USABILITY INSPECTION FOR SONIFICATION APPLICATIONS

Ag. Asri Ag. Ibrahim

Supervisor: Dr. Andy Hunt

Thesis submission for
the degree of Doctor of Philosophy

Electronics Department
The University of York

APRIL 2008

Abstract

Sonification is the representation of data using mainly non-speech sound for the purpose of communication and interpretation. The process and technique of converting the data into sound is called the sonification technique. One or more techniques might be required by a sonification application. However, sonification techniques are not generally suitable for all kinds of data, and often custom techniques are used - where the design is tailored to the domain and nature of the data as well as the users' required tasks within the application. Therefore, it is important to assure the usability of the technique for the specific domain application being developed.

In previously reported research, most designers of sonification applications have needed to develop at least a prototype for user testing. The result are interpreted and analysed to look for potential problems and solutions to improve the design. This dissertation has developed a new systematic usability inspection approach called the Task Interpretation Walkthrough (TIW) for the design of sonification application *before* they go to the initial development phase. It is hypothesized that designers of sonification applications will be able to detect significantly more important potential usability problems before the implementation phase by analysing the interaction between the user and the application as well as paying attention to the different stages of how the data is transformed into sound. It uses two new models – the Sonification Application (SA) model and the User Interpretation Construction (UIC) model.

Four experiments with human subjects were carried out to study the feasibility and effectiveness of Task Interpretation Walkthrough inspection by comparing it against two widely used techniques; Heuristic Evaluation and Cognitive Walkthrough. The sonification designs being inspected were a Mobile Phone Joystick Text-Entry with Sound (Experiments I and II), a Diagnosis Tool for Analysis of The Motion and Usage of a Patient's Arm (Experiment III); and an Audio-Visual Analysis Tool of Cervical Sample Slides (Experiment IV). The participants included sound researchers (Experiment II); and students with a background in music technology and software engineering (Experiments I, III and IV), acting either individually or in 2-person groups. The results have shown that the research hypothesis is supported, where the significantly important usability problems were able to be detected before the implementation phase. From the inspection method comparison study, results showed the Task Interpretation Walkthrough to be more effective than the existing techniques (Heuristic Evaluation and Cognitive Walkthrough).

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank all of the participants of the experiments for their time and effort.

I am most grateful to my supervisor, Dr. Andy Hunt, for his help and guidance on this work, and his patience through the painful process of writing up. I would like to thank Prof. Dr. David Howard and Dr. Alistair Edwards, for supporting this research in anyways. And also to Genevieve and Iain - application designers, for such a creative and incredible sonification application design.

I thank all the members of D013 for being such wonderful lab mates. Many people have helped in different ways in this work including members of Garrowby Castle V and York Malaysian Society Association (YMSA).

Finally, I thank my parents for their full support, love and patience.

Declaration

I hereby declare that this thesis contains research which is for the most part my own. In cases where I have drawn upon the ideas of others I have clearly stated so and given references.

I also declare that some parts of this thesis have already been published and presented at international conferences and also as a chapter in LNCS Springer Link book. The papers are:

Ag. Asri Ag. Ibrahim & Hunt, A. 2006a. *HCI Model for Usability of Sonification Applications*. Proceedings of TAMODIA'2006 post-proceedings. Published in 'Task Models and Diagrams for Users Interface Design', Springer LNCS, Springer-Verlag, ISBN 978-3-5540-70815-5, pp. 245 - 258.

Ag. Asri Ag. Ibrahim & Hunt, A. 2006b. *A General HCI Framework of Sonification Application*. Proceedings of The 8th International ACM SIGACCESS Conference on Computer & Accessibility. pp.221-222. Portland Oregon USA. October 2006.

Ag. Asri Ag. Ibrahim. 2006c. *Task Analysis for Sonification Applications Usability Evaluation*. Student Research Competition (SRC). Proceedings of The 8th International ACM SIGACCESS Conference on Computer & Accessibility. pp.289-290. Portland Oregon USA. October 2006.

Ag. Asri Ag. Ibrahim & Hunt, A. 2007a. *Systematic Usability Inspection for Sonification Application*. Proceedings of The 13th International Conference on Auditory Display. pp.453 - 460. Montreal, Canada, 26 - 29 June 2007.

Ag. Asri Ag. Ibrahim & Hunt, A. 2007b. *Combining Data State and CCT Diagrams to Model Sonification Applications*. Proceedings of The 25th ACM SIGDOC 2007 Conference. pp.148-154. El Paso, Texas. October 2007.

CHAPTER 1 : INTRODUCTION

This chapter introduces the two main research fields in this thesis, namely Sonification and Usability Inspection. Thereafter, the research focus is discussed through three sections including the current usability problems, sonification, objectives of the research and the Hypothesis. Furthermore, the thesis strategy is presented at the end of this chapter.

1.1 Background

In this section a brief overview of sonification and its design processes is presented. This includes the current practice of user testing of sonification applications which normally takes place after the development phase. As an alternative, a new form of usability inspection is introduced where the design is inspected for potential errors and problems before the initial development.

1.1.1 Sonification

The increasing volume of data or information nowadays has led to a certain ineffectiveness and inefficiency of existing data processing applications. Storage is no longer a problem due to the availability of high storage capacity devices and lower market prices. As a result, the question has now mutated into 'how can we effectively handle, present and understand the data?' Much research has been carried out to investigate the best way to understand and deliver a massive amount of information to users. This has encouraged research activities in other fields such as data mining, data exploration, data visualization and so forth. However, researchers have also realised that besides looking at the data, we could, in fact, also **listen** to the data. This new field of data representation is known as **Sonification.**

Sonification is defined as "the presentation of information using non-speech sound to help in understanding of data or processes by listening" (*Kramer, 1994*) or "the transformation of data relations into perceived relations in an acoustic signal for the purposes of facilitating communication or interpretation" (*Kramer et al., 1997*). Therefore, as a reference, which will repeatedly be used in this thesis, any software programs which use sound to represent data or information for communication or interpretation will be called **Sonification Applications.** As examples of the range of possible sonification applications, sounds could be used to represent sales data for a large retailer; a list of transactions of card credit holders; a list of prices in the stock market; or streams of data from sensors attached to the human body.

Graphical representation currently dominates the field of external representation, but sound is now seen as an alternative and its complement. Previous research has shown the success of using sound in several areas, especially for blind or visually impaired users; or in situations where the user's eyes are occupied with other tasks such as looking at a patient in medical diagnosis; or something which is difficult to represent using graphics, such as multidimensional data. As an example, instead of representing 2-dimensional data using graphics on a 2-dimensional graph (with x and y axes), the data could instead be represented using time (t) for the x axis and (for example) frequency for the y axis. By playing the data like this as a sound output, the user might be able to detect the increment or decrement of the data by listening to the higher and lower pitch of the sound. The goal (in this case) of detecting the increment and decrement in the data is the **user task** of the application (*more examples of sonification applications are discussed in Section 2.5 Auditory Display Tools and Applications*).

1.1.2 The Sonification Design Process

Generally, there are two main characteristics of sonification applications that make them different from other applications; the **application output** and the **sonification technique**. The **output** of the application is sound, which involves the human hearing sense in order to interpret it. The **technique** is the method by which the data or information is changed into sound. There are many techniques currently available in data sonification such as audification, parameter mapping, model-based sonification etc. (*and these are explained further in Section 2.3 Auditory Display Techniques*). Different techniques will produce different sound outputs and thus will influence the user's interpretation. The choice of technique for a particular sonification application is normally guided by considering the *type of data* being used as an input and the *user tasks* required of the application. Therefore, for sonification applications, it is important to look at and analyse the **tasks** and **data** that influence the generation of these different sound outputs. The generation of data into different sound outputs is referred to as the transformation process.

This thesis looks at the sonification process from different *perspectives* according to the ways in which data is handled and transformed (the transformation tasks). These transformation tasks are further split into different *views* depending on who or what is doing the task and how the task is done. These task *views* are tasks carried out by the user, those done purely by the application and those involving interaction between the two. These **perspectives** and **views** are the main concept of the proposed **inspection technique**, which will be explained later in this section.

Perspectives

One of the basic characteristics of sound is that it is time dependent. This means that it needs to be played based using time-based parameters such as duration, tempo or the time interval between sound samples. Researchers have found that when the real-world data is time dependent (such as stock market data evolving over the course of a day, or volcanic energy changing over several months) it is highly appropriate to portray this data as a sound representation. Unfortunately, not all data in this world is time dependent, for instance cancer cell images, or information relating to credit card applications. Such data first needs to be changed into something that is more suitable for sound transformation. This stage is called **data transformation.**

After this data transformation process, the transformed data needs to be converted into a form where it is ready to be played as sound. As an example, let us assume that we use the most popular conversion technique, the parameter-mapping technique. In this technique, the data can be mapped directly into sound or, more commonly, into some intermediate acoustic parameters. Examples of such acoustic parameters include amplitude, pitch, timbre and so forth. Other conversion techniques are described later in *Section 2.3 Auditory Display Technique*. Such a conversion is called an **acoustic parameters transformation.**

The outputs from the above transformation are then mixed to form a final sound and listened to by the users. The simplest way to play them is rather like playing music from an audio player, where a file is played from start to finish. However, the user could also manipulate the output from the acoustic parameters transformation through *interaction*. For instance, the user might be able to repeat any selected sounds as a repeating loop; play the sound either faster or slower; play only the selected area; play the sound either forwards or backwards etc. In other words, the same result of the acoustic parameters transformation can be further manipulated and played as different sound representations. This process of manipulating the output from the acoustic parameters transformation is called the **final sound transformation.**

In summary, **sonification techniques** consist of at least three different processes that transform the data into sound. These include data, acoustic parameters and final sound transformations as described above. All these three transformations significantly influence the final sound output of the application, which needs to be interpreted by the user. Therefore, it is important to ensure that the most suitable transformations are used. As a reference in this research, the three different transformation processes are referred to as three different **perspectives**; data, acoustic parameters and final sound perspectives, as they denote the three most important ways of looking at how the data is transformed during the process of sonification.

Views

For further understanding of sonification applications, it is also important to understand their end **users**. The user might (or might not) have in mind thoughts about *the capability of the application,* how can the application help them achieve a goal? Or *how and where they should start?* Also, the designers themselves might also have in their mind *what they think the end users should do* and *should know* prior to using the software. All these things occur in the users' and designers' heads. If the designer introduces a new function or interface that is important but is not in the list of user requirements, the users will need to know about it and to learn it. In sonification applications, it is important to assess the knowledge base (or expertise) of the user in relation to the data. If, for example, the user is a physiotherapist, their knowledge about muscles and the way that the data is gathered could affect their understanding and interpretation of the sound representation. The perception of acoustic parameters is also important, as different users might perceive sound differently. For instance, the same pitch of sound can be judged and levelled differently by different listeners. The arrangement or the combination of different acoustic parameters can also produce different kinds of sound representation and so perceptions; therefore it is important to consider these in the design and thus the inspection.

It is also important to have knowledge of the **system's** main function, as it clearly helps to understand the characteristics of the system's intended use. This is especially helpful in rationalizing the design. For example, if the system reduces multidimensional data into two-dimensional data, the rationale and reasoning behind this would be important for an understanding of the system. This also applies to the selection of acoustic parameters and how the sound is to be played.

Finally, as mentioned above, the extent to which users can **manipulate** the application inputs can determine how the final sound output is generated. However, some applications do not provide any input manipulation at all, especially for 'monitoring applications' where the sound is played in the background (or as ambient sound).

It is also important to understand sonification applications from the three aspects – user, system main function and inputs/outputs manipulation. Paterno (1997) refers to these aspects as three tasks called users, application and interaction tasks. Therefore, as a reference in this research we refer these aspects as three **views** called **users, application**, and **interaction**.

In summary, the three **perspectives** (stages of transformation processes: *data, acoustic parameters, and sound transformation*); and the three **views** (the user, the application and the interaction between the two) are highly important for a better understanding of sonification applications as each significantly influences the final sound output.

Therefore, I propose to describe sonification applications from these three different perspectives and three different views. These produce nine different aspects (*each perspective has three different views*) which I use to describe sonification applications in my **Sonification Application model** (*which is explained in detail in Section 4.5*).

User Testing in sonification applications

In order to best determine the most suitable and usable sonification technique(s), each application requires a separate and dedicated **usability study**. In the brief history of sonification applications so far, the sonification technique tends to be rapidly chosen, the application is developed (programmed), and testing takes place with the end user in order to evaluate the application's effectiveness (*explained further in Section 3.3 Existing Evaluation in Sonification*). User feedback is important for the designer to determine whether or not the sounds are working and achieving the intended purposes. If, on testing, the application does not work effectively, the designer goes back to the drawing board, and selects an alternative sonification technique. Sometimes, this 'choice-development-testing' process needs to be repeated many times until the required and suitable sounds are produced and the **user task** is attainable.

Since the User Testing is typically carried out at the stage when a working prototype (and sometimes a full program) is available, this kind of develop-test process can be costly and time consuming. Furthermore, if the usability evaluation is only carried out when the application is almost complete, there will probably be little chance to make significant changes or corrections to any deficiencies and errors found. This is especially true if the project involves a very tight schedule and deadlines. It will probably end up with a higher overall cost and longer than expected development time, particularly if it requires major changes. **This cost could be avoided if the major problems were to be detected in the earlier stages.**

Because of the above problems, I believe that the field of sonification requires an alternative, not to replace but at least to enhance the evaluation techniques in order to predict anomalies or problems *before* the expensive development phase. Therefore, a cheaper and faster technique is required.

1.1.3 Usability Inspection

Usability inspection can be such an alternative for evaluating sonification applications because it can be done towards the *start* of the development process, and without involving end users. Usability inspection is a generic name for a set of methods based on having human 'evaluators' or 'inspectors' examine usability-related aspects of a user interface (Nielsen et al, 1994). It is an expert-based evaluation, which is carried out by human specialists, and is normally implemented at the design stage before it goes to the implementation or development stage. It requires fewer participants (typically usability experts) than controlled end-user experiments.

Examples of existing inspection techniques are Cognitive Walkthrough, Consistency Inspection, Pluralistic Walkthrough, Standards Inspection, Heuristic Evaluation and Formal Usability Inspection (*explained further in Section 3.2.5 Predictive Evaluation*). They are distinctive from each other in various aspects such as the purpose and focus of the technique; the type of problems or anomalies the technique addresses; and how the technique guides the inspector to do the inspection. For example, Cognitive Walkthrough focuses on the goals and knowledge of a user while performing a specific task, whereas Heuristic Evaluation emphasises a list of 'usability principles' to be followed as a guideline.

All these techniques tend to produce qualitative results including the early identification of usability problems, anomalies, comments, suggestions and so forth. Nielsen et al. (1994) defined **usability problems** or **anomalies** as aspects of the user interface or functionality of the application that may cause the resulting application to have reduced usability for the end user. The encountered problems might affect different users in different ways. A small problem for expert users could be a big problem for novice users. Some problems could be serious enough to prevent users from accomplishing the task successfully; or they could just make the user a bit slower in performing the task. However, it is important to note that the general definition of 'usability problem' in this thesis is based on that by Nielsen et al. (1994, *p*.3). A usability 'problem' or 'anomaly' is considered to be any aspect of the design which, if changed, would lead to an improved system.

The problems found by this process will be used to make recommendations on how to fix and improve the design. Studies of usability inspection methodology have found that many usability problems are overlooked by user testing. However, such user testing also finds problems that are overlooked by the inspection (Nielsen et al., 1994). Therefore, the best result is attained by *combining* both empirical user testing and inspections. However, in this thesis, the focus is only on **inspection of sonification applications.**

Usability studies should be conducted as early as possible in the design stage. If problems can be detected earlier, there is a good chance that they can be fixed and corrected before the expensive implementation phase.

Inspection materials is the name given to the package that contains descriptions of the application to be evaluated; steps and instructions for inspection; forms to write the encountered problems, and so forth. Different inspection methods might use different inspection materials. For example, the Cognitive Walkthrough technique requires 'tasks scenarios' that represent the structure and flow of goals and actions; and in Heuristic Evaluation a prototype of interface with usability heuristics guidelines are used etc.

In summary, this research attempts to provide inspection materials and inspection technique guidelines specifically for sonification applications for the purpose of usability inspection.

1.2 Problem Statements

This section explains current problems in usability evaluation for sonification applications.

1. Existing usability evaluations for sonification applications are mostly based on empirical user testing with a working system or at least a working prototype. The time and cost to develop the system or prototype could be a waste if the design was found to have high severity problems that require it to be completely redesigned.

2. Evaluations are often implemented in an ad-hoc way by individual designers or researchers, which could leads to them being ineffective.
3. Empirical user testing often focuses on concrete tasks that can be measured and quantified, such as whether an object can be detected, or the speed of response when searching for data. More abstract and perceptual tasks are harder to deduce and quantify, such as how the user understands and analyzes the data.
4. Existing usability inspection techniques are not suitable for inspecting sonification applications because they:
   a. do not consider *sonification techniques, data* and *sounds.*
   b. only focus on WIMP (*Windows, Icons, Menus and Pointing devices*) and GUI (*Graphical User Interface*).
5. Certain measurements are often not suitable for sonification applications. For example, in monitoring applications (where sound is used simply to alert the user), a *memorizing* criterion probably is not very important.
6. Since a sonification application can use different sonification techniques, it is important to evaluate each technique for its effectiveness before proceeding to other criteria such as efficiency and satisfaction.

1.3 Research Objectives

The objectives of this research are therefore to:

1. Look at the possibility of inspecting usability aspects of sonification applications in the early stages of the design process;
2. Review the issues, capabilities and limitations of current sonification applications in terms of usability;
3. Propose a technique for analysing the tasks in sonification applications and develop a model which allows us to understand how users interpret the sound output of sonification applications.
4. Propose a systematic usability inspection technique for sonification applications.
5. Provide recommendations for usability inspection/evaluation of sonification applications.

1.4 Thesis Hypothesis

This section states and discusses the Hypothesis, which will guide the focus of this research.

1.4.1 Hypothesis Statement

The following hypothesis will be investigated in this thesis:

Designers of sonification applications will be able to detect significantly more important usability problems/anomalies before the implementation phase by analysing the task through different views[a] and paying attention to different perspectives[b] in the data state transformations.

Where;

[a] *Views include user view, application view and interaction view (refer to Section 4.5).*

[b]*Perspectives include data perspective, acoustic parameters perspective and final sound perspective (refer to Section 4.5)*

This hypothesis will be supported by showing that:

1. the proposed usability inspection will be able to detect **significantly more potential usability problems** in overall performance.
2. the proposed usability inspection will be able to detect significantly more **important** usability problems.
3. the proposed usability inspection will be able to detect significantly more potential important usability problems **in each perspective** (data, acoustic parameters and final sound) compared to existing usability inspection techniques.

1.4.2 Discussion of Hypothesis

The above hypothesis states that the designer will be able to detect significantly important potential usability problems by inspecting the design of the sonification applications. In user testing, usability normally refers to how easy it is for the user to learn a system, how efficiently they can use it, and how pleasant it is to use. However, for sonification applications, it seems to be more important to determine the ability of the sounds to fulfil the intended tasks with high accuracy, efficiency and pleasantness. Therefore, we can consider potential usability problems to be any aspects of the design that may result in low accuracy, efficiency and pleasantness of the final sound representation.

As the problems detected might vary in terms of their relative importance, the lists of encountered problems need to be prioritized. In the existing usability inspection research to date, all the encountered problems are classified and counted. By doing this, the researcher is able to measure the effectiveness of the inspection technique itself. So, for example, if more potential problems are found in the most critical problem category by the proposed technique compared to an existing technique, we can conclude that the proposed technique is more effective in detecting critical problems than the existing technique. In this thesis, the problems found will be rated using a severity level (0 to 4) as follows (adapted from Nielsen J., 1993, p.103):

0 = this is not a potential usability problem at all
1 = potential cosmetic problem - only needs to be fixed if extra time is available on project
2 = potential minor problem – fixing this should be given low priority
3 = potential major problem – important to fix, so should be given high priority
4 = usability catastrophe – imperative to fix this before product is usable

It can be quite difficult to make such a distinct categorisation of the problems found. However, in this thesis, the classification will be done as fairly as possible by giving the list of problems to several usability experts for rating. The average of their ratings will be used as the final rating. This is important in order to clarify the reliability of the classification method, as different people might have different opinions on which problem is the most serious.

The hypothesis also states that by analysing the tasks through different views (user, application, and interaction) and paying attention to different perspectives (data, acoustic parameters, sound transformation) the inspector can understand more about the application. This analysis should help them to rationalise the design and encourage them to give comments and feedback about it. These could be potential problems, which might affect the usability of the application. We can use these concepts to 'walk around' the application looking at it from each perspective and within each perspective, taking all the different views. Here is an example of how this might work.

1. From the *data transformation* perspective, the following example questions could be asked from each of the different views:
   - What does the user want to know about the data? (user's view);
   - What does the application need to do with the data? (application view); and,
   - How can the user change and interact with the data? (interaction view).

2. From the perspective of the *acoustic parameters transformation,* the inspector might like to know:
   - Which acoustic parameters will the user perceive? (user's view);
   - What kind of acoustic parameters will the application convert the target data into? (application view); and,
   - How can the user interact with the acoustic parameter settings? (interaction view).

3. And finally, from the perspective of the final sound transformation, it is important to understand:
   - What does the user need to know about the different ways of presenting the sound? (user's view);
   - What kind of final sound will the application produce? (application view); and,
   - How can the user *interact* with the final sound representation? (interaction view).

This thesis introduces a novel technique for enhancing existing usability inspections for sonification tasks: determining what and how to inspect and look for significantly important potential problems of sonification applications while they are still in the design stage. The core idea of the technique is to understand the **design rationale** of the sonification applications being inspected. This can be done by critically analysing the design from the points of view of a) the user, b) the application, and c) the interaction between the user and the application; and also through different perspectives based on the transformations, which occur in sonification techniques, namely data, acoustic parameters and sound. These perspectives and views are described through the **Sonification Application (SA) model.**

During inspection, the results of the analysis will be used to look at and understand how users interpret the sound output of sonification applications. The analysis and construction of the possible interpretation will be done by using the specially developed **User Interpretation Construction (UIC) model** (*explained in detail in Section 4.7.2*).

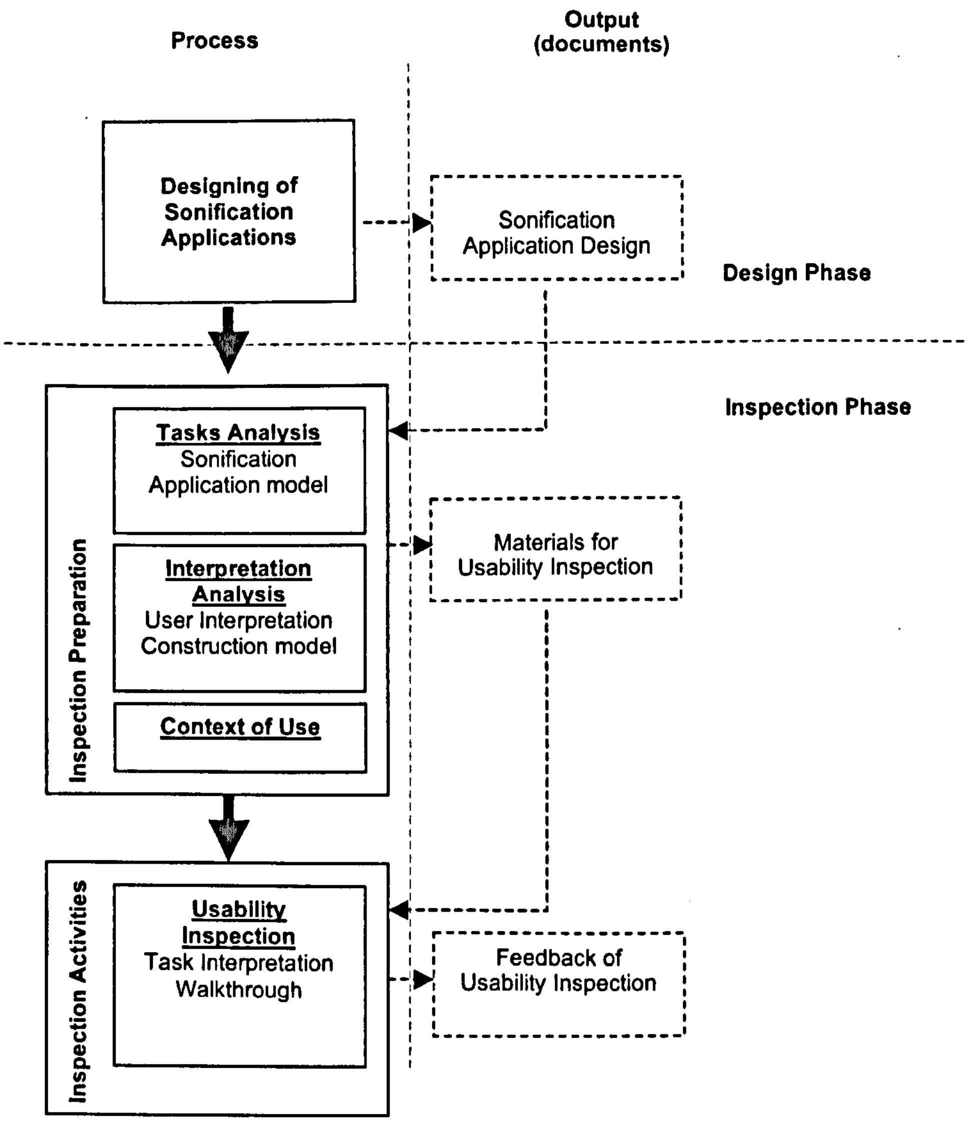


Figure 1-1: Overview of Inspection Strategy for Sonification Applications

Figure 1-1 above shows an overview of the proposed inspection strategy for sonification applications. The design of sonification applications will be analysed and rationalised through the Sonification Application model. The result of the analyses will be used by the User Interpretation Construction (UIC) model to understand and analyse how users interpret the sound output, which will later be used for usability inspection. The potential problems found as the feedback will be used to improve the design.

In summary, by understanding the rationale behind the design and critically analysing and discussing them, significant design anomalies, problems, comments and feedback can be gathered, which can be used to improve the design later on. This can become an integral part of the iterative design process in the development of future sonification applications.

1.5 Thesis Strategy and Structure

This section explains how a study will be carried out to investigate and support the research objectives as well as its hypothesis. An overview of all the chapters in this thesis is also given.

1.5.1 Thesis Strategy

This research addresses the following two general questions:

1. How can we help a designer or inspector to understand and explore the tasks and usages of their sonification application?
2. How can we help a designer or inspector to inspect, detect and identify significantly potential important usability problems in their sonification application?

To answer these questions, it is important to understand the two major research fields of this thesis -*Sonification* and *Usability Inspection*, which are explained in detail in Chapter 2 and Chapter 3 respectively. By analysing and criticising several aspects of these major fields, I propose a novel systematic usability inspection for sonification applications called the Task Interpretation Walkthrough (TIW). The proposed technique is explained in detail in Chapter 4.

For the next step, a study and several experiments will be carried out to understand the feasibility, effectiveness, and reliability of the proposed usability technique. In order to do this, several questions need to be answered in each criterion as follows:-

Feasibility:

- Is inspection of sonification applications and techniques feasible?
- Is the proposed technique practical?

Effectiveness:

- Are inspectors able to detect potential usability problems?

- How serious are the potential problems that can be detected?
- How effective is this new inspection method compared to the existing inspection techniques?

Reliability

- Are the results (repeated experiments) consistent?
- How reliable is the inspection?

1.5.2 Thesis Structure

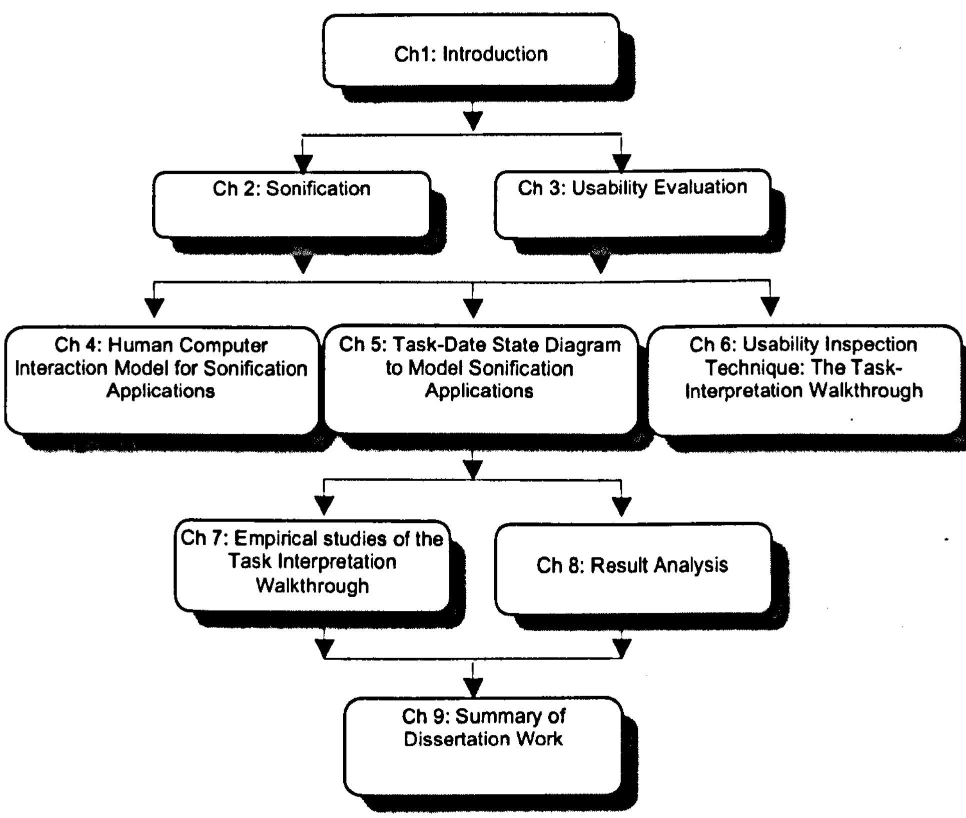


Figure 1-2: Plan of Thesis Structure

Figure 1-2 below shows the plan of the thesis structure. Each chapter is briefly explained below to give an overview of the overall content of this thesis.

Chapter 2: Sonification

A definition of sonification can be drawn up from:

- the processes involved in converting data into sound,
- the inputs and outputs involved in these processes, and,
- the objectives of the sonification technique itself.

These three components are referred to as *technique, input/output* and *objectives* respectively. Examples of such techniques are introduced: audification, parameter mapping and model-based sonification. Differences in the type of input/output and the objectives of particular programs produce a variety of sonification applications and tools. To design these applications, several approaches can be used such as a 'syntactic approach', 'semantic approach', 'task oriented approach' etc. All of these techniques, design approaches and examples of applications are explained fully in this chapter.

Chapter 3: Usability Evaluation

This chapter explains several definitions of usability and introduces existing usability evaluation methods. These methods include observation, interviews, experiments, interpretation and prediction (inspection). Previous evaluations of sonification applications are mostly based on end-user testing with at least a working prototype. The testing is based on experimental design, which is used to predict a relationship between variables being investigated. Several tasks such as matching, comparison and classification are used in this testing to manipulate the variables for subsequent analysis. Several issues in designing sonification are also discussed in this chapter, such as issues in the type of acoustic parameters used, sound aesthetics, sound structure and so forth.

Chapter 4: Human Computer Interaction Model for Sonification Application.

This chapter discusses the Human Computer Interaction (HCI) Model for Sonification Applications. The HCI model comprises two sub-models called the Sonification Application (SA) model and the User Interpretation Construction (UIC) model. These two sub-models are used to describe the design of sonification applications.

Chapter 5: Task-Data State Diagram to Model Sonification Applications

This chapter discusses a new diagrammatic way to describe the design of Sonification Applications, which we have called the *Task-Data State Diagram*. Some related work, including the Data Flow Diagram, the Data-State Diagram and the ConcurTaskTree Diagram, will be discussed briefly.

Chapter 6: Usability Inspection Technique: Task Interpretation Walkthrough

This section explains our new usability inspection technique called Task-Interpretation Walkthrough (TIW). The instructions and inspection materials required by the technique will be discussed, which include design descriptions in the Task-Data State diagram form (Chapter 5); interpretations of the predicted outputs; the context in which the application will be used; and other documents such as list of user requirements and graphical user interfaces.

Chapter 7: Empirical Studies of the Task Interpretation Walkthrough

This chapter discusses four series of experiments, which were conducted with three different sonification applications. Objectives, variables, experimental design, procedure and materials of the experiments are also described.

Chapter 8: Analysis of Results

This chapter will explain the result analysis from the experiments described in Chapter 7. The analysis includes inspection performances of Task Interpretation Walkthrough, Heuristic Evaluation and Cognitive Walkthrough. The results are explained based on the three supporting hypotheses that can be categorized by overall performance; level of severity; and transformation perspectives.

Chapter 9: Summary of Dissertation Work

This chapter explains the implication and conclusion of the analysis from the previous chapter. It also explains the contribution of this research and several suggestions for future research.

CHAPTER 2 : SONIFICATION

This chapter introduces sonification including its definition and current development. This includes existing auditory display techniques such as audification, earcons, parameter mapping and model-based sonification. The chapter goes on to explain the previous design approaches to sonification such as syntactic, semantic, pragmatic, task-oriented and data-state transformation. Finally, it gives several examples of existing applications and tools in sonification. To inspect usability in sonification applications, these three aspects are important for giving an overview of what sonification is, how to design applications and for what purposes they could be used.

2.1 Introduction

Sonification is a new interdisciplinary research field. Figure 2-1 shows the related research disciplines, which can be involved in the process of creating and producing sonification systems. In the data domain for instance, a statistician could be involved in producing new techniques of data processing and reduction for a better data perspective.

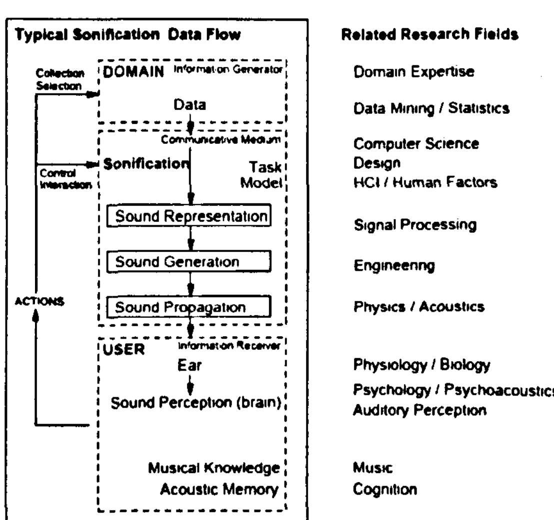


Figure 2-1: Schematic illustration of the Information Flow in Auditory Displays and Related Research Disciplines (Hermann, T., 2002)

Computer scientists might contribute to a new and more effective sonification algorithm. Human Computer Interaction specialists might be more concerned with creating guidelines and design techniques, and in establishing the usability of the applications. Engineers and acousticians might look into the details of sound processes, signal processing and sound manipulation. Researchers from the social sciences, such as psychology and cognitive science, could focus on how humans perceive and are emotionally affected by the sound. Altogether, the combination of these different disciplines will produce various research findings, which are informative for the future direction and development of the field of sonification.

2.2 Terms and Definitions

This section explains several terms and definitions that will be used in this research.

Sonification

Below are several definitions of sonification that are most commonly referred to:

1. "Sonification is the **presentation of information** using **non-speech sound** to help in **understanding of data or processes** by listening" (*Kramer, 1994*).
2. "Sonification is a **mapping of numerically represented relations in some domain under study to relations in an acoustic domain** for the purposes of **interpreting, understanding, or communicating relations** in the domain under study" (*Scaletti, 1994*).
3. "Sonification is the **transformation of data relations into perceived relations in an acoustic signal** for the purposes **of facilitating communication or interpretation**" (*Kramer et al., 1997*).

Figure 2-2 below describes some important elements and issues that should be considered in the construction of sonification displays. It shows that sonification consists of **data** and **auditory display**; and **how to manipulate it** (Tasks and Human-Computer Interaction) for **the purpose** of perception (Sound and Perception).

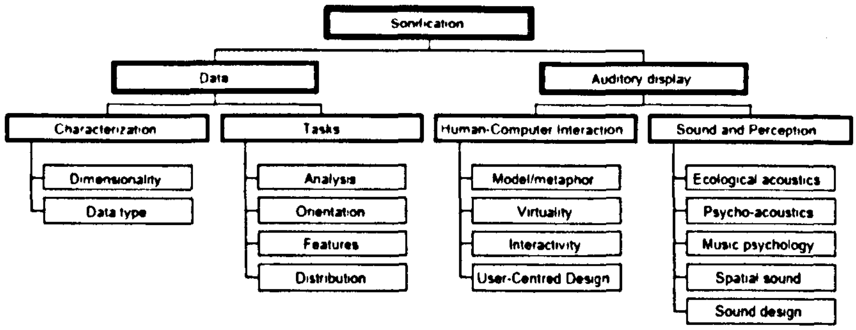


Figure 2-2: Elements of Sonification Display (Saue, 2000)

Sonification Technique

A Sonification technique is a concept and algorithm used to convert data into a sound representation.

Sonification Application

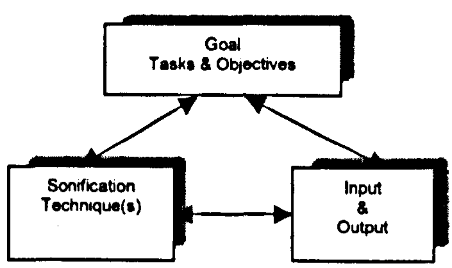


Figure 2-3: Sonification Applications block diagram

Referring to the definition of sonification above, sonification applications can be said to have the three main elements as shown in Figure 2-3:-

1. the goal, tasks and objectives to achieve (*e.g. interpretation, communication etc.*);
2. input (*data*) and output (*e.g. non-speech sound*); and
3. the technique(s) for data transformation.

Therefore, a sonification application can be defined as a piece of software with a specific usage in a specific domain and using one or more specific sonification technique(s) to transform data into a sound representation.

2.3 Auditory Display Techniques

This section explains the sonification **technique**, which is how the data is converted into sound. It also explains briefly several research foci and techniques within auditory display. The techniques covered are audification, earcons, auditory icons, parameter mapping and model-based sonification.

2.3.1 Audification

Audification is said to be the most direct auditory display technique because the sound samples can be directly obtained from the data values. It can be as simple as the direct conversion of signal-to-sound through an audio amplifier. However, to ensure that the conversion output can be heard, the designers should manipulate the data to be in the frequency range of human hearing, which is, at best between 20Hz and 20,000Hz. But this simple technique is not suitable for all data, especially for small amount and slow changing series of data.

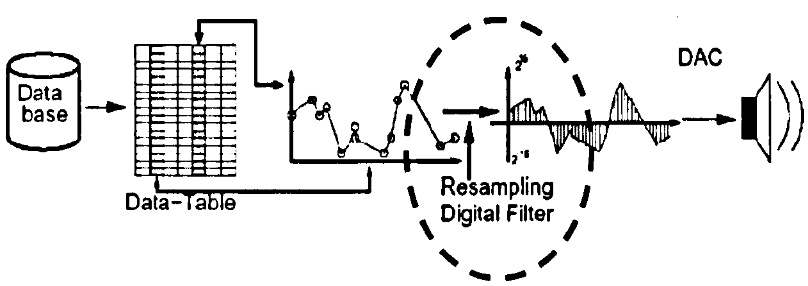


Figure 2-4: Audification (Hermann, 2004)

However, several other transformation techniques such as re-sampling, filtering, time stretching, pitch scaling, dynamic compression, translations and extraction can also be applied to the signal. These are shown as the dashed circle in Figure 2-4. An example of audification is in planetary seismology by Dombios (2001). In this research, audification is used to listen to the data from the earth's activities, making earthquakes audible. The earthquakes were found to produce specific acoustic characteristics, which are difficult to display visually. Those findings have made auditory seismology more promising in analysing seismological data.

In terms of production, it is easier and faster to produce sound using audification compared to other techniques such as Parameter Mapping, Model-Based Sonification etc (*these techniques are explained later in this section*). It is able to compress hours of data into few minutes or even seconds of sound, especially if the technique uses a high sampling rate. The disadvantage of audification is that it requires a lot of data values even for a short audification.

2.3.2 Earcons

Blattner et al. (1989) define earcons as "non-verbal audio messages that are used in the computer/user interface to provide information to the user about some computer *object, operation* or *interaction*". In terms of design, Hankinson et al (1999) proposed the usage of musical grammar in earcons. Musical grammar is a set of rules that describe how basic units such as notes, rhythms and pitch can combine to form larger phrases. In general, there are four types of earcons: one-element, compound, hierarchical and transformational (Blattner et al., 1989) as described briefly below:

One-element earcons – are the simplest type and can be used to communicate only one bit of information e.g., a single pitch earcon.

Compound earcons – are formed by concatenating one element earcon with another to form messages that are more meaningful. For example, Brewster (1994) combined several one-element earcons such as 'save', 'open' and 'file' to form compound earcons by playing one after another to represent a bigger phrase such as 'save file' and 'open file'.

Hierarchical earcons – consists of several nodes in a hierarchical form where each node represents an earcon. Each node inherits all of the node's properties that link to it. There could be more than one node at each level. As an example, Figure 2-5 shows a system of hierarchical earcons with three levels and with three parameters, namely; rhythm, pitch and timbre. Level 1 is the Rhythm Family and has no pitch. This rhythm is then inherited by Level 2, which is termed the Pitch Family. At this level, the timbre is just a plain sine wave. Both the rhythm and pitch are then inherited by Level 3, which is termed the Timbre Family.

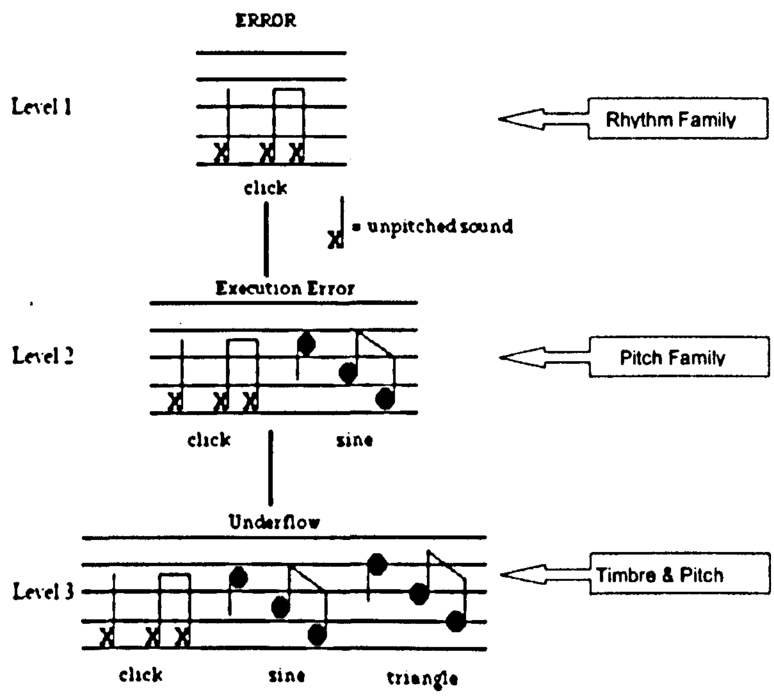


Figure 2-5: Hierarchical Earcons (Brewster et al, 1992, *adapted from Blattner (1989)*)

Transformational earcons – are also constructed around a "grammar" which is similar to Hierarchical earcons. Each auditory parameter such as rhythm, pitch and timbre can be altered to change the meaning of an earcon. This common grammar is actually the strength of this technique as less learning (of multiple individual earcons) is required (Blattner et al., 1998).

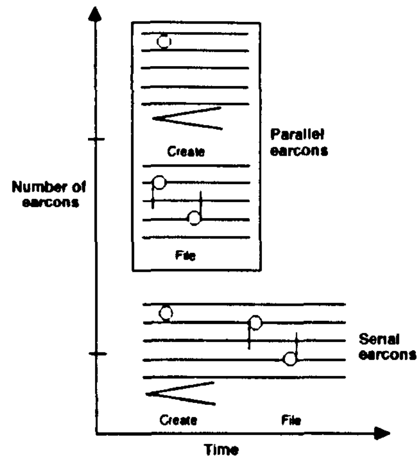


Figure 2-6: Comparison between Parallel Earcons and Serial Earcons (Brewster et al., 1995)

Parallel earcons – are introduced to overcome the problem of compound earcons or serial earcons that can take long time to play. It is done by playing two earcons at the same time (parallel) rather than in sequence (serial) as shown in Figure 2-6. By doing this, the same kind of information can be delivered with a faster and shorter sound representation compared to compound earcons.

2.3.3 Auditory Icons

Auditory icons involve the design of everyday sounds to convey information about events by analogy to everyday sound events (Gaver, 1994). Everyday sound is used in order to aid and improve its learnability and comprehension. For instance in Buxton et al.(1994), a sound that would be made when a real-world object is touched can be used for object selection; and a scraping sound can be used for an object moving. Both auditory and visual icons that use the same analogy will increase interface redundancy, which can help users to learn and remember it. This contrasts with the use of earcons, where it is quite difficult to achieve this audio-visual mapping, because of the reliance on musical parameters. This means that users need to learn two different sets of rules for auditory and graphic components.

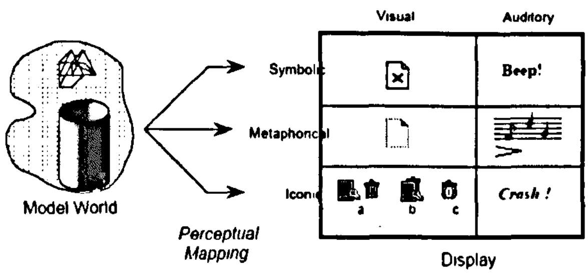


Figure 2-7: Type of Mapping (Buxton et al., 1994)

Buxton et al (1994) highlighted three issues in designing auditory icons: *mapping, vocabulary* and *annoyance*. The Mapping strategy is very important in designing auditory icons to ensure that they are intuitive or natural to the user. Designs can be symbolic, metaphorical or iconic as shown in Figure 2-7. **Symbolic** mapping means using an arbitrary sign to convey meaning, such as a *'beep'* sound for auditory icons indicating that the process is completed. **Metaphorical** is the process of duplicating the analogy of real-world activities into electronic representation. For example, the *'decreasing volume'* of sound is used to indicate the file deletion (disappearing). **Iconic** is the representation of the objects and also the process it needs to accomplish any particular task. For example, the sound (*'Crash'*) of an object dropping into a full trash can. Symbolic mapping is difficult to learn because it relies on social convention. For instance, the siren of an ambulance has become accepted as an 'emergency' sound because it has been used for so many years. Metaphorical and iconic are easier to learn as both focus on the similarities between the sound representation and thing it represents (Buxton et al., 1997).

In terms of vocabulary, Buxton et al. (1997) also mentioned that the sounds to be used must be 'everyday sounds' whenever possible. For instance, a 'tapping' sound should be produced if a file is clicked or selected, and a 'dragging sound' could be appear if we grab and then drag the file.

The final issue is a challenge to designers as to how to produce sounds that will not be annoying to users. But this might not be an issue for certain applications such as warning systems. Can we imagine what might happen if an emergency sound was represented by a pleasant musical sound? The sense of urgency would not be there. Therefore, annoyance is still important especially if the warning system requires a certain level of urgency. Edworthy et al (2002), in their experiments on acoustic parameters, have shown that there are four major parameters that can be used to influence the level of urgency (e.g. warning sound). These are speed, fundamental frequency, repetition and inharmonicity.

2.3.4 Parameter Mapping

Parameter mapping is a popular technique for representing high dimensional data as audio. Generally, it involves mapping data or data properties into sound properties such as pitch, volume, duration, timbre and location to produce different types of sound (indicated in Figure 2-8 as 'parameter-vector'). Therefore, high dimensional displays can be obtained by mapping different data variables to different acoustic parameters.

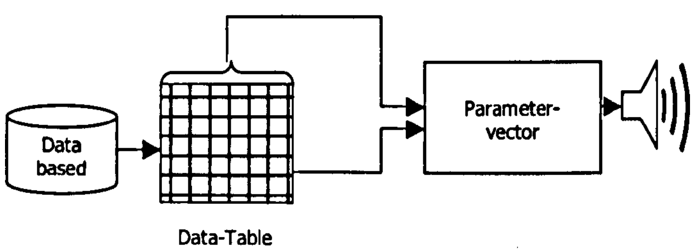


Figure 2-8: Parameter Mapping

All parameters can be listened to simultaneously or could be changed using the same data to give different sound representations. Allowing the user to change the type of sound parameter to be used gives the users flexibility and more options to explore and understand the data under study.

Besides mapping the data to acoustic parameters, **interaction** with the application is said to enhance the perception of the data-sound mapping (Hunt et al., 2004). Hunt et al. (2004) also emphasized that besides mapping the data, elements of real-time interactivity are also important in sonification, instead of just the typical playback button e.g. start and stop playback. By continuously controlling interactively the position within the dataset (or the mapping strategy, or both at the same time) more dimensions of data perception can be listened to and observed.

However, there is no universal mapping technique of data into acoustic attributes. The sounds produced are different for different mapping of different acoustic attributes, which makes parameter mapping difficult to learn even though it is said to be more flexible.

2.3.5 Model-Based Sonification

This technique was introduced by Hermann (2002) where data spaces are sonified by taking as a model the environmental sound production from real world. As parameter mapping focuses on sound and acoustic attributes, model-based sonification is said to focus more on manipulating the data set and its properties. The data set is used to create an 'instrument and its acoustic structure' as a *model*. For example, Figure 2-9 below shows that the data can be converted into a surface area of a 'drum' with 'hitting' as the interaction mode. The interaction modes are specified to allow user to interact with it.

