (1995) "Window Concepts: an introductory guide for visually disabled users", London, Royal Institute for the Blind. ISBN: 1 85878 067 5
- [Morley 98a]** Morley S, Petrie H, O'Neill A-M and McNally P : (1998) "Auditory Navigation in Hyperspace: Design and Evaluation of a Non-Visual Hypermedia System for Blind Users", In Proceedings of ASSETS '98, ACM.
- [Morley 98b]** Morley S : (1998) "Digital Talking Books on a PC: A Usability Evaluation of the DAISY Playback Software", In Proceedings of ASSETS '98, ACM.
- [Muchnik 91]** Muchnik C, Efrati M, Nemeth E, Malin M and Hildemeister M : (1991) "Central Auditory Skills in Blind and Sighted Subjects", Scandinavian Audiology, Vol 20, Part 1, pp19-23.
- [Mynatt 92a]** Mynatt E and Edwards W.K : (1992) "The Mercator Environment: A Nonvisual Interface to X Windows and Unix Workstations", Graphics, Visualization and Usability Center, College of Computing, Georgia Institute of Technology, Atlanta GA, 30332-0280, Technical Report GIT-GVU-92-05, February 1992.
- [Mynatt 92b]** Mynatt E.D and Edwards W.K : (1992) "Mapping GUIs to Auditory Interfaces", Proceedings of the Fifth Annual Symposium on User Interface Software and Technology (UIST'92), ACM, New York, pp61-70.
- [Mynatt 93]** Mynatt E.D and Edwards W.K : (1993) "Metaphors for Nonvisual Computing", in Edwards A.D.N (Ed), Extra-ordinary Human-Computer Interaction, New York: Cambridge University Press, 1995.
- [Mynatt 94]** Mynatt E.D : (1994) "Auditory Presentation of Graphical User Interfaces", in Kramer G (Ed), Auditory Display: Sonification, Audification and Auditory Interfaces, Santa Fe. Addison Wesley: Reading MA.
- [O'Gorman 93]** O'Gorman L : (1993) "The Document Spectrum for Page Layout Analysis", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 15, No.11, November 1993, pp1162-1173.
- [O'Neill 99]** O'Neill A-M, Petrie H, Gallagher B, Hunter H, Lacey G and Katevas N : (1999). "Initial evaluations of a robot mobility aid for frail and elderly visually impaired persons", In C. Buhler & H. Knops (Eds.), Assistive Technology on the Threshold of the New Millennium. (AAATE 99). Amsterdam: IOS Press.
- [Oppermann 94]** Oppermann C : (1994) "Opening Windows of Access with JAWS for Windows", A Henter-Joyce White Paper about Speech Access and Microsoft Windows, Henter-Joyce, Inc., 2100 62nd Ave N., St.Petersburg, FL 33702.

**[Peterson 91]** Peterson C.D : (1991) "Editres - a graphical resource editor for X toolkit applications", in Conference Proceedings, Fifth Annual X Technical Conference, Boston, Massachusetts, January 1991.

**[Petre 92]** Petre M and Price B.A : (1992) "Why computer interfaces are not like paintings: the user as a deliberate reader", Proceedings of East-West HCI'92: The St.Petersburg International Conference on Human-Computer Interaction, Vol I, pp217-224.

**[Petrie 93a]** Petrie H, Heinila J and Ekola H : (1993) "A Comparative Evaluation of Computer Input Devices for Blind Users", Proceedings of ECART2 Conference, Stockholm, Sweden, ISBN 91 88336 19 0.

**[Petrie 93b]** Petrie H and Gill J : (1993) "Short Reports - Current research on access to graphical user interfaces for visually disabled computer users", European Journal of Special Needs Education, Vol 8, No.2, pp153-157.

**[Petrie 96]** Petrie H and Johnson V : (1996). "Methodologies for the evaluation of navigational aids for blind persons", In Tellevik and Haugum (Eds), Proceedings of International Mobility Conference 8. Trondheim, Norway: Tambartun.

**[Petrie 97a]** Petrie H, Johnson V, Strothotte T, Michel R, Raab A and Reichert L : (1997). "User-centred design in the development of a navigational aid for blind travellers", In S Howard, J Hammond and G Lindgaard (Eds), Human-Computer Interaction INTERACT '97. London: Chapman & Hall. ISBN 0 412 80950 8.