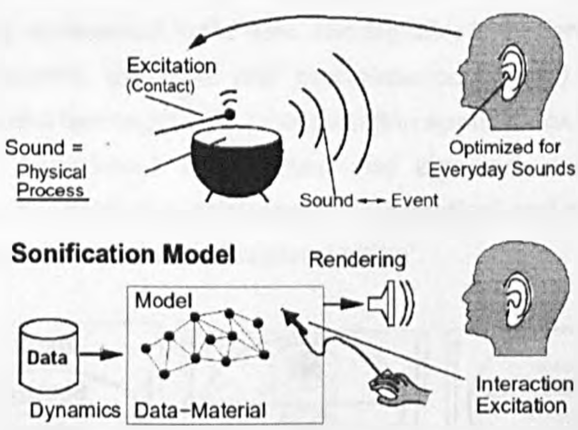


Figure 2-9: Model-Based Sonification (Hermann, 2004)

Hermann (2002) gave an example model of a 'high-dimensional dataset', which was transformed into a '2-dimensional rectangular membrane' as a model. The user can explore the model by striking-interaction (as part of the model design) which puts kinetic energy into a surface element. This excitation causes a dynamic motion of the membrane, resulting in a sound. This sound is taken as the sonification and is presented to the user as a sonic feedback of each interaction. Besides striking, other possible modes of interaction are plucking, hitting, striking, rubbing, scratching, shaking and deformation.

Hermann (2002) gave a few examples of models such as the Particle Trajectory Sonification model, Data Sonograms and Principal Curve Sonification. Since data can be represented using more than one model; and a model can be used by different types of data, a proper 'model selection' for certain problems and data types is important and is becoming a sonification design issue.

2.4 Previous Auditory Design Approaches

This section explains briefly several existing approaches to auditory design. The approaches are Syntactic, Semantic, Pragmatic, Task-Oriented and Task Analysis Data Analysis (TADA). Some key terms and concepts in this section will be based on the *Model of Sign* as shown in Figure 2-10 below. The concept of 'Sign' is that which refers to something other than itself (Chandler, 1997). It has two parts; *Signified* and *Signifier*. Signified is the concept being represented to the user, and Signifier is the form that the sign takes. For instance, the word and pronunciation (sound) of "tree" is a signifier of a real tree (signified). In sonification applications, the 'input and output' can be referred to 'signified and signifier' respectively. The signified can be 'data, data attributes or information' and signifier can be 'acoustic parameters and sound representation'.

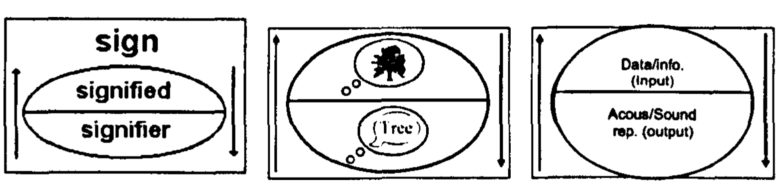


Figure 2-10: Model of Sign (Chandler, 1997 adopted from Saussure, 1974)

2.4.1 Syntactic approach

Syntactic approaches focus on the way the 'signifier' (*acoustic parameters and sound representation*) are organised to produce meaning. An example of a syntactic approach is the use of *earcons*, as explained in section 2.3.2 where non-speech sound is used to represent the information (Brewster et al., 1992). The sound or signifier does not necessarily correspond (have a close similarity) to the object being signified. Therefore any objects, including those are unable to produce any sound physically, can still be represented using sound under this approach.

Different objects and information can be represented by manipulating the structure of sound. The manipulation can be organized around three different structures; transformation, combination, and inheritance (Brewster et al., 1992). In the combination structure for example, two different sounds, which represent 'open' and 'file', can be combined and played in sequence to represent 'open file'. However, there is no standard syntax or lexicon of 'signified' and 'signifier' to this approach and therefore, a significant effort of learning is required and this becomes a major challenge of this approach.

2.4.2 Semantic approach

The Semantic approach tries to solve the learnability problem (of the syntactic approach) by focusing on the information being signified rather than the acoustic properties (signifier). An example of the semantic approach is the *auditory icon* as explained in section 2.3.3 It is a method to map objects or events in the user interface with everyday sound as the signifier.

The design could begin by analysing the interaction between objects in the interface and determining the corresponding sound, which might be produced using the same interaction in physical world. For instance, a file dragging across a computer desktop can be represented by the sound of a real file being dragged across a real desk, as demonstrated by Gaver (1994).

However, even though Gaver claimed that auditory icons are better than earcons in terms of learnability, Lucas (1994), in his series of experiments found that there was no significant difference for both auditory icons and earcons. In fact, he found that the most influential factor towards the accuracy of recognition of the auditory icons and earcons was an explanation of the *rationale behind the sound design.* Also Ballas (1994) found that the listener's *expectation, context and experience* were very important and would influence the speed and accuracy of identification of the source of a sound.

2.4.3 Pragmatic approach

As in the Syntactic approach, the Pragmatic approach also focuses on acoustic attributes - the signifier. However, the emphasis is more on the material, which forms the signifier i.e. acoustic parameters. A set of signifiers in a lexicon can be developed based on previous research results and can be used as a guideline by auditory display designers. In order to represent different signified and to avoid ambiguous signifiers, the lexicon must be discriminable (Barrass, 1997). For example, McCormick et al. (1983) introduced compatibility principles in the design of sounds, where the selection of signal dimensions should naturally be readily understood and easily discriminated by the end users. For instance, low and high sound volume (signifier) could be associated with down and up (signified).

Brewster et al.(1994) introduced some guidelines on how to use acoustic parameters for earcons. Some examples of these are as follows;

1. Timbre – timbres are used with multiple harmonics to help users' perception and to avoid masking. For instance, two different timbres can be used to represent and differentiate between two different things.
2. Pitch – Pitch is useful if used with rhythm or another parameter to differentiate earcons. However not all instruments can play all pitches. Therefore, a careful selection of timbre for certain pitches is required.
3. Rhythm and duration – The rhythms should be arranged to be as different as possible. It can be further improved by putting different numbers of notes in each rhythm.
4. Intensity – intensity is the main cause of annoyance. It should not be used on its own as people are not good at making absolute intensity and pitch judgments.

Since the focus is more on the signifier (acoustics) and its parameters, it is important for designer to ensure that its learnability is high, especially if it involves more complex information representations.

2.4.4 Task-Oriented Approach

The Task-Oriented approach concerns itself with the task (or goal) that the user is trying to accomplish. An example of Task-Oriented approach is TaDa (*Task Analysis Data Characterization),* which was developed by Barrass (1997) for sound design to support an information processing activity. It is a way to design an auditory display by analysing the task to be accomplished and the data to be converted into auditory display.

Figure 2-11 shows an example of TaDa analysis. The analysis is derived from a scenario or story, which is related to the application being designed. From the story, detailed requirements are analysed based on the 'task to be accomplished', 'information to be delivered' and 'data to be converted into sound'. Task analysis focuses on the characteristics of a task itself such as its purpose (e.g., to identify, to compare etc.), style (e.g., for exploration, monitoring etc.) and other information as shown in 'TASK' tab of Figure 2-11. The Data analysis part focuses on the characteristics of the data such as its type (e.g., nominal, ordinal etc), range (e.g., unlimited for real time data), and other information as also shown in 'DATA' tab. The 'INFO' tab refers to the characteristics of information to be delivered to the user in the form of auditory display. All these characteristics are used as requirements and guidelines for the design of auditory displays.

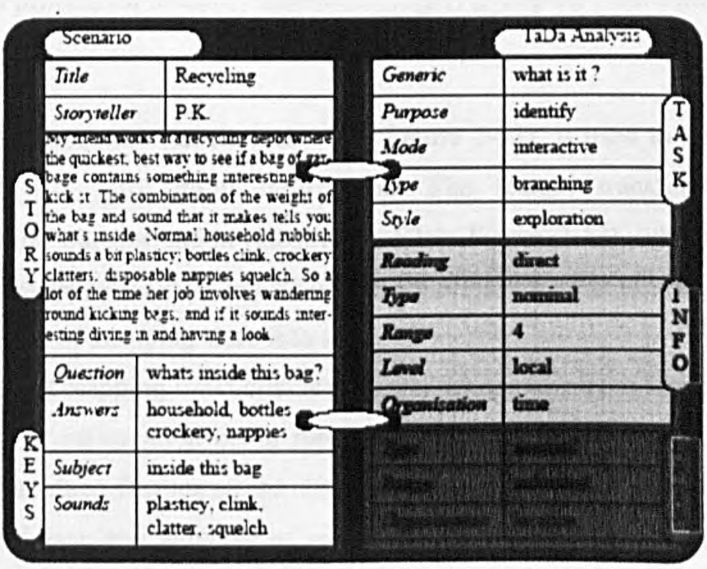


Figure 2-11: TaDa approach to auditory information design (Barrass, 1997)

2.4.5 Data State Transformation

This approach is introduced by Sylvain et al. (2003), which in turn, is inspired by the data pipeline for information visualization Chi et al. (1998). Figure 2-12 shows that there are three transformation processes; data transformation, sonification transformation and auditory display transformation.

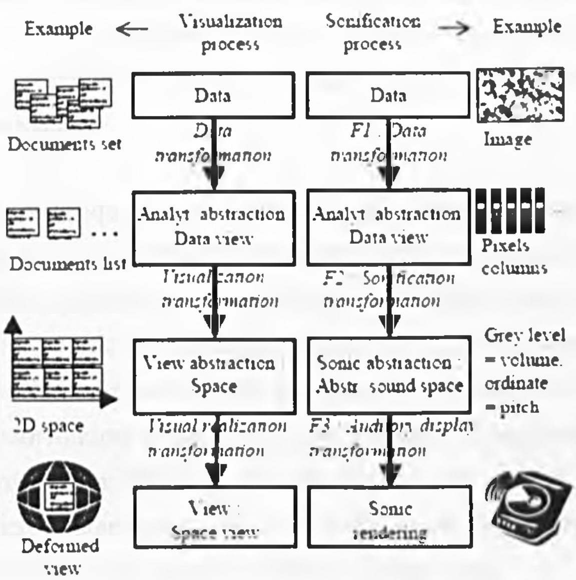


Figure 2-12 : Chi's data pipeline for information visualization (left) and Design process for auditory interfaces (Right) (Adopted from Daude et al. 2003)

Data transformation (labelled as F1 in Figure 2-12), is used to change the original data into useful information. The second transformation is sonification transformation (labelled as F2 in Figure 2-12), which consists of element representation and sound coordination. Element representation is the process of mapping data into a sound representation. It involves three independent mapping functions: *mapping of semantics* which focuses on the meaning or information being represented (such as earcons); *mapping of structures* which focuses on the structure of the data such as column, row or groups of data; and *mapping of values* which focuses on the values of the data as single elements. In sound coordination, the outputs from 'element representation' are then played either simultaneously or alternately. For instance, two sounds could be played either panned left-right through two stereo sound speakers simultaneously or separately one after anothe

The final transformation (labelled as F3 in Figure 2-12) is **auditory display transformation,** which involves the sound rendering technique. It is divided into three main steps; (1) computation of device parameters from sound space parameters, (2) computation by the sound engine of the sound signal according to device parameters; and (3) display of the signal on a physical device. The designer can only control step (1) and the rest are considered as a constraint to the overall design and dependent on the sound devices used.

By using this technique, the transformation processes of data (signified) into acoustics (signifier) can be repeated easily as a pattern and can be reused for other applications. For instance, the same data can be used by several existing sonification transformations (F2), or the same sonification transformation (F2) can be used to process several different outputs from the data transformation process (F1). Or the same transformation of data (F1) and sonification (F2) can be rendered and played with several techniques such as stereo separately or at the same time. This will produce a variety of different designs to be tested and observed.

2.5 Auditory Display Tools and Applications

This section explains briefly several existing applications and tools in sonification. These include applications and tools for programming languages, virtual reality, data exploration, monitoring applications, education, the world wide web and also several other applications. These categories show that auditory display in fact can be implemented and used in various applications.

2.5.1 Programming Language: Source Code and Language Structure

Sonification for programming language development (either visual or text environments) has been reported by several research groups in recent years. It is used especially for the purpose of program monitoring or debugging by understanding the program behaviour and data analysis. The simple method that is normally used involves mapping the program properties into sound. Examples of applications are Sonnet, ADSL, CAITLIN, SKDtools etc., which are elaborated below.

Sonnet

Jameson (1994) introduced Sonnet, an audio-enhanced system that is used to monitor and debug programs written with a visual programming language - SVPL (Sonnet Visual Programming Language). It constructs run-time actions that can be associated visually with statements or data in running programs. SVPL is also responsible to control operations such as running and halting. It was developed mainly to add audio capabilities to a debugger to help a programmer to debug a very large program easily using sound. As an example, Figure 2-13 shows that the MIDI note number 64 is sounded when line 34 is reached and the note will stop when line 39 is executed.

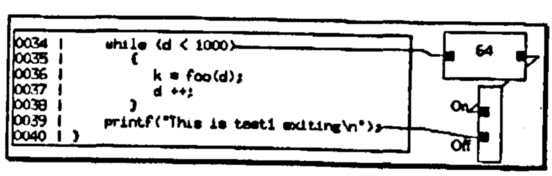


Figure 2-13: Triggering Actions at Run Time (Jameson, 1994)

ADSL

Bock (1994) introduced ADSL -the *Auditory Domain Specification Language.* It is a program auralization specification language that was developed to aid programmers in understanding how a program works, and also to determine whether or not the program is running correctly. This is done by allowing the user to create customized sound domains for their software components in 'tracks', which contain a list of program constructs, associated predefined audio cues and conditional run-time constraints. For example, Figure 2-14 shows a program auralization track used to monitor a loop operation where two different sounds are assigned to two different types of loops; *for* and *while.* The difference between ADSL and Sonnet is that ADSL adds audio to programs at the pre-processing stage.

```
Track_Name=Loop
{
1 Track=Status('for'):Snd("for_sound");
2 Track=Status('while'):Snd("while_sound");
}
```

Figure 2-14: Example of a program auralization track used to monitor software loops (Bock, 1994)

CAITLIN

CAITLIN is an auralization system that provides musical feedback of Turbo ® Pascal Programs [Vickers et al., 1996; Alty et al., 1997] as shown in Figure 2-15. The system provides a debugging environment for Pascal Programs, especially for novice programmers.

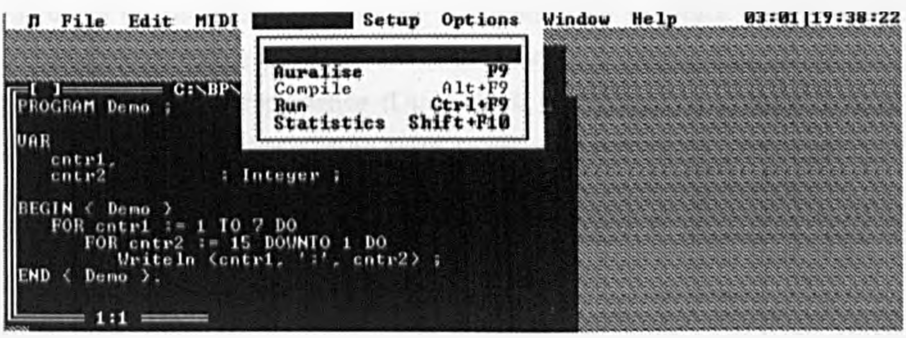


Figure 2-15: CAITLIN main screen (Vickers et al., 1996)

CAITLIN focuses on the auralisation of constructs in Pascal programs e.g. WHILE, REPEAT, FOR, CASE, IF..THEN.. ELSE and WITH. The auralisation can be divided into three parts – *beginning, execution* and *end* of the construct. The output of the execution part is influenced by the construct's characteristics, which is referred to as a Point of Interest. The term Point of Interest (POI) is used to describe features of constructs which are important for the programmer to monitor during execution. For instance, the IF construct has 4 POIs:

- *entry to the IF construct*
- *evaluation of the conditional expression*
- *execution of selected statement; and*
- *exit from the IF construct.*

The outputs of the beginning and end parts always represent the first and the last POI. A Signature tune is used to differentiate between other constructs.

Vickers et al. (1997) in their experiments found that the auralization did not significantly affect the time taken to locate bugs in the programs. They also reported that there could be also a possibility for individual who have difficulty with the debugging process itself may not benefit from using auralisation. However, in summary, all the techniques have been reported as useful especially in error detection for large programs that involve thousands of source code lines.

2.5.2 Virtual Reality

Normally, virtual environment designers try to create a feeling of presence for users by producing convincing 3D interactive graphics. Recent work shows that the usage of spatial audio in virtual environment could enhance the human sense of presence (Dinh et al, 1999). Pair et al. (1997) have produced a toolkit that can be used to create a virtual environment (VE) called the *Complete Object Oriented Library for Virtual Reality* (COOLVR). This toolkit is used by programmers to help them in creating virtual environments that are cross-platform, by not only focussing on the visual senses but also on the sense of hearing. Sounds are rendered and are stored in audio virtual reality objects. When an audio object is 'attached' to a graphics object it gives the user the illusion that the object is a real one that can produce its own unique sounds.

Investigations have also been done on the usage of Virtual Audio Reality (VAR) as an interface to computers. Ohuchi et al. (2003) for instance, have developed a game called "Hoy-Pippi" for visually impaired children, which intends to improve children's capability of spatial recognition. Huopaniemi et al (1996) have also introduced a model of real-time VAR called DIVA (Digital Interactive Virtual Acoustics), which was then further improved by Savioja et al. (1997). It can be used by acousticians and architects to plan and design a concert hall through auralization and animation. This VAR can be used also to model the interface for interaction with a personal computer (Frauenberger et al., 2003). Several big research centres like NASA Ames Research Centre and NASA Langley Research Centre are both using this VAR in their simulation experiments for aircraft operations [(Begault et al, 1996),(Rizzi et al. 2003)], looking into how sound cues could help a pilot fly an aircraft.

Inman et al. (2000) have also presented the usage and design of 3-D spatial audio for teaching *orientation* and *mobility* skills to visually impaired persons. They include sound identification, localization and tracking skills. This kind of program is very important especially if it can help blind people to learn and develop both skills faster. Even though the research in using sound in this field is relatively new, the initial results show a good indicator of improvement.

2.5.3 Data Exploration

"Exploratory data analysis is a process of sifting through data in search of interesting information or patterns" (Derthick et al, 1997). Kramer (1994) states that one of the possible applications for sonification is in data exploration. In data exploration, we are looking for something whose precise form is as yet unknown. What we want to find is an unusual pattern that probably contains valuable information. Raw data are probably easy to display but hard to perceive.

Most research on data exploration focuses on visualization. Very few researchers are focusing on exploration through sound. Some of them have embarked on sonification techniques (as explained in section 2.4) and have produced a variety of application tools, several of which are elaborated briefly below:

Listen

Listen (Wilson et al., 1996) is used in data exploration and intended to be an interactive, flexible and portable environment for sound mapping. It is implemented on a SGI platform and uses both audio and MIDI libraries (Wilson et al., 1996). The possible drawback of Listen is that the sounds produced are non-melodic, which can cause fatigue especially when exploring huge data sets for a long period of time. In terms of usability, Listen is trying to accommodate a step by step learning process by providing four basic programs; Listen 1, 2, 3 and 4. Listen 1 and 2 allow the user to get started using sonification with minimum difficulty; and as the user gains more experience, they can start to customize and increase functionality in Listen 3 and 4. For example, Listen 4 allows the sonification and all modules to be integrated with visualization programs to enhance the interaction and display of the sonification application.

MUSE

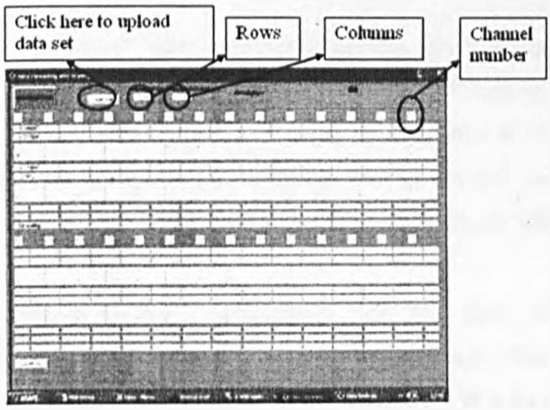
Musical Sonification Environment (MUSE) (Lodha et al., 1997) is a toolkit used to map scientific data to musical sounds. A team of computer scientists and musicians designed it for a scientific audience with two main components: music composition and a graphical user interface (GUI). It allows interactive and flexible mapping of data to six different sound parameters namely timbre, rhythm, volume, pitch, tempo and harmony (Lodha et al., 1997). One of the objectives of this toolkit is to look at whether or not musical sound could overcome the irritating or fatiguing sound problem as in *Listen*. From the results, it was claimed to be better than the non-melodic sound representation e.g. Listen, in exploring large data sets for long periods while at the same time preserving the meaning of the data.

MUSART

MUSART (*MUSical Audio transfer function Real-time Toolkit*) (Joseph et al., 2002) is another general purpose mapping technique sonification toolkit. It is used to manipulate several sound parameters such as register, pitch, timbre, loudness, and panning to produce a melodic sound map. It can also be used to sonify univariate (as well as multivariate) data. The technique and parameter settings to be used are open to users to choose and decide.

Interactive Sonification Toolkit

(a)



(b)

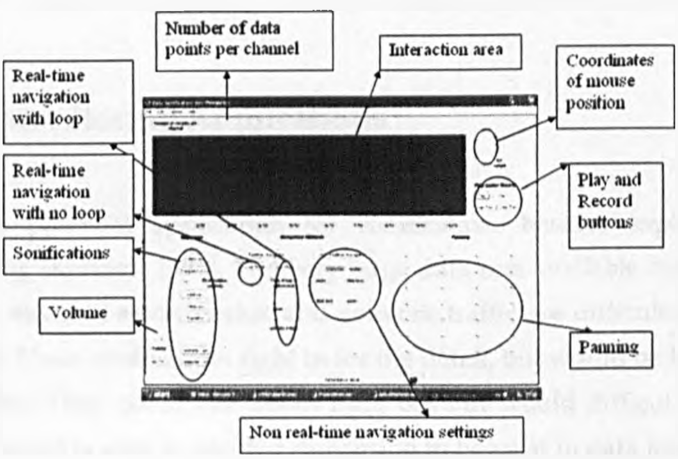


Figure 2-16: (a) Data scaling window and (b) Interactive sonification window of sonification toolkit (Pauletto et al., 2004a)

Pauletto et al. (2004a) have introduced a new interactive toolkit to help in general data set analysis. The toolkit consists of a *scaling window* and an *interactive sonification window.* The scaling window is used to upload and scale one or more data sets which depend on the sonification method. For the interactive sonification window, this toolkit provides three types of navigation: real-time, real-time with loop and non-real-time navigation. These different techniques of navigation provide users with different options for exploring data. They can also produce different types of feedback in terms of sounds and will help in data interpretation or pattern detection as the user is able to interact directly with the sound outputs. This is useful especially to attract the user's attention to investigate further certain areas of interest.

The ability to use different types of sonification methods with many channels of data has made this toolkit more flexible. It has already been tested with two types of data: physiotherapy movement and helicopter flight analysis and the results have shown the potential of sound to improve the understanding of both data types (Pauletto et al., 2004a).

In general, the problems in data exploration are: the data are not well understood and the problems are also not well specified. The larger the amounts of data that become available the more difficult it is to understand and to make sense of it. But the usage of sound as one of the optional solutions really does seem to open a new dimension to data exploration.

2.5.4 Monitoring Applications

Another potential application for sonification, besides exploration, is *monitoring* (Kramer, 1994). The very large data sets available from systems such as medical, stock market and network traffic are difficult to perceive visually. Users could zoom right in for the detail, but would be lost and out of context. They could also zoom right out but would difficult to see the detail. Sound is seen as another dimension to be used in data monitoring of this complex and multidimensional data. A few examples of monitoring applications are described briefly below.

Janata et al. (2004) presented a system called *Marketbuzz*, which is used to monitor the movement of market indices. Normally the monitoring process is done in front of two to fifteen screens showing market indicators, electronic trading platforms and proprietary spreadsheets. Traders normally make up to four hundred trades in a day, which sometimes require them to make decision within few seconds. If they fail to respond at the right time, it may result in an increased risk or a lost opportunity. Based on several experiments, it is suggested that the movement of volatile market indices can be monitored effectively by using auditory display.

A research group Bielefeld University, Germany in collaboration with their campus radio station, have been conducting a pilot project on the usage of sonification to render and present auditory weather forecasts (Hermann et al., 2003a). The information to be sonified includes expected weather events (e.g., thunder or snow) and a summary of temporal weather changes during the day. Generally, this research is more like the use of auditory icons to represent wind, rainfall, temperature, cloudiness, humidity and events. An extra effect to the auditory icon used is called Emo-Marker, which is used to evoke certain emotional affects connected with the typical weather conditions.

Malandrino et al. (2003) introduced *NeMoS*, a program to monitor distributed client/server network systems using sounds. The information is provided by the Simple Network Management Protocol (SNMP), a standard monitoring device on the network. In this program, sounds are used to complement the existing visual feedback interface. This network monitoring system is very important to ensure the efficiency and effectiveness of the resources available to the user. Since sound is time based, it is good for monitoring such real-time behaviour of the system.

By looking at these examples of applications from business, weather, network to robotics together with the promising results given; we can see that sonification is important and useful in monitoring applications especially for complex data.

2.5.5 Education

In education, several applications using sonification have been developed and the results are very promising and encouraging (Upson, 2001, 2002; Gardner et al., 1996; Walker et al., 2003; Stevents et al., 1994). Upson (2001,2002) found in his experiments that the use of sonification in teaching mathematics is engaging and fun. This indirectly helps and encourages students to learn this subject. This is a positive indicator for a new dimension of sonification application in curriculum enhancement.

Another example of sonification for education is TRIANGLE (Gardner et al., 1996) that offers an alternative to traditional graphing methods by producing them in stereo data sonification. It has been released by the Oregon State University Access Project. It includes mathematics and a science word processor, a graphing calculator, a viewer for x-y plots, a table viewer and the Touch-and-Tell program for audio. The Touch-and-Tell Program is for audio representation of tactile figures on an external digitizing pad. To use this program, a data set needs to be entered or imported into TRIANGLE's table viewer, and then it can be graphed and listened in the same plot viewer.

Bonebright et al. (2001) were also conducting a series of tests into the effectiveness of sonified graphs for education. They found that rhythm components need to be incorporated into sonification graphs to make it more musical and thus assist in people's ability to follow the auditory stream. Pares et al (2003) in their experiment found that temporal mapping is better than pitch or panning mapping for the auditory presentation of statistical graphs.

Walker et al. (2003) presented another toolkit for auditory graphs called *Sandbox*. Sandbox is a multipurpose toolkit that can be used for science and mathematics, data exploration and experimenting with various sonification techniques and parameters. It allows users to independently map several data sets to timbre, pitch, volume and panning. It also can be set to play at intervals of data points or seconds. For playback output, this toolkit only provides standard play, stop and pause buttons with interactive progress bar. Since the provided interactions are very limited, the success of data interpretation is very much dependent on the success of sound mapping and final representation that can only be played and stopped.

Another example is Algebra earcons by Stevens et al. (1994). The algebra notation is sonified to provide an 'audio glance' to facilitate planning prior to reading. This is useful especially for blind users in algebra equation representation.

In conclusion, more research and applications are required for sonification in education especially in content development. The scope must not only focus on graphical representation or mathematics subjects but also to other subjects such as geography, biology, physics and so forth.

2.5.6 World Wide Web Applications

The Internet nowadays offers huge amounts of information in many forms such as text, images, sounds, multimedia etc. Mostly this information is in a visual form: and this has created a new major accessibility problem for blind and visual impaired users. Several research streams as well as technologies have been introduced and developed to make Internet browsing possible for visually impaired people. For instance, Roth et al. (1998) produced *AB-Web,* an audio web browser to facilitate blind users to access to the WWW. By interacting through a touch-sensitive screen, the browser generates a virtual sound of the information, which is not only the text but also the images.

Petrucci et al. (2000) have also introduced *WebSound,* a web browser for blind and visually impaired users. It is a new generic web sonification tool, which applies 3D audio augmentation to an Internet browser (Internet Explorer 5.0). Teppo et al. (2001) have introduced another solution for Internet document (html file) by converting it into VoiceXML so that it can be navigated through a speech interface. VoiceXML is designed for creating audio dialogs that feature synthesised speech; digitised audio; recognition of spoken input; recording of spoken; input; telephony; and mixed-initiative conversations[1]. This is not only good for blind people but also useful for Internet browsing through small visual display devices such as Personal Digital Assistants (PDA) and mobile phones.

The strength of the world wide web is its interactive hyperlinks, which interlink documents to each other producing a network of information. Therefore, Braun et al. (1998) tried to take the advantage of this capability by introducing *sonic hyperlink,* which is a sound annotated link. In terms of searching purposes, there are so many search engines currently available such as Google, Altavista, Cari, Yahoo, Netscape Search etc. However, none of them provide facilities for blind people to do searching on the Internet. Ferworn et al. (2000) were trying to solve this problem by introducing a new auditory WWW search tool. This search engine will organize search results into a voice menu format.

---

[1] *VoiceXML Forum, http://www.voicexml.org*

From the browser and digital content to the search engine, the usage of auditory display in World Wide Web sounds very promising even though it is still in its infancy.

2.5.7 Other Applications

Sonification is also used in multimedia performing arts and live musical performance. Nagashima (2002) in his PEGASUS (*Performing Environment of Granulation, Automata, Succession and Unified-Synchronism*) project looked at a real-time interactive performance (in both graphic and sound) between human performer and computer systems. The system used many sensors placed on the human body as interfaces for the interactive communication and producing sound. Examples of performing approach are sensing with breathing in performing arts, muscle performing music and body hearing sounds (Nagashima, 2002). Fernström et al (2001a) have shown their creative technique of producing music from arbitrary numerical data sets. The data used were from the Irish meteorological service; with over 77,000 data points (rainfall data) to be sonified to become a creative musical piece. Previously, Fernström et al. (1998) have introduced an interactive floor-space called *LiteFoot*. It is used to track dancers' steps and convert them into an auditory and visual display.

Sonification is also useful in representing geographical information especially for blind people. An example of such system is KnowWhere ™ System by Krueger et al.(1997), which was developed to present geographic information data using sound. It is done by substituting the tactile sensation of touching a virtual object with sound. It is designed mainly for blind people to help them to navigate a virtual map. The sound feedback provides information about the point on the map at the tip of the finger, and this system can support up to 40 points. Even though the system has the ability to display more points, the maximum number which human can perceive, analyze and interpret is also very important to assure the effectiveness of this system.

Using sounds in medical applications is not new. The use of stethoscopes to listen to breath noise and heart tones is good enough to demonstrate that sounds are very useful in several medical applications. Pauletto et al.

(2004a) have been using their toolkit for physiotherapy movement analysis. The toolkit helps therapists to listen and assess a complex set of movements, for instance the difference in tension between two muscles. The advantage of this toolkit is that the analysis can be done in real-time, or alternatively without the patient being around – by analysing the recorded data.

2.6 Summary

This chapter has given an overview of sonification including its definitions, techniques, design approach and examples of application. Examples of existing techniques are audification, earcons, auditory icons, parameter mapping and model-based sonification. Audification is a direct data to sound conversion that can be used to sonify a huge volume of time series data. Earcons and Auditory Icons are representations of information, objects, operation or interaction by using sounds, which are normally used as to help visually impaired or blind users. Parameter mapping manipulates acoustic parameters, which allows designers to come up with many mapping strategies. In terms of the approach to sonic design, this chapter discussed five main approaches – namely syntactic, semantic, pragmatic, task-oriented and data state transformation approaches. The syntactic approach focuses on the sound representation and how it is organized to convey messages that are more complex. The semantic approach focuses on the data or information to be sonified and the metaphorical meaning of sound. The pragmatic approach focuses on acoustic attributes that emphasise the material used to form the sound. The Task-oriented approach places more emphasis on the function or the tasks to be accomplished. Data state transformation focuses on how the data is transformed from its original form into sound representations. Finally, this chapter gives several examples of where sonification can be used such as in programming language, virtual reality, data exploration, education and so forth.

In the next chapter, we will discuss existing usability evaluation techniques, and how sonification applications are currently designed and evaluated.

CHAPTER 3 : USABILITY EVALUATION

This section gives an overview of usability evaluation. Several definitions of usability, as well as current methods in usability evaluations, will be explained (*Sections 3.1 and 3.2*). We then look at scenarios of evaluation in existing applications or in previous research in auditory displays (*Section 3.3*). Finally, several design issues that need to be pointed out in sonification applications are discussed (*Section 3.4*).

3.1 Usability Definition

The term *usability* is intended to replace the generic term "user friendly". Many definitions have been given to this word due to the various approaches in making a product usable. Below are several definitions of usability from the ISO standard which concerns product and process oriented standards (Bevan, 2001):

1. *"Usability is the extent to which a product* ***can be used*** *by* ***specified users*** *to* ***achieve specified goals*** *with* ***effectiveness, efficiency*** *and* ***satisfaction*** *in a* ***specified context of use*** [ISO 9241-11: Guidance on Usability (1998)]" (ibid)

2. *"Usability is the* ***capability*** *of the software product to be* ***understood, learned, used*** *and* ***attractive*** *to the user, when used* ***under specified conditions*** [ISO/IEC FDIS 9126-1: Software Engineering – Product quality – Part 1: Quality model (2000)]" (ibid)

3. *"Quality in use: the* ***capability*** *of the software product to enable specified users to* ***achieve specified goals*** *with* ***effectiveness, productivity, safety*** *and* ***satisfaction*** *in a* ***specified context of use*** [ISO/IEC FDIS 9126-1]" (ibid)

Definition 1 explains usability by measuring user performance and satisfaction in a 'specific context of use'. According to this standard, the measurement of system usability consists of three main attributes;

effectiveness, efficiency and satisfaction. *Effectiveness* deals with the degree to which a system fulfils its intended function or goal. *Efficiency* deals with the application resources, which are required and used in order to achieve its intended function or goal. It can be measured either by human mental or physical performances such as the time taken to get things done etc. Finally *Satisfaction* is about how the users feel about their use of the system.

Definition 2 is taken from software engineering and the word *'capability'* is used instead of *'can be used'* as in definition 1, which indicates that there is no definite answer of whether or not the product is usable, but rather a capability to be used in a specific context or conditions. In definition 3 this has been expanded to become more general by using a different term "Quality in use" instead of "Usability".

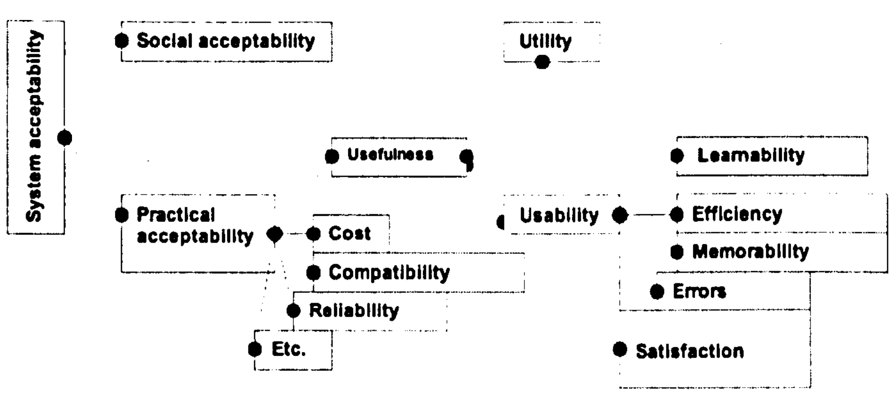


Figure 3-1: A Model of the attributes of system acceptability
(Adapted from Nielsen, 1993, pp.25)

Another definition of Usability is given by Nielsen (1993) by dividing it into five different attributes as shown in Figure 3-1 above. The attributes are:

1. *Learnability* – the application should allow new users to easily start using it
2. *Efficiency* – the application should be able to increase users' performance as compared to similar existing applications.

3. *Memorability*[1] – the application should be memorable and easily recalled, to allow users who have used it before to use the application without having problems.
4. *Errors* – the application should be free from errors especially catastrophic errors.
5. *Satisfaction* – the users should be satisfied with the application and enjoy using it.

The level of importance and criticality of these attributes depends on the type of application. For instance, efficiency will be more vital for time-critical applications. If we look at these five attributes, Nielsen did not mention *effectiveness* which is included in all ISO standard definitions above. But instead, he adds three extras measurements: learnability, memorability, and errors. He also defines Usefulness as "*the issue of whether the system can be used to achieve some desired goal*" (Nielsen, 1993). It is further broken down into utility and usability where "*utility is the question of whether the functionality of the system in principle can do what is needed* [..while..] *usability is the question of how well users can use the functionality*" (Nielsen, 1993). Similarity exists between both definitions of effectiveness and utility, which focus on the functionality or the intended purpose of the application.

Grudin (1993) explains that the requirements and the functions of a product are normally predetermined by the managers and marketing people before giving them to the developers. Therefore, the system's utility is no longer the concern of the developers. Both Nielsen and Grudin consider usability engineering as performance attributes of the system that can be measured without considering its utility. On the other hand, Bevan (1995) thinks that usability must be seen in a broader way: whether or not the user achieves their intended goal when they use the product.

[1] *The difference between Memorability and Learnability is in the focus of user level. Learnability focuses on novice users who are new with the application. Memorability focuses on casual users who have used the application before, and who just need to remember how to use it based on their previous learning.*

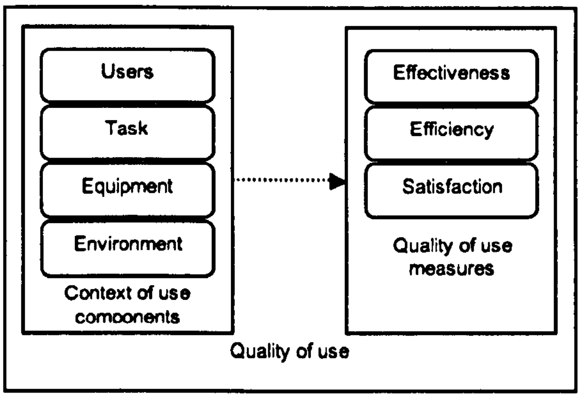


Figure 3-2: Usability Factors (Bevan et al., 1994)

Figure 3-2 above shows four 'context of use components' that need to be taken into consideration in determining quality of use. The components are users, task, equipment and environment. These components are evaluated based on 'quality of use measures' (*effectiveness, efficiency and satisfaction*) by gathering metrics performance. Examples of metrics are *the time taken to achieve a task, time spent on errors, percentage of errors, number of times that user express frustration* etc. (Bevan et al., 1994). These results are then analysed to check the level of effectiveness, efficiency and satisfaction. Evaluation might produce different levels of usability performance for the same piece of software when it is used in a different context. For example, by asking different levels of user (such as novice and intermediate), the results such as time spent and percentage of tasks accomplished might also be different.

The focus on certain 'quality of use measures' could also be different, depending on the type of application being evaluated. For instance, memorable functions are essential for the user if the application is to be used infrequently (Scholtz, 2004). If the application is time-critical then efficiency will be important. For sonification applications, the most critical attribute seems to be effectiveness (will be explained further in Chapter 4), which depends on the sonification technique used.

In summary, different applications require a different focus on usability evaluation. The focus is dependent on the nature of the application, its required goals as well as its context of use.

3.2 Evaluation Methods

Figure 3-3 below shows the timeline of the development of usability evaluation over the past 30 years. It began with methods that require the user's involvement, such as *usability labs* and *metrics for user performance* (shown on the timeline). User-centred evaluation is accomplished by identifying representative users, representative tasks and developing a procedure for detecting potential usability problems with the tasks.

This was later followed by model representation such as the Goals Operators Methods Selection (GOMS) model (John et al., 1996). Models were used to explain a more complicated application which involved many objects, tasks or procedures. Models can be constructed using computer-aided tools such as the ConcurTaskTreeEnvironment[2] (CTTE) for task based modelling, the GLEAN3[3] tool for procedure-based modelling; and the Unified Modelling Language[4] (UML) for object-based modelling.

This was then followed by expert-based evaluation methods which focus on inspection during the design and development process, such as Heuristic Evaluation (Nielsen J., 1993) and Cognitive Walkthrough (Wharton et al. 1992). These evaluations are similar to software review by experts. They are qualitative evaluations where a review can be carried out through guidelines and scenarios.

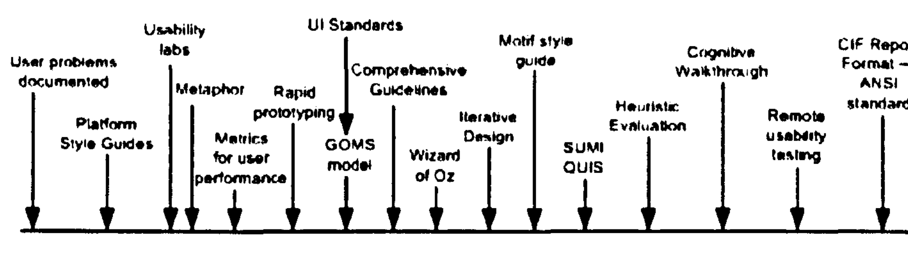


Figure 3-3: 30 years of highlights in the development of desktop computing user evaluations from 1971 - 2001 (Scholtz, 2004)

2 *http://giove.cnuce.cnr.it/CTTE /predownload.html*
3 *http://www.cs.uoregon.edu/education/classes/00W/cis677/GLEAN3.html*
4 *http://www.uml.org*

In general, usability evaluation methods can be also categorized based on which phases they could possibly be implemented in the Royce's waterfall model (Royce, 1970), as shown in Figure 3-4. The model shows five phases in the software development process.

Methods which primarily use **Inquiry** are usually implemented during the *Gathering Information* phase. Such methods can be used to obtain information about the user's understanding and requirements of an existing or proposed system. They are useful for producing a user's task analysis for the *Design* Phase.

Methods that can be termed **Inspection** are implemented during the *Design* phase. Such methods are also categorized as Formative or Analytical evaluation, which is normally conducted by experts (Bell College, 2004). They are useful for guiding the process of redesign and for solving problems before the product is completed.

Methods which mainly concern **Testing** are normally user-centred and conducted with the finished product. Such methods are also categorized as Summative or Empirical evaluation, which is conducted by end users (Scholtz, 2004). They are useful for getting real problems and feedback.

Examples of evaluation techniques are given in the Figure 3-4 below and some are described in the next section.

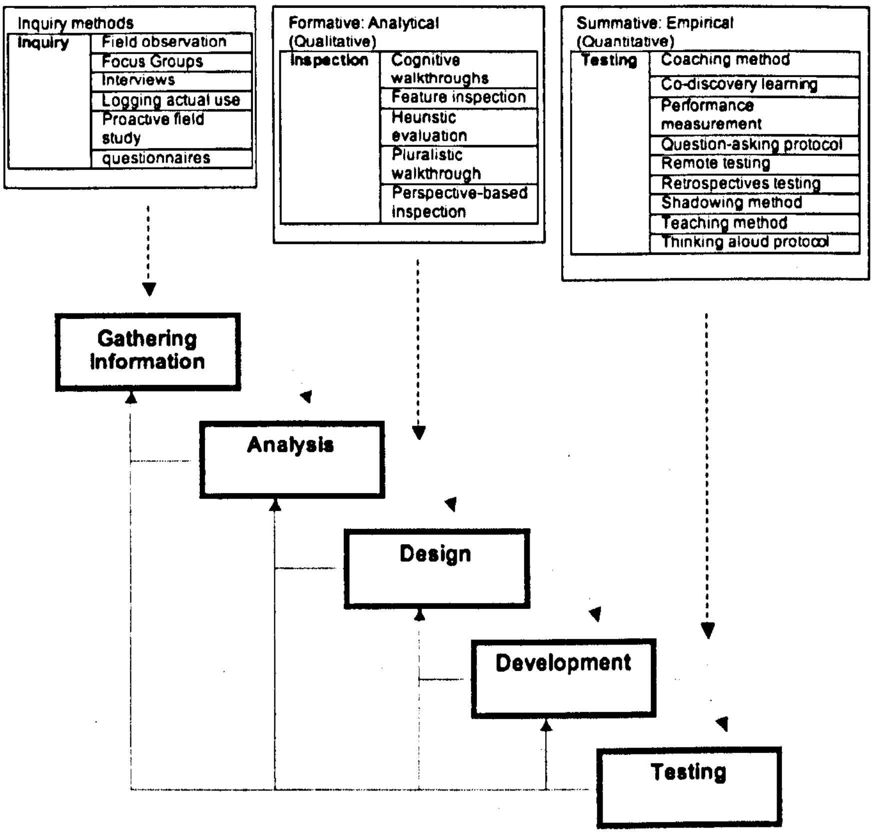


Figure 3-4: Usability Evaluation in the Royce's waterfall model (Royce, 1970)

For this thesis, several methods will be explained briefly based on the following main sections:

1. Observation and monitoring
2. Questionnaires and Interviews
3. Interpretive evaluation
4. Predictive evaluation

3.2.1 Observation and Monitoring

Observation and monitoring evaluations are normally conducted in the lab where interactions of the user with the product are observed and monitored for subsequent analysis. There are two important keys in these methods: "seeing" and "listening" (Taylor-Powell et al., 1996). These allow the designers to gather information that can be observed directly as the output of the interaction between users and an application. For instance, if we were observing people drawing and mixing colour while listening to sound, then the drawing, the way they mix colours etc. are examples of the useful information.

The evaluation data can be gathered and collected either through *direct observation* or *indirect observation*. In *direct observation*, the user's performance is recorded and checked directly by the researcher straight away (Taylor-Powell et al., 1996). However, this method can lead to a few problems such as data misinterpretation and the *Hawthorne effect* which is a phenomenon where the performance of the user might no longer be genuine due to the presence of researcher during the experiment (Draper (2006), Macefield (2007)).

Indirect observation can be done without the presence of researcher by using a video camera, sound recording or logging system where the computer will record any keystrokes or mouse movements in log files (Gediga, 2002). Since observation is done indirectly, further information can still be obtained by encouraging the users to speak their thoughts while doing the experiment. This method is called Thinking Aloud (Someren et al., 1994) or Pre-event Protocol (Gediga, 2002). In another method, called Post-event Protocols, users are allowed to give their comments on what they are trying to do by looking back at the recorded video after the experiment sessions (Gediga, 2002).

All observed activities need to be recorded (in case further evaluation is needed). The recording can be in various forms depending on the needs and type of application being evaluated. For instance, evaluators can use paper (writing), picture, voice and video recording or a combination of those techniques. To gather more and detailed information, the usage of high-tech equipment might be required, but this may lead to an expensive evaluation and huge amount of output data to be analysed. Laakso et al. (2002) were trying to solve this problem by introducing Discount User Observation (DUO), which reduces the amount of evaluation data without affecting its important information. It is a timeline-based documentation, where 'data samples' and 'time-stamped notes' are recorded with a digital camera.

In summary, both observation and monitoring techniques often produce huge amounts of data, which can cause difficulty and become time consuming during the analysis phase. The *Hawthorne effect* could also occur, especially if the users know that they are being monitored during the evaluation.

3.2.2 Questionnaires and Interviews

Users' opinions about a product or the software being developed or tested are important in order for developers to know their requirements and whether or not they like it. These can be done through *questionnaires* and *interviews*. These techniques can be used either before the development for requirements gathering or after the product is completed.

Questionnaires can be in open-ended or closed form. Closed questionnaires are where the respondent needs to choose from a set of given answers. Open questionnaires are a bit more flexible, where the respondent can provide his or her own answers (Taylor-Powell, 1998). Several guidelines on how to construct a good questionnaire are available, for instance in Taylor-Powell (1998). They suggest that information to be obtained through questionnaires must be divided into four different types: knowledge, beliefs-attitudes-opinions, behaviour and attributes (Taylor-Powell, 1998). Each of these types can be used as a reference and guideline in designing a questionnaire. In addition, a pilot study of the questionnaire can be also carried out to allow any changes and amendments to the questions before distributing them.

One of the potential problems of the questionnaire technique is the possibility of ending up with *limited information*. This is due to the number of questions in a questionnaire being finite. However, Root et al. (1983) found that open-ended questions can be added to reduce this limitation problem. Even though a lot more questions could be added, it might make the respondent feel uncomfortable. Sometimes the respondent has very limited time to answer many questions, especially if it involves experts or busy people. This human factor can lead to invalid and unreliable results. In addition, the number of respondents is also important to fulfil statistical validation and reliability issues of the results.

Besides questionnaires, users' opinions can be also gathered through interviews. There are two sorts of *Interview*: structured and flexible. A *Structured interview* has pre-determined questions to be asked to all respondents. In contrast, a *Flexible interview* is not concerned with pre-determined questions but rather is dynamically based on feedback from the respondent (McNamara, 1999). McNamara (1999) has written guidelines on how to conduct such an interview, which are also inspired by Patton (1990). He suggested that the question designer should in the first place clarify what are "the problems to be addressed" using "the information to be gathered" before they start designing the questions. By doing this, it is easier to determine the independent variables and dependent variables for the analysis of results later on.

Research on questionnaires and interviews in evaluation has kept improving in efficiency. In questionnaires for instance, they are not only done on paper but also sent through email and published as interactive web sites. Today's technology has made the questionnaire technique easier to implement, to distribute and to collect.

3.2.3 Interpretive Evaluation

Interpretive evaluation is used to understand how users use the systems without any formal instruction. This technique is conducted in an informal situation and tries to observe users' activities in their natural environments. The available methods in this technique include contextual inquiry, cooperative evaluation and participative evaluation (Sharp et. al., 2002).

Contextual inquiry is a qualitative technique for information gathering and analysis. This technique is adapted from the fields of psychology, anthropology and sociology (Raven et al., 1996). In contextual inquiry, problems are identified through evaluation within the user's real working environment. The evaluation is done together with users and researchers. According to Raven et al. (1996), contextual inquiry is based on the principles of context, partnership and focus:

1. *Context* – the data should be gathered from users' working environment.
2. *Partnership* – the design should be explored together with the users as a partner
3. *Focus* – the activities should focus on a particular situation, which could be based on the evaluator's presumptions.