**[Petrie 97b]** Petrie H : (1997). "User-centred design and evaluation of adaptive and assistive technology for disabled and elderly users", Informationstechnik und Technische Informatik, 39 (2), 7 – 12.

**[Petrie 98]** Petrie H and Morley S : (1998) "The Use of Non-Speech Sound in a Non-Visual Interface to the MS-Windows GUI for Blind Computer Users", In Proceedings of ICAD'98, International Conference on Auditory Display, .Glasgow, November 1998.

**[Pitt 95]** Pitt I.J and Edwards A.D.N : (1995) "Pointing in an Auditory Interface for Blind Users", Proceedings of the 1995 IEEE International Conference on Systems Man. and Cybernetics, Vol 1, pp280-285.

**[Pope 93]** Pope S.T and Fahlen L.E : (1993) "The Use of 3-D Audio in a Synthetic Environment: An Aural Renderer for a Distributed Virtual Reality System", IEEE Virtual Reality International Symposium, VRAIS '93, Part 72, pp176-182.

**[Portigal 94]** Portigal S : (1994) "Auralization of Document Structure", Master of Science Thesis, University of Guelph, Canada.

**[Pugh 92]** Pugh K, Harness S.J, Sherkat N and Whitrow R.J : (1992) "Icon recognition for providing access to WIMP interfaces for the blind and partially sighted", Workshop on Iconic Communication, Rediffusion Simulation Research Centre, University of Brighton, 11-12 December 1992.

**[Rahgozar 95]** Rahgozar M.A : (1995) "An overview of document recognition technologies", IS&T's 48th Annual Conference - Imaging on the Information Superhighway, Final Program and Proceedings, Ch 151, pp102-106.

**[Raman 92]** Raman T.V : (1992) "An Audio View of (LA)TEX Documents", TUGboat, Volume 13, No.3, Proceedings of the 1992 Annual Meeting, pp372-379.

**[Salminen 98]** Salminen A-L and Petrie H : (1998). "Evaluating assistive technology prototypes: laboratory or real life contexts?", In I. Placencia-Porrero and E. Ballabio (Eds.), Improving the quality of life for the European citizen: technology for inclusive design and equality. Amsterdam: IOS Press.

**[Saslow 67]** Saslow M.G : (1967) "Frequency discrimination as measured by AB and ABX procedures", Journal of the Acoustical Society of America, Vol 41, pp220-221.

**[Scadden 84]** Scadden L.A : (1984) "Blindness in the information age: Equality or irony?", Journal of Visual Impairment and Blindness, 78, pp394-400.

**[Scadden 86]** Scadden L : (1986) "Facilitating access of future generations of computers. Symposium on human interface design and the handicapped user", Proceedings of the CHI'83 or 86: ACM Conference on Human Factors in Computing, Boston: ACM Press.

**[Scheifler 87]** Scheifler R.W : (1987) "X window system protocol specification, version 11", Massachusetts Institute of Technology, Cambridge, Massachusetts, and Digital Equipment Corporation, Maynard, Massachusetts.

**[Schweitzer 84]** Schweitzer J, Schubert R, Plaisance E, Herrington M, Cobb K and Bruley D.F : (1984) "Research for the Visually Impaired Computer User", Proceedings of the 2nd International Conference on Rehabilitation Engineering combined with the RESNA 7th Annual Conference, Ottawa 1984, Vol 4, pp579-580.

**[Schwerdtfeger 91]** Schwerdtfeger R.S : (1991) "Making the GUI Talk", BYTE, December 1991, pp118-128.

**[Sensory Systems]** Sensory Systems, UK Supplier of the following products: Frank Audiodata, Versabaille, Optacon.

**[Sherkat 93]** Sherkat N, Pugh K, Harness S.J and Whitrow R.J : (1993) "A Window and Icon Recognition Engine for On-Line WIMP Screen Analysis", 7ICAP, Seventh International Conference on Image Analysis and Processing, Bari, Italy, 20-22 September, 1993.

**[Sherkat 95]** Sherkat N, Derwas P.C and Whitrow R.J : (1995) "Widening Accessibility of Graphics User Interfaces for the Visually Impaired", The European Context for Assistive Technology, Proceedings of the 2nd TIDE Congress, 26-28 April 1995, Paris, Vol 1, pp379-382.