These contextual processes are suitable for designing applications, which involve a real working environment and a type of work that are controlled over by the users (Mirel et al., 1996). Some examples of contextual inquiry in practice are Revere et al. (2001) in their clinical information tool and Cross et al. (2000) in their *PDA Control Presentations.*

Cooperative evaluation is designed to reduce the overall cost of usability evaluation. It can be carried out between designers and users without any HCI specialist[5]. The users are involved in deciding what the evaluations should cover as well as in result analysis together with the observers. During the evaluation, outputs of Think Aloud procedure are gathered, followed by debriefing sessions and discussions to check users' opinions.

Since observers and users are going to work together, the number of users that will participate will be limited because an increasing number of users requires a corresponding increasing number of observers. It might not be possible to evaluate the efficiency factor (involving task completion time) due to bias from the observers themselves. It is also quite difficult to implement this during the early stage of the development process because of the requirement for at least a working prototype. An example of cooperative evaluation is by Aires et al. (2003) in their proposal of evaluation for information retrieval in the Portuguese language.

The final form of interpretive evaluation is *participative evaluation.* In this technique, researchers try to become part of the users' environment. Users and researchers are involved together in the process of data collection and analysis. This differs from cooperative evaluation since it is more open and subject to greater control by the users. This approach also allows opinions from different levels of interest in an organization (e.g. top management,

---

[5] *Corporative Evaluation. Copyright European Multimedia Usability Services 1999. Website: http://www.ucc.ie/hfrg/emmus/methods/ coop.html. Downloaded: April 2005.*

marketing department, developer etc.) to be taken into account in designing the application so that it meets its real business requirements (Remenyi et al., 1999). Both cooperative and participative evaluations are aimed at making interaction between users and evaluators as relax and natural as possible.

3.2.4 Predictive Evaluation

Predictive evaluation tries to think ahead to identify the problems that users might encounter when they use the system. It is an expert-based evaluation which is carried out by knowledgeable inspectors and without actually testing the system with users. Therefore it is a formative evaluation, carried out *during* the development process. An example of predictive evaluation is the *inspection technique.*

Inspection technique is the name for a set of techniques that involve inspectors, who are normally the experts that will review and look for potential problems of a user interface design (Nielsen, 1994). Some specific examples of inspection methods are:

1. Cognitive Walkthrough
2. Consistency Inspection
3. Standards Inspection
4. Pluralistic Walkthrough
5. Heuristic Evaluation
6. Formal Usability Inspections

These methods are used to find usability problems during the application design process. The inspection does not necessarily use a working prototype but can also be performed with an 'on paper design' such as an application storyboard. Heuristic, Cognitive Walkthrough and Standards Inspection are usually done by a single inspector at a time. Consistency Inspection and Pluralistic Walkthroughs are usually done by a group of inspectors. Formal Usability Inspection is a combined individual and group inspection.

Cognitive Walkthrough (CW) is a usability inspection technique that focuses on the user's goals and knowledge towards the task being inspected (Wharton et al., 1992). This technique inspects how a user tries to start and accomplish the task being evaluated. Its main focus is on the ability of the interface to be used by first time users. It explains human computer interaction in terms of the following four steps (Rieman et al., 1995):

1. Define a goal(s) for a user's task to be completed.
2. Inspect the interface and look for any available actions that can be used to perform the task.
3. Decide on the actions that most likely can be used to perform the task.
4. Execute the actions and compare the potential feedback with the goal of the task being inspected; and decide whether or not it is being accomplished.

Consistency and **Standards Inspections** (Wixon et al., 1994) are normally done by designers to see whether the designs of the whole system are consistent. They are also for the purpose of checking compliance with international standards.

Pluralistic Walkthrough (Bias, 1994) brings together representative users, product developers and human factors professionals into a design session to discuss new ideas. Scenarios and working prototypes are used for the inspection. During the evaluation, users are required to write down any actions in as much detail as possible on how they would like to carry out the designated task. After all participants have written down their actions or responses, a verbal discussion is held to discuss their responses and any potential usability problems. All written responses and feedback from the discussion are used for the improvement of the application being evaluated. The drawback of this method (which also happens to CW) is that besides being unable to simulate all possible actions, it will go as slowly as the slowest person.

Heuristic Evaluation is a technique to inspect an interface design for its potential problems by considering several usability principles that have already been widely used and practiced (Nielsen, 1993). This method does not require advance planning and can be implemented in the early phase of the development. Nielsen listed several usability principles for heuristic evaluation, which include *"simple and natural dialogue; speak the user's language; minimise the user's memory load; consistency; feedback; clearly marked exists; shortcut; precise and constructive error messages; prevent errors; and help and documentation"* (Nielsen, 1993). The evaluators are required to inspect the interface independently of each other, by using those principles as a guideline. The results can be used as part of the iterative design process.

Nielsen (1993) found that this evaluation is quite difficult and cannot be relied upon with the results of only one evaluator. The performance will be substantially better with an increase in the number of evaluators. Figure 3-5 below shows that the *proportion of usability problems found* will increase with increased *number of evaluators.*

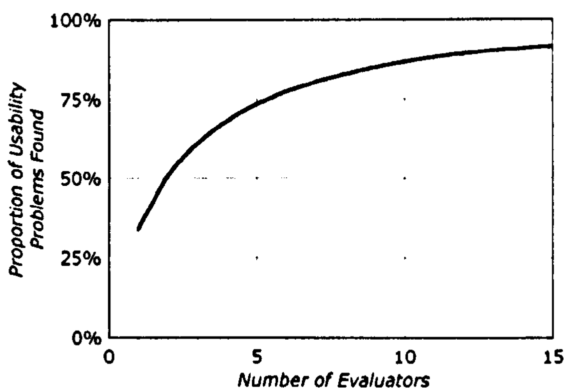


Figure 3-5: The Proportion of Usability Problems Found vs Number of Evaluators by Heuristic Evaluation (Nielsen, 1993)

But by adding more evaluators, the graph levels off gradually, showing that it will not give proportionally more benefits. Nielsen suggested that the optimum number of evaluators is about three to five. An example in practice is Kantner et al. (1997), who found that Heuristics Evaluation had the ability to detect the most visible usability problem of their web sites.

Formal Usability Inspection (Kahn et al., 1994) is a review of users' potential task performance with a piece of software. It is designed to help developers, especially engineers and programmers, to find a number of usability errors. The process consists of the following six phases:

1. *Planning* – includes selecting inspectors; and preparing inspection instructions and requirements.
2. *Kick-off Meeting* – distributing and briefing of all the inspection instructions and requirements to inspectors.
3. *Review* – inspectors are given some time to review and understand all the inspection instructions and requirements.
4. *Logging Meeting* – inspectors check each task's scenario being inspected and log all potential problems.
5. *Rework* – discussing and proposing potential solutions for the found potential problems.
6. *Follow-up* – designer will collect, evaluate and finalise the outcome of the inspection process.

Hewlett-Packard has already implemented this method successfully and they found that their engineers (*who are not trained in human factors*) were able to help improve the ease of use as well as detect usability defects (Gunn, 1995).

In conclusion, user testing on a finished product is the most complete usability evaluation. But inspection or expert-based usability methods appear to be quite effective for generating useful insight into the usability of a developing (or not yet existing) interface. They allow problems to be detected in the very early stages of development, and thus fixed earlier.

3.3 Existing Evaluation in Sonification

This section gives an overview of existing evaluation procedures for sonification applications. Several tasks that are normally used in existing evaluations are also discussed.

3.3.1 Brief overview of existing evaluation

Most previous evaluations of sonification applications are based on end-user testing with at least a working prototype. The testing is based on experimental design, which is used to predict a relationship between variables (*any characteristics that vary in different conditions of the experiments*).

To check the validity of the experiment, the researcher uses statistical tests which show the probability of the result to be significant. In general, such testing can be divided into four important requirements – participant, stimuli, procedure and results analysis.

Participant

Participants are the people who are involved directly with the experiment, who could also influence the validity and reliability of overall evaluation results. These influences include the number of participants (sample size), background knowledge, emotion felt and so forth. However, some influences are still being debated such as user's *background musical knowledge*, as several inconsistent results have occurred in previous evaluation experiments involving sonification (Edwards et al., 2000).

It is also not clear how to differentiate between users who are musically trained or not. Most of the existing evaluations assumed that musically knowledgeable participants are those who have attended formal musical classes. However, what about those who are non-formally musically trained but clearly have musical ability? Therefore Edwards et al. (2000) introduced the MAT (Musical Aptitude Test) to provide a benchmark for any auditory experiments by relating experimental results with the MAT score. For example, if the testing found out that a particular design was performed better only by users with a high MAT score, this indication suggested that the design was not good for a general audience (Edwards et al., 2000).

Stimuli

Stimuli are the sounds that users are given during the testing sessions for one or more specific task(s). The tasks and stimuli are dependent on the hypothesis and the application being tested. From the task activities, several performance measurements (metrics) are collected as the data to be analysed. For example, usability attributes such as effectiveness, efficiency, memorability and error rate can be evaluated through the **number** of correct tasks as well as the **time** taken to finish the tasks as the performance metrics.

Procedure

Experimental Procedure concerns how the user does the experiments. The same tasks and stimuli can be used to collect different data in different experimental procedures. For example, in *longitudinal* design testing, the period of training or practice can be used as the performance measurement – such a study follows the participant over an extended period. The assumption is that people need practice in order to perform a task effectively. Changes in performance after giving more practice can be used to deduce the learnability of the application being evaluated. The period of practice (or number of training sessions) can be fixed either in advance (by the evaluation designer) or by allowing the user (subject) to continue until they are ready to stop. By fixing the period and number of training sessions, a standard is set for all subjects. On the other hand, by giving flexibility to the user for practice, the period and number of training sessions can be observed as one of the performance measurements. The simple assumption is that the shorter the training period for the user to use the application successfully, the better the application.

An example in earcons, Brewster et al. (1992) divided their test experiment into four phases. Phases 1 and 2 were conducted with training and followed by testing; phase 3 was conducted without training; and phase 4 was conducted by repeating the test exactly as in phases 1 and 2 but without training. By giving tasks with and without training with different groups of participants; and repeating the same tasks with and without training for the same group of participants, the performance results were used to evaluate memorability and learnability.

Result analysis

In summary, for user testing of sonification, users are given one or more specific tasks to do, which are dependent on the hypothesis and the objectives of the evaluation. Subjects' performances are captured during the testing and this can be done: automatically by the system (*such as logging the number of errors and the time to finish*); using questionnaires such as satisfaction level; taking verbal opinions (*think-aloud evaluation method*) about the problems in using the system; recording body movement using video camera; and so forth. The data will be gathered, analysed and an experimental conclusion drawn up based on the hypothesis and objectives of the evaluation. The level of significance, validity and reliability of the results are calculated through statistical tests such as ANOVA analysis. Several unpredicted problems (*other than in the hypothesis and objectives*) can also be observed through open-ended questionnaires, verbal opinions and video recording. Such results are usually useful for improving the application or its sound design.

3.3.2 Tasks in evaluation

This section discusses several tasks that have been used in previous evaluations of sonification applications. These tasks are used to generate performance metrics as the data for measuring usability properties such as effectiveness, efficiency and satisfaction. Each task will be explained based on its purpose (reason of task) and issue (quality of task).

Matching Task

A Matching task is used to estimate whether or not a pair corresponds to each other at some aspects. This task can be used to investigate the effectiveness of sounds in data or information representations. For example, Bonebright et al. (2001) used a matching task between visual and sonified graphs to measure effectiveness of using sound in representing graphs. During the evaluation, the participants were basically required to select the visual graph that best matched the sound being played.

Another example was by Brewster et al. (1992) to evaluate their earcons design. The played sounds need to be matched by the user with the graphic icons it represents as shown in Figure 3-6 below.

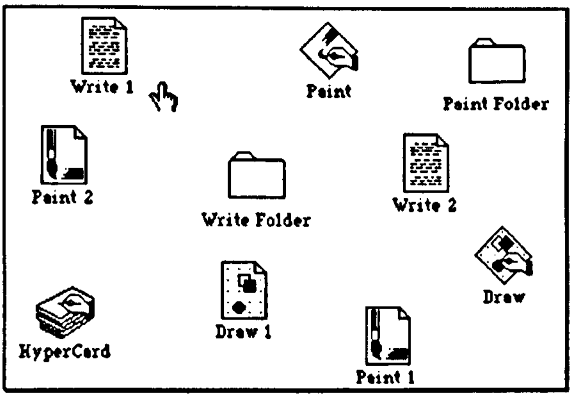


Figure 3-6: Icon Screen (Brewster et al., 1992)

In these examples, the issues were how to increase the score of correct matches and how to speed up the task. These two quality aspects can be used to measure the effectiveness of sounds in object (graph, icon etc) representations.

Comparison Task

A Comparison task is used to estimate the similarities and differences between two or more sounds; or a sound with the object it represents (information, images, data etc). The result could be either they are 'similar' or 'different'; grades of difference; detail how they differ etc. For example, Bonebright et al. (2001) used comparison results of how users understood visual and sonified graphs to evaluate the effectiveness of their sonified graph design. In the experiment, each graph was given with a written description of the data set as well as several questions for the user to answer. One of the questions required the user to compare the data points in the graph while listening to the auditory display of the graph.

A comparison task can be used to measure effectiveness and efficiency by measuring the number of correct comparisons (e.g. between visual and sonified graphs) as well as the time taken to finish the task.

Classification Task

A Classification task is used to arrange or organize sounds according to its categories. This task can be used to investigate the effects of sound properties. For example, Martins et al. (1997) conducted an experiment to evaluate whether or not sounds could be classified according to some different basic visual shapes that they should be representing (e.g. circle, square, ellipses and hashes).

In this task, the effectiveness could be measured by looking at the percentage of users that successfully classifed the sounds as the correct set.

Ordering (Sorting) Task

An Ordering task is used to arrange the separable elements of a group based on their properties. This type of task can be used to investigate the perceptual effects for a large number of stimuli. For example, Bonebright et al. (1998) used this type of task to observe the most important attributes of sounds by analyzing how the subjects group the sound stimuli. In the experiment, subjects were required to perform the sorting task by putting the auditory stimuli into groups according to what they felt the group should be and how they perceived the relationships between stimuli. From this experiment, the most important attributes of the sound can be observed by analyzing at how the subjects group the stimuli.

Association Task

An Association task is used to make connections or relations between sounds and information or data. For example, Stevens et al. (1994) used this type of task to see whether or not the user could associate the sounds (audio glance) being evaluated with any algebra expressions. The assumption for this testing was that if a listener can recover sufficient information by being able to select an expression from a series or similar alternatives, the audio glance is said to be effective in presenting syntactic information. Syntactic information is the information that is represented with alphabetic characters or symbols e.g. mathematical representation.

Prediction Task

A Prediction task is used to forecast knowledge or to guess about the future. It can be influenced by users' prior knowledge, previous observation, reading etc. Prediction tasks are especially important for sonification applications, which involve data analysis or data mining, where users are required to predict patterns or knowledge from sounds. For example, Nesbitt et al. (2002) investigated the influence of sonification in analyzing and predicting the price movement of market stock data. The task was to predict the next position of data value – go up, go down or remain the same. In this evaluation, the number of correct predictions was used to measure the effectiveness of the sonification technique used for the market stock data.

Finding Task

A Finding task is used when the user wants to look for or discover something. It is also very useful for such data mining sonification applications where the user needs to find patterns and deduce possible new knowledge from them. Nesbitt at al. (2002) also used this type of task to observe whether or not the user could find consistent patterns in the stock market data being observed. Current traders read the stock data manually, from numeric tables, to find patterns that will assist them in decision making. As compared to this manual way, the Finding task was used to measure the user's performance by observing whether they could find the auditory patterns.

Memorization Task

A Memorization task concerns the user remembering something such as the meaning of certain sounds. This is important for sound-based applications especially if they require the user to compare data that cannot be played simultaneously (as the user will need to listen sequentially to different pieces of data, and thus remember the first when comparing to the second). Maffiolo et al. (2002) used a memorization task to evaluate the efficiency of the sonified vocal server (*called Avantys*[6]) as compared to their original vocal server. The memorization task was observed by looking at the number of "elementary actions" the user had to do to achieve a specific task. The system is said to be efficient if it could reduce the number of "elementary actions" for the same task.

Brewster et al. (1992) also tested their earcons design in terms of memorization. It was used to test the learnability and memorability of the earcons by investigating whether the subject could remember the original set after having learnt another similar set. This was done by allowing the users to get familiar with the sound and its iconic representation; and then giving another, similar, set for real testing.

---

[6] Maffiolo et al. (2002)

Navigation Task

A Navigation task involves planning and controlling the position of the user in the system. The easier the user finds it to navigate the system, the better the application is in terms of user-friendliness. For example, Maffiolo et al. (2002) used the number of keys pressed from different scenarios to evaluate the ease of navigation within the system. They also recorded the amount of time that the user spent on the help menu. The assumption was that if the number of keys pressed increased, this will also increase the difficulty of the system. Also, less time spent on the Help menu, the more we can assume that subjects have understood well what they have to do.

Identification Tasks

An Identification task involves the act of determining the properties of particular sounds. This technique can be used to investigate the ability of sounds to be perceived and recognised uniquely. This can be done by determining whether subjects can *correctly identify* objects or events with their associated sounds. For example, Bonebright et al. (1998) used this technique to determine whether or not the response towards sound stimuli was identical among subjects. In this case, response results were used to determine whether or not the subject would correctly identify the sound stimuli. The response frequency from these experiments was used to indicate the significance of the identifications. It also provides information about sounds properties that potentially confuse the users.

In summary, this section has discussed the previous evaluation techniques and tasks involved in measuring usability properties of sonification applications. For the evaluation, the designer basically **requires several participants** with at least a **working prototype** of the application to be tested (e.g. sound stimuli).

3.4 Issues in Sonification Design

This section explains several design issues that have arisen from the previous development of sonification applications. The issues are based on existing designs and evaluations of sonification applications. The issues are as follows.

Data reliability and resolution

The reliability of sound representation (output) in carrying the correct and precise information for sonification applications is dependent on the reliability of its data (input). For example, the auditory weather forecasts by Hermann et al. (2003a) require reliable data for the weather predictions.

This is even more critical for applications that involve data exploration and analysis, where the results of the applications are dependent on data quality. In this kind of application – the more data can be generated within a second (sampling rate), the more precise the source for the sonification as well as the sound output. For instance, if the important information or pattern happens within microseconds of the data stream, and if the system is only able to generate the data every second, therefore, this important information will not be captured in the data. As a result, this information cannot be revealed by the sound and this is not because of the ineffectiveness of the sonic representation or the sonification design but rather the quality of the data source itself. Therefore, reliability and acceptable resolution of data source is important for sonification application.

Data set reduction

Some sort of data reduction technique is required in some sonification applications, especially if they involve large and high redundancy data sets. Hermann (2002) gave several examples of model-based sonification applications where only a few selected important features of multivariate data were used as the input to create the sonification models. The selection of significant features was done by using a statistical method called Principal Component Analysis[7] (PCA), which exploits redundancy in multivariate data. This method could help in reducing data dimensionality without significantly losing its important information or content. Therefore, the reduction of redundancy without affecting the meaning of the data will increase the effectiveness and efficiency of the sonification application for data exploration and analysis.

Data scaling

Some data needs to be scaled to assure the compatibility of the data for sonification transformations. For example, in a direct data-to-sound

[7] *http://149.170.199.144/ multivar/pca.htm*

conversion technique such as audification, the data might need to be scaled to ensure that the output amplitudes and frequencies are within the range of human hearing. Pauletto et al. (2004a) introduced two different methods of data scaling in their sonification toolkit; *"(1) defining new minimum and maximum values and (2) defining new transposition and stretching factors"*. By doing this, several new sets of scaled data from a source of data can be produced to suit the sonification techniques.

Data insufficiency

Some techniques require a lot of data to make them more effective. For example, audification requires huge amounts of data, especially if the transformation is at a higher sampling rate. Because in audification, the sound samples is obtained directly from the data values. Although this makes it a very useful technique for dense data sets, problems will occur for small non-stream or non-temporal data as well as slow-changing series of data. As a result, the important pattern might not be revealed due to data insufficiency.

Mental image of data

"A mental image is an experience that significantly resembles the experience of perceiving some object, event, or scene, but that occurs when the relevant object, event, or scene is not actually present to the senses" (Finke, 1989). The mental image of data is important especially in understanding the sound representation of non-temporal data. Users could face problems in understanding and interpreting sounds if they are unable to build a correct mental image or representation of the data. Dufresne et al. (1996) supported this in their experiment of multimodal access to windows for blind users, which found that the user's understanding level was significantly increased if the sound used helped to build a mental representation of the object being presented.

Some designers incorporate different modalities of feedback into their design, such as haptic, to further improve users' mental representation of data. As an example, Petrucci et al. (2000) introduced a web browser for blind users called WebSound, which was a "3D audio augmented internet browser". The browser used both Auditory and Haptic modalities to help in developing the user's mental representation of the HTML document layout. This mental representation using sound is important as alternative representation techniques of the spatial and visual related information in an HTML document especially for blind and visually impaired users.

Musical Knowledge Requirement

Based on the Causal Framework of Usability as shown in Figure 3-7, the user's knowledge is one of the factors that could influence their reaction towards applications, which would affect their performance in accomplishing the task. In the field of Auditory Display, many researchers have explored the effects of musical knowledge towards the interface design such as Brewster (1994), Stevens (1996) and Vickers and Alty (2000). They wanted to know whether the user with musical knowledge would perform better for the given tasks as compared to those who have not.

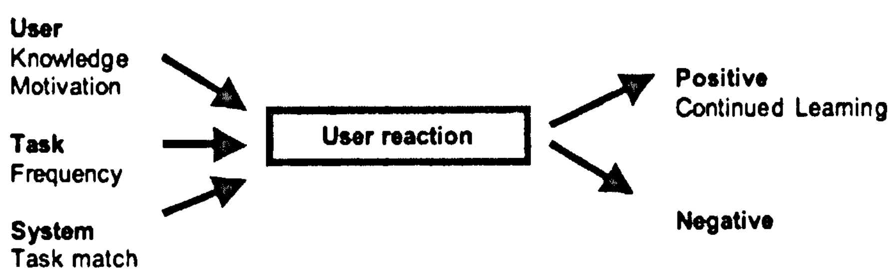


Figure 3-7: A Causal Framework for Usability [(adapted from Eason, 1984), Lowgren, 1995)]

Edwards et al. (2000), describes the ambiguity that exists as two of his PhD researchers have produced two different opinions on the significance of the influence of music knowledge towards the usage of earcons. Brewster (1994) found that it significantly influenced the results, whereas Stevens (1996) found that it did not. Stevens et al. (1994) found in their research of algebra for blind readers that subjects who had musical training performed significantly better. On the other hand, in auralization, Vickers et al. (2000) found that if the subjects had knowledge and experience of music this did not significantly affect their experiment results. Due to the inconsistent results, Edwards et al. (2000) proposed a standard test of musical ability called Musical Aptitude Test (MAT) to become the benchmark of musical knowledge and ability for participant (as discussed before in Section 3.3.1 Participant).

Sound Type – Melodic and Non-melodic

One of the potential drawbacks of using non-melodic sound is user fatigue, especially in data exploration applications, which involve exploration of large data sets for a long period of time. The effects of using non-melodic sound was reported by Brewster et al. (1992) in their experiment where the performance of musical (melodic) earcons was better in terms of understanding graphs acoustically even though the differences were not statistically significant. The issue was also reported by Wilson et al. (1996) as the potential drawback of their sonification tool (LISTEN). Lodha et al. (1997) then introduced MUSE, a toolkit to map scientific data to melodic sounds. They found out that it was at least better to use melodic than non-melodic sound type in exploring large data sets for a long period as it could avoid user fatigue.

Number and Type of Acoustic Parameters

The number and type of acoustic parameters used to represent information are very important, due to the limitations of human hearing for perceiving different parameters at the same time. Brewster et al. (1992) found out that five acoustic parameters are the maximum number for designing effective auditory icons. They also found that musical timbres are more effective than simple tones. This was also supported by Stevens et al. (1994) who found that timbre and timing were important and could be used to convey different meanings. Stevens et al. (2004a) found that there was possibility for recognition accuracy to be decreased as the number of parameters increases. However, the results were not statistically significant.

Another potential problem is sound annoyance, which is often related to sound density. The sound density can be perceived as the thickness of a sound, with contributory factors such as timbre, duration, intensity, number of instruments etc. Previously research by Leplatre et al. (2004) found that this annoyance problem can be reduced and avoided by carefully designing the sound density.

Since the number and type of parameters affects the accuracy and effectiveness of the sound, the selection of the most suitable and effective parameters is important for the sound design.

Perceptual Issues

A correct perception of sound is important for developing the user's mental representation as well as improving their task performances. Previous research has shown that careful choice and manipulation of acoustic parameters can aid this perception. For instance in warning systems, Edworthy et al. (2002) found that the perception of urgency was influenced by the following four parameters - speed, fundamental frequency, repetition and inharmonicity. Another example comes from the experiment by Nesbitt et al. (2002), which looked at the fluctuation effect of frequency and amplitude towards the perception of stock market data. Based on the results, subjects seemed to be able to predict down-trades better rather than up-trades for both parameters.

Alty et al (1998) mentioned that there are three important levels of perception in auditory interfaces – namely *"detectable mapping, perceptual context* and *reasoning"*. These have been interpreted by Vickers et al. (2000) as *"uniqueness level, metaphorical level* and *semantic level"*. Detectable mapping (uniqueness level) is related to the ability of sounds to be identified uniquely. Perceptual context (metaphorical level) refers to an object carrying different meanings depending on the domain or context of its use. Reasoning (semantic level) refers to the construction of the meaning that is carried by the sounds. This is where the listeners start to reason about what the audio messages actually means.

Sound Aesthetics

Aesthetics relates to the perceived pleasantness, which is associated with sound representations. It plays a role in the sense of satisfaction experienced by the user of a system. It can be judged with term 'pleasantness experience', with assumption that a high design quality results in a pleasant experience (Khaslavsky et al., 1999). Leplatre et al. (2004) found that the relationship between the 'functional' and 'aesthetic value' were correlated. A low rating for the functionality of a sound can result in a correspondingly low rating of aesthetic value too.

This study is still new and faces the difficulty of tackling and evaluating the aesthetic value of an auditory display. However, the previous studies can at least provide an insight of potential aesthetic properties of a software interface.

Number of signals that can be played at the same time

The number of sound sources to be played at the same time could potentially make the sound cluttered and cause difficulty for the user in discriminating the individual sounds. Some work has been carried out to determine the optimum number of signals, which can be played at the same time. For example, Fernstrom et al. (1998a) found that the users who were supported by 'multiple-stream audio' performed significantly faster than users with 'a single stream audio' in browsing tasks.

Spatialization of audio (e.g. in 3D environment) is also found to be helpful in discriminating several sound signals. Rober et al. (2004) found that in interactive virtual auditory environments, spatialised sound could help listeners in the process of differentiating several audio signals that are coming from different directions and locations.

Therefore, the number of signals that are played at the same time by an application will influence the user's perception and discrimination of the sounds.

Sound Structure

Sound structure refers to the way that acoustic parameters are arranged to create or carry certain meanings. From previous experiments, Brewster et al. (1992) found out that the more 'structured sounds' were better than the 'unstructured sounds' for communication purposes. Their structured sound was divided into several levels, representing different acoustic parameters such as Rhythm in level 1, Pitch in level 2 etc. (details found in Chapter 2, section 2.3.2). This result suggested that to produce an effective sound representation, a clear sound structure needs to be considered carefully.

In summary, this section has discussed several design issues that need to be taken into consideration in designing sonification applications. These design issues can also be used as a guideline in the planning and designing of end-users evaluation by determining the issues and suitable tasks (as explained in section 3.3.2) to test them.

3.5 Summary

This chapter has given an overview of usability evaluation techniques. In general, usability evaluations can be divided based on three different phases in the Royce's waterfall model: gathering information, designing and after development (testing), which are called inquiry, inspection and testing method respectively. Inquiry methods include field observation, interviews, questionnaires etc. Inspection methods are analytical (qualitative) studies or a predictive evaluation, which includes cognitive walkthrough, heuristic evaluation, perspective-based inspection etc. Finally, testing methods are empirical (Quantitative) studies, which include thinking aloud, performance measurement, remote testing etc. Most of the previous evaluations for sonification are in the testing method with several subjects (participants) and at least a working prototype (sound stimuli). This testing involves four important requirements – namely participants, sound stimuli, procedure and results analysis.

This chapter has also discussed several tasks that normally need to be done during an evaluation: and several design issues that could influence the usability of sonification applications. The tasks can be manipulated in many different ways and situations depending on the design issues to be investigated. The feedback will be analyzed and become a reference to improve the sonification application design. Through this practice, the process of evaluation will become part of an iterative design process for sonification applications.

One way to evaluate usability is by asking the users themselves during tests on a finished product or highly developed prototype. However, substantial time and resources are required to design and run controlled user experiments, especially if they involve special equipment and dedicated room settings. Some sonification applications have so many potential tasks to evaluate, that (due to time constraints in preparing for evaluation such as preparing a prototype and looking for test subjects) only a certain and selected set of specified tasks can be tested at one time. By introducing an inexpensive and faster evaluation technique, more evaluations could be performed and earlier in the development process where it is less expensive to make changes. The previous inspection techniques were found to be efficient in detecting potential problems of software design.

Therefore, this thesis suggests extending usability inspection for sonification application design. Inspections are interesting as they can be performed quickly in the early stages of development with low cost compared to end-user evaluations. It is interesting to consider whether we can evaluate sound representations without even listening to them. Therefore, to make such an inspection possible for sonification, we need to understand what is a sonification application?; and how to describe its design?; and finally to plan how to use it for inspection. These are addressed in the following Chapter.

CHAPTER 4: HUMAN COMPUTER INTERACTION MODEL FOR SONIFICATION APPLICATIONS

This chapter discusses the Human Computer Interaction (HCI) Model for sonification applications, which has been developed by the author. The HCI model comprises two sub-models called the Sonification Application (SA) model and the User Interpretation Construction (UIC) model. These two sub-models are used to describe the design of sonification applications, and this will be explained further in this chapter. The content of this chapter has been presented as a paper and published in Springer-Verlag's Lecture Notes in Computer Science (Ibrahim and Hunt, 2006a).

4.1 Introduction

Chapter 2 showed the advantages and capabilities of sonification in various applications, especially in data representation and exploration. There are many techniques currently available for data sonification such as parameter-mapping (Kramer, 1994), Model-Based Sonification (Hermann, 2002), Audification (Dombois, 2001) and so forth. These techniques are normally guided by the type of data to be presented and the required user tasks that the sonification can support such as programming debugging (Vickers, 1999), multi-channel data display (Pauletto, 2004a), stock market prediction (Janata et al., 2004), computer network auralisation (Malandrino et al., 2003) etc.

The issues of usability should no longer be an option, but rather a requirement for the design of sonification applications. A proper design method should be involved from the very beginning of the development phase. However, only a few researchers embark on how to design this type of application in a more systematic way, such as *TaDa* by Barrass (1997) and Designing Process for Auditory Interface by Daude et al. (2003). In addition, the type of 'tasks' that the user needs to accomplish is sometimes not clear, especially if it involves data exploration. This makes the usability aspects quite difficult to implement and evaluate in such applications.

In previous practice, usability testing normally took place at the end of the development period. Therefore, the problems can only be detected when at least a prototype is ready. This puts more cost and time into the development process. As a result, it might be useful for the designers if they could have detected the problem earlier, so at least that they could have produced a better design before proceeding to the development process. The effects could be even more significant if the development involves special and more expensive equipment.

In Chapter 3, several previous testing strategies in software engineering have been discussed, some of which are still in practice. The existing techniques can be implemented at different phases of the software development life cycle e.g. over the 'Information Gathering phase', 'Designing phase' and after the 'Development phase'. Researchers have reported that usability inspection was able to successfully detect problems of software as early as the design phase (Nielsen, 1995). Usability inspection is a technique where inspector(s) examine and predict any potential problems of the software. This kind of technique is also referred as *expert-based* evaluation, which is carried out by experienced people and normally implemented at the design stage before it goes to the implementation or development phase. It requires fewer participants (typically usability experts) than controlled end-user testing. Examples of existing inspection techniques are Cognitive Walkthrough, Consistency Inspection, Pluralistic Walkthrough, Heuristic Evaluation, etc.

This thesis will look at the potential of implementing usability inspection as part of the design process for sonification applications. However, most of the existing inspection techniques are optimised towards the evaluation of *graphical* user interfaces (GUI), and with little focus on the sort of issues that are related to acoustic parameters and sound representations. Therefore, this thesis introduces a novel inspection technique called the Task-Interpretation Walkthrough (TIW) that gives more attention to the design of sonification applications. This inspection technique will be discussed further in Chapter 6.

The concepts behind the TIW are the **Human Computer Interaction (HCI) Model for Sonification Applications,** which consisting of two sub models called **Sonification Application (SA)** model and **User Interpretation Construction (UIC)** model. The SA model is used to describe and represent the design of sonification applications in a diagrammatic form, which we have called the **Task-Data State Diagram.** This diagrammatic representation will be discussed further in Chapter 5. Before we go into further details of the HCI Model, the next section will describe and give some ideas on the definition and overview of usability evaluation in current designs of sonification applications.

4.2 Definition and Purpose of Usability Evaluation for Sonification Applications

This section gives a specific definition of usability for sonification applications. The issues of usability, utility and context of use are also briefly discussed. Some issues will be discussed by posing several (numbered) questions, which we will refer to later in the thesis.

Usability Issues

The definition of usability has already been discussed in detail in Chapter 3, where several usability experts have introduced a number of usability parameters such as effectiveness, efficiency, satisfaction, memorability etc. The importance level of each parameter is dependent on the type and purpose of the application involved. For instance, the effectiveness parameter might be more important for critical applications or expert tools such as data analysis applications, which require precise outputs.

In general, these parameters are measured to answer these questions:

1. **How well can the users use the system?** (Q1)
2. **Can the system, in principle, do what is needed?** (Q2)

The ISO Standard has treated these questions as 'usability', which can be measured through three attributes effectiveness, efficiency and satisfaction (Bevan, 2001). However, researchers in usability engineering have treated these questions as two different issues – usability and utility for the first and second questions respectively. For example, Nielsen (1993) viewed *usability* as how well the users can use the system (the first question) with five attributes called efficiency, learnability, memorability, low error rate and satisfaction. The second question is treated as *utility,* which is a question of whether the application is actually capable of doing what the user required. The combination of these two (usability and utility) is called *usefulness,* which focuses on the ability of a system to accomplish its desired goal. Chapter 3 has shown that the utility parameter has already been debated before, whether or not it needs to be included as part of usability parameters.

Utility Issues

Sonification is an interface technology like information visualization but using sound to represent data or information. Figure 4-1 shows that this process requires a 'sonification technique' (to convert the data into sound) as well as an 'interface & interaction' (to allow users to control the software and communicate with it). Since the sound is the core output of the application and very important to be interpreted and understood correctly, it is essential for the sound output to be effective. This can be done by making sure that the sonification technique is effective (that it is inherently capable of portraying the data to the user), because if it is not, no matter how easy the application is to use, users will not get any benefit from it. *Effectiveness* is thus an important attribute of usability for sonification applications.

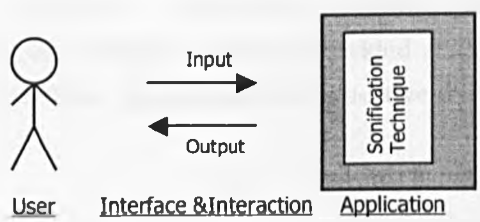


Figure 4-1: General overview of Human Computer Interaction for Sonification Applications

Therefore, instead of evaluating usability as an issue separate from effectiveness, we should consider both issues and focus on how the usability attributes will support the effectiveness of the technique(s) used. Questions such as, *'how can learnability increase the effectiveness of sonification applications?'*, *'how can memorability help the effectiveness of sonification* *applications?'* and so forth should be taken into consideration in the evaluations.

Context of Use

Bevan (1995) mentioned that usability is not only determined by the product but also by the context in which it is used; particular users, tasks and the environment. The previous research in auditory display by Alty et al. (1998) also found that context played an important role in assisting sound interpretation and understanding.

In this research, the contexts of use to be considered in the proposed extended usability inspection are inspired by Bevan et al., (1994). The contexts include users, interface and interaction, equipment and environment (explained later in detail in Chapter 6).

Usability for Sonification Applications

Therefore, for this research, the definition of usability for sonification applications is: *the capability of sonification applications to enable specified users to achieve specified goals using effectively the capability of the human auditory system in the process of perceiving the sound of sonified data (input/output) into a useful mental representation; with efficiency, learnability, memorability, low errors and satisfaction in a specified context of use.*

As illustrated in , this research treats effectiveness and usability (includes learnability, efficiency, memorability, errors, satisfaction) as two independent issues but that need to be tackled at the same time to make sure the sonification applications are effective with a high degree of usability.

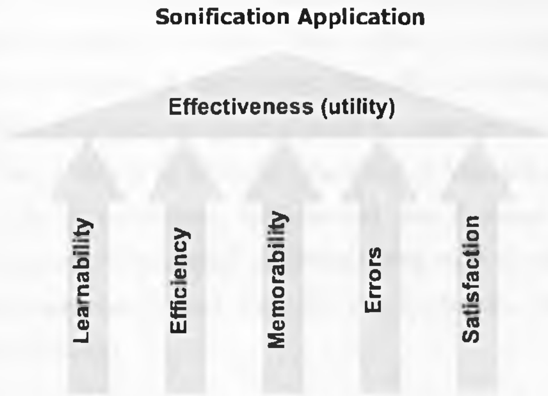


Figure 4-2: Usability for Sonification Applications

4.3 Overview of Usability Inspection for Sonification Applications

As explained in Chapter 2, most evaluations for sonification applications are based on *user testing*. The quantitative evaluation can be generalized into several steps in a flowchart as shown in Figure 4-3.

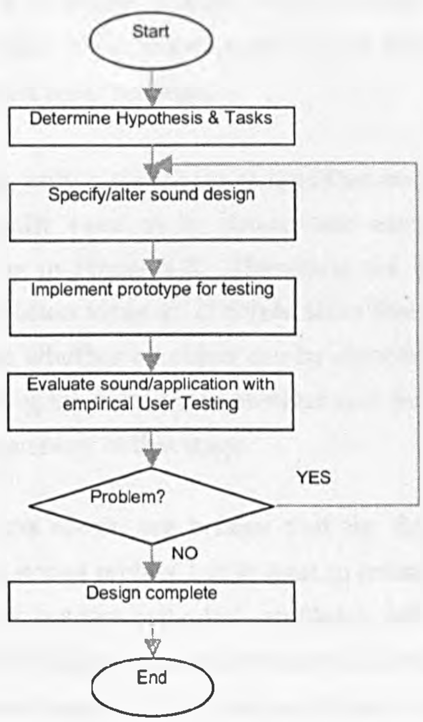


Figure 4-3: Flow Chart for User Testing

Before user testing, designers or developers need to determine at least one hypothesis. The hypothesis contains independent and dependent variables for statistical analysis. Several user tasks are required in order to investigate the impact of dependent variables on independent variables. There can also be a series of general tasks to compare the application under study with an existing application (*examples of tasks are given in Chapter 2*). For each task (*in sonification applications*), the designer might require to specify its sounds. The design of sounds and tasks are determined by the hypothesis statement(s) and variable manipulations (*both dependent and independent variables*).

Upon agreement of tasks and sounds, prototypes are developed such as sound samples, interfaces and interactions. Depending on the complexity of testing, special equipment might also be required, such as a 'haptic glove' device if it involves sounds interaction in virtual reality. When everything is ready, the empirical user tests will be carried out, and the results will then be analyzed and validated by using appropriate statistical mathematics such as ANOVA and the T-Test.

Even though the best technique for usability evaluation involves testing the developed application on end-users, the implementation to get to this stage is slow and expensive. Substantial time and resources are required to design and run controlled experiments especially if they involve special equipment and dedicated room settings.

Due to time constraints, only a certain set of specified tasks can be tested at one time. These normally need to be determined earlier in 'Determine Hypothesis & Tasks' as in Figure 4-3. Therefore, the tasks to be tested could be limited. They often focus on concrete tasks that can be measured and quantified, such as whether an object can be detected, or the speed of response when searching for data. More abstract and perceptual tasks are harder to deduce and quantify at this stage.

Because of the problems above, we believe that the field of sonification requires an alternative, not to replace but at least to enhance the evaluation techniques in order to predict potential problems *before* the expensive development phase. Referring to several evaluation methods in Chapter 2, ***usability inspection*** can be such an alternative for evaluating sonification applications because it can be done towards the start of the process of development, and without involving end users. This method tends to produce qualitative results and requires fewer participants than controlled experiments

Most of the existing usability inspection techniques focus on the inspection of Graphical User Interface (GUI) designs. Some methods are too open, general and rather dependent on the expertise of the inspectors. The technique that provides inspection guidelines such as Heuristic Evaluation is also suitable only for concrete tasks and graphical interfaces design. Cognitive Walkthrough also requires precise steps and task descriptions as inspection materials.

As mentioned in section 4.2 above, the main elements that differentiate between sonification applications and other applications are the *sonification technique* and the *sound output*. These differences prevent existing inspection techniques from being suitable. But it is believed that the same existing inspection *concept* will be very useful for detecting potential problems in sonification applications. This can be done by enhancing the inspection methods to suit the characteristics and differences of auditory displays. This is what this thesis is all about; *extending usability inspection for sonification applications.*

Based on the definition of usability for sonification applications (*Section 4.2*), we can summarize that the purpose of usability inspection for sonification applications is as follows:

> *To understand and investigate how usability factors (efficiency, learnability, memorability, satisfactory and error handling) will support the effectiveness of sonification applications in manipulating the human auditory system's capability in the process of perceiving and interpreting the sound (sonified data) and its structure into useful mental representations or information.*

Therefore, the application is said to be effective if the user's intended tasks can be accomplished with high accuracy and completeness. This happens if the users can gain a useful mental representation from the sound, which can be achieved if the intended structure of the data and the perceptual structure of the sound coincide. Thus, focus should be given to the data and to human perception as in the questions below:

1. **How is the data or information transformed into sound representation?** (Q3)

2. **How does the application help the user to perceive and interpret the sound as a useful mental representation of the original data?** (Q4)

To answer the above questions, a Human Computer Interaction (HCI) Model for Sonification is created to understand the interaction process between the user and the application. The overview of this model will be discussed in the next section.

4.4 Overview of Human Computer Interaction (HCI) Model for Sonification Applications

This section proposes a new Human Computer Interaction Model for Sonification Applications. This model is the basis of the novel extended usability inspection – Task Interpretation Walkthrough (TIW) (will be explained in detail in Chapter 6). It is used to explain and understand sonification applications (at the design stage) and how the users might interpret the sound outputs.

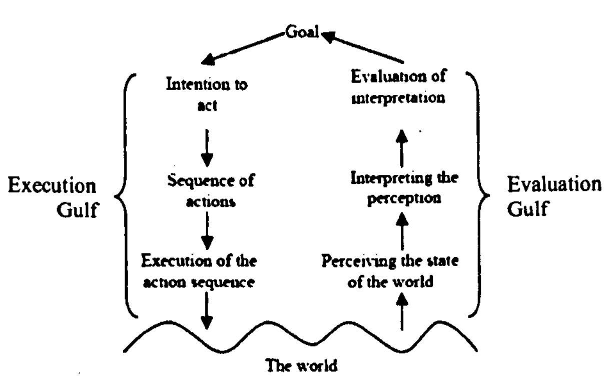


Figure 4-4: Seven Stages Norman's Model of Human Computer Interaction (Norman, 1988)

The model was inspired by Norman's HCI Model as shown in Figure 4-4. Generally, Norman's Model consists of two gulfs namely the 'execution' and the 'evaluation' gulf (Norman, 1988). The *execution* gulf is the gap between the effects that the user intends to achieve and the actions provided by the system. In *evaluation* gulf, the user perceives any feedback or output from the application (*the world*). It is then followed by the interpretation of the perception and finally evaluation of the interpretation to see whether it is what was intended. In general, this model focuses on the 'users' requirements' (*what the users wanted to do?*) and the 'users' perception and interpretation' (*what the users will interpret from the output?*)

'The world' is considered as the application that provides the functionalities, solutions as well as feedback to the user. In this thesis, the 'world' represents the sonification application under examination. In the HCI Model for Sonification, the 'world' part will be used to describe the solutions provided by sonification application (*what the application can give?*).

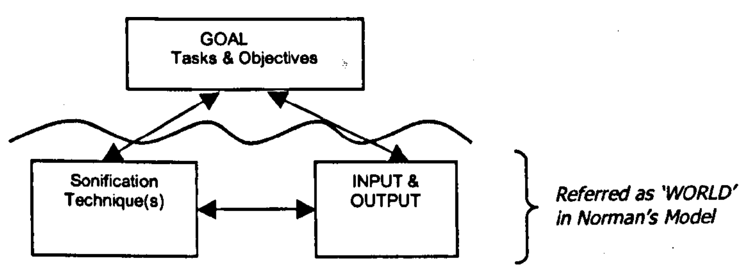


Figure 4-5: Sonification Application block diagram

To describe the HCI model for sonification applications, Norman's Model (Figure 4-4) is adapted together with the 'sonification application block diagram' (Figure 4-5) to explain the interaction between the users and the application. By comparing the two diagrams, an overview of sonification applications can be described from the following three questions.

1. **What do users want to do? (User requirements)** (Q5)
2. **What will users interpret from the output? (User perception and interpretation)** (Q6)
3. **What solution does the application offer?** (Q7)

These three questions are used to model the interaction between a user and a sonification application as shown in . The model shows how both Norman's model of HCI and the three elements of sonification applications are blended together.

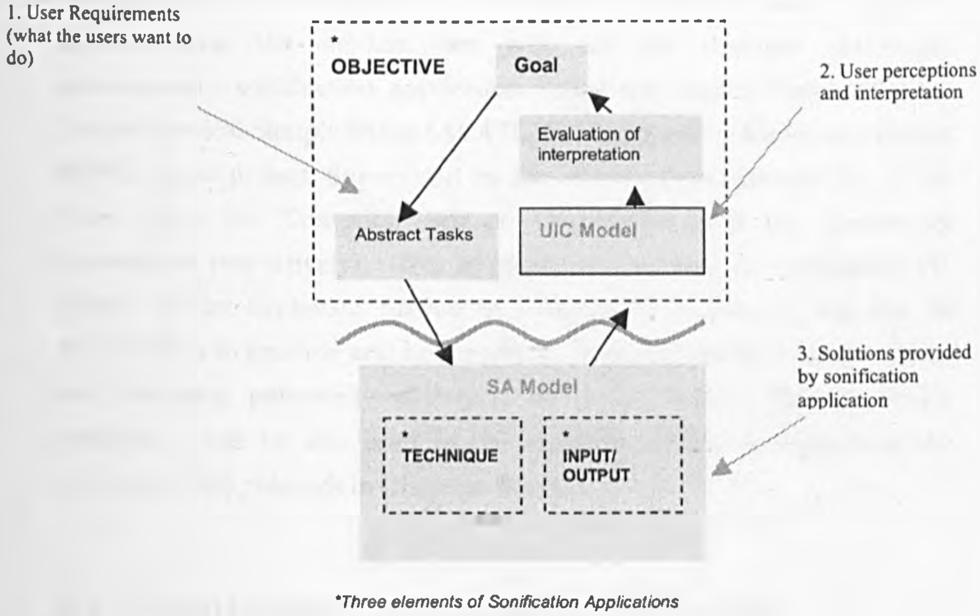


Figure 4-6: HCI Model for Sonification Applications

In this thesis, it is assumed that the user requirements (as in question Q5) have already been gathered by the designer. Therefore, the focus will be given to the solution provided by the application (as in Q7) as well as the potential user's interpretation of the output (as in Q6).

Based on the block diagram, the sonification technique(s) and input/output are considered as the 'world' in Norman's HCI Model. The 'world' is the application that the user needs to interact with. The model for the application design ('world') is called the **Sonification Application (SA) model**, which will be explained in detail in the next section. This model is used to describe what solutions the application can give (to answer the Q3 and Q7).

The user's view of goals is divided into two as mentioned by Norman's Model as two gulfs. These include the user requirements and the user perception and interpretation. The possible user interpretation of the sound output is modelled as the **User Interpretation Construction (UIC) model** (which is used to answer questions Q4 and Q6).

In summary, the application solutions will be explained through the Sonification Application (SA) model. This model is used to describe the design of sonification applications. The users will then need to interpret the output of the application. The user interpretation of the output is modelled through the User Interpretation Construction (UIC) model. To help in understanding the models, we will use an example real-world contemporary sonification application called the Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS) (Designed by Genevieve Hines, ESPRC Grant project, Supervised by Dr. Alistair Edwards and Dr. Andy Hunt, from the Computer Science Department and the Electronics Department respectively). This application was used in experiment IV, which will be explained further in Chapter 7. In general, the aim of AVATCSS is to provide and help users in analysing cervical sample slides and detecting potentially abnormal or cancer cells. This particular application will be also used as the example to help in explaining the concepts of this research in Chapters 5 and 6.

4.5 Sonification Application (SA) model

The sonification application is described as the Sonification Application (SA) Model in the HCI Model for Sonification Application. The details of the SA Model are based on the following questions (*repeated from the previous section*):

(Q3) How is the data or information [Input] transformed [Technique] into sound representation [Output]?
(Q7) What solution does the application offer?"

Based on Q3 above, the 'data or information' are referred to as the 'input' of a sonification application and 'sound representation' as the 'output'. The 'transformation' of these input and output is referred as the 'sonification technique'.

It is proposed to answer these questions by looking at the existing illustrations and descriptions of auditory display designs. Figure 4-7a shows a schematic framework to illustrate the existing research on how to analyse an auditory display. Figure a) shows the schematic by Kramer (1994), which describes an auditory display as an 'Information generator' (where the data come from), 'Communication Medium' (where the data is changed into sound) and 'Information Receiver' (where the user will listen and try to interpret and understand the data). In Figure 4.7b, Daude et al. (2003) described their auditory interface design framework into three transformation processes namely – 'Data Transformation', 'Sonification Transformation' and 'Auditory Transformation' (this framework is explained in detail in Chapter 2).

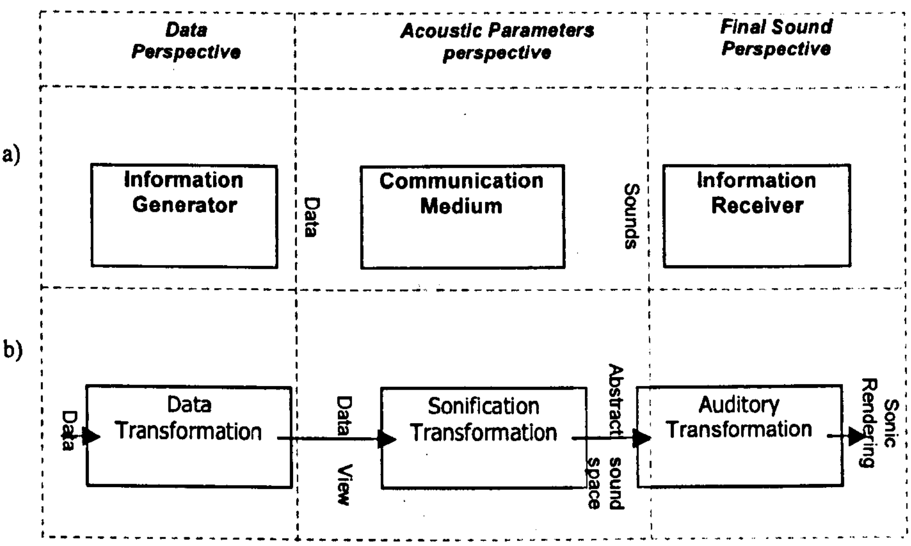


Figure 4-7: Sonification Applications: a) Schematic of an Auditory Display System (adapted from Kramer, 1994); b) Design process for Auditory Interfaces (adapted from Daude et al., 2003)

For our research, it is beneficial to classify the Kramer schematic and the Daude framework as three different general perspectives; **Data Perspective, Acoustic Parameters Perspective** and **Final Sound Perspective** as shown in Figure 4-7. From these three perspectives, we can see how previous research agrees that the data is transformed from its original form via an intermediate "ready to play" form and then into the final sound.

The Interaction between a Sonification Application and its users is also important as some applications allow interactivity in at least one of the perspectives i.e. interaction at the data perspective (Pauletto et al., 2004a; Herman, 2002; Janata, 2004)); interaction at the acoustics parameters perspective (Pauletto et al., 2004a); and interaction at the final sound representation perspective (Zhao et al.,2004).

Therefore, to answer the questions above, we propose to explain the sonification applications by at least *a transformation process, its input and output* as well as *the interaction between the sonification application and its users* in each perspective. The transformation processes explain how the data is transformed from its original form into the final sounds, and thus these transformation processes form the sonification technique of sonification applications.

As we have seen, the transformation processes consist of:

- Data Transformation,
- Acoustic Parameters Transformation, and
- Final Sound Transformation.

As mentioned in Chapter 1, the interaction between a sonification application and the user is explained by three aspects called users, system function and input/output manipulation. These aspects will be explained based on tasks, which are also referred by Paterno (1997) as users, application and interaction tasks. Therefore, in this research, each of the transformation processes will be explained:

- by the **application**, (transformation processes that only involve the tasks of the machine without any interruption from the user)
- by the **user** (performed by the user without interacting with the system), and
- by the **interaction** between the two (performed only by the user *with* the system).

Therefore, the above three types of tasks can be derived from each perspective to describe its transformation process as follows:

1. **Data** Perspective – a) *Data-User Tasks, b) Data-Application Tasks and c) Data-Interaction tasks.*

2. **Acoustic Parameters** Perspective – a) *Acoustic-Parameters-User Tasks, b) Acoustic-Parameters-Application Tasks, c) Acoustic-Parameters-Interaction Tasks.*
3. **Final Sound** Perspective – a) *Final-Sound-User Tasks, b) Final-Sound-Application Tasks, and c) Final-Sound-Interaction Tasks.*

The data and sound parameters involved in all the transformation processes are considered as the input and output (I/O). In general, the I/O includes the Raw Data, Processed Data, Acoustically-Prepared Data and Final Sound. Referring to Figure 4-8, we can see that Raw Data and Processed Data are the general input and output of any transformations involved in the data perspective. The output will become the input of the transformations involved in the Acoustic Parameters Perspective, which produces the output called Acoustically-Prepared Data. Ultimately, the Acoustically-Prepared Data is transformed into Final Sound in the Final Sound Perspective.

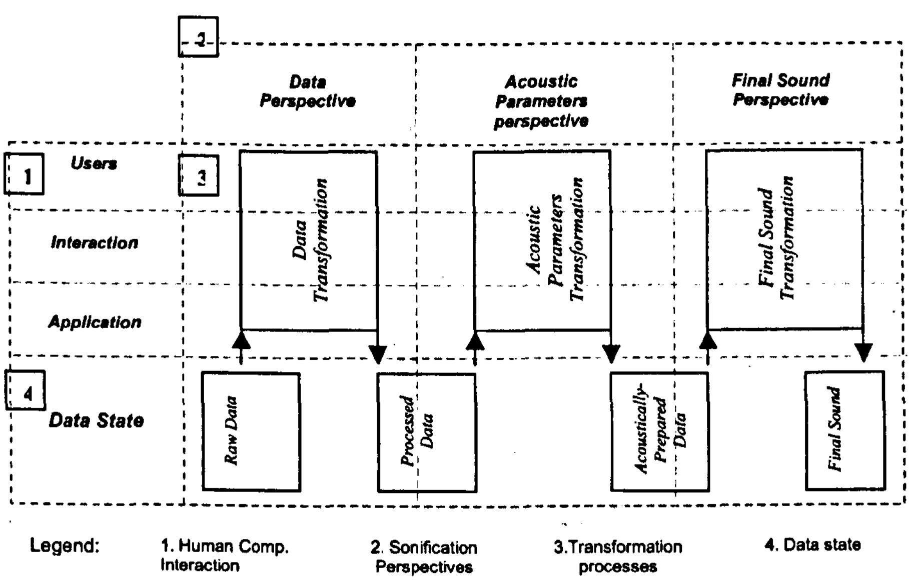


Figure 4-8 : The new way to describe Sonification Applications

In summary, to ensure that detailed attention is given (during usability inspections) to the transformation processes, the Input/Output and the user's interactivity, we have created the Sonification Application model based on the following main aspects as illustrated in Figure 4-8:

1. Human Computer Interaction – to explain the application from the points of view of the application, the user and the interaction between the two. These are referred to as Views in the Hypothesis statement in Chapter 1.
2. Sonification Perspectives– looking at sonification applications from three different perspectives - namely data, acoustic parameters and final sound perspectives. These are referred to as Perspectives in the Hypothesis statement in Chapter 1.
3. Transformation Processes – considering at least one transformation process in each perspective – i.e. data transformation, acoustic parameters transformation and final sound transformation.
4. Input/Output (Data State) – this includes the input and output of each transformation process that shows how the raw data is changed into the final sound. The different states of data include a raw data state, a processed data state, an acoustically-prepared data state and a final sound state.

All of the aspects above will be discussed in each perspective in the next section. Each perspective will be described based on the Human Computer Interaction, transformation processes and data state. As mentioned earlier, each perspective will be explained using an example of a specific sonification application design, which is used in Experiment IV – called the *Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS)*. The examples will be given in *italics.*

4.5.1 Data Perspective

The data transformation might not be required if the raw data is already suitable for direct conversion into acoustic parameters. However, this is not always the case for all sonification applications. The data will often need to be changed or transformed in order to make it more suitable for the sound conversion (sonification technique) e.g. data re-scaling and filtering.

Some reasons why the data transformation process is important are – to come out with data that are more meaningful; to ensure the data is always in the audible range; to get rid data redundancy; to change the data into a different form, and so forth.

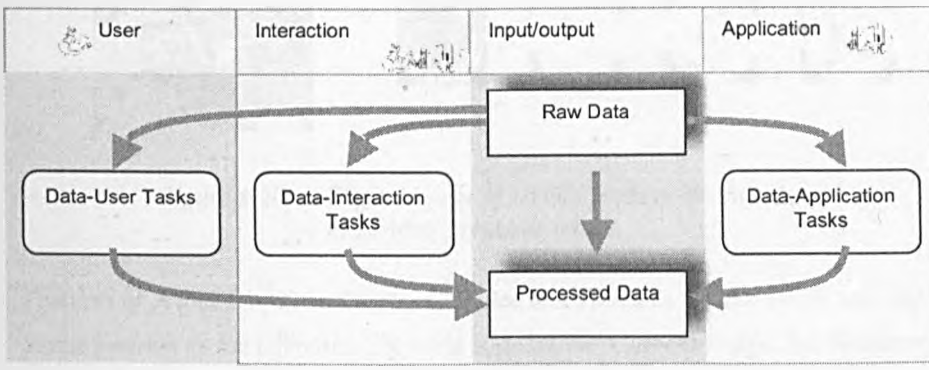


Figure 4-9: Data Perspective of the SA Model

Figure 4-9 above shows the Data Perspective of the SA Model, which is taken from Figure 4-8 by taking a vertical slice in the Data Perspective column and rotating it sideways. This, again, allows us (and thus the designer and usability inspectors) to focus for a while solely on the Final Sound Perspective. It consists of the following categories (will be expanded on below):

1. Input/output [*Raw Data* and *Processed Data*];
2. Data-Application Tasks;
3. Data-Interaction Tasks; and
4. Data-User Tasks

Below is the example of Data perspective for AVATCSS.

In AVATCSS, the input is a digital image of a microscope slide with a sample of human cell tissue that contains potentially normal and abnormal cells, as shown in Figure 4-10 a). The image is processed and segmented into 'regions of interest', each of which is a 16 grey level image, as shown in Figure 4-10 b). Through this process, most of the cell body and very small contaminating objects are discarded. The remaining regions are mostly nuclei (the main part of the cell) and sections of nuclei. The texture of the nuclei (which is a major indicator of whether a cell is normal or abnormal) is analysed by calculating two values called "A and B features"[1]*. The value of the A and B features of each cell are required for the next transformation, and are considered as the output of the data transformation process.*

[1] *The technique is based on "Statistical geometric features – Extensions for cytological texture analysis", R Walker and P Jackway, ICPR '96. The mis-classification is reported around 7%. Therefore, in the experiment, it is assumed that the X and Y features will work well and the user does not need to know exactly how to calculate the X and Y features.*

Figure 4-10: a) Digital image of cervical slide as the input
b) 16-level greyscale image

Features of A and B of the cells are calculated and stored in a table which uses the pixels position as the reference. The table is called the 'Current Image A-B Features Lookup Table' which contains - cell position (coordinate) and A-B values as shown in Figure 4-11. Assume that the range of the x-axis is between 0 and X_{max} and the range of the y-axis is between 0 and Y_{max}. Each range is then divided into 10 bands to produce 100 squares. The range is dependent on the band number as well as the maximum value of features A and B. This table is called the 'Reference A-B Features Look-up Table', which contains the pre-processing of previously analysed normal and abnormal cells, which are also stored based on their A and B feature values. The number of normal and abnormal cells in the same square is used to represent the density of cells (density of normal cells and density of abnormal cells) that belong to that square.

Current Image A-B Features Lookup Table

Coordinate	*A Feature*	*B Feature*

Reference A-B Features Look-up Table

y \ x	*Band 1*	...	..	*Band 10*
Band 1		*Number of normal and abnormal cell*		
..				
..				
Band 10				

Figure 4-11: Look-up Tables in Pre-Processing

The two tables are linked to each other so that the information from the 'Reference A-B Features lookup Table' can be accessed through the coordinates of the cell being sonified.

Input and Output (Raw Data and Processed Data)

Figure 4-9 above shows the input of the transformation process is 'Raw Data'. For an application that requires data transformations, the 'Raw Data' will be converted into 'Processed Data' such as data attributes that could be more suitable for sound conversion.

Several existing examples of data and processed data are shown in Table 4-1. For instance, in CAITLIN (Vickers et al., 1996), the program source code is divided into constructs (IF, FOR etc) which contain several Points of Interest (POI) as its features. An example POI is the 'result of IF construct that could be either true or false'. This POI is considered as the valuable information (which is also the focus of the auralisation) for the application and needs to be detected by the users.

Table 4-1: Examples of Input/Output in Data Transformation

Examples of Input (Raw Data)	**Examples of output (Processed Data)**	**Reference**
- Turbo Pascal program source code	- List of constructs: IF-ELSE, FOR, WHILE etc. - Points of Interest (POI) of each construct: o Entry to construct o Evaluation of conditional expression o Execution of selected statement o Exit from construct	Vickers et al. (1996)
- Stream of data (Sensors from Helicopter)	- New stream of *scaled* data	Pauletto et al. (2004a)
- Multivariate Data set	- Data position - Features vector	Bovermann et al. (2005)
- Muscle movements	- Real time data from sensors on human body (in live musical performance)	Nagashima (2002), Humon et al. (1998)
- Bids and Asks from stock market data	- Sorted bids in descending order and sorted asks in ascending order	Janata et al. (2004)

In the example of AVATCSS, the input/output (Raw Data/ Processed Data) includes:

- *the 'digital image of cervical slide'*
- *the '16 grey levels of the image cell'*
- *list of cells with A and B feature values*

Data-Application Tasks

As mentioned earlier, data-application tasks are those performed by the system, without any interruption from the user, and which involves transformation of Raw Data (input) into Processed Data (output).

Some applications need to process the raw data to reveal several valuable features as the input for sound transformation. However, this can be quite difficult for data *exploration* applications, especially where we do not even know in advance what type of information that the user will be looking for. Some techniques might not require any data transformation but rather use directly the original unchanged data for a sonification transformation, such as in the 'direct conversion technique' for Audification.

As an example from previous research, Janata et al. (2004) introduced a sonification of stock market data, where the bids and asks *need to be sorted in descending and ascending order* respectively. In this example, the application task is 'sorting', which includes 'to sort in ascending order' and 'to sort in descending order'. The input (raw data) of this transformation is the 'bid and asks'; and the output (processed data) is the 'sorted bids and asks' in ascending and descending order.

In the example of AVATCSS, the application needs to do the following tasks:

- *to process the digital image of microscope slide by segmenting it into 'regions of interest', which is a 16 grey level image.*
- *this is then followed by calculating the A and B features of each cell image.*

In this example, the application tasks are ***'to segment'*** *the image and* ***'to calculate'*** *the A and B features of a cell image. These segmentation and calculation tasks are done by the application without any interruption from the user.*

In general, the purpose of this transformation is to prepare the data for sound conversion.

Data-Interaction Tasks

Some applications provide flexibility for users to manipulate the data. Allowing the user to interactively manipulate the data can make the application highly responsive to the user's needs. Interaction on this level might be able to help users to relate their pre-conceptions and pre-interpretation with the sounds that they hear. Users are normally unable to understand the process of how their raw data is turned into the output sound. Therefore, manipulating down to data level will at least help the users to know about the data being processed and to relate this to the final sound.

Below are several examples of applications or toolkits that provide data manipulation:

1. Pauletto et al. (2004) introduced a toolkit that allows the data scaling process to be done and determined by the user. For example, the toolkit allows a user to define a new minimum and maximum data value, which will ensure the sound to be within perceivable range.
2. Janata et al. (2004) introduced the sonification of stock market that allows users to change the threshold value of the data.

In the example of AVATCSS, the application could allow the user to do the following:

- *To adjust and manipulate the grey levels, which will change the value of the A and B features of a cell image;*
- *the user could change the number and range of band, which could change the number of cells in each square box.*

However, these functions have not yet been implemented in the design, which is in its earliest stages. These are just examples of the kinds of interaction that could be done at this stage, and which come to light by the very process of discussing these issues with the designer in a structured way. By doing these sorts of manipulations, the 'processed data' is changed, which will also change the sound output of the application.

Data-User Tasks

Data-User Tasks are those performed only by users *without* interacting with the system, and which are related only to data transformation. The focus of such tasks is normally to help explain the user's cognitive ability, which includes consideration of what the user is required to do, understand, be aware of, think, perceive, interpret etc. in order to accomplish a certain application goal. These sorts of tasks are important as they influence the user's understanding towards the 'overall output' (sound output).

Generally, it is better for the user to understand the characteristics of the data. It would be particularly helpful if the users are already aware of what to look for in the data, such as the ability 'to understand the absolute movement in data stock market' (Janata et al., 2004), to recognize weather events (Hermann et al., 2003a) or to detect structure problems in program code (Vickers et al., 1996). However, this might not always be true for all applications but at least an initial or general task is needed. In data exploration, for instance, it may not be obvious ahead of time what the user specifically needs to look for, but can state that the initial task concerns finding a pattern or something strange that attracts the attention for further investigation.

In the example of AVATCSS, the user needs to do the following:

- *In the first place, be aware that the purpose of the application is to detect potential cancer cells.*
- *Be aware that each cell will be categorised by its two features A and B.*
- *Understand that the density of cells being inspected is based on the number of normal and abnormal cells in the same square box.*

The A and B values of a cell are required only as a reference to find which band the cell is belonged to. The data to be sonified is not the A and B values of the cell but rather the band it belongs to and the number of pre-processed normal and abnormal cells that belong to the same bands. By understanding this, it could help the user to understand what the sounds are representing, which is actually the proportion of real normal and abnormal cells that have more or less the same values of A and B as the current cell being inspected. Otherwise, the user might misinterpret the sound output, for example that the sound relates to the colour of the cell itself.

4.5.2 Acoustic Parameters Perspective

In the Acoustic Parameters Perspective, the processed data from the previous transformation (Data Perspective) will be converted into acoustically-prepared data, which is ready to be played as sound. Some techniques require a specific conversion of the processed data into any specific acoustic parameters such as a specific pitch and timbre, which is in general called parameter mapping. However, for certain sonification techniques, the conversion is not always directly mapped to a specific acoustic parameter. For example, an audification technique does not specify any acoustic parameters, but rather converts the signal directly into a basic audio output.

The Acoustic Parameters Perspective will be also explained from the three different tasks (user, application and interaction). Figure 4-12 shows all these tasks and the input and output for the Acoustic Parameters Perspective of the SA Model. This figure is also taken from Figure 4-8, by taking a vertical slice in the Acoustic Parameters Perspective column and rotating it sideways. This, again, allows us (and thus the designer and usability inspectors) to focus for a while solely on the Acoustic Parameters Perspective.

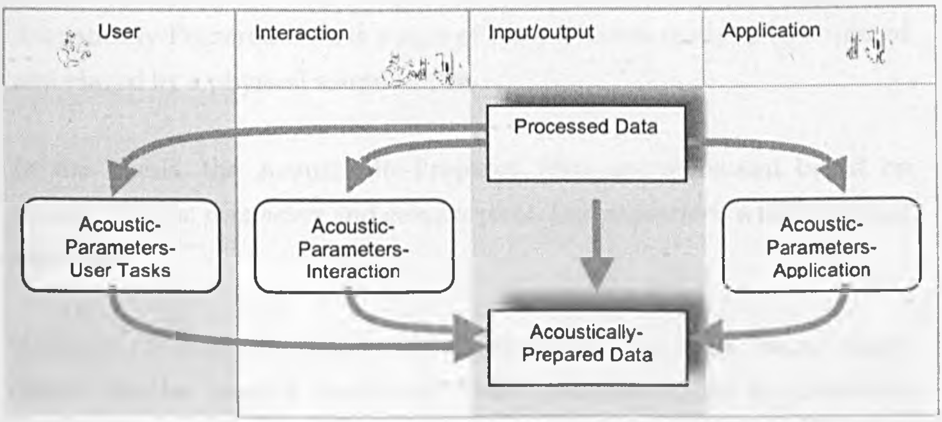


Figure 4-12: Acoustic Parameters Perspective of the SA Model

It consists of:

1. Input/Output [Processed Data and Acoustically-Prepared Data];
2. Acoustic-Parameters-Application Tasks
3. Acoustic-Parameters-Interaction Tasks
4. Acoustic-Parameters-User Tasks

Figure 4-13 shows an example of the Acoustic Parameters mapping process in AVATCSS. 10 ranges of pitch are used to represent the 10 bands of y or 'values of feature B' and 10 different rhythms are used to represent the other 10 bands of x or 'values of feature A'. These pitches and rhythms are used to represent the position of the cell within the square feature space.

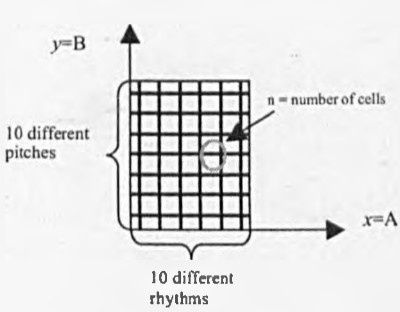


Figure 4-13: Acoustic Parameters Mapping

The numbers of sound for normal and abnormal cells are based on the number of normal and abnormal cells at a particular square. The normal and abnormal cells *are differentiated by mapping them with two different timbres (timbre 1 and timbre 2).*

Input and Output (Processed Data and Acoustically-Prepared Data)

Figure 4-12 shows that the input of acoustic parameters transformation is Processed Data and the output is Acoustically-Prepared Data. The Acoustically-Prepared Data is a state of data, which is ready to be rendered and played by a physical sound device.

In this thesis, the Acoustically-Prepared Data are explained based on *acoustic physical parameters* and *acoustic perceptual parameters*, which are now explained.

Acoustic physical parameters refer to the properties of the sound wave, which can be directly measured. These parameters can be explained technically and directly with standard measurements and scales, such as sound magnitude in decibels (dB) and duration in milliseconds (ms). These physical parameters are correlated with the subjective sensation of the sound. For example, the frequency is correlated with pitch and timbre; and the sound magnitude is perceived as loudness by humans. However, the relation is not straightforward. For the same value of sound magnitude for instance, a different listener might perceive a different level of loudness.

Therefore, in this thesis, this loudness is referred as acoustic *perception* parameters. Acoustic perception parameters refer to how humans perceive the sound. These parameters cannot be measured directly because of their subjective nature. Some examples of processed data and acoustically-prepared data from previous research are shown in Table 4-2.

Table 4-2: Examples of Input/Output in Acoustic Parameters Transformation

Examples of Input (Processed Data)	Examples of output (Acoustically-Prepared Data)	Reference
- Type of Indices (stock market data)	- Type of Instrument (timbre)	Janata et al. (2004)
- Array of Scaled data	- Array of Frequency values	Pauletto et al. (2004a)
- Five value categories	- Five String pitches	Zhao et al. (2004)
- Selected Aura (selected and surrounded point) 1) The distance between surrounded point and selected point. 2) The distance among surrounded point.	- Cloud Density, Grain Duration, Grain Oscillator Frequencies, Grain Amplitudes and Onset Delays	Bovermann (2004)
- data sets (for graph)	- Timbre, pitch, volume etc.	Walker et al. (2003)

In AVATCSS, the Acoustically-prepared data are:

- *a list of 10 pitches,*
- *a list of 10 rhythms,*
- *the timbres for both normal and abnormal cells*
- *the number of channels for normal and abnormal cells*

Acoustic-Parameters-Application Tasks

Acoustic-Parameters-Application Tasks are those performed by the system to transform the processed data into acoustically-prepared data without any interruption from the user. The transformation depends on the technique used as well as the output of the previously data transformation process (Processed Data). Parameter mapping for instance, will map the individual data into acoustic parameters such as pitch, volume, timbre and so forth. For instance, Hankinson et al. (1999) proposed a musical grammar in earcon design, where a set of grammatical rules are mapped into notes, chords, rhythms and pitch, which then form a larger phrase.

In our example of AVATCSS design, the cell's A and B values (feature A and B) are used to determine the pitch and rhythm of the band it belongs to. Examples of Acoustic Parameters-Application Tasks in this application include:

- *Mapping both normal and abnormal cells into two different timbres.*
- *Obtaining the pitch and rhythm for the band of A and B features.*
- *Reproducing the same timbre based on the number of normal and abnormal cells in the same square.*

Generally, the acoustic parameters tasks in this application are mostly to 'map' the Processed-Data [type of cells (normal and abnormal); number of normal and abnormal cells in a particular square; 10 ranges of both x and y axis] into Acoustically-Prepared data [10 ranges of pitch; 10 different rhythms; timbre and number of channel].

Acoustics-Parameter-Interaction Tasks

Acoustics-Parameter-Interaction Tasks describe tasks performed by users *interactively* with the system, which are related to acoustic parameters transformation. This type of task allows users to manipulate directly how the scaled and processed data is converted into a 'ready to be played' form. Not so many applications provide flexibility to the users at this level. This is a pity, because such interaction between users and the application might help the users to intuitively understand how their data is being transformed into sound. As different physical acoustics parameters produce different perceptions, the ability to change the acoustics parameter mapping for instance might help the user to understand the same data from different 'views' under their control. This would make the system more open, flexible and not too rigid; and this is good especially for data exploration applications.

An example of previous research that gives flexibility at this level is Walker et al. (2003) in their auditory graphs toolkit *Sandbox* which allows users to independently map several data sets into any acoustic parameters that are available for selection.

In our example AVATCSS design, there is no interaction given by designer at this level of task. However, as examples, interactions could be added into the design, to:

- *Provide the user with an option to customise the timbre of both normal and abnormal cells.*
- *Allow the user to customise the different of pitches for each band.*

This is because to compare the pitch and rhythm of two cells (two sounds) being inspected with different timbre might be a bit difficult. Therefore, by allowing the normal and abnormal cells to be mapped with the same timbre, and playing them on different speakers (e.g. left and right), this could help users to differentiate and compare the pitch of the sounds better.

Acoustics-Parameters-Users Tasks

Acoustic-Parameters-Users Tasks are those performed only by users (without any interaction with the system), which are related to acoustic parameters. Such tasks focus on user perception of acoustic parameters, which can vary for different users, for instance pitch, loudness and timbre.

These parameters seem to be most easily observed in a parameter-mapping sonification technique because the specific parameters are predetermined by the designer. However, this is quite different in Audification where the data are directly converted into amplitude values at a certain sampling rate. Therefore, it is important for designers to consider these differences in their sound design.

In the example of AVATCSS, it is important for the user to be able:

- *to differentiate clearly the two timbres;*
- *to differentiate the 10 pitches and rhythms.*
- *Identify roughly the position of pitch from the 10 ranges (band) of pitch*
- *Identify roughly the position of rhythm from the 10 different (band) rhythms*

The user's act of differentiating the acoustic parameters is what we mean by the Acoustic-Parameters-User tasks in AVATCSS. This is important as each timbre represents different cell: either normal or abnormal. Failure to differentiate the timbre will cause the user to fail to differentiate between normal and abnormal cells, and thus fail in the main aim of the application.

4.5.3 Final Sound Perspective

This perspective focuses on how to play and manipulate the Acoustically-prepared data (that were produced in the Acoustic Parameters perspective) as the final sound. It also concerns how to put them together as a sound that might carry useful information. This is the point where the user should be able to relate the sound with the data, acoustic attributes and the purposes of the sonification application.

Figure 4-14 shows the transformation process of Acoustically-Prepared data into the final sound representation. This Figure is also taken from Figure 4-8, by taking a vertical slice in the Final Sound Perspective column, and rotating it sideways. This, again, allows us (and thus the designer and usability inspectors) to focus for a while solely on the Final Sound Perspective.

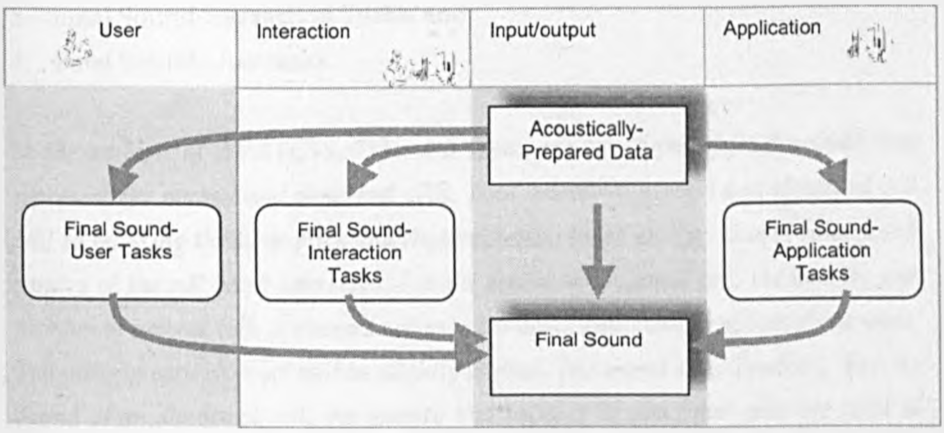


Figure 4-14: Final Sound Perspective of the SA Model

The final sound is the audio that is played on a physical device such as headphones or speakers. The Final Sound Perspective allows us to consider the manipulation of Acoustically-Prepared data according to the meaning or objectives that the application needs to represent or achieve. However, not all applications produce the final sound representation with explicit meaning. For instance, in data exploration applications, the sounds are dependent on the data being explored. Barrass (1997) mentioned that researchers in both sonification and visualization have recognized two different styles of information processing tasks. The first is the exploration of data sets for interesting and unknown features such as data mining and data exploration. The second is the presentation of known features or information such as in earcons and auditory icons. It is important and useful to know the style of sound representation (either for exploration or only for presentation) especially to relate it with what it should be signifying.

The Final Sound Transformation describes how acoustically-prepared data are transformed to final sounds from the same three different point of views as before; application, user, and the interaction between the two.

Figure 4-14 shows the Final Sound Perspective of the SA Model. It consists of:

1. Input/Output [Acoustically-Prepared Data and Final Sound]
2. Final Sound- Application Tasks
3. Final Sound-Interaction Tasks; and
4. Final Sound-User tasks.

In the example of AVATCSS, Table 4-3 shows the two types of final sounds that represent the normal and abnormal cells. Both sounds of normal and abnormal cell will be referring the same pitch and rhythm, which based on the value of band x and band y of the cell being inspected. For the sound of a normal cell, the density and number of normal cells are used to change the delay and number of sound channels. The pitch of each channel will be slightly shifted. The sound uses Timbre 1. For the sound of an abnormal cell, the density and number of abnormal cells are used to change the delay and number of sound channels. The sound uses timbre 2 and the pitch of each channel will also be slightly shifted.

Table 4-3: Two type of sounds output

Acoustic Parameters	Sounds of Normal Cells	Sounds of Abnormal Cells
Pitch	*Band x*	*Band x*
Rhythm	*Band y*	*Band y*
Delay	*Density of Normal Cell*	*Density of Abnormal Cell*
Number of Channels	*Number of Normal Cell*	*Number of Abnormal Cell*
Timbre	*Timbre 1*	*Timbre 2*
Pitch shifting	*All sound channels*	*All sound channels*

By mixing these two sounds together, it should produce a sound with the same pitch and rhythm but with different timbre and thickness. Since these sounds are going to be played at the same time, the user should be able to detect the two different timbres and recognize which one is representing normal and which the abnormal cell. Besides differentiating timbre, the user should also be able to observe the thickness of the sound and recognize which timbre is thicker. The thickness of sound (chorus effect) is influenced by the delay value (density of cells) and number of channels. The higher the density and number of channels, the thicker the sound will be produced. The thickness of the sound should be perceived by the user as the number (density) of normal and abnormal cells which are having more or less the same value of A and B features of the cell being inspected or sonified. Depending on which timbre is thicker, the user should be able to get a clue of whether or not the cell being sonified is likely to be an abnormal or a *normal cell. The users could also compare the sounds among different cells through their pitch and rhythms. For instance, if the trend of 'abnormal cell' is to produce a higher pitch, therefore, even though the chorus effect produces a 'thinner' sound but if the pitch was higher or more or less the same as the sound trend of 'abnormal cell', it is more likely to be a potential abnormal cell.*

Input and Output (Acoustically-Prepared Data and Final Sound)

The sound representation is the final sound that will be heard by users as the output of the application. It is dependent on how the application manipulates the Acoustically-Prepared Data through either the Final-Sound-Application Tasks (produced by the application without any interruption from the user) or the Final Sound-Interaction Tasks (sound produced due to user actions). Below are some examples of final sound representations from previous research:-

1. Brewster (1992) introduced Hierarchical Earcons, where three different parameters are used to carry three pieces of information that are related hierarchically to each other. The three parameters give information that is more meaningful when they are played together in sequence.
2. Pauletto et al. (2004a) introduced a Sonification ToolKit that allows users to experiment with a large variety of Parameter Mapping sonification techniques. This tool combines more than one acoustic parameter together as one sound representation.
3. The Musical Sonification Environment (MUSE) combines six different parameters, namely timbre, rhythm, volume, pitch, tempo and harmony to produce a musical sound representation (Lodha et al., 1997).

Generally, the final sound can be manipulated in various ways depending on the output of the Acoustic-Parameters transformations (Acoustically-Prepared Data) as well as the goals and objectives of the application itself. It can be manipulated in terms of: the *number of acoustic parameters, number of sound channels, sound coordination and sound scope (all explained in the following sections)*. However, these are also dependent on the technique used; for instance, the number of acoustic parameters is easier to manipulate in a Parameter Mapping technique than in an Audification technique.

Number of acoustic parameters

There are many acoustic parameters, which can be used to portray sound in sonification. Either they are a) 'purposely selected' as in Parameter Mapping sonification techniques; b) natural representations (e.g. an auditory icon based on everyday sounds); or c) 'existing without the designer's intention' (e.g. dependent on data characteristics as in Audification or the physical model used as in Model-based sonification techniques). For example, a Hierarchical earcon combines *up to five different acoustics parameters* (Brewster, 1992); and the sonification toolkit by Pauletto et al.(2004a) can be used *to sonify the same set of data with different parameters* to produce a single sound output. The number of acoustic parameters shows the complexity of the sound especially if each parameter carries different information.

Number of Channels

This refers to the number of data streams that can be represented independently (at different times) or that can be combined and displayed together (at the same time) as a single sound output. Some applications have more then one group or data stream to be sonified and played at the same time. For example Pauletto et al.'s toolkit (2004a) provides *multi channel sonification*. By playing two different streams of data in stereo, the relationship between the two sets of data can often be revealed.

Coordination

Sound coordination is how the acoustic parameters (as well as any multiple data channels) are organized and coordinated to produce the sound output. The phrase 'sound coordination' is adopted from Daude et al. (2003), which is part of their model of sonification transformation. However, in our SA Model, the coordination occurs at the final sound transformation, and describes how different sounds are 'linked together'. The combinations are shown below:

Table 4-4: Sound Coordination, adopted from Daude et al. (2003)

Coordination	Descriptions
Same place- simultaneous	Sounds mix and play as one
Same place – alternate	Sounds mix and play one after another
Separate places –Simultaneous	Has position(s); and play(s) at the same time
Separate places – alternate	Has position(s); and play(s) one after another

Sound Scope

The Scope of the sound representation is based on the 'duration of sound' and the concept of 'data level'. The data level ranges from a single point of data (Point), some part of data (Regional) or an overview of all data (Global). For instance, a bit of data (point) can be represented using a single pitch. The information from a section of data (regional) can be then represented by a few seconds of sound containing a stream of varying pitch values. This is summarized in Table 4-5.

Table 4-5: Scope of Sounds

Level	Descriptions
Point	Sound that can be retrieved through a single element of data or information. *E.g. clicking on an element that will produce sound.*
Regional	Sounds that can be retrieved through an area of data. It is normally produced due to user action(s). *E.g. Scratching an area in a Model-Based Sonification application.*
Global	Sound that can be retrieved through the whole set of data. It can be a continuous sound or an overall sound based on the user's selection. *E.g. playing all sound at the same time*

Point means that the sound comes from a single piece of data or information. For example, one-element earcons use only a single pitch to represent one bit of information (Blattner et al., 1989). *Regional* means that the sound comes from a portion of the data (made up of more than one Point). For instance, in CAITLIN (Vickers et al, 1996) a regional scope might involve listening to a sound from a loop structure of programming source code; and in Pauletto's ToolKit (2004a) regional scope might entail listening repeatedly to part of the data in real-time with loop navigation. *Global* means that the sound of the entire data is played continuously or for overall overview. For example, the NeMoS (Malandrino et al.,2003) is a real-time behaviour monitoring system using sound for distributed client/server network system. Global scope is also found when listening to an overall graph in Sandbox (Walker et al. (2003)).

In the example of AVATCSS, the final sounds are:

- *the sound of normal cells with chorus effect, which consists of pitch, rhythm, delay, number of channels and timbre (as in Table 4-3).*
- *The sound of abnormal cells with chorus effect, which also consists of pitch, rhythm, delay, number of channels and timbre.*

- *These acoustics parameters are combined together to create two different final sounds but played together at the same time.*

Final-Sound-Application Tasks

Final-Sound-Application Tasks are those performed entirely by the system to transform the Acoustically-Prepared data into the output sound. The same acoustically prepared data can be used to reproduce various different final sound representations. The type and number of acoustic parameters can be easily manipulated as the final sound in a parameter mapping technique but difficult in audification and model-based sonification. Designers could also choose spatial sound using stereo or more speakers. For existing example, Miele (2003) introduced SKDtools (Smith-Kettlewell Display Tools) for MATLAB, which uses a two-speaker representation method; left and right.

In the example of AVATCSS, the Final Sound-Application tasks include:

- *To obtain and combine all pitch, rhythm and timbre*
- *To set delay values and create chorus effect*
- *To mix both sound of normal and abnormal cells*

Final-Sound-Interaction Tasks

Final-Sound-Interaction Tasks involve the tasks performed by user *interactively* with the system, which relate to final sound reproduction. Once the data is already in the form of *Acoustically-Prepared Data,* it can be manipulated and listened to by interacting with it. The simplest interaction is by pressing the 'play' button, and the computer will then process it and the sound can be heard.

The final 'tempo' (speed of playback') can be either constant or variable. Variable tempi usually occur due to the user's interaction with the application. Interactivity at this level gives a user the flexibility to hear varieties of final sound representation and this is useful especially for data exploration applications. For example, Pauletto et al. (2004) provided three types of interaction in their sonification toolkit - namely *real-time, real-time with loop* and *non-real-time navigation.* In real-time navigation, the user needs to click on and drag the mouse pointer within an on-screen interaction area. The coordinates of the mouse drive a pointer to the sound/data array (*acoustically prepared data*) and the sounds are only heard when the mouse is moving. Different speeds of mouse movement produce different speeds of sound representation. **This creates *different* and *user dependent* sound representations from the same data and acoustic parameters.** Interaction is also important in giving a natural (real) feeling of a virtual 3D environment. For example, Pair et al. (1997) introduced COOLVR where the sound object is attached to a graphic object (sound will be produced through interaction) in a virtual environment to give the user the illusion that the object seen is real.

In the example of AVATCSS, the user will be allowed to:

- *Change the overall volume of final sound.*
- *Change the complexity of final sound by enabling and disabling the pitch, rhythm and timbre.*

Final-Sound-User Tasks

Final-Sound-User Tasks are those performed only by users without interacting with the system, which are related to final sound. At this stage, the user tries to interpret the sounds that are produced by the application. The interpretation is also influenced by the context of where the application will be used. As in the software engineering field, the contexts of use that will influence the usability of a product include the user (knowledge, experience etc); task (to be completed); equipment (input, output or any necessary devices); and environment (situation in which the application will be used). These contexts of use will be explained further in Chapter 6. The contexts of use are required for the user to relate any sound event with its possible meaning.

However, some sonification applications have no clear objectives and tasks such as in data exploration. The tasks are complex, and cannot be clearly defined in advance, because the patterns being sought in the data are not yet specified. For instance, a data-interpretation task and a data-understanding task are highly interactive and normally have no clear sequence of actions to do the tasks. Sometimes the decision on what to do next is also dependent on the previous feedback e.g. outputs from user's interaction. In this case, the interpretation is dependent on the domain of the application as well as user's knowledge. The user will perceive the sound and try to relate it to their previous knowledge in that domain. For instance, a fluctuation of pitch could be due to a fluctuation of market prices if the application is for the stock market data domain; or it could be due to an irregularity of muscle tension if the application is for the physiotherapy data domain. If the users are not able to relate the sound event with their previous knowledge, it remains as something that interests them for further analysis. In general, this ability to detect an area of interest in the data (*e.g. the ability to detect pitch fluctuations*) could also become the task and objective of the sonification applications. Therefore, how the user perceives the sound representation in general is very important in this kind of application. In terms of design, regardless of the data domain being explored, the general objectives of acoustics parameters transformation and final sound transformation must be taken into consideration.

In AVATCSS, some examples of Final-Sound-User Tasks include:

- *To understand that 'thickness' of sounds represents the density of the respective cells.*
- *The ability to differentiate the timbres either for normal or abnormal cells*
- *To identify which timbre produces the thicker sound*
- *Decide whether the inspected cell is potentially a cancer cell.*

4.5.4 Summary of the SA Model

In summary, the previous sections discussed the three different perspectives, three different views and different stages of data transformation into sound. The combination of each perspective is shown in Figure 4-15, which we call the Sonification Application Model. This diagram is re-worked from Figure 4-8.

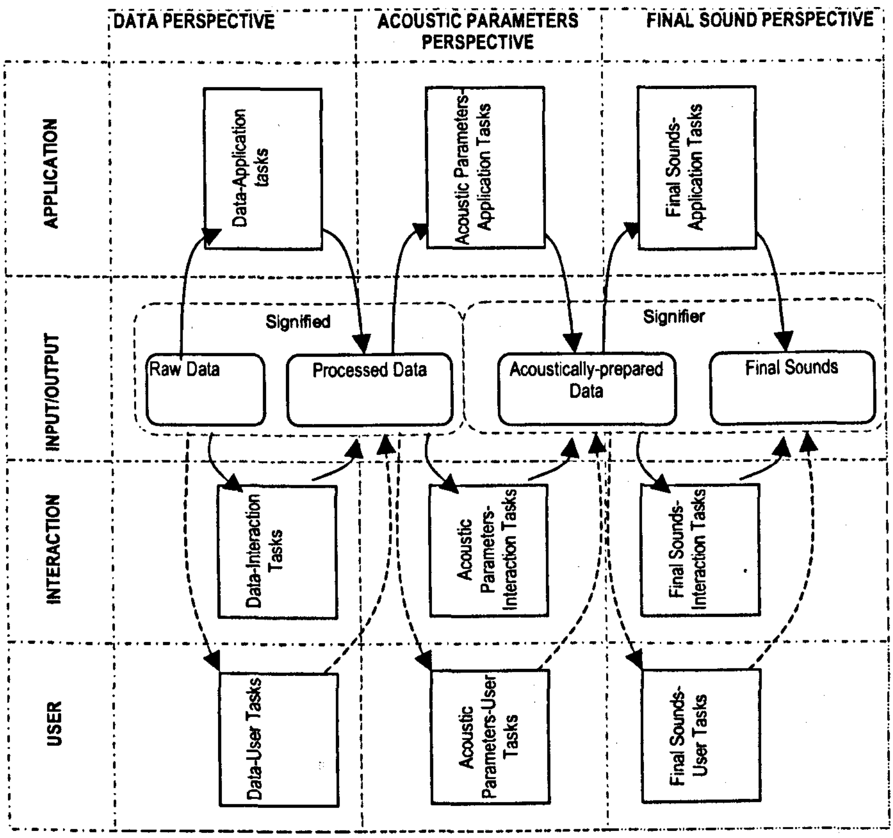


Figure 4-15: Overview of Sonification Applications (SA) Model

In each perspective, there is at least one transformation task. The whole system is described through the *views* of –

- **User** (what the user needs to do without interacting with the application);
- **Interaction** (what the user needs to work with the application); and finally
- **Application** (what the application does without user involvement).

The input and output of the transformation processes in each perspective include data or information which is:

- input to the system (raw data);
- suitable for conversion into sound (processed data);
- in the form of acoustic parameters and ready to be played (acoustically prepared data); and finally,
- the sound that will be heard by users (final sound).

Based on the model of sign (Chandler, 1997 adopted from Saussure, 1974), both the Raw Data and the Processed Data are considered as the Signified (the data being represented); and the Acoustically-Prepared Data and Final Sound are considered as the Signifier (the sound that is used for the representation).

Each task and data state of this model will be represented in a diagrammatic form called the Task-Data State Diagram, which will be described in detail in Chapter 5.

Now that we have introduced the Sonification Application (SA) model, we now explain how the user perceives and interprets the sound output of a sonification application, by introducing the User Interpretation Construction (UIC) model. Together, the SA and UIC models are used in the Task-Interpretation Walkthrough (TIW), which is the novel inspection technique developed for this thesis (and which is covered in Chapter 6).

4.6 User Interpretation Construction (UIC) model

The user's interpretation of sound is described as the User Interpretation Construction (UIC) model in the HCI Model for Sonification Application. The details of the UIC model are based on the following questions (*repeated from the previous section*):

> *(Q4) How does the application help the user to perceive and interpret the sound as a useful mental representation of the original data (Goals/Objectives)?*

Based on Q4 above, the general goal and objective of a sonification application is to help a user to perceive its output as useful information. To answer this question, we have created a model to describe how the user tries to perceive and interpret the output of the sonification application, which is sound, into something that is useful (and thus is what the application is intended for). It is inspired by the definition of Information processing (Coren et al., 1999) as shown below.

"Information processing concerns the interaction of the various levels of mental processing from the sensory through to the cognitive mechanism that finally leads to identification and interpretation of stimuli. This includes a registration or sensory phase, an interpretation or perceptual phase and a memoric or cognitive phase" (Coren et al.,1999).

This definition attempts to integrate **sensation, perception** and **cognition** within a common framework. The interpretation process occurs starting from the first contact between user's sense organs with the sound. Initially, this is simply a *sensation* (a term, which deals with the more basic aspects of experience such as 'how loud does the sound appear to be?' (Coren et al., 1999)). It is then followed by *perceptions,* which is how we form a conscious representation of the outside environment and in the accuracy of that representation (Coren et al., 1999). Examples of questions are 'How far away is it?' and 'How large is it?' These kind of questions have a variable that can be perceived differently by different people, such as 'loudness'. The same amplitude level of sound might be perceived as different loudness levels by different people. Finally, *cognition,* which is how we try to understand and make sense of the sound by relating it with previous knowledge, experience etc.

Through the concepts above, we created a model to explain the user's point of view about how they listen to the sound; focus on the sound that they are interested in; and finally try to learn and understand what the sound is representing. This model is called the User Interpretation Construction (UIC) model as shown in Figure 4-16.

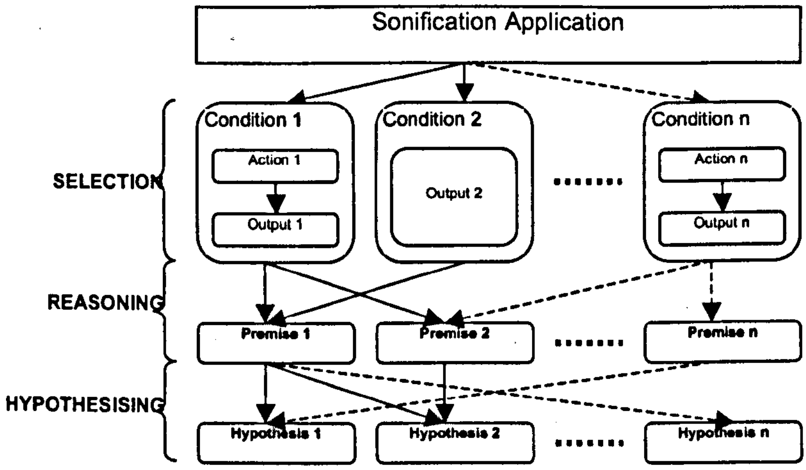


Figure 4-16: Overview of User Interpretation Construction model (UIC)

An overview of the UIC model is shown in Figure 4-16. The output from a sonification application needs to be interpreted by the user into useful information. From the figure above, the vertical axis of the model shows three stages involved in the process of constructing the interpretation: *selection, reasoning* and *hypothesising.* Each of these stages is described in detail in sections 4.6.1, 4.6.2 and 4.6.3 respectively.

People always tend to ignore sounds that they already familiar with or get use in their everyday life as well as low level such as the sound from an air-conditioner or from a computer fan. However, they could easily be alerted if these devices suddenly produce a different sound. In our model, we refer to this ability to focus or pay attention to a certain sound as **selection.** The different sound of air-conditioner is selected, as it is more attractive and different from its normal sound. This different sound we refer as **condition.** The conditions are the potentially useful outputs that might carry important information e,g,, the changes of pitch in the sound of air-conditioner that might indicate that it needs to be serviced. There could be more than one condition, which is illustrated horizontally as Condition 1, Condition 2 and so forth as shown in the figure above.

From the sound that they pick up and to pay attention, they could try to build up and deduce some knowledge or information that related to the sound. We refer this stage in UIC model as **reasoning,** for example, to deduce any possible information about the reason why the pitch in the sound of air-conditioner become higher than normal. We refer all the possible information as **premises.** The premises are statements to support the reason why the selected conditions are important (potentially important). They could be more than one premise, which is illustrated as Premis 1, Premis 2 and so forth as shown in the figure above. Finally, from all the list of premises, their experience and prior knowledge – they will then try to interpret and make an overall statement explaining about the selected sound. This statement we refer in our model as **hypothesis.** The hypothesis is a statement of the overall idea about how the relationship between premises and their conditions will work. The list of hypotheses is also illustrated horizontally as Hypothesis 1, Hypotehsis 2 and so forth in Figure 4-16.

The input sound of UIC model is the output sound from the Sonification Application (SA) model. As a reminder of SA model, the output sound is produced by transforming data through three transformation processes called Data, Acoustic Parameters and Final Sound transformations. Each transformation has process tasks that will produce output (referred to as the Data State in the SA model). Therefore, we propose the input for the Selection stage in the UIC model are the output of tasks involved in each transformation process, which is the states of data in each perspective of SA model. These states of data are used to create a list of conditions, premises and hypothesis as the output of the UIC model.

In general, this model can be used to list out the possibilities of a user's interpretation of the output of a sonification application. From the designer's point of view, this model can be used to describe and explain their view of what the user should be getting from and interpreting from the application. Therefore, this will help them to explain their expectations and rationale behind the design as described in the SA Model. Each stage of the UIC model is explained further below. As in the SA Model, the AVATCSS will be also used as an example.