**[Shu 88]** Shu N.C: (1988) Visual Programming, New York: Van Nostrand Reinhold.

**[Smither 93]** Smither J.A : (1993) "Short term memory demands in processing synthetic speech by old and young adults", Behaviour and Information Technology, Vol 12, No.6, pp330-335.

**[Sola 90]** Sola I and Shepard D : (1990) "A voice recognition interface for a telecommunications basic business group attendant console", Human-Computer Interaction: Interact 90, Ch 168, pp779-785.

- [Stevens 94]** Stevens R.D, Brewster S.A, Wright P.C and Edwards A.D.N : (1994) "Providing an Audio Glance at Algebra for Blind Readers", In Kramer G and Smith S (Eds.), Proceedings of ICAD'94, Santa Fe Institute, Santa Fe: Addison-Wesley, pp.21-30.
- [Strauss 1990]** Strauss A and Corbin J : (1990) Basis of Qualitative Research: Grounded Theory, Sage.
- [Strothotte 93]** Strothotte T, Kurze M, Fellbaum K, Krause M and Crispian K : (1993) "Multimedia-Interfaces for Blind Computer Users", In Ballabio E, Placencia-Porrero I and Puig de la Bellacasa R (Eds.), Rehabilitation Technology, Proceedings of the 1st TIDE Conference. ISBN: 90 5199131 2, IOS Press, Amsterdam, 1993, pp 30-34.
- [Strothotte 98]** Strothotte T, Petrie H, Johnson V and Douglas G : (1998) "Evaluation of an Orientation and Navigation Aid for Visually Impaired Travellers", Improving the Quality of Life for the European Citizen, I.Placencia Porrero and E.Ballabio (Eds), IOS Press, 1998.
- [Sumikawa 85]** Sumikawa D.A : (1985) "Guidelines for the integration of audio cues into computer interfaces", unpublished Master's Thesis, University of California, Davis, CA. (Also published as a Lawrence Livermore National Laboratory Technical Report, UCRL 53656).
- [Telesensory]** Telesensory Systems, Inc., 3408 Hillview Avenue, Palo Alto, Calif., 1984.
- [Thatcher 93]** Thatcher J : (1993) "The Problems and Challenges of the Graphical User Interface", IBM Research.
- [Theberge]** Theberge S.A : "Make Windows Talk", Speech Systems for the Blind, 76 Wheaton Drive, Attleboro, MA 02703.
- [Vanderheiden 89]** Vanderheiden G.C : (1989) "Nonvisual Alternative Display Techniques for Output from Graphics-Based Computers", Journal of Visual Impairment and Blindness, Vol 83, Part 8, October 1989, pp383-390.
- [Vanderheiden 90]** Vanderheiden G.C and Kunz D.C : (1990) " Systems 3 - An interface to graphic computers for blind users", RESNA 13th Annual Conference, Washington DC, 1990.
- [van Schaik 97]** van Schaik P, Petrie H and Japp J : (1997). "The ATM speaks: the design and evaluation of an automatic teller machine with voice output", In G Anogianakis, C Buhler and M Soede (Eds), Advancement of assistive technology. Amsterdam: IOS Press. ISBN 90 5199 361.
- [Walker 92]** Walker E, Tobin M and McKennell A : (1992) Blind and Partially Sighted Children in Britain: The RNIB Survey Vol 2, London: HMSO.
- [Weber 94]** Weber G, Petrie H, Kochanek D and Morley S : (1994). "Training blind people in the use of graphical user interfaces", In W.L. Zagler, G. Busby and R.R. Wagner (Eds), Computers for handicapped persons. Berlin: Springer Verlag.
- [Wenzel 92]** Wenzel E.M : (1992) "Localization in Virtual Acoustic Displays", Presence: Telepresence and Virtual Environments, Vol 1, Part 1, pp80-107.

**[Yeung 80]** Yeung E.S : (1980) "Pattern recognition by audio representation", Anal. Chem., 52(7) p120.

# Acknowledgements

I am very grateful to the staff and students at the Royal National College for the Blind, Hereford who assisted in evaluating the developed systems, and to Greg Hale and Rachel Thomas for co-ordinating the visits.