4.6.1 Selection Stage

Some sonification applications are able to produce different sound representations, which can be changed through *'user's interactions'* (for example as in Model-Based sonification); or *'changes in acoustics parameters'* in parameter-mapping. Therefore, as mentioned previously, the output will be based on the state of data at each perspective – namely the data, the acoustic parameters and the final sound perspectives. These data states can be observed through the SA Model.Figure 4-17 shows the Selection Stage (taken from the Selection row of Figure 4-16) where users need to select the outputs to become a '*condition*'. Selection is a discriminating process where the user will choose the potential useful output. Condition is a mode or state of data at particular time. Among all the data states (outputs), probably only a few of them are important and attracted the user's attention, and so these will likely become the final conditions chosen by the users. Therefore the Selection stage is like a filter, where the user attends only to important output that potentially contains the required and useful information.

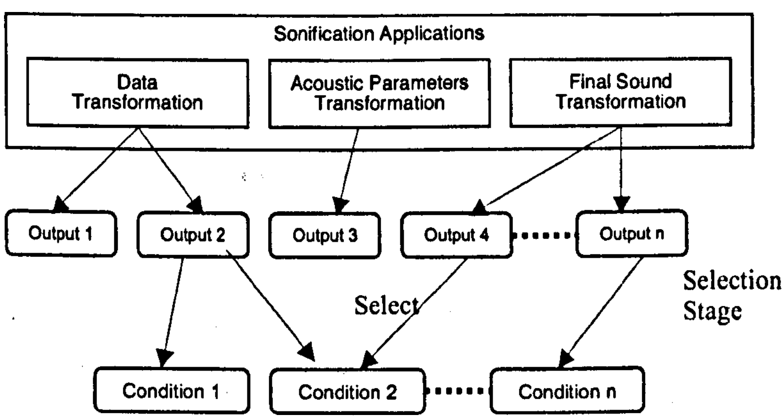


Figure 4-17: Selection Stage

As an example,

Table 4-6 shows several conditions that could be important in the process of interpreting the output of AVATCSS. The conditions are divided according to the perspective of the data states. The examples are the list of conditions that the designer wanted the user to detect, select, attend to, focus on etc. In the Data Perspective, the user might need to know that each cell will be using the two features A and B as the reference; the x axis and y axis are divided into 10 bands; etc. In the Acoustic Parameters Perspective, it is important for the user to be aware of the different timbres (1 and 2); the 10 ranges of pitch; the 10 different rhythm beats; etc. And in the Final Sound, it is important for the user to be aware of several important conditions that potentially carry important information, *knowledge, meanings etc. such as the number of different sounds, the type of timbre, the delay, level of pitch and so forth.*

Table 4-6: Example of Lists of Conditions

Data Perspective	Acoustic Parameters Perspective	Final Sound Perspective
• Feature A • Feature B • x and y axis • x axis for feature A • y axis for feature B • 10 bands of x axis • 10 bands of y axis • number of normal cells in each square • number of abnormal cells in each square.	• Timbre 1 • Timbre 2 • Sound thickness • 10 ranges of pitch • higher pitch • lower pitch • 10 different rhythms • Fast beat rhythm • Slow beat rhythm	• Number of sounds • Delay of sounds • Type of Timbre • Level of Pitch • Chorus effect

4.6.2 Reasoning Stage

Figure 4-18 shows the Reasoning Stage (taken from the Reasoning row of Figure 4-16) where the users organize, construct, or put together one or more conditions to form a statement or premise. Reasoning is the activity where users are required to construct, arrange or put together several conditions into an information statement or structure. These structured reasoning conditions or information statements are called premises. These premises are statements to support the reason why the conditions are being selected and also to describe what is represented by the conditions.

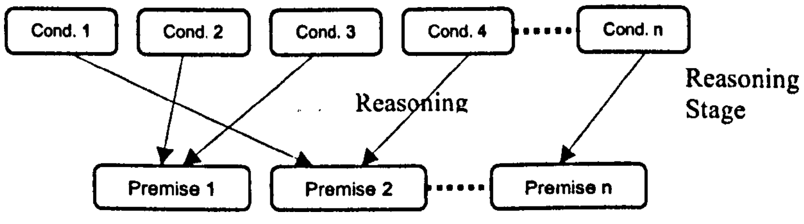


Figure 4-18: Reasoning Stage

Among all the conditions, probably only a few are important for use in the reasoning process. Some conditions could also be used repeatedly in different premises. At this stage, the user will only give attention to the potentially important premises at each perspective that might help in the hypothesising stage.

From the list of conditions above, some examples of premises that could be drawn in AVATCSS are shown in Table 4-7.

Table 4-7: Example of list of Premises

Perspectives	Example of Premises
Data Perspective	• Feature A is represented by x axis. • Feature B is represented by y axis. • Cells will be stored and categorized based on their A and B features. • x and y axes are divided into 10 bands to produce 100 square boxes. • Normal and abnormal cells are distributed among the 100 square boxes.
Acoustic Parameters Perspective	• The higher the pitch, the higher the value of A • The faster the beat of rhythm, the higher the value of B • The thicker the sound, the higher the density.
Final Sound Perspective	• The sound will be thicker and more complex if the number of channels is increased. • The delay of the sound will increase the complexity of the sound. • If the sound of timbre 1 is thicker and complex then the other sound timbre, it is more likely the cell being sonified a normal cell. • If the sound of timbre 1 has no chorus effect at all, therefore, there is no normal cell found in the square box. • If there is no chorus effect, the trend of pitch level and type of rhythm can be observed and compared with the previous sound – as the clue for the user.

In the Data Perspective, both Features A and B are important as it will be used as the reference for x and y axes. The x and y axes need to be divided into 10 bands to produce 100 square boxes, which will be used to categorise the distribution of the normal and abnormal cells.

In the Acoustic Parameters Perspective, the pitch is used as it will indicate the fluctuation of feature A. The rhythm is chosen to differentiate the axis and also to indicate the fluctuation of feature B's values continuously.

In the Final Sound Perspective, for instance in the last statement, the pitch and rhythm are important to be used in giving a clue about the condition of the cell being inspected, especially if there is no chorus effect.

4.6.3 Hypothesising Stage

Figure 4-19 shows the Hypothesising Stage (taken from the Hypothesising row of Figure 4-16) where the user tries to make sense, conceptualize or conceive the significance of the premises. This is normally influenced by their prior knowledge, previous experience or even through their instinct.

This stage requires at least one or more premise from the Reasoning Stage. The combination of several premises forms a knowledge statement called a *hypothesis*. A hypothesis is a proposed explanation of a phenomenon based on several premises (in each perspective) of the application. It describes the overall idea of the relationship between the outputs, conditions and premises.

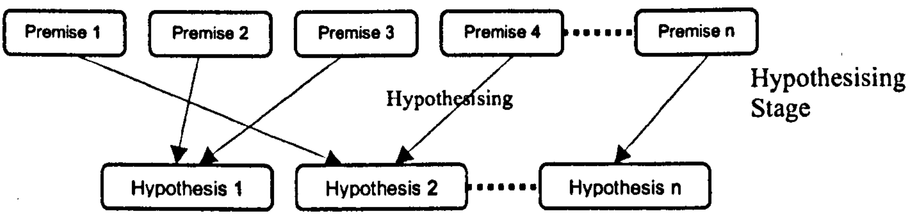


Figure 4-19: Hypothesising Stage

Some examples of hypotheses in AVATCSS are shown in Table 4-8. In the Data Perspective, the first hypothesis tries to relate the density of cells and the square box. In the second hypothesis, the user should be able to relate their decision with the density of cells. In the Acoustic Parameters Perspective, the first hypothesis tries to relate the features and the acoustic parameters (pitch and rhythm). And the second hypothesis tries to relate the range settings of acoustic parameters with the position of the square box. In the Final Sound Perspective, in the first hypothesis for instance, the user should be able to relate the complexity of the sound and the type of cell that is represented by the respective timbre.

Table 4-8: Example of List of Hypotheses

Data Perspective	• The density of normal and abnormal cells is represented by the number of normal and abnormal sample/reference cells in a square box. • The inspector's decision towards the cell being inspected is influenced by the density of normal and abnormal cells.
Acoustic Parameters Perspective	• The value of the features A and B can be predicted through the level of pitch and rhythm. • The range of pitch and rhythm will determine the position of the square box
Final Sound Perspective	• The more complex and thicker the sound of timbre 1, the higher the possibility of the cell being sonified to be a normal cell. • The more complex and thicker the sound of timbre 2, the higher the possibility of the cell being sonified to be an abnormal cell. • In a condition where, there are no reference cells in the square box, the level of pitch and rhythm can be used as the clue to decide whether or not it is more towards a normal or an abnormal cell.

4.6.4 Summary of the UIC Model

In summary, the UIC Model tries to predict the user's perceptions and interpretation of the outputs of a sonification application. Based on this model, a user selects several *conditions* based on the available output (data states) of the application. The user then organizes the condition to from *premises*. Finally, one or more premises will be interpreted by the user to create a *hypothesis*.

Based on these three stages of interpretation construction activities, the possible users' interpretation or hypothesis is created from one or a series of action(s), data states (outputs), condition(s) and Premise(s) as illustrated in Figure 4-20.

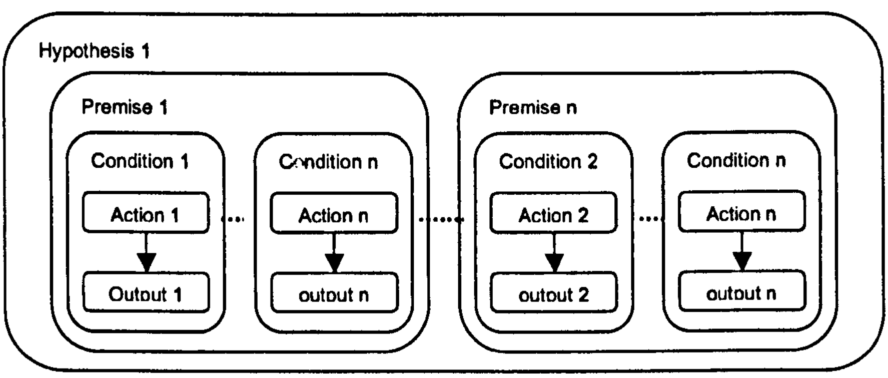


Figure 4-20: Conditions, Premises and Hypothesis of Sonification Applications

4.7 Summary

In summary, this chapter has presented a novel and extensive framework for analysing the design of sonification applications. It is intended to analyse the design from several different angles. The overall framework is called the Human Computer Interaction (HCI) model for Sonification Applications. This model encompasses of another two sub models called the Sonification Applications (SA) model and the User Interpretation Construction (UIC) model.

The SA Model considers all the tasks that are carried out by the computer, by the human user and by the human interacting with the computer (this is referred as *views* in the research hypothesis statement). In addition, it is also analysed according to how the data is transformed from its original form via an intermediate "ready to play" form, and then into the final sound (this is referred as *perspectives* in the research hypothesis statement). All these tasks in each perspective and input/outputs descriptions are used to describe the design of sonification applications. In this research, this design (*input/output; and tasks in all views and perspectives*) will be illustrated in a form of diagram. The diagrammatic way of representing this sonification design is called the Task-Data State Diagram, which will be explained in the next chapter 5.

The UIC Model is used to describe what and how the user should interpret the sound output. The interpretation covers all the outputs from all the perspectives, which is described in SA model. The model will be explained based on the three stages on how the user could interpret the output of the application, which include the selection, the reasoning and the hypothesising stages. Through this model, the designer could express the rationale behind their design.

As explained earlier in the definition of usability for sonification application where if the intended structure of the data and perceptual structure of the sound coincide, the user might get a useful mental representation as to what it should and could be. If this has happened, the task could be accomplished with high accuracy and completeness and therefore the application is said to be effective. By comparing Sonification Application (SA) model *with* User Interpretation Construction (UIC) model – 'what the application or designer would like the user to do and to know' with 'what the user might perceive and understand', the sonification application is said to be effective if these two (models) coincide.

Both models will become part of the inspection materials that will be used in the proposed usability inspection technique called Task Interpretation Walkthrough. This technique will be explained further in Chapter 6.

Next, Chapter 5 will explain the diagrammatic way to describe sonification applications design called Task-Data State Diagram, which is based on the SA model.

CHAPTER 5 : TASK-DATA STATE DIAGRAM TO MODEL SONIFICATION APPLICATIONS

This chapter discusses a new diagrammatic way to describe the design of Sonification Applications, which we have called the *Task-Data State Diagram*. This diagram is based on the Sonification Application (SA) Model previously discussed in Chapter 4. This diagram represents all the perspectives, views and data states of the SA model. Some related work, including the Data Flow Diagram, the Data-State Diagram and the ConcurTaskTree Diagram, will be discussed briefly. The content of this chapter has been presented in Ibrahim and Hunt (2007b).

5.1 Introduction

A task model can be used in the creation of user-oriented interactive applications. An example of a task model within auditory display is *TaDa* (Task analysis Data characterization), which analyses the problem scenario of auditory displays by splitting the design into several tasks and data categories (Barrass, 1997).

The overall design process might involve people with a range of expertise, including graphic designers, domain experts, usability specialists and sound professionals. They all require a platform to help them understand the application design, user tasks and other requirements. For sonification applications, it is especially important to have a "diagrammatic representation" that can be used to describe the application in as much detail as possible. This makes the design and discussion process easier by minimizing the number of necessary cross-references to other documents. As a result, we propose to describe sonification applications in diagrammatic way with notations that are:

- easy to understand and use, which will help to improve communication among people of different disciplines who are involved in the design process.

- o able to explain various sonification techniques
- o able to provide an overview of the whole concept of the sonification application being designed
- o explicit in their representation of the involvement of both the application and its users.

This chapter introduces a new diagrammatic way to describe Sonification Applications called the Task-Data State Diagram to model sonification applications. The diagram is the combination between the Data-state Diagram (Chi, 1999) and the ConcurTaskTrees (CTT) diagram (Mori et al., 2002).

Before describing how the diagrammatic representation works, it is essential to know what should be represented in pictorial form. Therefore, this chapter will start by looking at the detailed characteristics of sonification applications, which need to be portrayed by the proposed diagrammatic representation.

5.2 Characteristics of Sonification Applications

The aim of the Task-Data State Diagram is to describe the design of sonification applications based on the Sonification Application model explained in Chapter 4. The characteristics of sonification applications from the SA model are used as the requirements for the proposed diagram, and which are now discussed.

5.2.1 Transformations and Input/Output

The Sonification Application model, considers the application based on three *perspectives*: Data, Acoustic Parameters, and Final Sound. Each perspective represents a data-to-sound transformation process and its input and output. The transformation processes include:

- raw_data-to-processed_data (*Data Transformation*);
- processed_data-to-acoustically_ready (*Acoustic Parameters Transformation*); and
- acoustically_ready-to-sound (*Final Sound Transformation*).

However, some applications require the data to be processed into several other intermediate states before it becomes the input for the next transformation. This creates several states of data as well as sub-transformation processes within each perspective. It is important to show these different data states in the model as the user often matches the relationship between the sound's characteristics and the data or information characteristics they represent. For instance, two different groups of data are easier to differentiate with two sounds with different timbres; than two sounds with different pitches. If both were not strongly matched, it could be difficult for the listener or user to interpret or understand what is being represented by the sound. Therefore, a focus on the data states should make it easier to understand and evaluate the *relationship* between data and sound.

In the example of AVATCSS, Table 5-1 shows the inputs and outputs of the transformation in each perspective (referred to as the 'state of data'). The initial input of AVATCSS is a digital image of a microscope slide, which contains cell samples to be inspected (looking for potential cancer cells). As a reference, several samples of normally healthy and typically cancerous cells are also used as an input to the data transformation. The output (referred to as processed data) will be used for the Acoustic Parameter transformation. Examples of the processed data in this application are 'the number of normal and abnormal cells'; band 'value of A and B features' etc. However, before it is ready for Acoustic Parameter transformation, the data needs to go through several processes, which also produce several states of data. For instance, the data is segmented into several images made up of a 16-level grey-scale. From this segmented images, the A and B features of the cell are calculated. The reference cells will be grouped into different bands based on their A and B feature values. This grouping process is based on 10 different ranges (bands) on both the x and y axes, giving a total of 100 possible boxes into which each cell can be categorized. Therefore, the 'number of normal and abnormal cells' (which is an input for the Acoustic Parameters transformation) is simply the number of pre-scanned cells which share the same banded value of A and B features with the inspected cell to be sonified.

Table 5-1: Example of Input and Output of Transformations in AVATCSS

Input/Output in General	List of Input/Output (States of Data)	Description
Raw Data	*- Digital Image of Cervical cells on microscope Slide.* *- Pre-existing normal and abnormal cells images*	*Digital Image of microscope slide containing the cells to be inspected for either cancer or not-cancer. Pre-existing normal and abnormal cell images are used as a reference.*
Processed Data	*- Number of Normal Cells* *- Number of Abnormal Cells* *- Band value of A features* *- Band value of B features*	*- Number of normal and abnormal cells is the number of pre-existing cells that share the same value of A and B features with the inspected cells. The Band values for A and B features are the bands that belong to the inspected cell, which are based on its A and B feature values.*
Acoustically-Prepared Data	*- Timbre 1* *- Timbre 2* *- Pitch for band X* *- Rhythm for band Y*	*Timbre 1 and Timbre 2 are used to represent the normal and abnormal cells respectively. The Pitch and Rhythm are based on the band of A and B features of the inspected cell.*
Final Sound	*- Final mixed sound*	*The final sound is a combination of both sounds from the normal and abnormal cells, which are played at the same time.*

The example above shows that the data needs to go through several sub-transformation processes before it finally becomes the input of the next transformation. Each sub-transformation process will produce different states of data as shown in Table 5-1. Therefore, these sub-transformation processes and different states of data should be represented on the proposed diagram.

Having looked at the input, output and data states, we next consider the type of task found in sonification applications, and to be illustrated by the proposed diagram.

5.2.2 Tasks

A goal describes what the users want to achieve with the application. It can be decomposed into several tasks and sub-tasks, which are the necessary activities for the users and application to achieve the goal. The goals of sonification applications are firstly decomposed into three main transformation processes (data, acoustic parameters and final sound) for each perspective (data, acoustic parameters and final sound) as described in the previous section. Each transformation process is further decomposed into several sub transformation processes, which explain in detail how the data is transformed into different states in each perspective. Finally, each sub-transformation process is further decomposed into three tasks – based on the points of view of the **application**, the **user** and the **interaction** between the two. These types of tasks have been also used by Paterno et al. (1997) in their task-modeling diagram called the ConcurTaskTree (CTT), which will be explained later in this chapter.

As repeated from the previous chapter, the tasks at the *application* level are those performed by a system to process, manipulate and transform data into sound representations without user interruption. Tasks at the *interaction* level are those performed by the user through interactions with the system. And the tasks at the *user* level are those entirely performed by the user independent of the system, and which often concentrate on the user's perception or interpretation of the data.

The following example (with the help of Table 5-2) shows the decomposition of sub-transformation processes into their tasks (user, application and interaction tasks). There could be at least one or more user/application/interaction tasks to describe how the 'sub-transformation process' tasks can be accomplished.

In AVATCSS, the user needs to move the field of view to explore and inspect the cells in the digital image of the slide stopping at the cell to be inspected. Once the user has stopped moving or scanning the image, the system will check the most striking object/cell in the field of view and get its pixel coordinate. The pixel coordinate is used to look up the A and B feature values of the inspected cell, which was pre-calculated in the earlier processes. These feature values are used to check the corresponding information from the Reference A-B Features Look-up Table. These include the value of band x and y; and the density and number of normal and abnormal cells in the same band. This information will be used to produce the sound output.

Table 5-2: Examples of Tasks and Sub-tasks in AVATCSS

sub-transformation process Tasks	Tasks
Selection of field of view	**User Task:** *Decide which area to inspect* **Interaction Task:** *move field of view* **Application Task:** *Get position (coordinate) of the area*
Getting the correct cells	**Application Task:** *Get all cells in the field of view* **Application Task:** *Get the most striking object* **Application Task:** *Get its coordinate* **Application Task:** *Get its A and B feature values* **User Task:** *Observe the potentially most striking cell in the field of view*
Accessing information from Reference Look-up Table	**Application Task:** *Get the A and B values of the inspected cell.* **Application Task:** *Get band values* **Application Task:** *Get the number of normal and abnormal cells* .....etc.

Each of these sub-transformation process tasks produces several tasks as well as different states of data, which are normally interrelated and very dependent on each other. For example, before the application is able to calculate the number of normal and abnormal cells of an inspected cell, the user must decide in the first place on which area of the digital image to inspect. It is important that this temporal relationship between tasks is also described in the proposed diagram.

Other instances of temporal relationships between tasks include: the ability of several tasks to be implemented at the same time; the requirement of some tasks to be run repeatedly; or, the ability of the user to choose optional tasks. The types of temporal relationship between tasks are based on the ConcurTaskTree task diagram (Paterno, 1997), which will be explained in the next section.

Below is the summary of the characteristics of sonification applications from the SA Model that should be covered by the proposed diagram representation:

1. Portray clearly the three main transformation processes in each perspective i.e., Data, Acoustic Parameter, and Final Sound.
2. Represent the sub-transformation processes involved in each perspective.

3. Represent the different states of data as the input and output of the sub-transformation processes in each perspective.
4. Represent each sub-transformation process task based on tasks from three views –application, user, and interaction.
5. Represent temporal relationships, optional, repetition of processing tasks and sub tasks.

Next, three existing tasks and data flow graphical representations will be discussed. These include Data Flow Diagram, Data-State Model and ConcurTaskTree Diagram.

5.3 Related Work

There are several diagram notations or graphical representations that currently exist to express and explain an application design in terms of its processes, data flow, events and the relationship between tasks. This thesis will look at the *Data Flow Diagram,* the *Data-State Model* and the *ConcurTaskTree Diagram,* which will be discussed in the following sections.

5.3.1 Data Flow Diagram

A Data Flow Diagram (DFD) is a widely used diagrammatic representation technique in software engineering. It can be used to show the interaction between the system and outside entities such as users. It also shows the flow of data between external entities and the system, data flow from one process to another, and finally how the data is stored. In general, the notations consist of external entities, processes, data stores and data flows as shown in Figure 5-1.

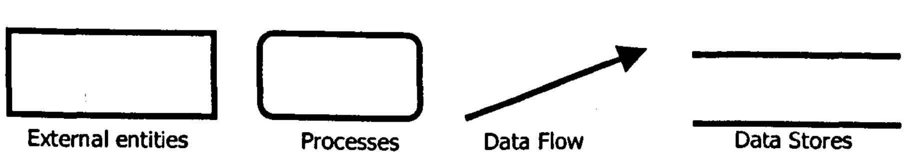


Figure 5-1: Data Flow Diagram Notations

The external entities are the sources of data such as users. Processes receive data as their input, do something to it and produce more data as the output. The arrow shows the direction of data flow. The open-ended rectangle shows a place to store the data.

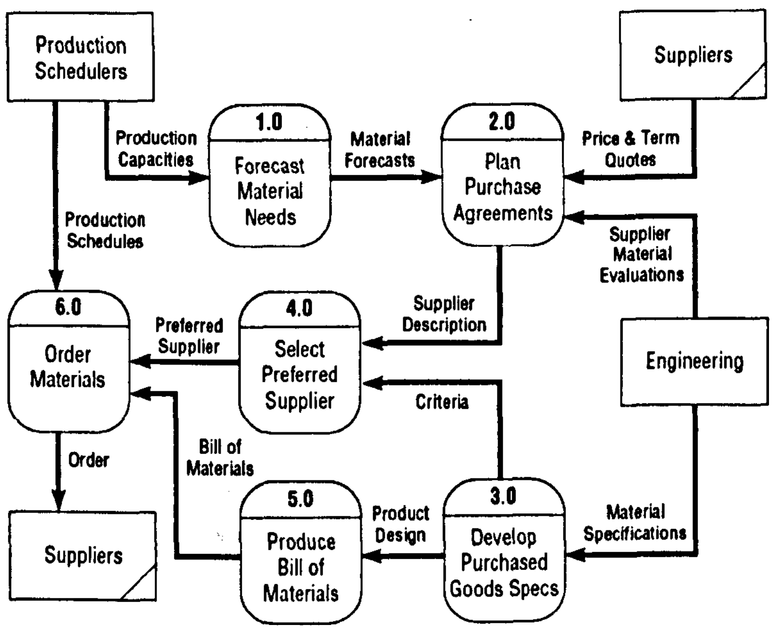


Figure 5-2: Example of DFD of Purchasing Fulfilment System from Hoffer et al. (1996)

The arrangement of diagram notations is normally scattered, where the external entities and processes are not arranged in sequence (top to down), as shown in Figure 5-2. In this example, processes 3.0 and 5.0 are located at the bottom of the diagram, while processes 4.0 and 6.0 are located in the middle of the diagram. It has to be arranged in this way to in order to avoid duplicating an external entity that receives and gives data. For instance, the diagram has two 'Supplier' external entities because the process 6.0 is located at the left side of the diagram. Process 6.0 has to be there to avoid having two 'Production Scheduler' entities at the same diagram. This makes it difficult to read and show smoothly the flow of the transformation processes as well as the different states of data, especially if it involves many processes and data changes.

In describing sonification applications, this diagram can be used to portray the general flow of the data transformation processes through its processes and data flows (arrow) notations. However, apart from showing process relationship through its sequence, it is difficult to represent the other temporal relationships such as the process tasks that can be implemented simultaneously. These difficulties are due to unavailability of notations to represent the temporal relationships between processes. For example, the diagram above shows that there is no information about the relationship between processes 4.0 and 5.0 e.g., whether both processes can be done simultaneously; and whether process 4.0 needs to be completed before process 5.0 starts or vice versa.

It is also difficult to represent the user and their interaction tasks with the system. For example, the diagram above shows the 'supplier' entity (the user) giving the 'price & term quotes' (the data) to be processed by 'process 2.0'. However, it is difficult to understand how the user gives the data; and what is needed by the 'supplier' before giving the data for process 2.0 (e.g., the knowledge required to select the correct 'price & term quotes').

In DFD, nodes (round rectangles) are used to denote processes and edges (arrows) are used to denote data flow direction. The focus of the edges is to represent the movements of data and its stages. However, in sonification applications, the focus needs to be given to the states of data at certain points due to several transformation processes that it has to go through.

The next section discusses the Data-State Model ((Chi & Riedl, 1998), (Chi, 1999, 2001)).

5.3.2 Data-State Model

The Data-State Model was suggested by Chi ((Chi & Riedl, 1998), (Chi, 1999, 2001)) to describe data visualization – a transformation of data into visual graphic representation. Unlike DFD, DSM uses nodes to represent data states and edges to represent processes as shown in Figure 5-3. The lines that divide several sub-processes and data-states into sections represent the different transformation levels. Each level could have several nodes and edges indicating the different data states and its sub-processes. Therefore, this model is said to give more attention to representing the states of data than its transformation processes.

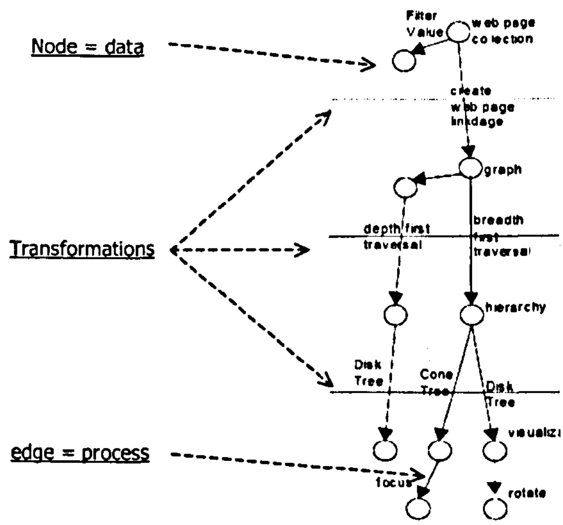


Figure 5-3: Example of Data State Model for Visualization, adapted from Chi & Riedl (1998)

In describing sonification applications, this sort of diagram can be used to describe the three main transformation perspectives (data, acoustic parameters and final sound), transformation processes and different data states. However, as in DFD, it too is not able to show interactivity and express the temporal relationships between tasks. The process tasks are portrayed in rather a general way as the focus is mostly given to the different states of the data.

5.3.3 ConcurTaskTree (CTT) Diagram

A *ConcurTaskTree* (CTT) diagram (Paterno, Mancini and Meniconi, 1997) is an example of task diagram that can be used to describe interactivity and temporal relationships between tasks. CTT describes tasks based on how a user accomplishes a goal within a specific application domain. The tasks consist of an abstract task, user task, interaction task and application task. Tasks are arranged in a hierarchical logical decomposition as shown in Figure 5-4. The advantage of this diagram is its ability to describe temporal relationships between tasks at the same levels, permitting the portrayal of interleaving, synchronization, enabling and iteration. It is also simple and easy to read as it reduces cross-references among tasks. These temporal relationships between tasks will be explained in detail in section 5.4.4.

In describing sonification applications, this diagram can be used to show transformation process tasks and their subtasks – user, application and interaction tasks. However, this diagram does not show clearly the different states of data. Referring to Figure 5-4, the data is explained as an entity. An action manipulates the entity to form a task. An example is circled below, where the object or entity was some 'results' and the action was to display them, so the task is written as 'ShowResults'.

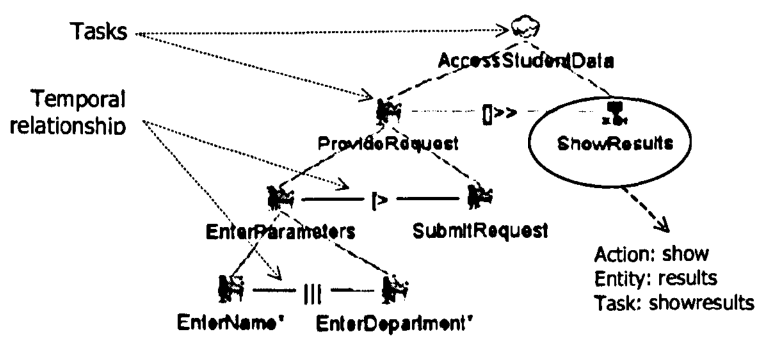


Figure 5-4: Example of ConcurTaskTrees (CTT), adapted from Paterno et. al. (1997)

Based on the requirements stated in Section 5.2 and the advantages and limitations of DFD, DSM and CTT, there is a need take advantage of these existing graphical representations for describing sonification applications. Therefore, in the next section, a new visual representation called the *Task-Data State diagram* is proposed, which combines the Data-State Model and the CTT Diagram to model Sonification Applications.

5.4 Task-Data State Diagram

The Task-Data State (TDS) diagram is created to show how the data is transformed into different states and the tasks involved in the transformation processes. This diagram combines the Data-State Model (DSM) and ConcurTaskTree (CTT) so as to focus on both data states and process tasks.

This diagram is designed to describe and represent the characteristics of sonification applications as mentioned earlier in section 5.2 It is also used to provide a visual representation of sonification application design. All the required information in this diagram comes from the designer's point of view, which can be gathered through series of interviews and discussions.

This diagram will be discussed based on four aspects of a basic framework for task modelling, which was introduced by Duursma and Olsson (1994). The aspects include agents, object, goal and activity ordering. An **agent** is one who performs tasks e.g., a user or the system. An **object** is the data or information to be manipulated e.g. the input and output of a process. The **goal** is the aim and the purpose of performing tasks. The **activity ordering** describes the relationship between tasks e.g., task ordering.

As in chapter 4, we will continue to use the example sonification application called the *"Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS)" shown in Italics.*

5.4.1 Agents and Task Type

TDS adopts the task types from CTT (Paterno et al., 1997), which include abstract, user, interaction and application tasks as illustrated in Table 5-3. These tasks are performed by agents. The agents in this diagram can be either human (user) or application (software).

Table 5-3: Notations of tasks [adapted from CTT Paterno et. al., (1997)] and its descriptions for sonification applications

Notations	Descriptions
 [Abstract Process Task]	An **Abstract Process Task** represents a sub-transformation process in each perspective. It is described in general, and can be further decomposed into user, interaction and application tasks. It is performed to achieve and accomplish the goal of the three main transformation processes (data, acoustic parameters and final sound)
 [User Task]	A **User Task** is performed by the user independent of the application in order to achieve and accomplish the goal of the sub-transformation process (Abstract Process Task) it belongs to.
 [Interaction task]	An **Interaction Task** is performed by the user through interactions with the application in order to achieve and accomplish the goal of the sub-transformation process (Abstract Process Task) it belongs to.
 [Application Task]	An **Application Task** is performed by the system to process, control, manipulate and transform input (data) into output without user interruption in order to achieve and accomplish the goal of the sub-transformation process (Abstract Process Task) it belongs to.

Figure 5-5 shows how the Abstract Process task (sub-transformation process) is further decomposed into its three type of sub-tasks (user, application and interaction, each represented by its own icon). Figure 5-5 a) shows the 'Abstract Process task' has done a sub-transformation task of its input and produces two outputs. The outputs of the abstract process task are normally produced by the application task as a feedback. This is represented by the arrows that are coming from the 'application task' as shown in Figure 5-5 b).

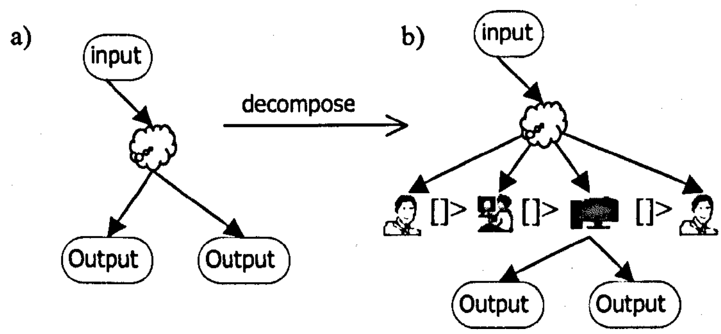


Figure 5-5: Decomposition of Abstract Process Tasks into Sub-tasks (Agent's Tasks) (inspired by DSM and CTT)

In summary, several Abstract Process tasks and their decomposition of sub-tasks (user, application and interaction) are used to describe three main transformation processes (data, acoustic parameters and final sound) that are involved in transforming raw data into final sounds in sonification applications. Starting from now on, sub-transformation process tasks will be referred as Abstract Process tasks.

As an example in AVATCSS (refer to Figure 5-6), the abstract task is to 'select the field of view', which is a rectangle that the user needs to move around the cells image to be sonified. The abstract task is further decomposed into a user task, an interaction task and two application tasks. To select the field of view, the user needs to decide which area to inspect. To select the area on the image, the user needs to interact with the application. The selected area will be processed by the application to produce a list of coordinate (pixels) of the view field and the selected area to be displayed.

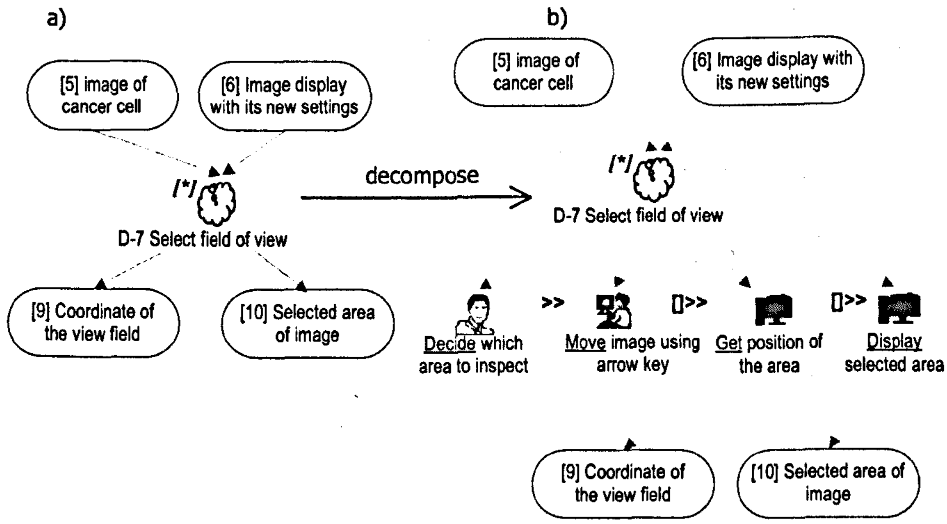


Figure 5-6: Example of Abstract Process Tasks decomposition into tasks in AVATCSS

The example above shows the **agents** and their **tasks** in processing the input (image cells) into its outputs (image coordinate). The input and output of process tasks will be discussed in the next section.

5.4.2 Objects

The Objects are the input or output of an Abstract Process task in sonification applications. Each Abstract Process task could have one (or more) input and output as shown in Figure 5-5. The output of the 'previous task' will become the input of its following (next) task. These inputs and outputs represent the different states of data. In the TDS diagram, the states of data are explained by adopting the Data-States Model (Chi et. al., 1998) as this model focuses on the data itself. Each input and output is denoted in node (round rectangle). It displays the changes and intermediate results of Abstract Process tasks and their sub-tasks.

In the example of AVATCSS as shown in Figure 5-6, there are two possible inputs of this abstract task – [5] image of cancer cells and [6] image display with its new settings. Both of these inputs are the output of the task beforehand. The outputs of this task are [9] coordinate of the view field and [10] selected area of image. The 'coordinate of the view field' is used as the reference to find the cells from the table *of pre-generated image cells. The 'selected area of image' is the area to be displayed and marked as a scanned area, which will be done by the next process task.*

5.4.3 Goal

The goal is described based on the 'general purpose' and 'specific issues' of each Abstract Process task shown in Figure 5-7. The general purpose explains what the Abstract Process task needs to achieve from its process activity. The specific issues explain the specific goals that need to be accomplished by the three different sub-tasks (users, application and interaction) in order to achieve the general goal of the abstract task.

Task: Abstract Process Task
General Purpose: [Describe the goal of Abstract Process Task]
Specific Issues:

1. **Users**: [Describe the goal of user tasks in supporting the general purpose]
2. **Application**: [Describe the goal of application tasks in support of the general purpose]
3. **Interaction**: [Describe the goal of Interaction tasks in supporting the general purpose]

Figure 5-7: Goal of Abstract Process Task

As a guideline for selecting the most suitable name for abstract process tasks and tasks, the name must reflect what it does and what it needs to achieve. For example in Figure 5-8 (a), if the image to be inspected is the input of the Display process, the goal of the abstract tasks can be described as to display the image to be inspected. The task can be written in one word (normally as a verb) if there is only one output and one or more input. However, if the abstract task has more than one output – in order to reflect the overall task goal, the name of the task can be written together with a general description of the output. For example in Figure 5-8 (b), if the two output coordinate of the view field and selected area of image were referred as field of view, the name of the abstract task can be written as select field of view. The goal of the task is to select field of view of cancer cell image.

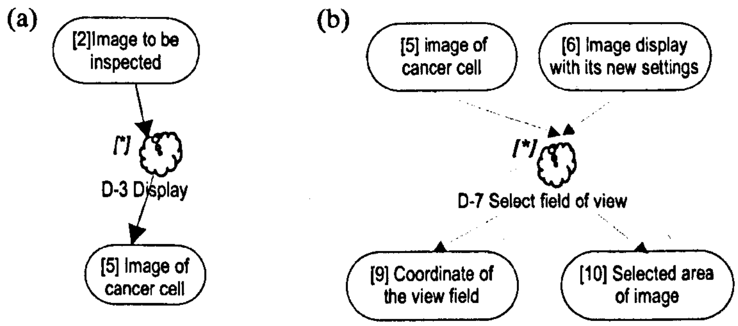


Figure 5-8: Example of abstract process tasks

The goals of each transformation process and the overall application are not explained directly, but the collection of goals of Abstract Process tasks in each perspective can be used to explain them. Therefore, the goal decomposition of the application into Abstract Process tasks is important to express the goal of the overall application.

In the example of AVATCSS (refer to Figure 5-6), the abstract task (Select Field of View) is further decomposed into several subtasks (User, Application and Interaction tasks). An example of the 'general purpose' and 'specific issues' *of the 'Select Field of View' abstract tasks is shown in Figure 5-9. In order to produce the two outputs, a field of view needs to be selected. In order to do this, the user in the first place needs to decide which part of the image is to be inspected. When the user wants to move the field of view, the interaction should be as real as moving the image in the microscope. Once the user selects an area, the application should be able to identify the position area and the pixel coordinates of 'viewfield' for the next processing tasks.*

Task: *Select field of view*
General Purpose:
- *to display the image or some part of the image as selected by the user*
- *to allow the user to explore a 'zoomed in' image.*

Specific Issues:
1. ***Users:*** *decide where to explore the image and know the area that has not been scanned/inspected yet.*
2. ***Application:*** *be able to identify the position area so that the area could be dimmed if the user chooses to dim the scanned area.*
3. ***Interaction:*** *allow a smooth transition of image movement so that the user feels like they are looking, zooming and moving the image as they normally do with a microscope.*

Figure 5-9: Goal of Abstract Process Task 'Select Field of View' from the example of AVATCSS

5.4.4 Activity ordering

One of the strengths of CTT is its ability to describe *temporal* relationships between tasks. This temporal information is important especially among the three subtasks – user, application and interaction tasks. In our TDS diagram, all the temporal relationships introduced as CTT operators will be reused.

- Interleaving (| | |) – tasks can be done in any order.
- Enabling (>>) – the first tasks is required for the activation of second task.
- Enabling with information passing ([]>>) – the output of the first task is required by the second task.
- Synchronization (| [] |) – two tasks need to occur at the same time.
- [> : deactivation ([>) – the first tasks is deactivated when the second tasks is performed.
- [Task name] or *[opt]* – task is an optional
- Iteration ([*]) – task is performed many times
- *[loop n]* : iteration 'n' times
- *[once]* : task is performed only once

An example of these temporal representations can be seen in Figure 5-6, Figure 5-10, Figure 5-11 and Figure 5-12.

Unlike in CTT, TDS denotes the task's input and output as nodes; and arrows as the direction of data flow. Therefore, to show a temporal relationship between two abstract tasks that 'produce' and 'use' the same data state, we can either place them 1) vertically aligned as in Figure 5-10 a); or 2) side by side as in Figure 5-10 b). Both representations convey the same meaning and thus we can optimise the diagram's layout.

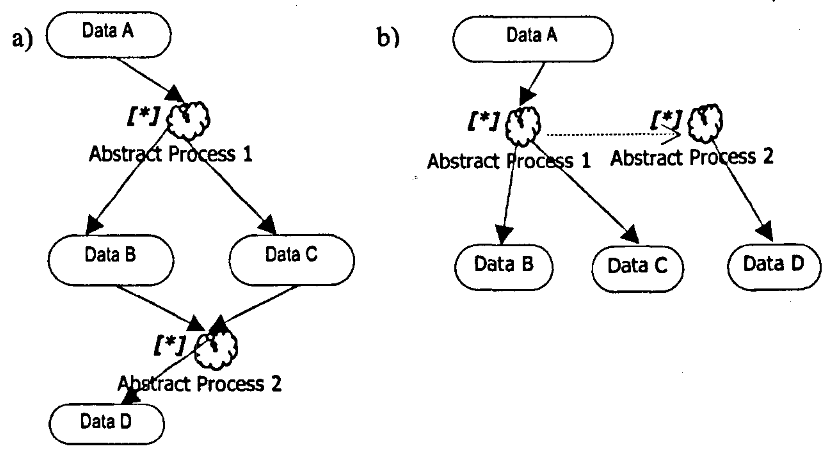


Figure 5-10: Temporal Relationship by Arrows

In the example of AVATCSS (refer to Figure 5-6) b), the relationship between the user task (Decide which area to inspect) and the interaction task (Move image using arrow key) is **enabling**, which is indicated by *'>>' sign. This means that the user should decide which area to inspect before moving the field of view. The application task (Get position of the area) will not able to proceed to the process tasks until the user decides the final 'field of view' to be inspected and sonified. Due to this constraint, the temporal relationship between the interaction task (Move image using arrow key) and the application task (Get position of the area) is indicated as '[]>>' sign, which is 'enabling with information passing',* as the application task requires the position of the 'field of view' to get and store its coordinates.

The next section explains the overall overview of the Task-Data State Diagram.

5.4.5 Overall Diagram

The TDS diagram is divided into sections as with the Data-State Model to ensure that the transformation processes can be easily observed. The number of sections is based on the number of transformations (perspectives) involved in the application to be modelled. Each transformation and its sub-transformations will be explained by using the four tasks, which include abstract, user, application and interaction tasks as in CTT diagram. To avoid the diagram to become too complicated, the application can be described at the macro level by only displaying its

Abstract Process task as shown in Figure 5-11. This is useful for giving a general and fast overview of the application design.

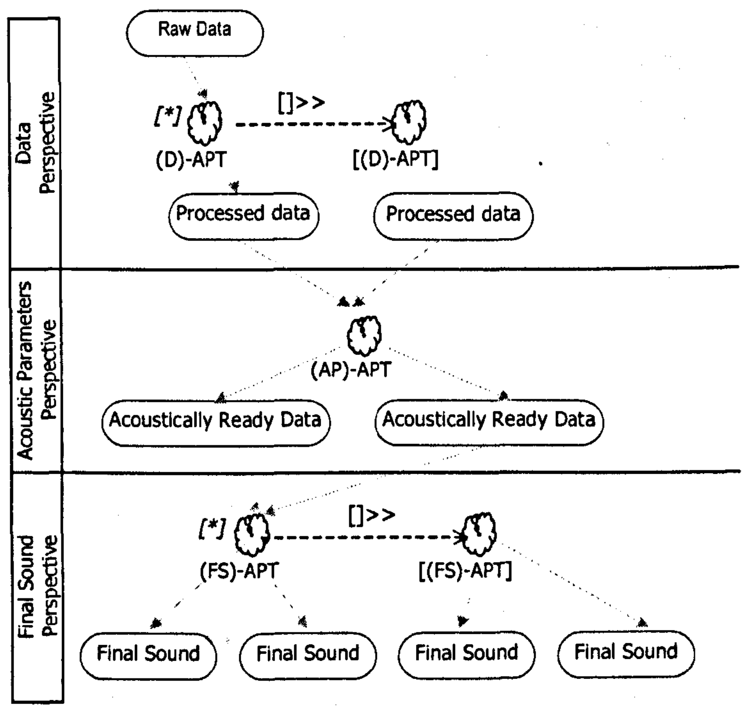


Figure 5-11: Macro level view of the Task-Data-State (TDS) Diagram

As a reminder, we have identified four main data states in Sonification Applications namely – raw data, processed data, acoustically-ready data and final sound. These different data states are produced at least by three transformation processes namely – data, acoustic parameters and final sound transformations. To model and describe a sonification application using our TDS diagram, it is divided into three sections to represent the three different perspectives as shown in Figure 5-11. Starting from the top, it shows the Raw Data being transformed into Processed Data, followed by the Acoustically-Prepared Data and then into the Final Sound. There are three types of Abstract Process Task (APT), where each focuses on its specific transformation process including Data (D)-APT, Acoustic Parameters (AP)-APT and Final Sound (FS)-APT. These three APTs are useful for describing the macro level of sub-transformations.

Figure 5-12 shows how the Abstract Process Tasks (APTs) are further decomposed into User (UT), Application (AT) and Interaction (IT) tasks to form a more detailed description of sonification applications. In Data perspective, the APT is further decomposed into Data-User Task (D-UT), Data-Application Task (D-AT) and Data-Interaction Task (D-IT). In Acoustic Parameters perspective, the APT is further decomposed into Acoustic-Parameter-User Task (AP-UT), Acoustic-Parameters-Application Task (AP-AT) and Acoustic-Parameters-Interaction Task (AP-IT). In Final Sound perspective, the APT is further decomposed into Final-Sound-User Task (FS-UT), Final-Sound-Application Task (FS-AT) and Final-Sound-Interaction Task (FS-IT). These three perspectives and four type of tasks are highly significant for a better understanding of sonification applications as each of them significantly influences the final sound output.

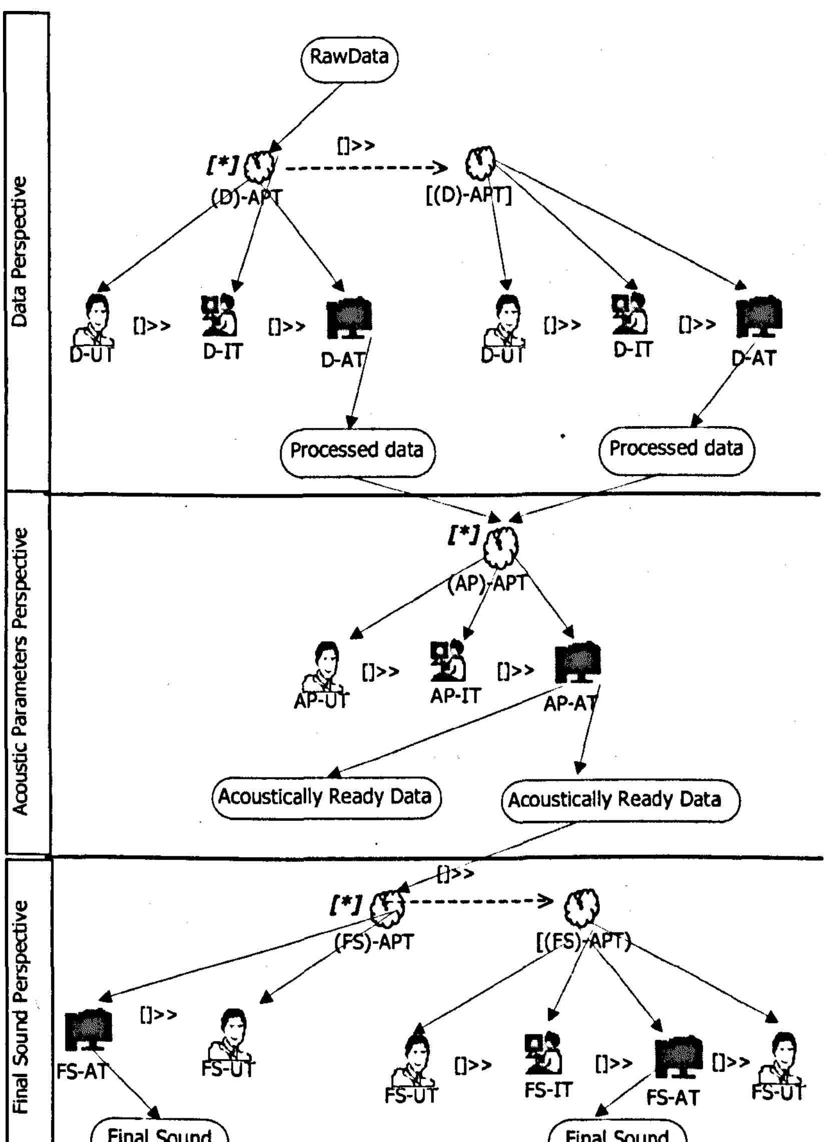


Figure 5-12: Overview of Task-Data-State Diagram to model Sonification Applications

In summary, TDS diagram explains the *processes involved in changing data into different states* in a form of process tasks (APT) as well as the agents who execute them (UT, AT and IT). Starting from an Abstract Process Task, this is then decomposed into User, Interaction and Application sub-tasks. In general, the diagram is able to show and describe the following information about sonification applications such as:

1. The flow of how data is transformed into different states and finally into sound
2. The sub-transformations involved in each main transformation
3. The involvement of users in changing the data states
4. The different states of the data itself
5. The goal of each transformation process and how the tasks achieve it
6. The final input and output of each main transformation process
7. Internal input/output manipulation in each transformation process before it becomes the final input of the next transformation process.
8. The optional and mandatory processes
9. The temporal relationship between tasks

TDS enhances both CTT and DSM (which concentrate only either on the task or data states) by giving attention to both tasks and data. As a result, we can observe the relationships between the data states and the process tasks at the same time. The example of this diagram in AVATCSS will be discussed in detail in Chapter 6.

5.5 Summary

This chapter has introduced the new Task-Data State (TDS) Diagram, which can be used to model sonification applications. The diagram shows how the data is transformed from its original state into a final sound representation. This diagram can be used by designers to describe their application design and as a discussion platform with other specialists such as application domain experts, usability consultants and graphical user interface professionals. In this thesis, the TDS will be used to graphically describe sonification applications, and thus will become one of the inspection materials to be used by the proposed usability inspection technique – Task Interpretation Walkthrough, which will be explained in the next chapter.

CHAPTER 6 : USABILITY INSPECTION TECHNIQUE: THE TASK-INTERPRETATION WALKTHROUGH

This section explains our new usability inspection technique called Task-Interpretation Walkthrough (TIW). This inspection technique is based on the Human Computer Interaction model for sonification applications explained in Chapter 4. The instructions and inspection materials required by the technique will be discussed. The inspection materials include design descriptions in the Task-Data State diagram form (Chapter 5); interpretations of the predicted outputs; the context in which the application will be used; and other documents such as list of user requirements and graphical user interfaces. The content of this chapter has already been presented publicly in Ibrahim and Hunt (2007a).

6.1 Overview of Usability Inspection Approach

Figure 6-1 shows an overview of how the proposed inspection technique works. The person who heads the inspection process is called the chief inspector. The chief inspector could in theory also be the designer himself, but an extra level of interaction and independence is gained by having a separate person do this task. The chief inspector uses the Sonification **design** to prepare the **Inspection Materials**. Inspection materials can consist of descriptions of software design, sketches of graphical interfaces, instructions and any documents that are required by the particular inspection method. Different inspection methods require different inspection materials. They are distinctive from each other in various aspects such as the purpose and focus of the method; the type of problems or anomalies the method addresses; and how the method guides the inspector to do the inspection. For instance, a Cognitive Walkthrough focuses on the goals and knowledge of a user while performing a specific task, whereas a Heuristic Evaluation emphasises a list of 'usability principles' to be followed as a guideline.

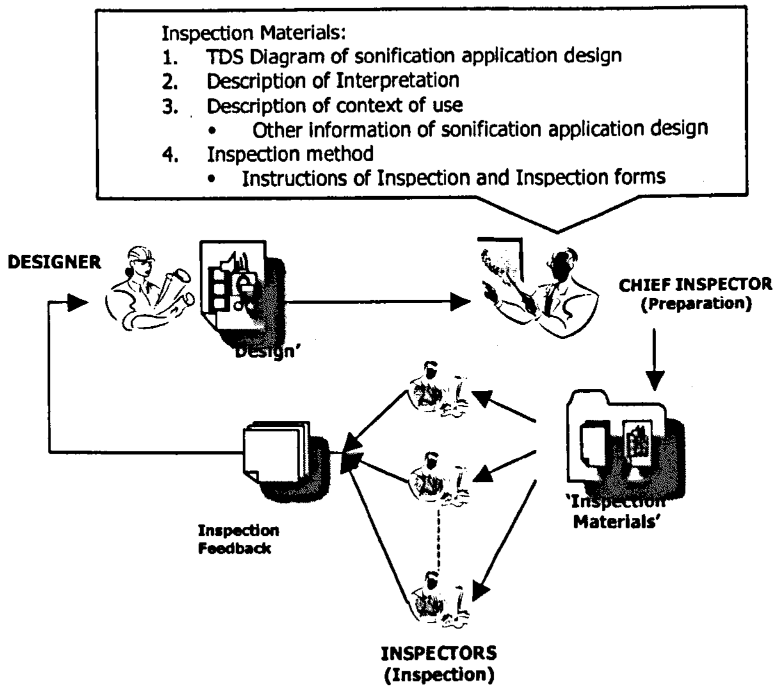


Figure 6-1: Overview of Usability Inspection Strategy of Sonification Applications

Figure 6-1 shows that the chief inspector is required to prepare **inspection materials**, which contain the following:

1. Description of the sonification application design to be evaluated by using the Task-Data State (TDS) diagram [based on the SA Model] *(explained in Section 6.2.1).*
2. Description of how the user should interpret the predicted application outputs [based on the UIC Model] *(explained in Section 6.2.2).*
3. User requirements and context of use *(explained in Section 6.2.3)*:
    a. List of user requirements;
    b. Description of contexts where the application might be used;
    c. Attachments (if any) of the sonification application design such as diagrams or sketches of graphical interfaces.
4. Description of inspection method *(explained in Section 6.3)*:
    a. Steps and instructions of the TIW inspection technique.
    b. A number of forms to write the encountered problems.

Several inspectors inspect the design using the inspection materials. As the output of this process, these activities produce qualitative results including the early identification of usability problems, anomalies, comments and sometimes suggestion of solutions. The potential problems found **(Inspection feedback)** by this process can be used to make recommendations on how to fix and improve the design.

Figure 6-2 shows where the proposed extended inspection can be placed in the evaluation steps. Upon agreement of tasks and sound design, and before a prototype is developed, the design will be used in the usability inspection. Two phases are involved during the inspection – **Inspection Preparation** and **Inspection Activities** (explained in detail in the next sections).

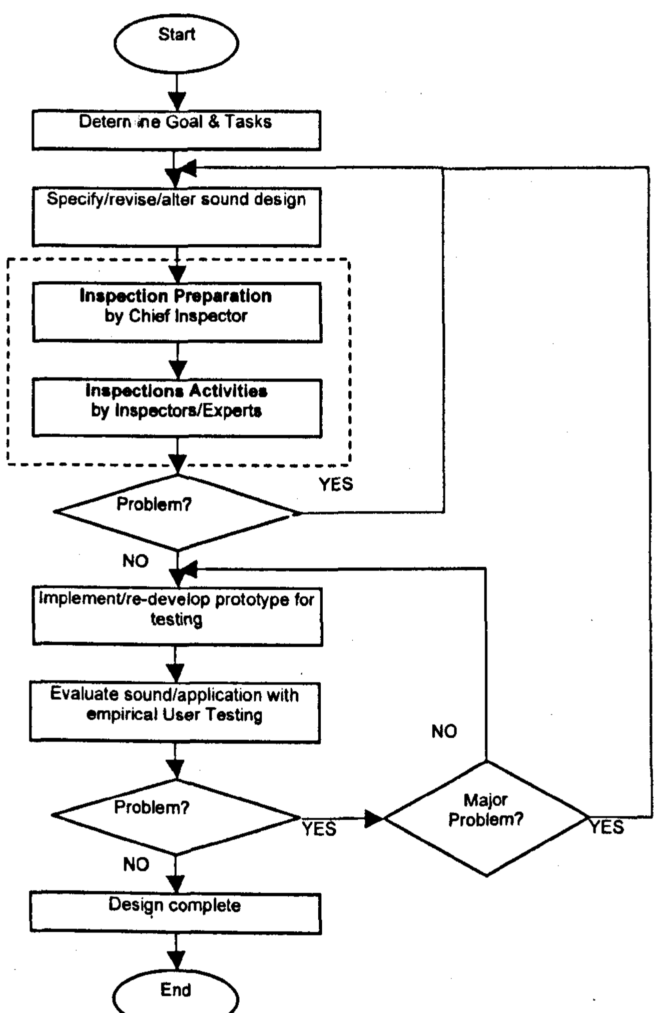


Figure 6-2: Flow Chart showing where the Inspection should take place

Inspection Preparation is where the designer or chief inspector produces the inspection materials. Inspection Activities are where the inspectors or experts use the inspection materials to detect potential problems with the design. The feedback (potential problems) are used as guidelines to revise and alter the design. The same processes can be repeated and become part of an iterative design process for sonification applications.

After all the necessary design changes have been carried out, the designers can proceed to the development phase. A normal round of end-user testing can be run to further refine the design before it is finally used. In general, the proposed usability inspection should:

1. *Help the evaluator to understand and explore how the sonification application can be used.*
2. *Help the evaluator to identify potential problems in the design of the sonification application.*

From this usability inspection, many potential design problems can be discovered and fixed before even implementing the prototype. As mentioned in all existing inspection techniques, this *proposed inspection for sonification applications* would not guarantee perfection or completeness in detecting all potential problems. But this technique provides an alternative to traditional usability evaluation with much lower costs (in money and time) and highly informative iterations early in the design process.

In summary, this chapter introduces a novel inspection technique, which purposely allows the inspection of sonification applications. The core idea of this technique is to understand the design rationale of the sonification applications being inspected. It is proposed to critically analyse the design tasks and understand how users interpret the output through the two models; the **Sonification Application (SA) model** and the **User Interpretation Construction (UIC) model.**

The next section explains the preparation phase of the inspection technique.

6.2 Inspection Preparation

This section describes the inspection materials required by the Task Interpretation Walkthrough (TIW). The materials include the user goal, context of use, task data state (TDS) diagram, interpretation and instruction of inspection. As in chapter 5, we will continue to use the example sonification application: *"Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS)"* shown in Italics.

6.2.1 Analyse Tasks and Data States of Sonification Applications

In general, a task is an action (or series of actions) that needs to be done to achieve its goal. Task analysis is an established technique in the field of HCI for understanding the way people perform tasks with a system. It includes the decomposition of task into subtasks; classification of tasks; and understanding of the current state of the application. In the field of auditory display, Barrass (1997) introduced TaDa (*Task Analysis Data Characterisation)* to design sounds that carry useful information. It focuses on identifying key features of the problem description and the requirements of a solution to be used in a design.

In TIW, the purpose of task analysis is to describe the design of a sonification application for the purpose of usability inspection. The focus is on 'what the application will offer to the user' rather than 'what the user wants from the application'. These two different focuses differentiate between the task analysis in TaDa and the task analysis for the TIW. The information is gathered from the designer's point of view.

This analysis will give us an overview of the designer's intentions for the sonification application design. To aid evaluation, it is important for the evaluator to understand the intention of the software. Even though the intention of the software is normally what is required by the users, but the requirements could be delivered and presented in a different way. The same user requirements can be presented with more than one techniques or software design. Therefore, the intention of the software focuses on how the designer plans for the application to deliver the users' requirements.

The tasks and data state analysis of sonification applications are implemented by using the Task-Data State (TDS) diagram. This shows how the data is transformed into different states and the tasks involved in the transformation processes.

In general, the purposes of this tasks and data states analysis are:

1. to understand the overall design of the sonification application;
2. to know the state of the inputs and outputs involved in the application;
3. to understand the tasks involved in changing the state of input and output;
4. to understand what the designer would like the user to know and to do;
5. to understand how the user will interact with the sonification application; and
6. to understand how the system does the transformation processes.

As we have seen in previous chapters, for sonification applications, tasks are categorised based on three different perspectives (data, acoustic parameters and final sound) and three views (user, application and interaction). These three different perspectives and views produce nine different types of task to describe sonification applications.

Before we proceed to the analysis part, we will look back at a few terms used in Chapter 5 to create the Task-Data State diagram. The processes involved in transforming data into sound in each perspective are called **Abstract Process tasks.** There are three categories of Abstract Process task – for data perspective, acoustic-parameters perspective and final-sound perspective. Each Abstract Process task has inputs and outputs. These inputs and outputs are referred to as **states of data.** Each Abstract Process task is further decomposed into three tasks called **users, application** and **interaction.**

In summary, the following information is gathered from the designer:

1. Abstract Process tasks in each perspective
2. Input(s) and output(s) of each Abstract Process task
3. Sub-Tasks for each Abstract Process task (user, application and interaction)
4. Goals for each Abstract Process task, which are described as 'General Purpose' and 'Specific Issues' respectively (explained in Chapter 5).

6.2.1.1 Goal, Task Type and Data States for Abstract Process Task

The analysis starts by looking at the Abstract Process tasks in each transformation perspective. This includes its goal, task type, task temporal relationship, input and output. Each piece of information is important for creating a TDS diagram for a particular sonification application.

Before looking at the details and examples of the analysis, Figure 6-3 shows the overview of activities that will be done during the tasks analysis of sonification applications. All sub-transformation-process tasks as well as their input and output will be identified in all perspectives. Then, each Abstract Process task is further decomposed into three tasks – users, application and interaction. However, depending on the type of Abstract Process task, it may not necessary for it to have all the three tasks. For instance, an Abstract Process task might not allow a user to interact with the application, which in this case, there is no interaction task involved. An Abstract Process task can also have more than one of each user, application and interaction task. Finally, temporal relationships between tasks are determined in both – 'between Abstract Process tasks' and 'between tasks under each Abstract Process task'. All the tasks will be described based on their Task Goal and Task Type, which will be explained later in this section.

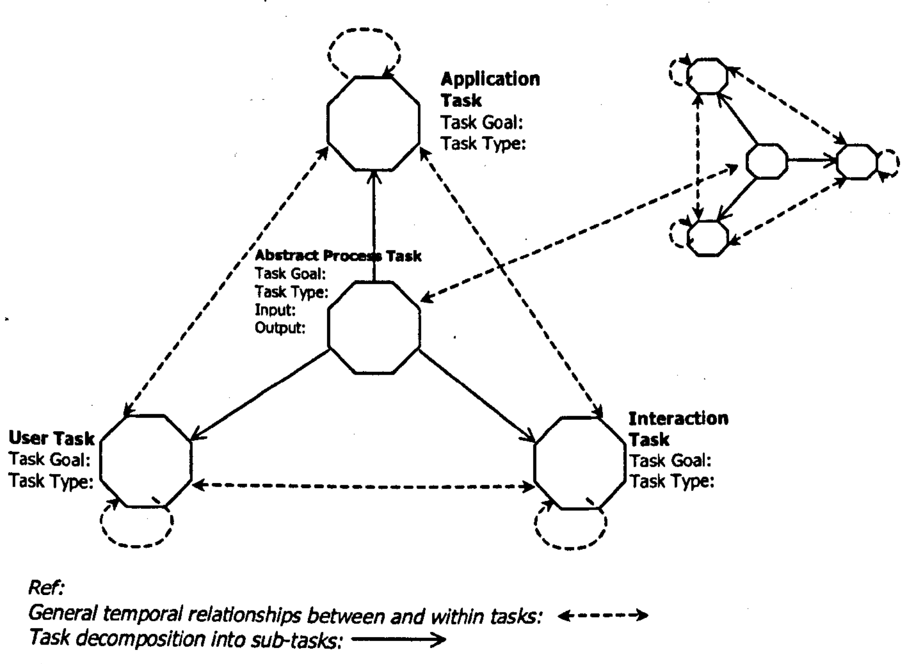


Figure 6-3: Overview of activities in data analysis for TDS diagram

Next, we will look at the task analysis by using the AVATCSS application design as an example.

Goal of Abstract Process Task

The Goal is what the process tasks are intended to achieve. It is a statement of what needs to be accomplished and normally contains the input and output of the task. The analysis can be started by determining the Abstract Process task of the first system input at the Data perspective. It is then followed by another Abstract Process tasks in the Acoustic-Parameters perspective and Final-Sound perspective.

The goal describes the 'purpose' of the 'Abstract Process task' as well as its input(s) and output(s).

Examples of task goal statements in each perspective in AVATCSS:

Data Perspective:

- *To get rid of most of very small contamination objects from the digital image slide to ensure it contains only nuclei and sections of nuclei.*
- *To display the image or some part of the image (field of view) that is selected by the user; allowing the user to explore a 'zoomed in' image.*

Acoustic Parameters Perspective:

- *To represent the 10 bands of the x-axis with 10 ranges of pitch.*
- *To map the current B feature value of the inspected cell to a rhythm based on the list of 10 predefined rhythms.*

Final Sound Perspective:

- *To combine all acoustic parameters including timbre 1, pitch of A feature, rhythm of B feature, delay and number of channel to produce a sound of normal cell*
- *To change the complexity of both normal and abnormal sound sections by enabling and disabling certain acoustic parameters*

Task Type of Abstract Process Task

The Type of Task is used to identify and categorize tasks in sonification tasks analysis. These are expressed as 'verbs' or 'actions', which can be obtained from the goal statement. An Abstract Process task does one or more actions on inputs to produce outputs.

From the examples above, the type of Abstract Process tasks are as follows:

Data Perspective:

- *The tasks are 'to get rid of' and 'to display'.*

Acoustic Parameters Perspective:

- *The tasks are 'to represent' and 'to map'.*

Final Sound Perspective:

- *The tasks are 'to combine' and 'to change'.*

Input and output of Abstract Process Task

An Abstract Process task processes input(s) to produce output(s), which then will be used by the next Abstract Process task. This is continuous activity, which will only stop when the final sound is produced. These input(s) and output(s) are different states of data, which will be used and illustrated in the TDS diagram.

From the examples above, the states of data are as follows:

Data Perspective:

- *The data states are 'Digital image slide', 'nuclei and sections of nuclei', 'field of view' and 'zoomed in image'.*

Acoustic Parameters Perspective:

- *The data states include '10 bands of the x-axis', 'list of 10 ranges of pitch', 'list of 10 predefined rhythms', 'Current B feature value', 'inspected cell' and 'a rhythm from the list of 10 predefined rhythm'.*

Final Sound Perspective:

- *The data states are 'timbre 1', 'pitch of A feature', 'rhythm of B feature', 'delay', 'number of channel', 'sound of normal cell section' and 'normal and abnormal sound sections with enabled/disable acoustic parameters'.*

Summary of Abstract Process Task Descriptions

In summary, Figure 6-4 shows the required information to explain an Abstract Process task in each perspective. The descriptions of an Abstract Process task include 'the task category based on the perspective it belongs to'; 'the goal it should achieve'; 'the type of task it needs to do'; 'the input it requires to process'; and 'the output it will produce'.

Abstract Process Tasks Category:
[*Data Transformation; Acoustic Parameters Transformation; Final Sound Transformation*]
Task Goal: [*What the task needs to achieve*]
Task Type: [*the intention of the task*]
Input:
State of data [*Raw Data, Processed Data, Acoustically-ready Data, Final Sound*]
Output:
New State of data [*Raw Data, Processed Data, Acoustically-ready Data, Final Sound*]

Figure 6-4: Descriptions of Abstract Process Task

Figure 6-5 shows three examples of Abstract Process task in each perspective in AVATCSS. The information is derived from the above examples.

Abstract Process Tasks Category: *Data Transformation*
Task goal: *to display the image or some part of the image as selected by the user and allow the user to explore a 'zoomed in' image.*
Task type: *to display*
Input(s): *digital image slide*
Output(s): *zoomed in image*

Abstract Process Tasks Category: *Acoustic Parameters Transformation*
Task goal: *to map the current B feature value of the inspected cell to a rhythm based on the list of 10 predefined rhythms.*
Task type: *to map*
Input(s): *Inspected Cell, Current B feature value, list of 10 predefined rhythms*
Output(s): *Customized Rhythm List, A rhythm from the list of 10 predefined rhythms*

Abstract Process Tasks Category: *Final Sound Transformation*
Task goal: *To combine all acoustic parameters including timbre 1, pitch of A feature, rhythm of B feature, delay and number of channel to produce a sound of normal cell section*
Task type: *to combine*
Input(s): *timbre 1, pitch of A feature, rhythm of B feature, delay and number of channel*
Output(s): *sound of normal cell section*

Figure 6-5: Example of Abstract Process tasks for AVATCSS

6.2.1.2 Decomposition of Abstract Process Task

Each Abstract Process task is further decomposed into user, application and interaction tasks. These tasks are also described by their 'goal' and 'task type'. The task decomposition forces us to look at how the 'user, application and interaction' tasks should help to accomplish the goal of Abstract Process task. This is done by relating the three tasks (**data, application** and **interaction**) with **the Abstract Process task (goal** and **task type** as well as its **input** and **output)** in each perspective.

The goal and task type need to be gathered from the designer's point of view. This is important, as the purpose of the evaluation is to evaluate their application design. This can be done through interviews with the designer, asking the following questions:

1. How will the 'application' help to achieve the goal of the Abstract Process task?

2. How will the 'interaction' help to achieve the goal of the Abstract Process task?
3. How will the 'users' help to achieve the goal of the Abstract Process task?

In summary, the tasks are explained based on the following descriptions:

Task Category:
[*Data-User Task; Data-Interaction Task; Data-Application Task;*
Acoustic-Parameters-User Task; Acoustic-Parameters-Interaction Task; Acoustic-Parameters-Application Task;
Final-Sound-User Task; Final-Sound-Interaction Task; Final-Sound-Application Task]
Task Goal: [*What the task needs to achieve*]
Task Type: [*The action to be taken to achieve the goal*]

The Task Category is based on the three perspectives (Data, Acoustic Parameter and Final Sound) and three views (users, application and interaction). The Task Goal describes what the task needs to achieve; and the Task Type describes an action to be taken to achieve the goal. Below are some examples of how to decompose an Abstract Process task into its user, application and interaction tasks in each perspective. We will continue using the AVATCSS application as the example, which will be in *italics*.

Data Perspective

The Data Perspective helps us to focus on where the 'Raw Data' is transformed into 'Processed Data'. The transformation tasks involve Data-User Tasks, Data-Interaction Tasks and Data-Application Tasks. Table 6-1 shows examples of the sort of questions that will help in the decomposition of Abstract Process tasks in the Data Perspective.

Table 6-1: Tasks for Data Transformation

Task Category	Example of Questions
Data-Application Task	1. How will the 'application' help to achieve the goal of the Abstract Process task it supports? 2. How will the 'application' transform the data (*RawData*) into its new states of data (*Processed Data*)?
Data-Interaction Task	1. How will the user interacting with the system help to achieve the goal the Abstract Process task it supports? 2. What do users need to do (and how) with the *Raw Data* and the *Processed Data?*
Data-User Task	1. What do users have to do to achieve the goal of the Abstract Process task it supports? 2. What do users need to know from the *Raw Data* as the input of the data transformation? 3. What do users need to know from the *Processed data* as the output of the data transformation? 4. What do users need to know about 'interaction' design in data transformation? 5. What do users need to know about how the 'application' changes the Raw data into the *processed data?*

The example task decomposition in AVATCSS below is based on the Abstract Process tasks for data transformation from Figure 6-5.

Task Category: *Data-User Task*

Task Goal: *to observe where to explore the image and the area that has not been scanned or inspected yet.*

Task Type: *to observe*

Task Category: *Data-Interaction Task*

Task Goal: *to allow a smooth transition of image movement so that it feels like looking, zooming and moving the image like a real microscope*

Task Type: *to move*

Task Category: *Data-Application Task*

Task Goal:

- *to identify the position and coordinate of the inspected area so that it could be dimmed if the user sets to dim a scanned area*
- *To display the selected 'field of view' of the digital image.*

Task Type: to identify, to display

Acoustics Parameters Perspective

The Acoustic Parameters Perspective helps us to focus on where the 'Processed Data' is transformed into 'Acoustically-prepared Data'. The transformation tasks involve Acoustic-Parameters-User Tasks, Acoustic-Parameters-Interaction Tasks and Acoustic-Parameters-Application Tasks. Table 6-2 shows examples of the sort of questions that will help in the decomposition of Abstract Process tasks from the Acoustic Parameters Perspective. The questions are more or less the same as for Data perspective, but the focus is more on Acoustic Parameters transformation.

Table 6-2: Tasks for Acoustic Parameters Transformation

Task Category	**Example of Questions**
Acoustic-Parameters-Application Task	1. How will the 'application' help to achieve the goal of the abstract task it supports? 2. How will the application transform the data *(Processed Data)* into *Acoustically-prepared Data?*
Acoustic-Parameters-Interaction Task	1. How will the 'interaction' help to achieve the goal of the abstract task it supports? 2. What do users need to do (and how) with the Processed Data and Acoustically-prepared Data?
Acoustic-Parameters-User Task	1. What do users have to do to achieve the goal of the abstract task it supports? 2. What do users need to know about the *processed data* as the input for the Acoustic Parameters Transformation? 3. What do users need to know about the Acoustically-prepared Data as the output of the Acoustic Parameters Transformation? 4. What do users need to know about the interaction design in the Acoustic Parameters Transformation? 5. What do users need to know about how the application changes the *Processed Data* into *Acoustically-prepared Data?*

The example task decomposition in AVATCSS below is based on the Abstract Process tasks for data transformation from Figure 6-5.

Task Category: *Acoustic-Parameters-User Task*

Task Goal: *to identify the level of rhythm based on the previously played sound.*

Task Type: *to identify*

Task Category: Acoustic-Parameters-Interaction Task[1]

Task Goal: to allow the user to optimise the rhythm levels for ease of differentiation.

Task Type: to customize

Task Category: Acoustic-Parameters-Application Task

Task Goal: to obtain the rhythm for band value (y-axis) of B feature.

Task Type: to obtain

Sound Representation Perspective

The Final Sound Perspective helps us to focus on where the 'Acoustically-prepared Data' is transformed into the 'Final Sound'. The transformation tasks involve Final-Sound-User Tasks, Final-Sound-Interaction Tasks and Final-Sound-Application Tasks. Table 6-3 shows examples of the sort of questions that will help in the decomposition of Abstract Process tasks in the Final Sound Perspective. The questions are more or less the same as for Data and Acoustic Parameters perspectives, but the focus is more on Final Sound transformation.

Table 6-3: Tasks for Final Sound Transformation

Task Category	Example of Questions
Final-Sound-Application Task	1. How will the 'application' help to achieve the goal of the abstract task it supports? 2. How will the application transform the Acoustically-prepared Data into the Final Sound?
Final-Sound -Interaction Task	1. How will the 'interaction' help to achieve the goal of the abstract task it supports? 2. What do users need to do (and how) with the Acoustically-prepared Data and Final Sound?
Final-Sound -User Task	1. How will the 'users' help to achieve the goal of the abstract task it supports? 2. What do users need to know about the Acoustically-prepared Data as the input for the Final Sound Transformation? 3. What do users need to know about the Final Sound as the output of the Final Sound Transformation? 4. What do users need to know about the interaction design in the Final Sound Transformation? 5. What do users need to know how the application changes Acoustically-prepared Data into Final Sound?

[1] There was no suggestion by the designer for the Acoustic-Parameters-Interaction task. However, for the purpose of this example, an interaction could be added at this level by allowing the user to customize the level of rhythm.

The example task decomposition in AVATCSS below is again based on the Abstract Process tasks for data transformation from Figure 6-5.

Task Category: *Final-Sound-User Task*

Task Goal:

- *the user should be able to recognize the timbre 1, which represents a normal cell.*
- *to observe the sound thickness, level of pitch and type of rhythm.*

Task Type: *to recognize, to observe*

Task Category: *Final-Sound-Interaction Task*[2]

Task Goal: *the user must be easily able to either enable or disable any acoustic parameter.*

Task Type: *to set*

Task Category: *Final-Sound-Application Task*

Task Goal: *to successfully render a sound for normal cell with its chorus effect.*

Task Type: *to render*

---

[2] There was no interaction involved the example of sub-transformation-process task, however, for the purpose of this example, the user could be allowed to enable and disable the acoustic parameters for the sound.

Table 6-4: Tasks Analysis for AVATCSS

	Data Perspective	Acoustic Parameters Perspective	Final Sound Perspective
Abstract Process Tasks	*Task Category: Data Transformation* *Task goal: to display the image or some part of the image (field of view) that is selected by the user; and to allow the user to explore a 'zoomed in' image.* *Task type: to display* *Input(s): digital image slide* *Output(s): zoomed in image, field of view,*	*Task Category: Acoustic Parameters Transformation* *Task goal: to map the current B feature value of inspected cell to a rhythm based on the list of 10 predefined rhythms.* *Task type: to map* *Input(s): Inspected Cell, Current B feature value, list of 10 predefined rhythms* *Output(s): Customised rhythm list, A rhythm from the list of 10 predefined rhythms (rhythm for B feature value)*	*Task Category: Final Sound Transformation* *Task goal: To combine all acoustic parameters including timbre 1, pitch of A feature, rhythm of B feature, delay and number of channel to produce a sound of normal cell section* *Task type: to combine* *Input(s): timbre 1, pitch of A feature, rhythm of B feature, delay and number of channel* *Output(s): sound of normal cell section*
User Tasks [Tasks that are entirely performed by the user without interacting with the system]	**A** *Task Category: Data-User Task* *Task Goal: to observe where to explore the image and the area that has not been scanned or inspected yet.* *Task Type*: to observe	**B** *Task Category: Acoustic-Parameters-User Task* *Task Goal: to identify the level of rhythm based on the previously played sound.* *Task Type: to identify*	**C1** *Task Category: Final-Sound-User Task* *Task Goal:* • *the user should be able to <u>recognize</u> timbre 1, which represents a normal cell.* *Task Type: to recognize* **C2** *Task Category: Final-Sound-User Task* *Task Goal:* • *to <u>observe</u> the sound thickness, level of pitch and type of rhythm.* *Task Type: to observe*
Interaction Tasks *[tasks performed by the users with the system Or tasks that are performed by the system due to the interaction or instruction from the user]*	**D** *Task Category: Data-Interaction Task* *Task Goal: to allow a smooth transition of image movement so that it feels like looking, zooming and moving the image on a real microscope* *Task Type: to move*	**E** *Task Category: Acoustic-Parameters-Interaction Task* *Task Goal: to allow the user to customize the rhythm to levels that are easy to differentiate.* *Task Type: to customize*	**F** *Task Category: Final-Sound-Interaction Task* *Task Goal: the user can easily enable or disable any acoustic parameters.* *Task Type: to set*

Application Tasks [Tasks that are performed by the system without user intervention]	**G1** ***Task Category:*** *Data-Application Task* ***Task Goal:*** • *To identify the position and coordinates of inspected area so that it could be dimmed if the user sets to dim a scanned area.* ***Task Type:*** *to identify* **G2** ***Task Category:*** *Data-Application Task* ***Task Goal:*** • *To display the selected 'field of view' of the digital image.* ***Task Type:*** *to display*	**H** ***Task Category:*** *Acoustic-Parameters-Application Task* ***Task Goal:*** *to obtain the rhythm for band value (y-axis) of B feature.* ***Task Type:*** *to obtain*	**I** ***Task Category:*** *Final-Sound-Application Task* ***Task Goal:*** *to successfully render a sound for normal cell with its chorus effect.* ***Task Type:*** *to render*

In summary, the results of the task decomposition above are shown in Table 6-4. As for the materials of the proposed inspection technique, these results are used to produce a 'List of goals' (*goals of abstract tasks and their breakdown into user, application and interaction tasks*) and 'Task Data State diagram' (*uses all task type and its input and output*).

The List of goals is presented in 'purpose' and 'issues'. The 'purpose' describes what is it required to do, which is related to the goal of the Abstract process task. The 'issues' describe the qualities that are important for the goal to be achieved, which relates to the goals of the user, application and interaction tasks.

For example in AVATCSS (referring to Table 6-4), the purpose and issues for one of the Abstract Process tasks in Data Perspective are as follow:

Purpose:

To display the image or some part of the image (field of view) that is selected by the user; and to allow the user to explore a 'zoomed in' image.

Issues:

User Task:

- *to observe where to explore the image and the area that has not been scanned or inspected yet.*

Application Task:

- *to allow a smooth transition of image movement so that it feels like looking, zooming and moving the image on a real microscope*

Interaction Task:

- *To identify the position and coordinate of inspected area so that it could be dimmed if the user sets to dim a scanned area.*
- *To display the selected 'field of view' of the digital image.*

To create the TDS diagram of this application, we need to determine the temporal relationship between the tasks. This includes the temporal relationship 'between Abstract Process tasks' and 'between sub-tasks (users, application and interaction)'. As a reminder from Chapter 5, the temporal relationships are adopted from the CTT diagram (Paterno et al., 1997), which include 'interleaving (| | |)', 'enabling (>>)', 'synchronization (|[]|)', 'enabling with information passing ([]>>)', 'deactivation ([>)' and 'iteration(*).

Referring to the tasks in Table 6-4, below are some example temporal relationships between the tasks in each sub-transformation-process.

Data Perspective:

- *Tasks A is required to do Task D;*
- *Task G1 requires inputs from Task D;*
- *Task G1 and G2 will only occur after Task D;*
- *Tasks G1 and G2 can be done in any order.*
- *Therefore, Task A >> Task D []>> Task G1 ||| Task G2.*

Acoustic Parameters Perspective:

- *Task H requires input from Task E;*
- *Task B can only be done after Tasks E and H;*
- *Therefore; Task E []>> Task H []>> Task B.*

Final Sound Perspective:

- *Task I requires input from Task F;*
- *Tasks C1 and C2 can only be done after Tasks F and I;*
- *Tasks C1 and C2 can be done in any order;*
- *Therefore; Task F []>> Task I []>> Task C1 ||| Task C2*

The next section will explain how to create the TDS diagram from results of the above task analysis.

6.2.1.3 Creating the Task-Data State (TDS) diagram for a Sonification Application

As a reminder from Chapter 5, the notations for the TDS diagram are as shown in Figure 6-6.

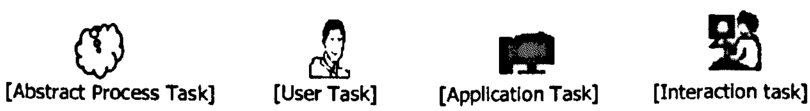


Figure 6-6: Notations for the TDS diagram

Figure 6-7 shows the example of TDS diagram based on the tasks, input and output from Table 6-4. The diagram shows the macro level of the TDS diagram where the tasks are only represented by the Abstract Process Tasks. The dotted line 'before the input' and 'after the output' of an Abstract Process tasks represent 'the input is produced by the previous Abstract Process task' and ' the output will become the input for the next Abstract Process task' respectively.

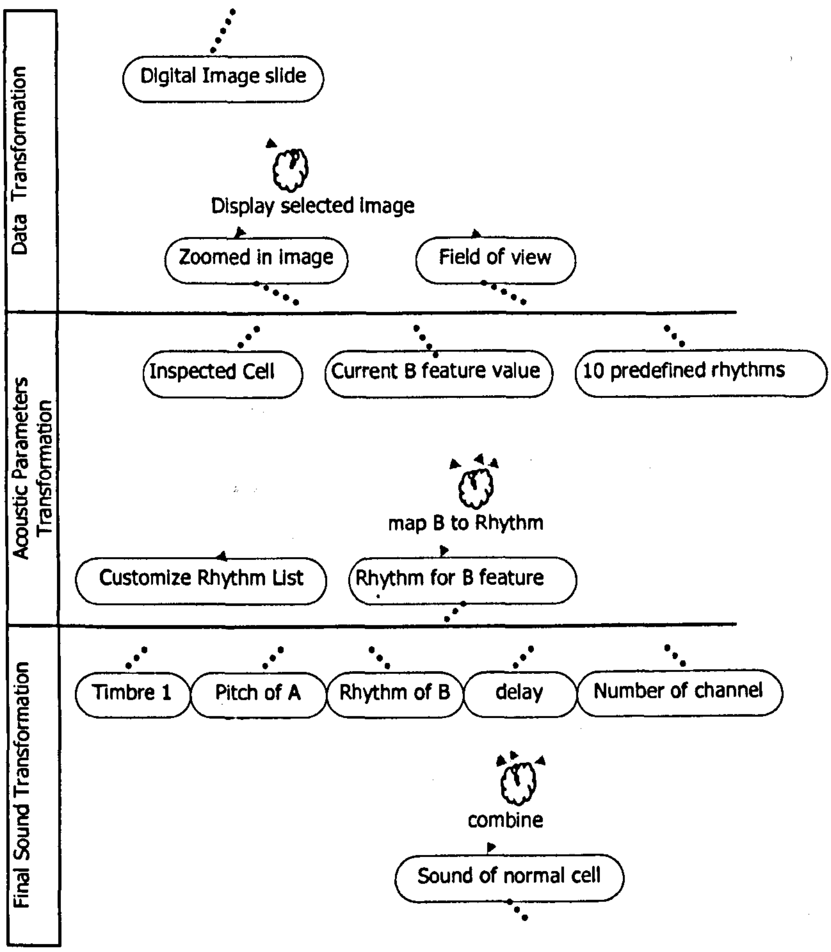


Figure 6-7: Macro level view of the TDS diagram for the example from Table 6-4

Figure 6-8 shows how the macro level views (abstract process tasks) from Figure 6-7 are decomposed into detail (user, application and interaction tasks) views within the TDS diagram. The macro level views can be used to describe in general how the application transforms raw data into its final sound representation. From top to bottom, the diagram shows the different states of data and the Abstract Process tasks involved. The complete diagram for the AVATCSS example can be found in Appendix 3.

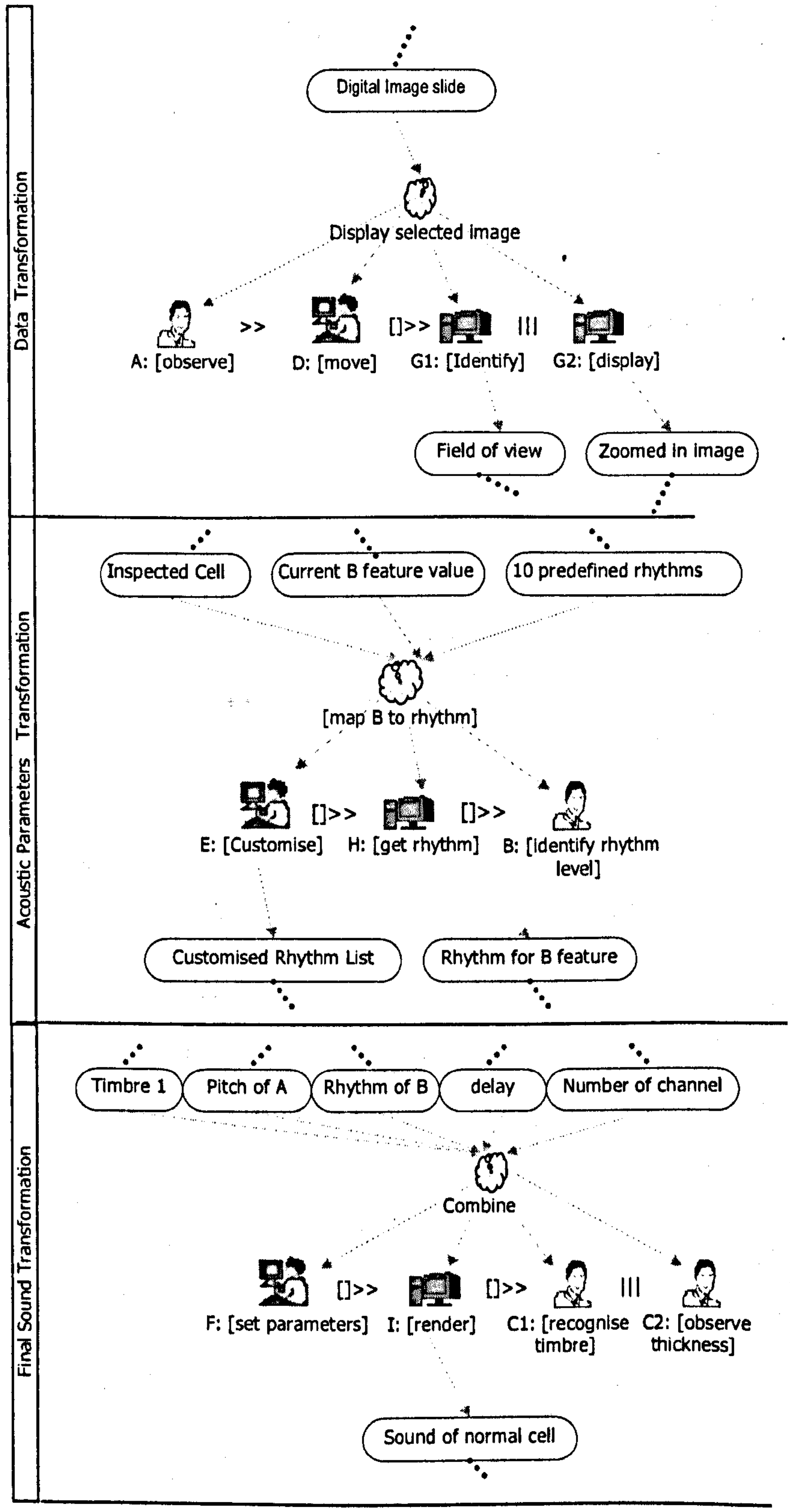


Figure 6-8: Decomposition of the Macro level view of the TDS diagram

Both the 'macro' and 'detail' levels of the TDS diagram for a sonification application design will be used as part of the inspection materials within the TIW inspection technique.

In general, the TDS diagram describes the solution provided by the sonification application from the designer's point of view.

Next, we will look at how and what the *user* might perceive and understand from the design.

6.2.2 Analyse Potential User Interpretations

The potential user interpretations of the sound produced by a sonification application are explained based on our User Interpretation Construction (UIC) model. This model focuses on the way the users should interpret the application's outputs. This is in contrast to the TDS diagram which focuses on 'what the application offers'. As a reminder from chapter 4, this model introduces three stages of how the sound representation will be interpreted into a useful mental representation; *selection, reasoning* and *hypothesizing.* Each stage produces *conditions, premises* and *hypothesises* respectively.

Each of Abstract Process tasks from the TDS diagram especially their input(s) and output(s) will be used in this potential user interpretation analysis. We will continue to explain these activities through the example of AVATCSS, which will be shown in *italics.* The Abstract Process task, input and output are based on Table 6-4.

Selection stage

At this stage, we want to think about how the users will discriminate the potentially useful output. There might be several output states that are important to the user and these are referred to as *conditions*. A condition is a mode or state of data at a particular time, which is potentially valuable and important to the users for the purpose of interpretation. Therefore, the chief inspectors together with the designer will try to filter out and attend only to the potentially significant outputs of Abstract Process tasks. They are asked merely to consider the observable artefacts that are potentially important, *without* taking into account the user's experience or knowledge. An example of the sort of question, which they will have to answer, is:

- What are the observable artefacts from the input and output of a particular Abstract Process task that are potentially important and valuable to the user?

Data Perspective:

Goal of Abstract Process task: *To display the image or some part of the image (field of view) that is selected by the user; and to allow the user to explore a 'zoomed in' image.*

Input(s): *digital image slide*

Output(s): *'zoomed in image', 'field of view'.*

Conditions:

Based on the goal, input and output, below are the observable artefacts that are potentially important to the user:

- *current image appearing on screen*
- *scanned image area*
- *zoomed in image*
- *'explorable' image*
- *selected area*
- *field of view*

Acoustic Parameters Perspective:

Goal of Abstract Process Tasks: *to map the current B feature value of inspected cell to a rhythm based on the list of 10 predefined rhythms.*

Input(s): *Inspected cell, current B feature value, list of 10 predefined rhythms.*

Output(s): *customised rhythm list, rhythm of B feature value*

Conditions:

- *Inspected cell*
- *Current B feature value*
- *10 different rhythms*
- *new customised rhythm*
- *Beat of rhythm (slow and fast)*

<u>Final Sound Perspective:</u>

Gaol of Abstract Process Tasks: *to combine all acoustic parameters including timbre 1, pitch of A feature, rhythm of B feature, delay, and number of channel to produce a sound of normal cell.*

Input(s): *timbre 1, pitch of A feature, rhythm of B feature, delay and number of channel.*

Output(s): *sound of normal cell section.*

Conditions:

- *Number of repeated sounds*
- *Delay of sounds*
- *Type of Timbre*
- *Level of pitch*
- *Type of rhythm*

Reasoning stage

In this stage, several conditions are used to construct, arrange or put together a *premise*. This premise is a statement to support the reason why the conditions are being selected and why they are potentially important. It is also describes what is being represented by the conditions. This can be done by using the available conditions to make a logical judgment. Some examples of questions to be asked at this stage are:

- Why are the conditions important to the user?
- What is represented by the conditions?

Continuing with our AVATCSS example. From the previously stated conditions, below are the reasons why those conditions are potentially important.

<u>Data Perspective</u>

Premises:

- *The image can be moved and scanned only in zoom-in mode.*
- *The Field of view will only display the zoomed image.*
- *The Cell in the current image closest to the centre of the screen is the one being inspected.*
- *The image is explorable only in 'zoomed in' mode.*
- *The areas already scanned (checked) should be able to be differentiated from those areas not yet checked.*
- *The Field of view displays only part of the overall image.*

Acoustic Parameters Perspective

Premises:

- *Rhythms are differentiated by their beat speed.*
- *On a scale of 1 to 10, the rhythms increase from a slow to a fast beat*
- *The rhythm is based on a value from the inspected cell*
- *The higher the value the faster the rhythm*

Final Sound Perspective

Premises:

- *The sound will be thicker and more complex as the number of sound channels increases.*
- *Adding delay to the sound will increase its complexity.*
- *If the sound of timbre 1 is thicker and more complex than the other sound timbre, it is more likely that the cell being sonified is normal.*
- *If the sound of timbre 1 has no chorus effect at all, there is no normal cell found in the square box.*
- *If there is no chorus effect, the trend of pitch level and type of rhythm can be observed and compared with the previous sound – as a clue for the user.*

Hypothesizing stage

This stage allows us to conceptualize, make sense of, or conceive the significance of the premises by relating them to the user's knowledge, previous experience or even their instinct. Several premises are combined to form a *hypothesis,* which describes the overall idea of the relationship between the outputs, conditions and premises. Examples of questions at this level are:

- What potential knowledge could the user pick up from the premises?
- What can the user learn from the premises?
- What is general idea from the list of premises?

Data Perspective

Hypothesis:

- *the image can only be explored if it is in zoomed-in mode.*
- *The field of view will only display part of the zoomed-in image.*

Acoustic Parameters Perspective

Hypothesis:

- *the 10 levels of rhythm can be differentiated based on beat speed.*
- *The higher the value, the faster the rhythm and vice versa.*

Final Sound Perspective

Hypothesis:

- *The more complex and thicker the sound of timbre 1, the higher the possibility of the cell being sonified being normal.*
- *If the sound of timbre 1 is complex and thick, it does mean that the cell being sonified is a normal cell.*
- *In a condition wher, there are no reference cells in the square box, the level of pitch and rhythm can be used as a clue for deciding whether or not it is more like a normal or abnormal cell.*

Each Abstract Process task as well as its input(s) and output(s) needs to be considered through all the above three stages to obtain its conditions, premises and hypotheses. These results will be used in the inspection materials for the TIW inspection technique.

Next, we will look at the application context of use.

6.2.3 Contexts of Use

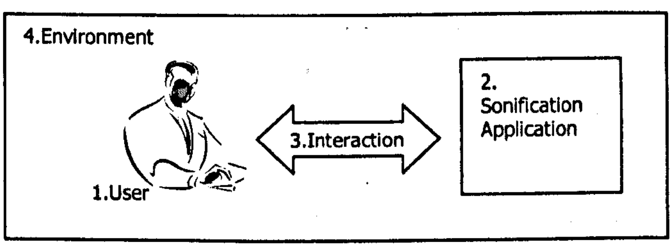


Figure 6-9: Context of Use Components

In terms of sonification applications, 'Contexts of Use' refer to a boundary of who is using the application; how the application will be used; what the application offers and where the application will be used. They are based on the basic components of user interaction with the application. These include the characteristics of **user (who)**, **interaction (how)** and **application (what)** as well as the situation and **environment (where)** where the application will be used, as illustrated in Figure 6-9. These contexts are suited to the focus of the proposed inspection technique (TIW), which gives more attention to the user, application and interaction tasks for each Abstract Process task. These contexts are important for the inspector to know so that the application is assessed fairly and appropriately. This information also provides contextual validity of any problems or anomalies found by the inspector.

Baven and Macleod (1994) and Maguire (2001) refer the contexts of use in four components, namely Users, Task, Equipment and Environment. Some characteristics from these existing components will be used as the criteria in the contexts of use for TIW.

Generally, we use the following four contexts of use- namely (1) users and user tasks; (2) application tasks, equipment and input/output; (3) Interface and Interaction; and (4) Environment. Further examples of each of these contexts are given below·

User and User Tasks Context

This context describes some information about the user background, experiences, knowledge and skills that are related to the application to be inspected. The user tasks can be referred to from the TDS diagram of the application to be inspected. The descriptions are based on the average or typical user. For example:

- User personality – describes the quality of the user such as age, physical capabilities and limitations.
- User Experiences – describes user experiences related to the application to be inspected.
- Users Knowledge – describes user task knowledge and domain knowledge related to the application to be inspected.
- User Skills – describes user's previous skills, motor skills, input device skill and requirements for training.
- User Perceptual system – describes the condition of the user's senses such as vision and listening.
- User task – describes the task that the user needs to accomplish.

Returning to our example for AVATCSS;

Criteria	Descriptions
User Personal attributes [*Age, gender ; Physical capabilities and limitations; Attitude and motivation*]	(1) Cytoscreeners, advanced practitioners and pathologists (2) User's age can be anything from mid-20's to retirement age. (3) The users are highly motivated towards the correct inspection of the slides. Their attitude is a priori highly skeptical as to the benefits of the audio system, but not hostile.
User's Practical Experiences *[Previous and related experiences]*	(1) The users are professionals trained in the visual inspection of cervical smears (2) User's experience in spotting visual patterns and using those as classification clues may be transposable to auditory clues (3) Users are experienced at navigating through a slide by means of microscope knobs (4) Users use computers for data entry (5) Some users enjoy listening to music or the radio during their work
User's Knowledge and cognitive system *[Previous and related knowledge]*	(1) The user might or might not have any formal or informal knowledge of music (2) They have formal knowledge of the application domain (e.g. pathology)
User's Skill and Motor System *[Previous and related Skills]*	(1) The user has no problem using a keyboard, mouse, knobs and buttons of a reasonable size
User's Perceptual system	(1) The user has no hearing problem (2) The user has no problem in visual perception e.g. colour blind.
User's tasks	**Please refer to the user tasks in the Tasks-Data State Diagram.**

Application Tasks, Equipment and Input/Output

This context describes the application; its inputs and outputs; and the equipment required by the application to be inspected. Some examples of relevant equipment are hardware, other software packages, and computer networks. The application and input/output can be referred to in the TDS diagram of the application to be inspected.

Continuing the example for AVATCSS;

Criteria	Descriptions
Application tasks	**Please refer to the application tasks in the Task-Data State Diagram.** Basically, (1) This application is to provide a cytologist (who is visually screening a smear slide) with complementary information with which to make his/her diagnostic. This information is provided aurally simultaneously to the visual inspection. (2) Might require the user to get used to the sonic metaphor and to choose his/her favourite settings. (3) This application will have two states: a. The pre-processing state where the slide is processed and the images analyzed. b. The live stage during the slide screening when the audio is rendered.
Equipment and Technical *[Hardware; Software; Network; Reference materials; other equipments]*	(1) In reality, the cells are fixed onto a glass slide, ready for inspection via a microscope, but here we will consider the situation where we use digital images of sections of slides (fields of view). (2) A computer interface (GUI) will be used at this preliminary stage for the visualization/navigation of the images and the synchronization of the visual field with the audio field (3) In-ear headphones will deliver the audio (4) Mouse/keyboard/joystick etc might be used to navigate through the images
Input/Output	**Please refer to the state of data in the Task-Data State Diagram**

Interface and Interaction

This context refers to the interface and interaction design of the application. The interface includes the graphical user interface, which (for the purposes of inspection) can be illustrated by a drawing of graphical windows, buttons and menus (a technique known as Paper Prototyping). Other interfaces to be considered could be haptic where a physical device is required to interact with the application. The details of the interaction can be inferred from the interaction tasks in the TDS diagram.

Again, continuing the example for AVATCSS;

Criteria	Descriptions
User Interaction *[objects; actions/steps; duration; constraints etc.]*	**Please refer to the interaction tasks in the Task-Data State Diagram.** 1. input is probably via the mouse and a toolbox menu 2. outputs are cell images and sound 3. inspection of each field of view can be expected to last no longer than 3 seconds 4. the user needs to be able to mark and save the image when problem areas have been identified
Other Interfaces *[e.g. Graphical interfaces]*	**- refer to attachment**

Environment

This context refers to either the physical environment or the social environment. Physical environment describes the realistic situation and conditions such as the workplace conditions and location. The organization and social environment includes culture and working structure.

Continuing the example for AVATCSS;

Criteria	Descriptions
Physical environment *[Workplace conditions Auditory environments Atmospheric conditions, Location, safety equipment, etc.]*	(1) The user will use the application while sitting at a desk (2) The user will be busy on an intense visual task while receiving the application audio output (3) The user will be in a quiet environment.
Organization/social environment *[Structure; Group working; Work practices Assistance; Interruptions; Communication structure; Attitude and culture]*	(1) The user is unlikely to be interrupted but will take regular, short, breaks at his/her workstation and regular/longer breaks, doing other tasks (2) Some interesting slides are presented at group meetings. It could be conceived that the audio output could be valuable in such a setting too.

All these four contexts of use give the inspector a defined scope and enable them to evaluate the application practically. As an example of AVATCSS above, in the context of 'User and User tasks' (under the sub-context of 'Users Perceptual System') - *'The users are presumed to have no hearing problem'*. By stating this, the inspector does not need to consider further a user who has hearing problems.

Next, we will look at other inspection materials that would help usability inspectors to understand the overall application design.

6.2.4 Other Inspection Materials

This section describes two other inspection materials that will be used during the inspection. These materials are described in 'other materials' because they are not prepared specifically for the proposed inspection technique but rather exist as part of the normal application development documentation. They are 'user requirements' and 'graphical interfaces' design.

6.2.4.1 User Requirements

In general, an application is developed to help users achieve their goals and to carry out certain tasks successfully. Therefore, it is important for the inspector to know what these goals and tasks are that the user wants to achieve. All these goals and tasks are from the user's point of view. This information should be gathered by the designer even before they start designing the application. The requirements can be gathered through interviews or questionnaires with the end users. Below are some examples of goals and tasks that the user might want to achieve in AVATCSS:

Goal 1: To produce (customized) optimum audio-visual settings

- *Task 1.1: Load and save image to be inspected*
- *Task 1.2: Modify Sound Settings*
- *Task 1.3: Modify Graphical and Navigation Settings*

Goal 2: To inspect a number of cells on a screen while listening to the auditory display of complex features of one or more interesting cells in the field of view

- *Task 2.1: Navigate graphical display and settle on a field of view*
- *Task 2.2: Visually inspect cells in the field of view*
- *Task 2.3: Listen to the audio and make a mental note of how 'normal' the cell sounds*
- *Task 2.4: Relate the audio to the visual field*
- *Task 2.5: Pause to mark any problem cell/cells for future reference*

6.2.4.2 Graphical Interfaces

Graphical interfaces for the application can be described by hand sketches on paper or graphic drawing on a computer screen. An example graphical interface for AVATCSS is shown in Figure 6-10.

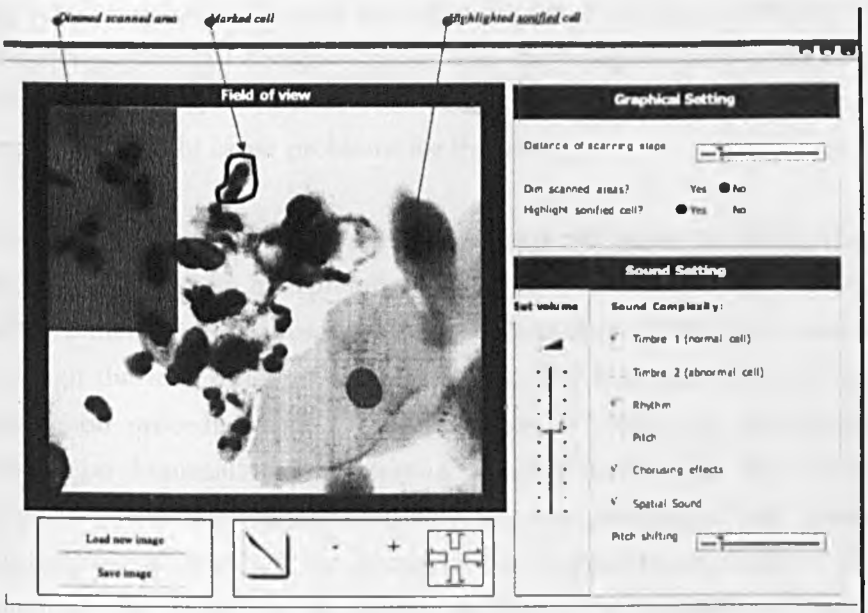


Figure 6-10: Example of Graphical Interface for AVATCSS

6.2.5 Summary of inspection materials

Section 6.2 has presented four types of inspection materials that will be used in the Task Interpretation Walkthrough (TIW) inspection technique. The materials include:

(1) Task Data State diagram;

(2) User Interpretation (list of conditions, premises and hypotheses);

(3) Contexts of Use; and

(4) other inspection materials [user requirements and graphical interfaces].

Next, we will look at how to carry out the new TIW inspection technique and how the above materials will be used.

6.3 Inspection Activities

As mentioned earlier in this chapter, the core idea of the proposed technique is to understand the design rationale of the sonification application. The inspection materials used by this technique give an idea of what the designer wanted to give the users, and their assumptions and expectations of what the user should understand. In other words, we try to understand how the designers rationalize the design of their application. If the designer's assumptions and expectations are inappropriate, the application might cause problems for the user.

The inspection process will critically inspect the goals of every Abstract Process task and its interpretation levels as well as the tasks required to achieve them. It is proposed to ask questions about these goals and tasks through the four different contexts of use. The inspector needs to follow inspection procedure and uses a given set of inspection materials. The Inspection Materials is therefore a package containing the necessary documents for inspections, such as 'inspection procedure' and 'problem writing forms' that will be given to the inspectors, as well as all the materials described in the above sections. The Inspection Procedure explains the rules and regulations on how to do the inspection.

6.3.1 Inspection Materials

The inspectors will be provided with an Inspection Materials package containing information and documents that will be used during inspection activities. The materials describe how the inspection should be done, and a description of the application being assessed. These include:

1) Task Data State diagram [explained in Section 6.2.1]
2) User Interpretations [explained in Section 6.2.2]
3) Contexts of use [explained in Section 6.2.3]
4) Other inspection materials e.g. User Requirements; sketches of graphical user interface; and any related and necessary documents such as User Requirements, etc. [explained in Section 6.2.4]
5) Inspection Procedure that describes step by step how to perform the inspection.

6) Feedback Form where the inspectors can write down the problems they discover.

6.3.2 Inspection Procedure

Below are the procedures to be followed by inspectors during an inspection session. The inspection is divided into two steps –
Step 1: General Inspection (looking at the *goals* and *tasks* on the TDS diagram)
Step 2: Thorough Inspection (a more detailed look at all the *sub-tasks, interpretation stages* and *context of use*).

At some points in the steps, the part that requires inspector to refer the inspection materials will be indicated as **<purpose>**, **<issues>**, **<condition>**, **<premise>** and **<hypothesis>**. As a reminder from the previous examples in this chapter, the <purpose> refers to the goal of Abstract Process task; the <issues> refer to the goal of user, application and interaction tasks; the <condition>, <premise> and <hypothesis> refer to the output of potential user interpretation.

Step 1 [General Inspection]

In this step, the inspector is required to go through the Abstract Process tasks using the Task Data State diagram of the application to be inspected. The materials required are simply the 'TDS diagram' and the 'list of user requirements'.

<u>Inspection Step 1.1</u>
This step becomes the checklist to ensure the application will fulfill the user requirements. For each of the user requirements, the inspector needs to step through the Abstract Process tasks of the TDS diagram and make sure that all the requirements are offered by the application. In the case that any requirements have not been offered, this should be reported.

<u>Inspection Step 1.2</u>
This step checks the important of the Abstract Process task. There could exist unnecessary Abstract Process tasks from the inspector's point of view. The inspector's reasoning here is important to be considered in the process of improving the design later on.

The inspector needs to step through the macro and detail levels of the TDS diagram. They need to check whether the goal of Abstract Process tasks is important and necessary to be performed. The goal is referred as **'purpose'**, which is explained in Section 6.2.1.2 (given in User Interpretation document during inspection). Should they find any potential problems, they need to report them. The inspector will be asked the following question:

Do you think the <**purpose**> (goal of Abstract Process task) is important and necessary for this application?

1. If NO, explain why it is not important. Describe the problem(s) if any.
2. If YES, do you think 'each sub-task' (User, Application and Interaction) of the Abstract Process task above is important?

For each unimportant sub-task, why it is not important? Describe the problem if any.

Step 2 [Thorough Inspection]

In this step, the inspector will examine the application in more detail. The inspection materials to be used include the TDS diagram, Contexts of Use, User Interpretation and other necessary materials (e.g., graphical user interfaces).

The inspector needs to walk through the Abstract Process task of the TDS diagram. In each Abstract Process task, the inspector will need to create several 'questions' based on the contexts of use and user interpretation, which will be explained in the next step. Based on the questions, the inspector is required to do the following:

a. Identify and detect any possible *cause*[3] of the potential problems that may hinder the effective, efficient and satisfying use of the application.
b. From the 'cause' above, look for any *potential failure stories* (effects[4]) that may influence the usability of the application;

[3] *The 'cause' is the potential error of the design that brings up the potential failure stories (effects).*
[4] *The 'potential failure stories' refer to the effects of the potential problems in the design that might happen to the user if they use the application later on*

For each Abstract Process task, the inspector needs to do the following:

<u>Inspection Step 2.1</u>

This step focuses on the 'USER TASKS' of the Abstract Process task. It requires the inspector to concentrate on the criteria from the Context of Use: Users and Users Tasks. For each Abstract Process task, its 'user task goal' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions to be used by the inspector to examine the design and detect any potential problems (could be the 'cause', 'effect' or both). The questions are:

From the criteria in the context of 'Users and Users Tasks', do you foresee any problem;

1. For the USER to achieve successfully the **<user tasks issues>**?
2. For the user to recognize, filter or attend only to the significant **<conditions>**?
3. For the user to construct, arrange or put together several conditions to form the **<premise>**?
4. For the user to make sense or conceptualize the **<hypothesis>**?

<u>Inspection Step 2.2</u>

This step focuses on the 'APPLICATION TASKS' of the abstract process task. It requires the inspector to concentrate on the criteria from the 'Context of Use: Application Task, Equipments and Input/Output'. For each 'abstract process tasks', its 'application task goal' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions to be used by the inspector to examine the design and detect any potential problems (could be the 'cause', 'effect' or both). The questions are:

From the criteria in the context of 'Application Task, Equipments and Input/Output', do you foresee any problem:

1. *For the APPLICATION to achieve successfully the* ***<application tasks issues>****?*
2. *That could influence the user to recognize, filter or attend only to the significant* ***<conditions>****?*
3. *That could influence the user to construct, arrange or put together several conditions to form the* ***<premise>****?*

4. *That could influence the user to make sense or conceptualize the **<hypothesis>**?*

Inspection Step 2.3

This step focuses on the 'INTERACTION TASKS' of the abstract process task. It requires the inspector to concentrate on the criteria from the 'Context of Use: Interface and Interaction'. For each 'abstract process task', its 'interaction task goal' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions to be used by the inspector to examine the design and detect any potential problems (could be the 'effect', 'cause' or both). The questions are:

From the criteria in the context of 'Interface and Interaction', do you foresee any problem;

1. *For the INTERACTION Design to achieve successfully the **<interaction tasks issues>**?*
2. *That could influence the user to recognise, filter or attend only to the significant **<conditions>**?*
3. *That could influence the user to construct, arrange or put together several conditions to form the **<premise>**?*
4. *That could influence the user to make sense or conceptualize the **<hypothesis>**?*

Inspection Step 2.4

This step focuses on the '**USER, APPLICATION AND INTERACTION TASKS**' of the abstract process task. It requires the inspector to concentrate on the criteria from the '**Context of Use: Environment**'. For each 'abstract process task', its 'user , application and interaction task goals' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions to be used by the inspector to examine the design and detect any potential problems (could be the 'cause', 'effect' or both). The questions are:

From the criteria in the context of 'Environment', do you foresee any problem;

1. *For all the [USER, APPLICATION AND INTERACTION Design] to achieve successful **<their issues respectively>**?*
2. *That could influence the user to recognise, filter or attend only to the significant **<conditions>**?*

3. *That could influence the user to construct arrange or put together several conditions to form the* ***<premise>****?*
4. *That could influence the user to make sense or conceptualize the* ***<hypothesis>****?*

All the problems found will be rated by inspector using a severity level (1 to 4) [adapted from Nielsen (1990)]. This rating is applied to prioritize the problems encountered. This is especially useful in deciding which problem is most critical and thus needs to be resolved first. The levels are:

1 = cosmetic problem - only needs to be fixed if extra time is available on project
2 = minor usability problem – fixing this should be given low priority
3 = major usability problem – important to fix, so should be given high priority
4 = usability catastrophe – imperative to fix this before application can be released

The inspector will need to repeat Step 2 (consisting of Inspection Steps 2.1, 2.2, 2.3 and 2.4) for every 'abstract process task' from the TDS diagram. Each potential problem found by inspectors needs to be written down along with its severity level.

6.3.3 Example of Questions and Problems

This section gives several examples of questions and potential problems that can be generated and found by TIW. The examples are still based on the inspection of AVATCSS application, which is the Experiment IV in this thesis. The full examples of questions and potential problems identified during the Experiment IV can be found in Appendix C and F respectively. To consider the questions below, inspectors should also refer to the information in the Context of Use.

Example of the First Questions for Inspection Step 2.1

Data Perspective

- *In terms of the user's personal attributes; practical experience; knowledge and cognitive system; skill and motor system; perceptual system; and user's tasks, do you foresee any problems for the user to decide and observe where to explore the image and know the area that has not been scanned/inspected yet?*

Acoustic Parameters Perspective

- *In terms of the user's personal attributes; practical experience; knowledge and cognitive system; skill and motor system; perceptual system; and user's tasks, do you foresee any problems for the user to identify the level of rhythm based on the previously played sound?*

Final Sound Perspective

- *In terms of the user's personal attributes; practical experience; knowledge and cognitive system; skill and motor system; perceptual system; and user's tasks, do you foresee any problems for the user to observe the sound thickness, level of pitch and type or rhythm at the same time?*

Example of the First Questions for Inspection Step 2.2

Data Perspective

- *In terms of application tasks; equipment and technical; and input/output do you foresee any problems for the application to identify the position and coordinate of the inspected area and display the selected 'field of view' of the digital image?*

Acoustic Parameters Perspective

- *In terms of application tasks; equipment and technical; and input/output do you foresee any problems for the application obtain the rhythm for band value (y-axis) of B features?*

Final Sound Perspective

- *In terms of application tasks; equipment and technical; and input/output do you foresee any problems for the application to successfully render a sound for normal cell with its chorus effect?*

Example of the First Questions for Inspection Step 2.3

Data Perspective

- *In terms of user interaction and graphical user interface, do you foresee any problems for the user to move the image using the arrow key with smooth transition so that they feel like looking, zooming and moving the image is similar to what they normally do with a microscope?*

Acoustic Parameters Perspective

- *In terms of user interaction and graphical user interface, do you foresee any problems for the user customize the rhythm to levels that are easy to differentiate?*

Final Sound Perspective

- *In terms of user interaction and graphical user interface, do you foresee any problems for the user to change/set the final sound setting such as to enable or disable any acoustics parameters?*

Example of the First Questions for Inspection Step 2.4

In terms of physical and social environments, do you foresee any problems:

- *for the user to decide where to explore the image and know that the area has not been scanned/inspected yet?*
- *for the user to identify the level of rhythm based on the previously played sound?*
- *for the user to observe the sound thickness, level of pitch and type or rhythm at the same time?*
- *for the application to identify the position and coordinate of the inspected area and display the selected 'field of view' of the digital image?*
- *for the application obtain the rhythm for band value (y-axis) of B features?*
- *for the application to successfully render a sound for normal cell with its chorus effect?*
- *the user to move the image using the arrow key with smooth transition so that they feel like looking, zooming and moving the image is similar to what they normally do with a microscope?*
- *for the user customize the rhythm to levels that are easy to differentiate?*
- *for the user to change/set the final sound setting such as to enable or disable any acoustics parameters?*

Example of the Second Questions for Selection Stage

Question 2 for Inspection Step 2.1

- *In terms of the user's personal attributes; practical experience; knowledge and cognitive system; skill and motor system; perceptual system; and user's tasks, do you foresee any problems for the user to recognize the following: (refer to the list of conditions for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 2 for Inspection Step 2.2

- *In terms of application tasks; equipment and technical; and input/output, do you foresee any problems that could influence the user to recognize the following: (refer to the list of conditions for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 3 for Inspection Step 2.3

- *In terms of interaction and graphical interface, do you foresee any problems that could influence the user to recognize the following: (refer to the list of conditions for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 2 for Inspection Step 2.4:

- *In terms of physical and social environments, do you foresee any problems that could influence the user to recognize the following: (refer to the list of conditions for Data, Acoustic Parameters and Final Sound Perspectives below)*

List of Conditions for Data Perspective

- *Current image appear on screen; scanned image area; zoomed in mode; 'explorable' image; selected area and field of view*

List of Conditions for Acoustic Parameters Perspective

- *Inspected cell; Current B feature value; 10 different rhythms; new customised rhythm; Beat of rhythm (slow and fast).*

List of Conditions for Final Sound Perspective

- *Number of repeated sounds; Delay of sounds; Type of Timbre; Level of pitch; Type of rhythm.*

Example of the Third Questions for Reasoning Stage

Question 3 for Inspection Step 2.1:

- *In terms of the user's personal attributes; practical experience; knowledge and cognitive system; skill and motor system; perceptual system; and user's tasks, do you foresee any problems for the user to know and understand that; (refer to the list of premises for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 3 for Inspection Step 2.2:

- *In terms of application tasks; equipment and technical; and input/output, do you foresee any problems that could influence the user to know and understand that: (refer to the list of premises for Data, Acoustic Parameters and Final Sound perspectives below)*

Question 3 for Inspection Step 2.3:

- *In terms of interaction and graphical interface, do you foresee any problems that could influence the user to know and understand that: (refer to the list of premises for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 3 for Inspection Step 2.4:

- *In terms of physical and social environments, do you foresee any problems that could influence the user to know and understand that: (refer to the list of premises for Data, Acoustic Parameters and Final Sound Perspectives below)*

List of Premises for Data Perspective

- *the image can be moved and scanned only in zoom-in mode.*
- *Field of view will only display the zoomed image.*
- *Cell in the current image appear on screen is the one being inspected.*
- *The image is only explorable in 'zoomed in' mode.*
- *Scanned (checked) area should be able to be differentiated*
- *Field of view displays only part of the image.*

List of Premises for Acoustic Parameters Perspective

- *Rhythms are differentiated by their beat speed.*
- *On a scale of 1 to 10, the rhythms increase from a slow to a fast beat*
- *The rhythm is based on a value from the inspected cell*
- *The higher the value the faster the rhythm*

List of Premises for Final Sound Perspective

- *The sound will be thicker and more complex as the number of sound channels increases.*
- *Adding delay to the sound will increase its complexity.*
- *If the sound of timbre 1 is thicker and more complex than the other sound timbre, it is more likely that the cell being sonified is normal.*
- *If the sound of timbre 1 has no chorus effect at all, there is no normal cell found in the square box.*

- *If there is no chorus effect, the trend of pitch level and type of rhythm can be observed and compared with the previous sound – as a clue for the user.*

Example of the Fourth Questions for Hypothesising Stage

Question 4 for Inspection Step 2.1:

- *In terms of the user's personal attributes; practical experience; knowledge and cognitive system; skill and motor system; perceptual system; and user's tasks, do you foresee any problem for the user to make sense and conceptualize the following: (refer to the list of hypotheses for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 4 for Inspection Step 2.2:

- *In terms of application tasks; equipment and technical; and input/output, do you foresee any problem for the user to make sense and conceptualize the following: (refer to the list of hypotheses for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 4 for Inspection Step 2.3:

- *In terms of interaction and graphical interface, do you foresee any problem for the user to make sense and conceptualize the following: (refer to the list of hypotheses for Data, Acoustic Parameters and Final Sound Perspectives below)*

Question 4 for Inspection Step 2.4:

- *In terms of physical and social environments, do you foresee any problem for the user to make sense and conceptualize the following: (refer to the list of hypotheses for Data, Acoustic Parameters and Final Sound Perspectives below)*

List of Hypotheses for Data Perspective

- *the image can only be explored if it is in zoomed in mode.*
- *The field of view will only display some part of the zoomed in image.*

List of Hypotheses for Acoustic Parameters Perspective

- *the 10 levels of rhythm can be differentiated based on beat speed.*
- *The higher the value, the faster the rhythm and vice versa.*

<u>List of Hypotheses for Final Sound Perspective</u>

- *The more complex and thicker the sound of timbre 1, the higher the possibility of the cell being sonified being normal.*
- *If the sound of timbre 1 is complex and thick, it does mean that the cell being sonified is a normal cell.*
- *In a condition where, there are no reference cells in the square box, the level of pitch and rhythm can be used as a clue for deciding whether or not it is more like a normal or abnormal cell.*

Example of Potential Detected Problems

The examples of potential problems below are taken from the results of Experiment IV based on the questions above. The transcripts and summary of data for all experiments can be found in Appendix E.

Examples of potential problems in Data Perspective category:

Problem 1: "difficult to measure the distance of scanning steps i.e. how far the navigator will jump".

Problem 2: "no reference for the user to know what size the image is or how much of it they are looking at".

Problem 3: "the user does not know how many times the image is being zoomed. Could have an indicator, or some kind of legend down on the interface?".

Problem 4: "image quality needs to be high especially when the user is zooming in, it should be clear and not blur (acceptable resolution)".

Problem 5: "in moving the image, the buttons are difficult to use as the user has to click multiple times".

Problem 6: "should provide scroll bar for zoomed image"

Problem 7: "the distance of scanning area should be below the direction button which will be more noticeable"

Problem 8: "no zoom ratio provided in the GUI. Therefore, no reference for the user to know what size the image is or how much of it they are looking at.."

Problem 9: "accuracy is very much reliant on the accuracy of A and B feature values, therefore, the values need to be done correctly"

Problem 10: "the highlighted cell should appear first to give attention to the user and prevent them from just scanning through faster"

Problem 11: " the look up table in pre-processing should also be displayed for better understanding of the image being examined"

Example of potential problems in Acoustic Parameters Perspective category:

Problem 12: "failure to detect gradual changes between different sounds may lead to inability to detect trends in the data produced by gradual variation, especially long-term trends. Wide pitch range probably required".

Problem 13: "there is no reference guideline in deciding the ranks. It cannot be just based on their perception from the previous sound they heard".

Problem 14: "difficulty to distinguish level of pitch and rhythm of two sounds with different timbres"

Problem 15: "difficulty in differentiating and determining the position of the small and gradual changes of rhythm".

Problem 16: "the concentration required to hear the difference might be too much. After 4 hours of listening to beep-beep-boop, the user might be mentally not fresh and lose focus.."

Example of potential problems in Final Sound Perspective category:

Problem 17: "on the screen. at sound setting, if users didn't tick the pitch box, the pitch shifting slides should be disabled for user to select e.g. using grey colour"

Problem 18: "The user might find it problematic to give attention to the thickness of the different timbres especially if they were played at the same time"

Problem 19: "potential of confusing the timbre (which belongs to which)"

Problem 20: "too much information to perceive at the same time"

Problem 21: "difficult to understand the trends especially if there is no chorus effect"

Problem 22: "..not allowing the user to choose whether to play the sound automatically or manually. Since the users are professionals trained in visual inspection, they may prefer to see the slide first then use sound feedback as an aid. So, I think it would be better if the user can choose when to give sound feedback"

Problem 23: " ..if the user is not able to hear any sound, they will move and scan the image faster and they might miss some cells"

Problem 24: "..difficult to differentiate either normal and abnormal just by using pitch and rhythm.."

Problem 25: "..there is no sound to represent zoom. E.g. the volume of sound could be used to represent zoom rate.."

Problem 26: "if the application just highlighted the potential cells to be sonified, the user could choose which cells will produce sound"

6.4 Summary

This chapter has introduced a new usability inspection technique that is dedicated for sonification applications. The technique is called Task Interpretation Walkthrough (TIW) as it requires inspector to step through every single task from the Task Data State (TDS) diagram and consider the user interpretation of the task output. The TIW requires a thorough analysis of tasks, which are used to produce the TDS diagram. The task analysis is based on the Sonification Application Model that explains what the sonification application is offering. The tasks include Abstract Process task, user, application and interaction tasks. The User Interpretation Construction (UIC) model is used to investigate what the user will perceive or interpret from the outputs of Abstract Process task. This produces a list of conditions, premises and hypothesis for each Abstract Process task. The contexts of use are required to ensure that the inspection is fair and within the scope of the application. The other inspection materials for the inspection include the user requirements and graphical user interfaces. Finally, this chapter discussed the steps which need to be done by inspectors to examine the design of sonification application and detect any potential problems by using all the inspection materials.

In this thesis, an empirical study of this technique has been done through four different experiments with three different application designs. These experiments will be explained in Chapter 7.

CHAPTER 7 : EMPIRICAL STUDIES OF THE TASK INTERPRETATION WALKTHROUGH

This chapter continues to describe the empirical studies of the Task Interpretation Walkthrough (TIW). Four experiments have been conducted with three different sonification application designs. The objectives, subjects, procedures, potential threats, variables and materials of the experiments will be discussed.

7.1 Introduction

In order to exemplify the approach of usability inspection for sonification applications, four experiments were conducted; Experiment I, II, III and IV. These experiments involved three different applications representing two potential main applications for sonification – data representation and data exploration. The following sections describe *the goal of experiments in supporting the research hypothesis* and the *experimental design of the four experiments that have been conducted.* The results of these experiments are analysed and discussed in Chapter 8.

7.2 Goal of Experiments

As explained in Chapter 1, the research **hypothesis** is as follows:

"Designers of sonification applications will be able to detect significantly important usability potential problems before the implementation phase by analysing the task through different views and paying attention to different perspectives in the data state transformations".

The goal of all four experiments is based on the research hypothesis above. The underlined statement of the hypothesis refers to the proposed inspection technique, Task Interpretation Walkthrough (TIW). Therefore, the overall aim of the experiments is defined as:

"To analyse and understand the capability of the Task Interpretation Walkthrough (TIW) to help a usability inspector to detect potential usability problems from the researcher's point of view (by increased attention to the different perspectives and views of sonification application design)".

The arrangement of experiments is based on the two underlined phrases of the goal; *'analyse and understand the capability of the proposed technique'* and *'to detect potential usability problems'*.

The capability of the TIW refers to its ability and performance in detecting potential usability problems. It can be measured by comparing the performance between the TIW and any other existing inspection technique e.g., Heuristic Evaluation or Cognitive Walkthrough. An example of measurable performance is the number of potential problems that each technique could detect. A usability inspection technique is considered to be doing well if it is able to inspect and encounter problems with high efficiency and effectiveness.

This word potential is important because in usability inspection, the inspectors could encounter problems that might not manifest themselves as actual problems to end-users. A problem is considered as 'real' if an end-user experiences it in a real application and it has a negative impact on their performance. In this research, the potential level of the problem occurring in the application is determined by the inspectors using a *severity rating* by Nielsen (1994). The rating is based on the effect and impact of the detected problem towards the final application.

The overall experimental results are used to support the three supporting hypotheses by showing that:

1. the proposed usability inspection technique will be able to detect significantly more potential usability problems in overall performance.
2. the proposed usability inspection will be able to detect significantly more potentially important usability problems.
3. the proposed usability inspection will be able to detect significantly more potentially important usability problems in each perspective (data, acoustic parameters and final sound) compared to existing usability inspection techniques.

7.3 Overview of Experimental Design

Four experiments were conducted to test the research hypothesis and investigate the quality aspects of our new Task Interpretation Walkthrough method. The experiments used designs from three different sonification applications. The experiments include:

1. Experiment I: Mobile Phone Joystick Text-Entry with Sound.
2. Experiment II: Mobile Phone Joystick Text-Entry with Sound [*With subjects who have research experience in audio*].
3. Experiment III: Diagnosis Tool for Analysis of the Motion and Usage of Patient's Arm.
4. Experiment IV: Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS).

These experiments had subjects ranging from students taking a Masters in software engineering, Masters in music technology, to participants involved directly in research related to audio as well as sonification. The respondents or subjects were asked voluntarily and they were all paid (except for inspectors in Experiment II) to guarantee their commitment through to the end of the experiments. The experimental arrangements had subjects either working alone or paired up with other inspector.

Experiment I involved twenty subjects. Experiment II involved six researchers who are involved in sonification design or audio research. Both Experiment III and IV involved three pairs and three single inspectors. All experiments were conducted in a lab room at the University of York with one subject (or a pair) at a time. All subjects undertook the inspection at different times and they were not allowed to discuss anything about the experiment with other people until the end of the last experimental session.

A major source of limitation of this work was the availability of more experienced inspectors to participate in Experiment II. However, it is feasible to have students taking a relevant class to participate as subjects, so long as they have enough knowledge to understand the task domain and the user interface domain. The environment of these experiments brought a few threats to validity of the results, and which are discussed later in this chapter.

Next, we will look at the objectives of the experiments, procedures, materials and data coding.

7.3.1 Objectives

In line with the three supporting hypotheses, the objectives of experiments are as follows. The experiment number in box bracket indicates that the objective only applies to the respective experiment.

1. To investigate the capability of Task Interpretation Walkthrough to detect potential usability problems.
2. To investigate the effect of each perspective (data, acoustic parameters and final sound perspective) towards the techniques used and the applications being inspected.
3. To evaluate the effectiveness of Task Interpretation Walkthrough.
4. To investigate the effect of inspectors' background in using the proposed and existing inspection technique. [Experiment I]
5. To analyze the effect of inspection order towards the experiment [Experiment I]
6. To investigate the most suitable existing inspection technique to be used in the next experiment III and IV. [Experiments I and II]
7. To analyze the effect of team size (solo or pair) towards the usage of the inspection techniques. [Experiments III and IV]

7.3.2 Independent and Dependent variables

The primary independent variable in all Experiments I, II, III and IV was the inspection technique. For Experiments I and II, the other independent variables were the inspection order (between Heuristic Evaluation and Cognitive Walkthrough); and Inspector Background (Knowledge in Music Technology and Software Engineering). For Experiments III and IV, another independent variable was the number of inspectors, either pair or single.

The only dependent variable in these experiments was the number of potential usability problems identified by each individual inspector. For the purpose of analysis, the identified potential usability problems were categorized into three different perspectives (data, acoustic parameters and final sound) as well as four severity levels (Levels 1, 2, 3 and 4).

The number of potential usability problems detected in each perspective was important as it shows the ability of each inspection technique to detect potential problems in each particular perspective. The results are used to suggest the benefit of inspectors focussing on different perspectives as suggested in the hypothesis. Table 7-1 shows the summary of independent and dependent variables for all experiments.

Table 7-1: Summary of Independent and Dependent Variables

	Experiment I	Experiment II	Experiment III	Experiment IV
Independent Variable	• Inspection Technique • Inspection Order • Inspector Background	• Inspection Technique	• Inspection Technique • Number of Inspectors	• Inspection Technique • Number of Inspectors
Dependent Variable	The number of potential problems identified • by each individual • by perspective • by severity level			

7.3.3 Subjects

This section gives an overview of the participants involved in all experiments – I, II, III and IV. All experiments were conducted with students, researchers and lectures from the University of York, U.K..

Experiment I

Experiment I was conducted by twenty Masters level degree students. Ten students were taking Software Engineering in the Computer Science Department, and who would have knowledge on application design and development. The other ten students were taking Music Technology in the Department of Electronics, and who would have knowledge in audio. All subjects were paid, volunteered, randomly chosen and assumed to have enough knowledge about the task domain and the user interface domain after a briefing before each experiment.

Experiment II

Experiment II was conducted by six subjects including lecturers and researchers from the Computer Science Department and the Department of Electronics. Since all subjects were working and doing research on sound and sonification, they would have an in-depth knowledge on sound related research. The original intention was to have ten experienced subjects or inspectors – five inspectors for each technique. Unfortunately, two subjects for the proposed technique were not able to make it. Therefore, the best three out of five inspection results from the existing techniques (control group) were used for the final analysis to compare with the three results from the proposed technique.

Experiments III and IV

Experiments III and IV were conducted by eighteen Masters students, who formed six pairs and six single inspectors in each experiment. Three pairs and three single inspectors were required and randomly chosen for each technique in each experiment. Both experiments were conducted with the same subjects. Twelve students were from Software Engineering and the other six were from Music Technology. All the single inspectors were from Software Engineering, while all the pair inspectors were a combination of both education backgrounds. All subjects were also paid, randomly chosen and assumed to have enough knowledge about the task domain and the user interface domain after a briefing before each experiment.

7.3.4 Experimental Design

This section gives the settings and arrangements of each experiment. The settings for Experiments I and II were used to provide information for the setting up of Experiments III and IV especially to determine the best existing inspection technique for comparison purposes.

Experiment I

The subjects were randomly divided into two different groups – control group and experiment group. Each group had ten inspectors, where five of them were students of Software Engineering (SWE) and another five were students of Music Technology (MT). Each subject inspected only one Sonification Application design. The experiment group was using our new Task Interpretation Walkthrough (TIW) while the control group used the Heuristic Evaluation (HE) and Cognitive Walkthrough (CW) techniques. Eight inspectors from the control group were given different inspection order either 1) HE followed by CW or 2) CW followed by HE. Each inspection order consisted of two inspectors from Software Engineering and two from Music Technology, which made up four inspectors for each order. Both groups were inspecting the same application design called 'Mobile Phone Joystick Text-Entry with Sound'. The summary of the experimental design is shown in Table 7-2.

Table 7-2: Summary of Experimental Design for Experiment I

	Experiment Group	Control Group
Application Design	Mobile Phone Joystick Text-Entry with Sound	Mobile Phone Joystick Text-Entry with Sound
Number of inspectors	10 inspectors	10 inspectors
Inspectors background	5 students of SWE 5 students of MT	5 students of SWE 5 students of MT
Inspection Technique	TIW	HE CW
Order of Inspection Technique	No Order	HE → CW (2 SWE and 2 MT students) CW → HE (2 SWE and 2 MT students)

Experiment II

The setting for the Experiment II was equivalent to Experiment I. The only differences between the two were the number and background of inspectors. As the number of inspectors was limited, there were only three inspectors in each experiment and control group. There were five inspectors for the control group, but only the best three were chosen for the results analysis, as two inspectors from the experiment group were not able to attend the experiment. The experimental group used the Task Interpretation Walkthrough (TIW) inspection technique while the control group used both the Heuristic Evaluation (HE) and the Cognitive Walkthrough (CW).

Based on the result analysis of Experiment I, there was no significant main effect for inspection order factor towards individual detection of potential problems. Heuristic Evaluation was found to perform better than Cognitive Walkthrough. The details of analysis results are discussed in Chapter 8. Due to these results and the limited availability of inspectors, the control group were only required to use *one* inspection order; Heuristic Evaluation followed by Cognitive Walkthrough. All subjects were inspecting the same application design called 'Mobile Phone Joystick Text-Entry with Sound'. The summary of experimental design for Experiment II is shown in Table 7-3.

Table 7-3: Summary of Experimental Design for Experiment II

	Experiment Group	Control Group
Application Design	Mobile Phone Joystick Text-Entry with Sound	Mobile Phone Joystick Text-Entry with Sound
Number of inspectors	3 inspectors	3 inspectors
Inspectors background	3 researchers in audio related	5 researchers in audio related (choose the best 3 results)
Inspection Technique	TIW	HE CW
Order of Inspection Technique	No Order	HE → CW

Experiments III and IV

In both these experiments, the subjects were randomly divided into two different groups – control group and experiment group. Each group had three paired inspectors (one from each education background) and three single inspectors from Software Engineering. The experiment group used Task Interpretation Walkthrough (TIW) while the control group used only Heuristic Evaluation (HE). Both experiments were carried out by the same inspectors. The summary of experimental design is shown in Table 7-4.

Table 7-4: Summary of Experimental Design for Experiments III and IV

	Experiment Group	Control Group
Application Design	Experiment III: Diagnosis Tool for Analysis of the Motion and Usage of Patient's Arm.	Experiment III: Diagnosis Tool for Analysis of the Motion and Usage of Patient's Arm.
	Experiment IV: Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS).	Experiment IV: Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS).
Inspectors background	6 students of SWE 3 students of MT	6 students of SWE 3 students of MT
Number of Inspectors	3 pairs (1 SWE+1 MT) 3 single (SWE)	3 pairs (1 SWE+1 MT) 3 single (SWE)
Inspection Technique	TIW	HE

Since there was the possibility of inspectors already being familiar with the existing inspection techniques (HE and CW), this might be unfair to the new technique (TIW), which was new to the inspectors. This problem was solved by giving all inspectors some 'reading material' for them to study and learn the inspection technique *before* they came for the real experiment. The reading material explained how to conduct an inspection using the respective inspection techniques. All inspectors from the experiment group and the control group were given the descriptions of TIW and HE with CW respectively. These reading materials can be found in Appendix IV.

Inspectors did the inspection at a time to suit their convenience. This could be a potential threat to the experiment, as it is possible that they might have a discussion with another participant before the real experiment. However, they were told not to discuss anything about the experiments with other inspectors before all of them had finished their inspection sessions. They agreed not to discuss about the experiments before all the inspection sessions over. A few other potential threats are discussed later in this chapter.

7.3.5 Materials

Two different inspection materials were given 'before' and 'during' the experiments. The inspection material that was given before experiment was the 'reading material' mentioned above. It introduced the inspection technique that the inspector would use. It explains the technique overview, inspection steps and an example. There were two sets of reading materials – for TIW and 'HE and CW'. Examples of the materials can be found in Appendix IV.

The inspection materials 'during' experiment were based on the inspection technique – TIW, CW and HE. Below are the lists of materials for each technique.

General Document

Regardless of the technique they used, all inspectors were given the same set of general documentation, which described the sonification application design to be inspected. The document contained four parts as follows:

Part 1: The description of the application

This part describes the idea of the application design as well as the illustration of graphical user interfaces. This should give the inspectors insight into the ideas that the designer has had about the application to be inspected.

Part 2: Objective of application

This part lists out the objectives of the application to be achieved.

Part 3: Description of users

This part gives a brief idea about the potential end users of the application.

Part 4: Goal, task and sub-tasks to be accomplished

This part lists out the goal and tasks that the users need to accomplish. This part is required by the Cognitive Walkthrough inspection technique.

All the documents for each application design can be found in Appendices I, II and III.

Heuristics Evaluation

In addition to the general documentation, the control group were given another document, which described the Heuristic Evaluation technique. In this technique, the inspectors were required to examine the application design (general document) and to see if any of the usability heuristic lists was violated. All the potential violations were reported by describing the cause of the detected potential problems. In general, the inspectors were required to do the following three steps:

<u>STEP 1:</u>
In this step, inspectors were required to understand the application by reading the attached general documentation.

STEP 2:
Inspectors were required to step through the example of tasks and use the list of usability heuristics below to identify any kind of anomalies or problems of the application.

List of Usability Heuristics by Nielsen (1993):

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

Each heuristic was further explained in the inspection materials, which can be found in Appendix I, II and III.

STEP 3:
In this step, all problems that were found by inspectors were rated based on Nielsen's (1993) Severity Level. The Levels include: -

Level 1: a cosmetic problem that only needs to be fixed if extra time is available on the project

Level 2: a minor usability problem, where to improve this could be set as a low priority.

Level 3: a major usability problem, which is *important to fix* and could be set as high priority

Level 4: a usability catastrophe, which *must* be fixed before it goes for development.

Cognitive Walkthrough

Cognitive Walkthrough was another technique used by the control inspectors, but only in Experiments I and II. In this technique, the inspectors were required to examine the application design (general documentation) and step through the action sequences to accomplish certain goals (Part 4 of the general document). During the action steps, inspectors were required to consider the four questions below and tried to detect any potential problems. All the potential problems were reported by describing the cause and the effect of the potential problems.

In general, the inspectors were required to do the following two steps:

<u>STEP 1</u>

To start the inspection, inspectors were required to step through the action sequences to accomplish certain tasks. For every action, inspectors need to ask the following questions:

1. Will the users try to achieve the right effect?
2. Will the user notice that the correct action is available?
3. Will the user associate the correct action with the effect to be achieved?
4. If the correct action is performed, will the user see that progress is being made toward solution of the task?

<u>STEP 2</u>

For every action of the task, inspectors were asked to <u>create and construct a success story</u> by referring all of the four questions above. When a success story cannot be told, inspectors <u>constructed a failure story</u> that was also based on the four questions above and the reason why the user may fail. All problems that were found by inspectors were also rated based on Nielsen's (1993) Severity Level.

Inspection materials for Cognitive Walkthrough can be found in Appendix I.

Task Interpretation Walkthrough (TIW)

The inspection materials for the Task Interpretation Walkthrough are as explained in Chapter 6. The materials include:-

1) Task Data State diagram
2) User Interpretations
3) Contexts of use

4) Inspection Procedure that describes step by step how to perform the inspection.
5) Feedback Form where the inspectors can write the problems.

Inspection materials for TIW can be found in Appendices I, II and III

7.3.6 Experimental Procedure

The experiments were divided into three main parts – *preparation, inspection process* and *potential problem analysis* experiments as described below. The experiment was carried out by only one inspector at a time.

Part 1: Preparation

Before the inspection session, all inspectors were given some 'reading material', which contained detail descriptions of the inspection technique that they were to use. There were no descriptions of the application design disclosed in any form to the inspector before the inspection session started. They were also not allowed to discuss the experiment with anybody after their experiment session.

Part 2: Inspection process

During the experiment, each inspector was given a package of inspection materials containing the 'general document' and instructions of the appropriate inspection technique. There were also some paper forms for them to record[1] any detected potential usability problems and their severity level. During the experiment, the inspectors were given a short briefing before they started the inspection tasks (see next section).

Brief Introduction

As an introduction to the experiment, a short brief was given to each inspector regarding the following information:

1) The overall goal of experiment, which was to find as many potential problems as possible so that a more usable and better application design can be produced and built.

[1] *In Experiments I and II, all detected problems were recorded directly on computer, but in Experiment III and IV, all detected problems were recorded in hand writing.*

2) The materials that were available for them to use, which included the 'general document', forms to record detected potential problems, a list of severity levels and the chosen inspection technique (either Task Interpretation Walkthrough or 'Cognitive Walkthrough and Heuristics Evaluation').
3) Explanation about how to use the usability inspection technique.
4) Brief overview of the sonification application design to be inspected.
5) A mention that the conversation and discussion between pair inspectors in Experiments III and IV would be voice recorded. We asked the inspectors to talk and discuss as usual.

After the brief introduction, the inspectors were allowed to ask any questions before they started the inspection process.

Inspection

The inspectors started the inspection tasks as soon as the briefing and question sessions finished. However, the observer was always available for any questions during the experiment.

For Experiments III and IV, the voice conversation and discussion for the pair inspectors were recorded to capture any important information that they might forget to write down. This also helped inspectors to describe certain problems that might be too long to be written or difficult to express in words. It also helped to form a useful record of the dynamics of the inspection process for further analysis. This recording was applied to pair inspectors for both the experiment and the control groups.

To ensure that the inspector really understood what they should do, the observer helped the inspector to start and detect the first potential problem. Each session took one to two hours depending on the inspectors. The plan was that after two hours, the inspector would be required to ask the observer to extend the time. However, it turned out that all inspectors required less than two hours in all experiments.

Part 3: Potential problem analysis

After all inspectors had done their inspection session, the results were finalised and coded as a list of detected potential problems. The final severity level of each potential problem was based on the average of severity level given by all inspectors. This process is explained further in the next section.

7.3.7 Data Coding and Analysis

After each experiment, all potential problems were gathered to build a master list of potential problems for each experiment. Each master list was further divided into three perspectives as suggested in the hypothesis (data, acoustic parameters and final sound perspective). Each perspective contains the following information – *problem ID, Problem Descriptions, Frequency* and *Severity Level* as shown in Table 7-5.

Table 7-5: Example of Coding Form for Each Perspective

Problem ID	**Problem Descriptions**		**Frequency**	**Severity Level**
	Cause	**Effect**		
[Unique ID for the potential problem]	*[describe the potential problem of the application design itself]*	*[the result of the potential problems]*	*[number of inspectors detecting the same potential problem]*	*[Severity level of the potential problem]*

The Problem Descriptions were further explained based on the cause and effect of the problem. The cause of error refers to the potential problems in the application design and the effect is the potential result of the problems especially to the end user. The description was explained in this way as some inspectors described certain problems based on 'the effects of the application error to the end user' rather than 'the error of the application itself'. Below is the example, of what we mean by 'cause' and 'effect':

Cause: *The mobile phone keypad size is too small*
Effect: *It is difficult to press and dial especially for older people.*

If the cause or effect of the detected potential problem had been raised before (either by this inspector, or by another inspector), it would be recorded under the same unique ID and the frequency would be increased by one. If it did not yet exist in the master list, it would be added with a new unique number. The severity levels of this potential problem, given by all inspectors, were also recorded. The average of these severity level values was used as the final severity level of the potential problem. The severity level and frequency are important for the designer as a guideline for deciding whether to improve the design.

For each of master lists from each experiment, statistical analysis was done to see whether or not the proposed technique was performing better and to verify how significant the results were. The result analysis will be explained in detail in Chapter 8.

7.4 Possible Threats to Validity

This section discusses several potential threats that might influence the results of the experiments. Threats are factors that can affect the dependent variables in an experiment.

Student Selection – In Experiments III and IV, the pair inspector consisted of two people from different backgrounds – Software Engineering and Music Technology. However, the single inspectors for both experiments were only from Software Engineering. It would have been better if some single inspectors were also from Music technology so that we could see the effect of subject background on the pair inspectors.

Testing duration – the time taken to do the inspection was within 1 to 2 hours. This might be quite long for the subject, and they might get tired, leading to them possibly performing worse.

Testing familiarity – the subjects might get familiar with the material and the technique and this might have an effect on subsequent results. In Experiments III and IV, the same inspectors used the same inspection technique but on different applications. By inspecting two 'totally different' applications, this effect was reduced.

Severity level assignment – the decision on categorizing severity level can quite vague, and is left to the judgement of the inspector, based on their knowledge and experience. However, any variation has been reduced by calculating the average of the severity level that was given by all inspectors regardless of the technique they used.

Technique comprehension – the subject might not fully understand the new technique and could affect the implementation of the experiments. To avoid this, all inspectors were given reading materials for them to become familiar with the inspection techniques. However, it was difficult to ensure that the inspectors had already understood the inspection technique before commencing the experiment. Even though the inspection techniques were explained during the experiment, it could have been better if the explanation by the observer was done before the real experiment e.g., a short meeting between all inspectors and observer to explain the reading materials.

Two techniques at the same time – In Experiments I and II, the control group used two techniques (Cognitive Walkthrough and Heuristic Evaluation) within the same session. Therefore, there was a possibility for the results of the second technique to be influenced by the first technique. During the experiment, the control group were further divided into two groups based on the order of inspection techniques (either CW followed by HE or HE followed by CW). However, as we will see, the statistical test did not show a significant main effect for inspection order factor. Therefore, both existing techniques would perform the same, either 'by the same inspectors for both techniques' or 'by different inspector for each technique'.

Recording of the discussion – the decision to record a discussion and conversation between inspectors (pair) might discourage them to discuss and talk. However, from the result analysis, this recording turned out to be very helpful as we found problems that had not been recorded properly on paper.

Inspecting at different time – the main reason of this type of arrangement was to allow the inspectors to choose their most convenient time to do the experiments. However, there was a possibility for the inspectors to discuss with other inspectors before the experiment ended. Since we want to provide time flexibility to the inspectors, all inspectors were required not to disclose or discuss anything related to the experiment until all inspectors had finished their inspection sessions.

7.5 Summary

In summary, this chapter has discussed the basis of the four experiments, which were carried out to support the three supporting hypotheses. There were always two groups in each experiment – the experiment group and the control group. All inspectors in each group were chosen randomly. The experiment group always used Task Interpretation Walkthrough and the control group used existing techniques, Heuristic Evaluation and Cognitive Walkthrough. The experimental design was arranged in such a way as to achieve the objectives of the overall experiments, which finally will be used to evaluate the hypothesis. There were three application designs used to test the proposed technique.

The results (potential problems) of the experiments were categorized based on the three perspectives (data, acoustic parameters and final sound) as well as their severity levels. The overall performance was used to test the first hypothesis, that the TIW should be able to detect more potential usability problems. The severity levels of each potential problem were used to evaluate the second hypothesis, that the TIW should be able to detect more important problems. Finally, the categorisation of the potential problems in three perspectives was used to assess the third supporting hypothesis, that the TIW should be able to detect more problems in each perspective especially in the acoustic parameters and final sound perspectives. A few potential threats to the validity of the experiments were identified, however, some efforts had been done to minimise these effects during the experiments.

By repeating the experiments and comparing with existing inspection techniques, different application designs, a mixture of inspectors background and different number of inspectors, several conclusions can be drawn and a statistical analysis done to verify the significance of the results.

This results analysis will be discussed in the next chapter.

CHAPTER 8 : RESULT ANALYSIS

This chapter discusses the analysis of the four experiments explained in Chapter 7. The analysis includes inspection performances of Task Interpretation Walkthrough, Heuristic Evaluation and Cognitive Walkthrough. The performances are explained based on the three supporting hypotheses, which can be categorized by overall performance; level of severity; and transformation perspectives.

8.1 Introduction

The results of all four experiments are analysed and explained in two parts – the results of 'Experiments I and II' as the first part; and the results of 'Experiments III and IV' as the second part. The explanation is arranged in such a way as to support the three supporting hypotheses mentioned in Chapters 1 and 7. Each part of the analysis explains the Task Interpretation Walkthrough (TIW), the Heuristic Evaluation (HE) and the Cognitive Walkthrough (CW)[1] in terms of;

1. The overall performances to test the first supporting hypothesis, H_1:

   H_1: the proposed usability inspection will be able to detect significantly more potential usability problems in overall performance

2. The overall performances in severity level to test the second supporting hypothesis, H_2:

   *H_2: the proposed usability inspection will be able to detect significantly more **important** potential usability problems.*

3. The overall performances in each perspective to test the third supporting hypothesis, H_3:

[1] Cognitive Walkthrough was only used in Experiments I and II

*H_3: the proposed usability inspection will be able to detect significantly more important potential usability problems **in each perspective** (data, acoustic parameters and final sound perspective) compared to existing usability inspection techniques.*

The problems detection performance of each inspection technique is represented based on the total number and percentage of unique potential problems; and the mean of individual detection performance. The unique potential problems detected in all experiments are calculated excluding overlap potential problems as explain in Chapter 7. The means are calculated including overlap potential problems by individual inspector. All probability values in these experiments refer to the means values unless otherwise stated.

8.2 Analysis of Experiments I and II

This section gives the analysis of results for the first analysis group – Experiment I and II. Both experiments are explained and analysed together as both inspected the same sonification application called the Mobile Phone Joystick Text-Entry with Sound. Each section is arranged and described based on the three supporting hypotheses as explained above. Summary and guidelines for further studies are also suggested. The overall performances and statistical tests of all inspection techniques (TIW, HE and CW) are discussed.

8.2.1 Supporting Hypothesis 1 (H_1)

This section analyses the experiments' results to evaluate the first supporting hypothesis (H_1), which mentioned, "*The proposed usability inspection will be able to detect significantly **more potential problems** in overall performance*". For this hypothesis to be supported, by giving the same number of inspectors with similar expertise, the TIW should significantly improve the detection of potential problems over the existing inspection technique.

8.2.1.1 Individual Detection Effectiveness for All Potential Problems

The detection performances for both experiments are summarized in Table 8-1. The table shows the number and percentage of potential problems detected by each technique as well as the mean of potential problems detected by inspectors - Task Interpretation Walkthrough (TIW), Heuristic Evaluation (HE) and Cognitive Walkthrough (CW). The total numbers of **unique** potential problems that were successfully detected by all inspection techniques are 40 in Experiment I and 36 in Experiment II.

Table 8-1: Detected Potential Usability Potential Problems by Inspection Technique

Inspection Techniques	**Experiment I** *(N=10 for each technique)*		**Experiment II** *(N=3 for each technique)*	
	Number and Percentage of Unique Potential Problems	**Mean**	Number and Percentage of Unique Potential Problems	**Mean**
Tasks Interpretation Walkthrough (TIW)	38 (95%)	10.2	30 (83%)	16.0
Heuristic Evaluation (HE)	16 (40%)	5.4	13 (36%)	6.0
Cognitive Walkthrough (CW)	15 (38%)	4.4	9 (25%)	4.0
Total Unique Potential Problems	40 (100%)		36 (100%)	

On average in Experiment I, TIW detected 95% of the total unique potential problems that detected by all the inspection techniques, which is more than double the problems detected by HE and CW, 40% and 38% respectively. The means also show that inspector with TIW detected more problems compared to HE and CW, with 10.2, 5.4 and 4.4 respectively. While in Experiment II, TIW detected 83% of the total unique potential problems, which is also more than double the problems detected by HE and CW, 36% and 25% respectively. The means also show that inspector with TIW was able to detect more problems with average of 16, which is more than double compared to HE and CW, 6 and 4 respectively. Based on these percentages and means, TIW outperformed both HE and CW. At the same time, we note that HE outperformed CW.

Generally, the results from both experiments in Table 8-1 support the first supporting hypothesis, in which the proposed inspection technique (TIW) should be able to help inspectors to detect more potential problems in overall performance.

Were these results statistically significant? This can be answered by calculating the probability of these results occurring by chance, or known as p value. All the p values are referring to and based on the means above unless otherwise stated.

As stated in the experimental design of Experiment I (in Chapter 7), besides the inspection technique (TIW, HE and CW), the inspection order (either HE followed by CW or CW followed by HE) was also an independent variable for the control group. Before TIW can be fairly compared with HE and CW, it is important to ensure that the 'inspection order' factor has no significant main effect on the results. Therefore, ANOVA[2] (two-factor mixed) was performed to test the effect of these independent factors (variables) and the interaction between the two on the individual detection scores – 'inspection technique' and 'inspection order' factors. Referring to Table 8-2, the results show that neither 'inspection technique' nor 'inspection order' have a significant main effect ($p>0.05$). This shows that both existing techniques have equivalent performance in individual detection of potential problems. It shows also that, using the two existing techniques in sequence by the same inspectors will not influence the detection performance of each technique. The interaction between the two was also found to be non-significant, which means the inspection order has no different effects upon the HE and CW techniques in individual detection of potential problems. Therefore, the effect of these variables can be tested separately.

[2] ANOVA stands for Analysis of Variance, which is used to investigate the effects of two or more independent variables. The effects could be (1) whether each independent variable has a significant "main effect" and (2) whether there is significant "interaction" between independent variables.

Table 8-2: Significant Differences and Effect of independent variables on HE and CW on the Individual Detection Scores for Experiment I

Inspection Technique (N=8 for each technique)	Mean *Inspection Order* (HE → CW)	(CW → HE)	Related t-test	Wilcoxon test	Inspection Technique (ANOVA)	Inspection Order (ANOVA)	Insp. Order x (Insp. Technique) (ANOVA)
Heuristic Evaluation	5.25	5.50	p>0.05	p>0.05	p>0.05	p>0.05	p>0.05
Cognitive Walkthrough	3.75	6.00	p>0.05	p>0.05			

Another way to deal with the order effect is to compare the performance of the two techniques based on the two different orders. Thus, two Related t-test and two Wilcoxon-test were run and the results are also given in Table 8-2. The results were also found to be non-significant, which means that the data points for both HE and CW were not related and influenced by the inspection order. Therefore, a comparison between the proposed inspection technique (TIW) with HE and CW can also be done separately.

The Related t-test and Wilcoxon test were run to evaluate the significant differences of individual detection scores for HE and CW. The detection score is the number of detected potential problems by each inspector. The sample sizes (number of inspectors) were ten and three for each technique in Experiments I and II respectively. Experiment I used the traditional 0.05 significance level. However, in Experiment II, the number of suitable and available subjects for running the experiment was limited. It was difficult to get people who were directly involved in research related to sound, especially in sound design. Because of this constraint, the study did not have enough statistical power to test the hypotheses at the traditional 0.05 significance level. The results are still useful and could be represented without looking at the significant level due to the lower number of subjects. However, it is still interesting to see the probability of the results to be happened by chance – by increasing the significance level. Therefore, a 0.10 significance level has to be used instead, so using a less rigorous level would not reject any potentially interesting results. This 0.10 significance level was in fact also used by Zhijun

Zhang in his PhD thesis. His thesis also introduced a new usability inspection technique called Perspective-Based Usability Inspection[3]. He also faced with the same problems of not enough expert inspectors, with only 3 inspectors for each technique in one out of his three experiments.

The results of the tests are shown in Table 8-3 (a) and (b) for Experiments I and II respectively. Even though the HE outperformed CW in both experiments (*refer to the second column of Table 8-3 (a) and (b)*), however, the statistical tests were insufficient to justify the difference in the individual detection scores as significant ($p>0.05$ in Experiment I and $p>0.1$ in Experiment II). This indicates that there is no difference in individual detection performances between HE and CW.

Table 8-3: Significant Differences in Individual Detection Scores between Heuristic and Cognitive Walkthrough (t-test and Wilcoxon test)

(a) Experiment I

Inspection Technique (N=10 in each technique)	**Mean, Number and Percentage of Potential Problems** *(N=3 for each technique) TOTAL = 40 (100%)*	**Related t-test (Parametric)**	**Wilcoxon test (Nonparametric)**
Heuristic Evaluation	mean=5.4, 16 (40%)	p>0.05	p>0.05
Cognitive Walkthrough	mean=4.4, 15 (38%)		

(b) Experiment II

Inspection Technique (N=3 in each technique)	**Mean, Number and Percentage of Potential Problems** *(N=3 for each technique) TOTAL = 36 (100%)*	**Related t-test (Parametric)**	**Wilcoxon test (Nonparametric)**
Heuristic Evaluation	mean=6.0, 13 (36%)	p>0.05	p>0.05
Cognitive Walkthrough	mean=4.0, 9 (25%)		

[3] Zhijun Zhang, 1999. The Design and Empirical Studies of Perspective-Based Usability Inspection. Phd. Thesis. The University of Maryland, College Park, US.

For TIW, an Independent t-test and a Mann-Whitney test were conducted to test the significance of differences in detecting performance between TIW and the existing inspection techniques. The statistical test results for Experiment I are shown in Table 8-4. Both statistical tests found the differences in individual detection scores to be significant, where $p < 0.05$ for both 'TIW - HE' and 'TIW - CW'.

Table 8-4: Significant Differences in Individual Detection Scores between TIW and HE; and TIW and CW for Experiment I

N=10 in each technique	Heuristic Evaluation (HE)		Cognitive Walkthrough (CW)	
	Independent t-test	Mann-Whitney test	Independent t-test	Mann-Whitney test
Task Interpretation Walkthrough (TIW)	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$

The same statistical tests were also conducted with the data points from Experiment II. The results are shown in Table 8-5, which also found the differences in individual detection scores to be significant (where all p values were below the 0.1 significance level).

Table 8-5: Significant Differences of Individual Detection Scores between TIW and HE; and TIW and CW for Experiment II

N=3 in each Technique	Heuristic Evaluation (HE)		Cognitive Walkthrough (CW)	
	Independent t-test (Parameteric)	Mann-Whitney test (Nonparametric)	Independent t-test (Parameteric)	Mann-Whitney test (Nonparametric)
Task Interpretation Walkthrough (TIW)	$p<0.1$ ($p=0.07$)	$p<0.1$ ($p=0.05$)	$p<0.1$ ($p=0.05$)	$p<0.1$ ($p=0.05$)

In summary, for Experiment I, when data from all twenty inspectors were considered, the differences in the individual detection of potential problems between TIW and HE as well as CW were found to be significant at the traditional significance level of 0.05. The same results also appeared to be significant in Experiment II at the significance level of 0.10. Therefore, the performance results as shown in Table 8-1 above are significant.

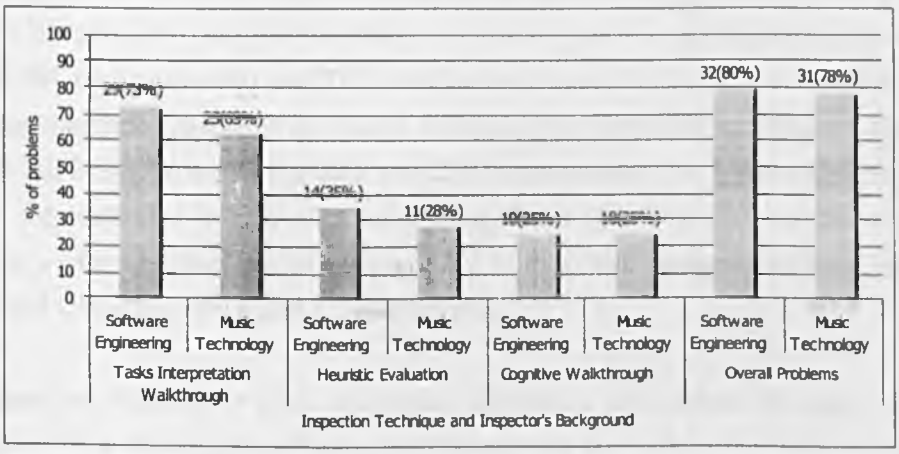
It is evident that TIW has significantly improved the inspectors' detection effectiveness for potential problems of sonification applications.

The next section investigates the influence of each inspector's background and prior knowledge towards the detection performance.

8.2.1.2 Effects of Inspectors' Knowledge Background on the Inspection Technique

As mentioned in the experimental design of Experiment I (Chapter 7), the inspectors were from two different knowledge backgrounds. At the time of experiment, ten of the inspectors were taking a Masters in Software Engineering and the other ten were taking a Masters in Music Technology. The first experiment was arranged in such a way as to find out the effect of these different backgrounds in detecting potential problems by each inspection technique.

Figure 8-1 shows the comparison of detected potential problems between the two different backgrounds for each inspection technique.



MEANS	TIW	HE	CW	Overall
Software Engineering	11.8	6.0	4.2	7.3
Music Technology	8.6	4.8	4.6	6.0

Figure 8-1: Detected Potential Problems by Participant's Background for Experiment 1

In overall performance (see 'Overall Problems' in the above Table 8-1), the numbers of potential problems detected (out of the overall unique potential problems) by both backgrounds are more or less the same with 80% and 78% for inspectors with a background in software engineering (SWE) and music technology (MT) with means 7.33 and 6 respectively. The inspectors with a SWE background detected 10% and 7% more potential problems than the inspectors with an MT background in TIW and HE respectively. The means also show the same results, where the inspectors with a SWE and an MT backgrounds were able to detect on average of 11.8 and 8.6 of potential problems respectively in TIW; and on average of 6 and 4.8 of potential problems respectively in HE. The CW inspection technique seems to give no effect towards the differences in background as both detected the same percentage of problems. However, the means show that the inspector with a MT background was able to detect a slightly higher number of potential problems than the inspector with an SWE background with 4.6 and 4.2 respectively. By comparing between the same backgrounds upon all the techniques, TIW was found to be able to detect more potential problems. In addition, TIW also helped the inspectors with no formal knowledge in usability to detect more potential problems. Their overall performances were higher than the performances of inspectors with a background in SWE that used HE and CW. This can be observed from Figure 8-1, where the inspectors with MT background (from TIW) were able to detect up to 63% of the overall unique potential problems compared to inspectors with SWE background from both HE and CW with only 35% and 25% respectively. The means also show that the inspector with a MT background from TIW was able to detect on average of 8.6 higher than to the inspector with a SWE background from both HE and CW with only 6 and 4.2 respectively.

In general, looking at each technique, the chart and means in Figure 8-1 suggests that knowledge of Software Engineering could help the inspectors to detect more potential problems in TIW and HE. This result was as expected that inspectors with knowledge in usability should detect more potential problems than inspectors without knowledge in usability. However, this seems not to be happened to CW. However, the knowledge background factor are not imply to the results between music technology for TIW and software engineering for HE, as TIW detected more problem than HE. This suggests that *TIW could also help and improve inspection performances of inspectors who have no formal knowledge in usability,* but at least having knowledge in the application domain (sound).

Were these results significant? (All the p values are referring to and based on the means above unless otherwise stated).

To answer this question, ANOVA (between subjects) was run to test the effects of the two factors – *inspection technique* and *inspector's background.* The inspection technique factor consists of TIW and HE, while the inspector's background factor consists of SWE and MT. Heuristic Evaluation is used in this test as it gave the same pattern of results as TIW, where inspectors with a background in SWE performed better than inspectors with a background in MT. The test results are shown in Table 8-6. The test suggests that the inspection technique and inspector's background factors have significant main effects on the overall detection effectiveness. This shows that the individual problem detection performance of TIW and HE were significantly influenced by the inspectors' background. The interaction between the inspection technique and inspector's background was found to be non-significant with a p value greater than the 0.05 traditional significance level. This shows that both backgrounds have the same effects upon TIW and HE in detecting potential problems. Therefore, the effect of each factor can be tested separately as both techniques have the same effects towards the inspectors' background.

Table 8-6: Effect of Inspector's background and Inspection Technique on Overall Detection effectiveness in Experiment I

	Inspection Technique		**Inspector's Background**		**Inspection Technique x Inspector's Background**
	TIW	**HE**	**SWE**	**MT**	
Mean, Number and Percentage of potential problems. *[Total potential problems=40 (100%)]*	Mean=10.2, 38 (95%)	Mean=5.4, 16 (40%)	Mean=7.33, 32 (80%)	Mean=6.0, 31 (78%)	$p>0.05$
Factorial ANOVA (N=10 for each technique)	$p<0.05$		$p<0.05$		

T-Test and Mann-Whitney tests were run to evaluate the significant differences between the two participant's backgrounds towards the individual detection of potential problems in each inspection technique. Table 8-7 shows the results of these tests. The results show that the differences in the number of detected potential problems for TIW and HE were statistically significant ($p < 0.05$). Therefore, **the participant's background significantly influenced the problem detection performance for both TIW and HE.** However, the differences in CW were found to be non-significant as the p-values were more than the 0.05 traditional significance level in both tests.

Table 8-7: Significant Differences of Inspectors Background on overall Effectiveness in Experiment 1 (p-value from T-test and Mann-Whitney test)

Inspection Technique (N=10 in each technique)	**Mean, Number and Percentage of Potential Problems** *Total potential problems=40 (100%)* *Inspectors background*		***p*-values (for different background) df=8**	
	SWE	**MT**	**Dependent T-test**	**Mann-Whitney test**
Task Interpretation Walkthrough (TIW)	*mean=11.8,2 9 (73%)*	*mean=8.6, 25 (63%)*	*p<0.05*	*p<0.05*
Heuristic Evaluation (HE)	*mean=6.0, 14 (35%)*	*mean=4.8, 11 (28%)*	*p<0.05*	*p<0.05*
Cognitive Walkthrough (CW)	*mean=4.2, 10 (25%)*	*mean=4.6, 10 (25%)*	*p>0.05*	*p>0.05*

Dependent T-test and Mann-Whitney tests were again used to evaluate the significance of differences between the inspectors with a background in music technology who were using TIW and the inspectors with a background in software engineering who were using HE. Based on the results as shown in Table 8-8, both tests show that the differences were statistically significant with p values less than 0.05 significance level. **This suggests that TIW significantly improved the inspection performance of inspectors who have no formal knowledge in usability but have knowledge in the domain of application (sound).**

Table 8-8: Significant Differences between Music Technology with TIW and Software Engineering with HE in Detected Potential Problems

Background of Inspectors (N=5 in each techniuqe)	Mean, Number and Percentage of Potential Problems *Total potential problems=40 (100%)*	p-values	
		Dependent T-test	Mann-Whitney test
Music Technology using TIW	***mean=8.6, 25 (63%)***	*$p<0.05$*	*$p<0.05$*
Software Engineering using HE	***mean=6.0, 14 (35%)***		

In summary, the overall results suggest that TIW was significantly able to detect more potential problems in overall performance, which supports the first supporting hypothesis, H_1. TIW was also found to significantly improve the detection performance of inspectors, who have no formal knowledge of usability but instead have knowledge in the domain of application, which is sound.

8.2.2 Supporting Hypothesis 2 (H_2)

This section analyses the experiments' results to evaluate the second supporting hypothesis (H_2), which *requires TIW to be able to detect significantly more **important** potential usability problems.*

8.2.2.1 Performance by Severity Level

The key word in the second supporting hypothesis is 'important', which was mentioned earlier in Chapter 7 as highly likely to become potential usability problems. This potentiality problem attribute is important because in usability inspection, the inspectors could encounter problems that do not always predict real usability problems for the end users. A problem is considered as real if end-users will experience it in the real application as well as it having a significant impact on their performance (user performance, productivity and/or satisfaction). Since this study was only comparing between TIW and the existing usability inspection techniques, therefore, the potential level was determined by using the Nielson (1993) severity level, which was rated by inspectors as described in Chapter 7. The severity levels are restated as follows:

Level 1 - cosmetic potential usability problem only, need not be fixed unless extra time is available.

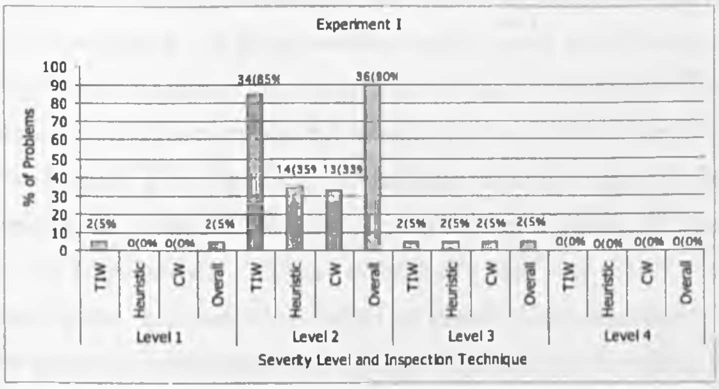
Level 2 - minor potential usability problem, fixing this should be given low priority.

Level 3 - major potential usability problem, important to fix so should be given high priority.

Level 4 - Usability catastrophe, imperative to fix this before product can be released.

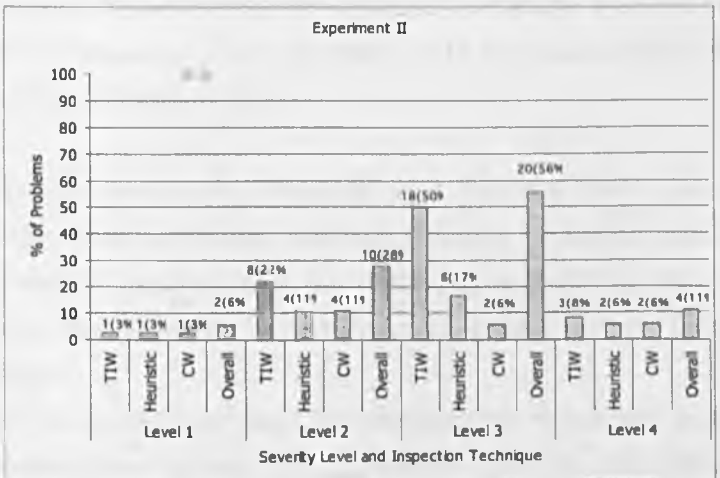
Therefore to satisfy the second supporting hypothesis, by using the same number of inspectors with similar expertise, TIW should detect significantly more important potential usability problems than the existing techniques. Figure 8-2 below shows the distribution of detected potential problems across the four severity levels for Experiments I and II. In Experiment I, most of the detected potential problems were rated as Level 2, which encompassed 90% of the overall unique potential problems. TIW detected more potential problems at Level I (mean=0.2) and II (mean=9.1) and equal number at Level 3 (but lower mean of 0.9). There was no potential problem at Level IV. In Experiment II, the potential problems were rated by using all levels of severity compared to Experiment I. Most problems were categorized as Level 3, which constituted 56% of the overall unique potential problems. It is followed by Level 2 with 20%; and Levels 4 and 1 with 11% and 6% respectively. TIW detected more potential problems at Levels II (mean=4.33), III (mean=9.33) and IV (mean=1.67); but equal numbers at Level I (but higher mean of 0.67). Based on these results, on average, both experiments could suggest that TIW was able to help inspectors to detect more potential problems at the higher level of severity (important potential problems). This supports the second supporting hypothesis of this research.

(a) Experiment I



MEAN	Level 1	Level 2	Level 3	Level 4
TIW	0.2	9.1	0.9	0.0
HE	0.0	4.3	1.1	0.0
CW	0.0	3.3	1.1	0.0

(b) Experiment II



MEAN	Level 1	Level 2	Level 3	Level 4
TIW	0.67	4.33	9.33	1.67
HE	0.33	2.33	2.67	0.67
CW	0.33	2.00	0.67	1.00

Figure 8-2: Detected Potential Problems by Severity Level for Experiments I and II

It is also interesting to see the different trends of judging the severity levels between inspectors in Experiments I and II. In Experiment I, the detected potential problems were mostly rated as level 2, which is a minor potential problem. As a reminder of inspectors' background, Experiment I was done by two groups of students – a group with a background in software engineering and a group with a background Music Technology. Experiment II was done by researchers as well as lecturers that already had experience in doing research (related to sound). The tendency of judging and giving a middle level of severity could be due to lack of experience and level of confidence of inspectors in Experiment I. They potentially did not want to judge the potential problems rigorously but rather to give a 'fair judgement'. As a result, most of the potential problems were categorized as Level 2, which was a minor potential problem that should be given a low priority. On the other hand, with more experience and confidence, inspectors in Experiment II used the severity level wisely. As a result, the potential problems were rated using all the available severity levels.

ANOVA was performed to test the effect and interaction within the two factors (inspection technique and severity level) in Experiment I, which is shown in Table 8-9. The results show that: -

1. Both factors 'inspection technique' and 'severity level' have significant main effects on individual detection potential problems. This shows that the detection performances of TIW, HE and CW were significantly different. The difference in performances in each severity levels was also significant.
2. There was a significant interaction between the 'inspection technique' and the 'severity level' factors. This shows that TIW, HE and CW could detect potential problems at any level of severity.

Table 8-9: The Effects of Technique and Severity Level in Individual Detection Potential Problems for Experiment I

ANOVA Testing Technique x (Severity Level)	p-value
Technique	$p<0.05$
Severity Level	$p<0.05$
Technique x Severity Level	$p<0.05$

A Two-factor mixed factorial ANOVA was performed to test the effect and interaction between 'inspection technique' and 'severity level' factors in Experiment II. The results are shown in Table 8-10.

1. The inspection techniques and severity level factors have significant main effects towards the individual detecting potential problems. These results are equivalent to the results of the Experiment I.
2. There was no interaction between the inspection techniques and severity level, which is opposite to Experiment I. This result shows that all techniques have the same effects upon each severity level. The reason could be due to the experience of inspectors that help them to detect common problems that are easy to determine their effects through the severity levels. This made the effects of individual detection performance in each severity level the same for each inspection technique.

Table 8-10: The effects of Technique, Severity Level in Individual Detection Potential Problems for Experiment II

ANOVA Testing **Technique x (Severity Level)**	**p-value**
Technique	$p<0.1$ ($p=0.07$)
Severity Level	$p<0.1$ ($p=0.05$)
Technique x Severity Level	$p>0.1$

In summary, TIW was able to detect more potential problems in almost all the severity levels, and these results were found to be statistically significant. **Therefore, the overall results suggest that the proposed usability inspection (TIW) was significantly able to help inspectors to detect more and unique 'important potential problems' in overall performance, which supports the second supporting hypothesis, H_2.**

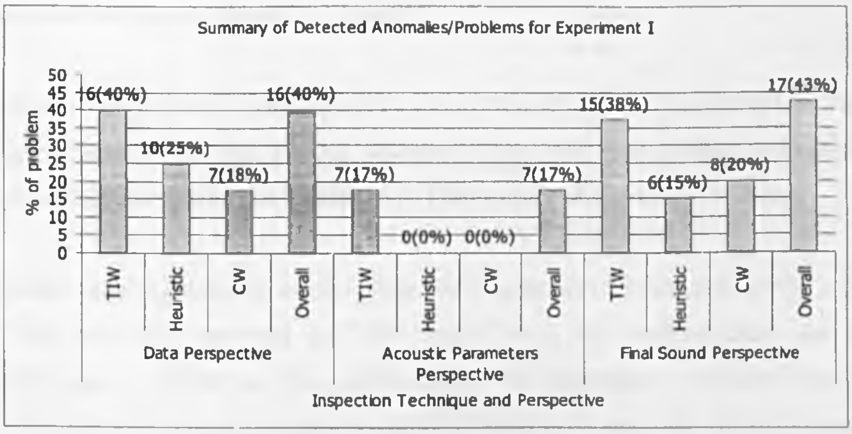
8.2.3 Supporting Hypothesis 3 (H_3)

This section analyses the results to support the third supporting hypothesis (H_3). The supporting hypothesis mentions, *"the proposed usability inspection technique (TIW) should be able to detect significantly more **important** potential problems **in each perspective (data, acoustic parameters and final sound)** compared to existing inspection techniques"*. Therefore, by comparing with the existing inspection techniques, focusing on different perspectives should significantly improve the detection performance of potential problems. In order to support this, the problems detected by each inspection technique were divided into three perspectives – namely Data, Acoustic Parameters and Final Sound perspective.

8.2.3.1 Performance by Perspective

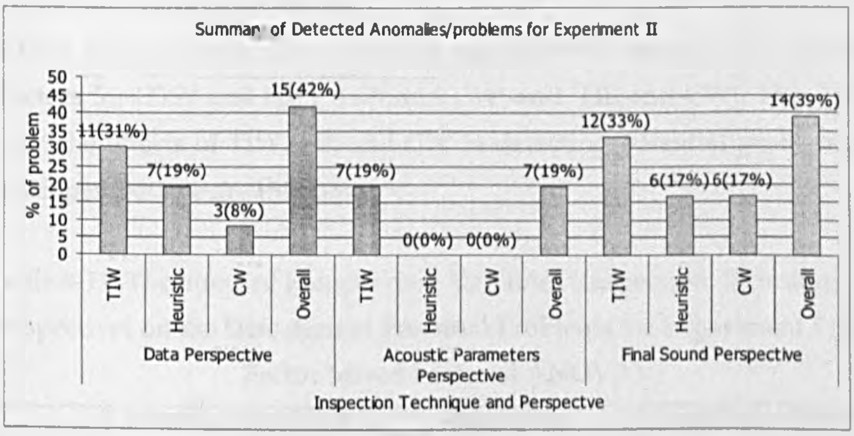
Figure 8-3 shows the summary of detected potential problems in three perspectives for Experiments I and II. In Experiment I, the graph shows that the majority of potential problems were found in the Data Perspective category with 40% from the total number of unique potential problems. It is followed by the Final Sound (43%) and Acoustic Parameters (17%) perspectives. TIW found the highest number of potential problems across the three perspectives, which was followed by HE and then CW. TIW also shows the highest mean values across all the three perspectives with 5, 2 and 3.2 for Data, Acoustic Parameters and Final Sound perspecuives respectively.

The trend also occurred in Experiment II, where TIW also detected the highest number of potential problems across the three perspectives – with 31% (mean=6.7), 19% (mean=4) and 33% (mean=5.3) for data, acoustic parameters and final sound perspective respectively. In general, both graphs and means (Experiments I and II) above show that TIW was able to help inspectors to detect more potential problems in each perspective.



Mean – Experiment I

	Data Perspective	Acoustic Parameters Perspective	Final Sound Perspective
TIW	5.0	2.0	3.2
HE	3.7	0.0	1.7
CW	2.7	0.0	1.7



Mean – Experiment II

	Data Perspective	Acoustic Parameters Perspective	Final Sound Perspective
TIW	6.7	4.0	5.3
HE	3.7	0.0	2.3
CW	2.0	0.0	2.0

Figure 8-3: Summary of Detected Potential Problems in Three Perspectives for Experiments I and II

Were these results significant? (All the p values are referring and based on the means above unless otherwise stated)

Factorial ANOVA was performed to check the effect of 'inspection technique' and 'perspectives' factors on the detection of potential problems in Experiment I. The results are shown in Table 8-11. The results show the following:

1. There was significant main effect for inspection technique factor between 'TIW and HE' as well as 'TIW and CW'. This means that one of the techniques differs in its performance in detecting potential problems, which from the graphs shows that TIW outperformed the other technique.
2. The Perspective factor has a significant main effect for 'TIW and HE'; 'TIW and CW' and 'HE and CW'. This shows that each perspective has different effects towards the detection potential problem by each technique.
3. There was no significant main effect for technique factor for 'HE and CW', which indicates that both techniques were performing the same in detecting potential problems.
4. There was no interaction between inspection technique and perspective factors for 'TIW and HE'; 'TIW and CW' and 'HE and CW'. This indicates that the effects of TIW, HE and CW in detecting potential problems upon each perspective are the same.

Table 8-11: The effect of Independent Variables (Inspection Technique and Perspective) on the Detection of Potential Problems for Experiment I (Two-Factor Mixed Factorial ANOVA)

	Inspection Technique	Perspective	Technique x (Perspective)
TIW and HE	$p<0.05$	$p<0.05$	$p>0.05$
TIW and CW	$p<0.05$	$p<0.05$	$p>0.05$
2-Factor within subject ANOVA			
HE and CW	$p>0.05$	$p<0.05$	$p>0.05$

(N=10 in each technique)

Factorial ANOVA was also performed to the results of Experiment II. The results are shown in Table 8-12. The results of main effects and interaction between all factors show the same pattern as in Experiment I. Except for the 'HE and CW', where the interaction between the two factors (technique and perspective) was found to be significant. This indicates that HE and CW have different effects upon each perspective.

Table 8-12: The effect of Independent Variables (Technique and Perspective) on the Detection of Potential Problems for Experiment II (Two-Factor Mixed Factorial ANOVA)

	Technique	Perspective	Technique x (Perspective)
TIW and HE	$p<0.1$ ($p=0.07$)	$P<0.1$	$p>0.1$
TIW and CW	$p<0.1$ ($p=0.05$)	$p<0.1$ ($p=0.02$)	$p>0.1$
2-Factor within subject ANOVA			
HE and CW	$p>0.1$	$p<0.1$	$p<0.1$ ($p=0.05$)

(N=3 in each technique)

Now, we look at the significant differences of individual detection of potential problems in each perspective. Independent T-Test, Related T-Test, Mann-Whitney and Wilcoxon were run to test the significant differences between two techniques ('TIW and HE'; 'TIW and CW'; and 'HE and CW') towards the individual detection of potential problems in each perspective. The statistical results for Experiment I are shown in Table 8-13. The results show that:

1. Referring to Table 8-13 (a), there were significant differences of results in the Acoustic Parameters and Final Sound Perspectives of individual detection of potential problems. However, there was not enough evidence of significant differences in the Data perspective.
2. Referring to Table 8-13 (b), the Independent t-test and Mann-Whitney test show significant differences for all perspectives in individual detection of potential problems for TIW and CW.
3. Referring to Table 8-13 (c), the statistical tests were insufficient to justify the differences in individual detection of potential problems as significant in all perspectives for HE and CW.

Table 8-13: Significant Differences between Techniques in Each Perspective for Experiment I

(a)

Perspective\technique	Task Interpretation Walkthrough (TIW) and Heuristic Evaluation (HE)	
	Independent t-test	Mann-Whitney test
Data	$P>0.05$ ($p=0.09$)	$p>0.05$ ($p=0.13$)
Acoustic Parameters	$p<0.05$	$p<0.05$
Final Sound	$p<0.05$	$p<0.05$

(b)

Perspective\technique	Task Interpretation Walkthrough (TIW) and Cognitive Walkthrough (CW)	
	Independent t-test	Mann-Whitney test
Data	$p<0.05$	$p<0.05$
Acoustic Parameters	$p<0.05$	$p<0.05$
Final Sound	$p<0.05$	$p<0.05$

(c)

Perspective\technique	Cognitive Walkthrough (CW) and Heuristic Evaluation (HE)	
	Related t-test	Wilcoxon test
Data	$p>0.05$	$p>0.05$
Acoustic Parameters	both techniques were unable to detect problems in acoustic parameters	
Final Sound	$p>0.05$	$p>0.05$

The same statistical tests were also done for Experiment II as shown in Table 8-14. The results show the same pattern as in Experiment I. All perspectives have shown significant differences in individual detection of potential problems for TIW and CW as shown in Table 8-14(b). However, as in Experiment I, the statistical test also failed to reveal significant differences in individual detection of potential problems for the Data perspective in TIW and HE as shown in Table 8-14(a). Both existing inspection techniques were also insufficient to justify their significant difference in individual detection of potential problems as shown in Table 8-14(c).

Table 8-14: Significant Differences between Techniques in Each Perspective for Experiment II (significance level is 0.10)

(a)

Perspective\technique	Task Interpretation Walkthrough (TIW) and Heuristic Evaluation (HE)	
	Independent t-test	Mann-Whitney test
Data	$p>0.1$	$p>0.1$
Acoustic Parameters	$p<0.1$ ($p=0.06$)	$p<0.1$ ($p=0.04$)
Final Sound	$p<0.1$ ($p=0.07$)	$p<0.1$ ($p=0.07$)

(b)

Perspective\technique	Task Interpretation Walkthrough (TIW) and Cognitive Walkthrough (CW)	
	Independent t-test	Mann-Whitney test
Data	$p<0.1$ ($p=0.07$)	$p<0.1$ ($p=0.05$)
Acoustic Parameters	$p<0.1$ ($p=0.06$)	$p<0.1$ ($p=0.04$)
Final Sound	$p<0.1$ ($p=0.07$)	$p<0.1$ ($p=0.08$)

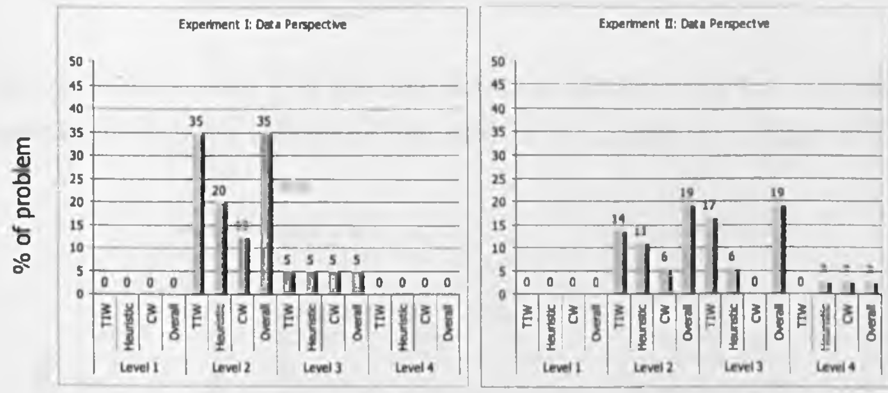
(c)

Perspective\technique	Cognitive Walkthrough (CW) and Heuristic Evaluation (HE)	
	Related t-test	Wilcoxon test
Data	$p>0.1$	$p>0.1$
Acoustic Parameters	both techniques were unable to detect problems in acoustic parameters	
Final Sound	$p>0.1$	$p>0.1$

In summary, the statistical tests above show that the 'inspection technique' and 'perspectives' factors have significant main effects on inspection performance. Apart from the Data perspective between TIW and HE, the statistical tests show significant differences of individual detection of potential problems in all perspectives. It is also found that there were no significant differences between TIW and HE for 'data perspective' in both Experiments I and II. This indicates that TIW was at least equivalent with HE in detecting potential problems related to data transformation. In addition, **TIW has significantly improved in detecting potential problems that are related to sound, which is important, as this is the nature of sonification application and the focus of the proposed technique.** Next, we will look at the severity level of potential problems in each perspective.

8.2.3.2 Performance by Perspective and Severity Level

Figure 8-4 shows the performance of TIW, HE and CW in detecting potential problems by severity level in the Data perspective for Experiments I and II. The graphs show that TIW has detected more potential problems in severity level 2 for Experiment I; and severity levels 2 and 3 in experiment 2. As repeated from the statistical test above, TIW has only shown significant differences with CW but not with HE in the Data perspective. As we can observe from the graphs that, TIW performs either equivalent or less than existing techniques for the higher level of severity, e.g. TIW performed equivalent at levels 3 and 4 in Experiment I; and performed less at level 4 in Experiment II.



Severity Level and Inspection Technique

Figure 8-4: Detected Potential Problems by Severity Level in the Data Perspective for Experiments I and II

Figure 8-5 shows the performance of the three inspection techniques in detecting potential problems by severity level in the 'Acoustic Parameters' perspective. **Both graphs show that only TIW detected potential problems. This indicates that TIW is able to critically analyse and allow the inspectors to look into the detail of design of acoustic parameters to detect any potential problems.**

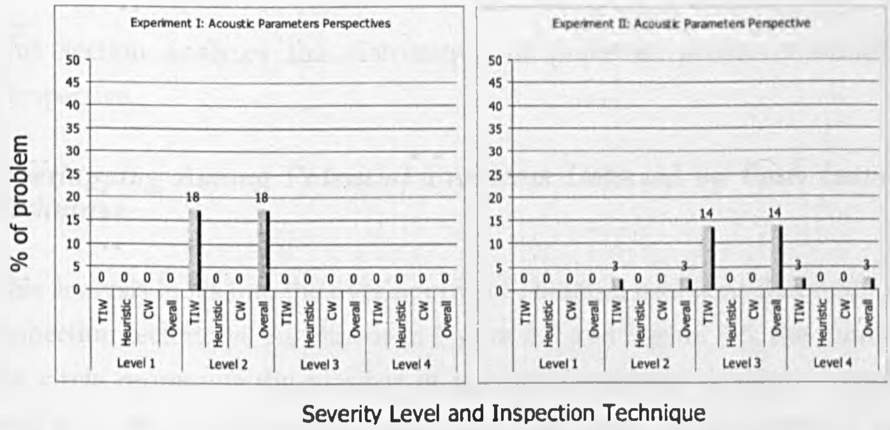


Figure 8-5: Detected Potential Problems by Severity Level in the Acoustic Parameters Perspectives for Experiments I and II

Figure 8-6 shows that TIW has detected more potential problems in almost all severity levels in the Final Sound perspective compared to existing techniques.

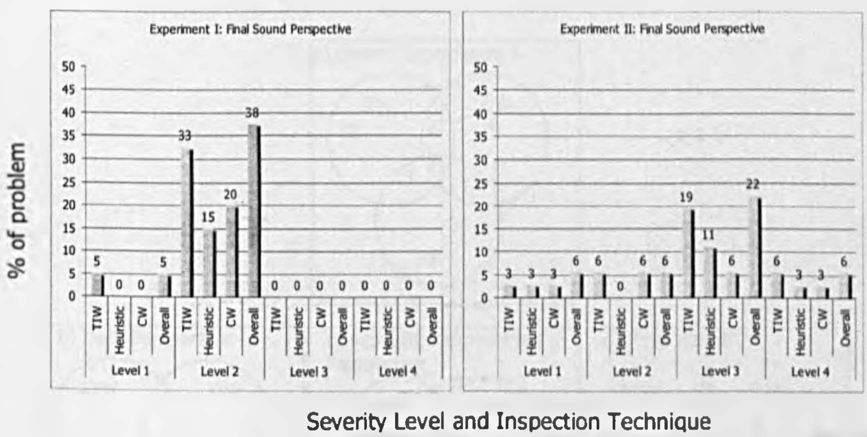


Figure 8-6: Detected Potential Problems by Severity Levels in the Final Sound Perspectives for Experiments I and II

In summary, TIW was found to be able to detect mostly at higher severity levels of potential problems in each perspective.

8.2.3.3 Distribution of potential problems in different perspectives

This section analyses the distribution of potential problems across each perspective.

Overlapping Among Potential Problems Detected by Each Inspection Technique

This analysis looks into the overlapping of potential problems detected by each inspection technique. As shown in Figure 8-7 and Figure 8-8, the numbers in the circle represents the number of potential problems detected uniquely as well as in the overlap of the inspection techniques. In Experiment I (Figure 8-7), for overall performance, there were only 11 potential problems detected by all inspection techniques, where half of them were in the Data Perspective and the other half were in the Final Sound Perspective. Almost all potential problems detected by existing techniques were also detected by TIW. TIW was able to detect about 87% of the potential problems detected by CW and 100% of those detected by HE.

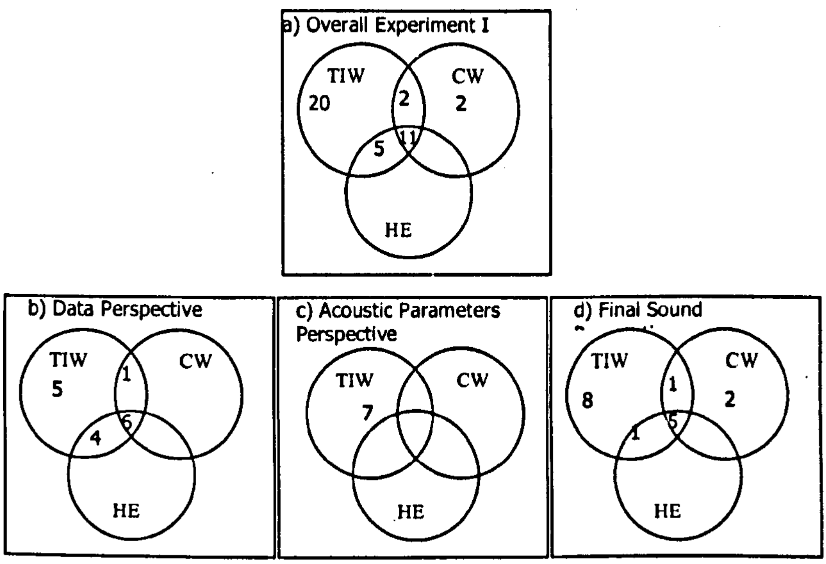


Figure 8-7: Overlap of Potential Problems by inspection techniques for Experiment I

Figure 8-8 shows the overlap of detected potential problems in Experiment II. TIW was also found to be able to detect most of the problems detected by all existing inspection techniques. Only six (17%) out of thirty-six detected potential problems could not be detected by TIW. About 67% and 62% of the potential problems detected by CW and HE respectively were also detected by TIW.

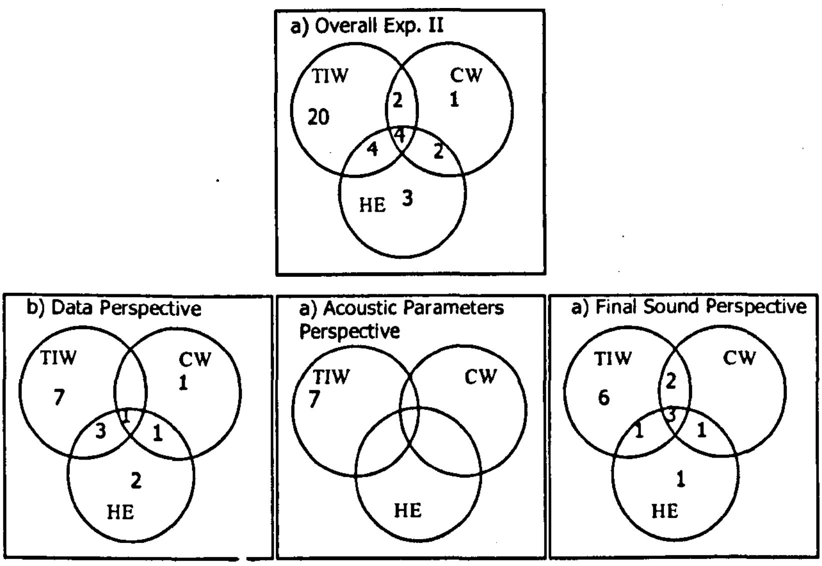


Figure 8-8: Overlap of Potential Problems by Inspection Techniques for Experiment II

As a summary of the overlapping figures above, it shows that most of the potential problems that were detected by both existing inspection techniques (HE and CW) can also be detected by TIW.

Potential Problems Detected by Only One Technique

Based on the Figure 8-7 and Figure 8-8 above, we can see the number of problems uniquely detected by only the respective inspection technique. The summary of this analysis for both experiments is given in Table 8-15.

Table 8-15: Potential Problems that Uniquely Detected by only one Technique in Experiment I and II

Technique	Percentage of problems	
	Experiment I	Experiment II
Task Interpretation Walkthrough	50%	56%
Heuristic Evaluation	0%	8%
Cognitive Walkthrough	5%	3%

From the table above, TIW was able to detect uniquely more than half of the overall potential problems. However, both existing techniques were not consistent, where CW detected 5% more unique potential problems than HE in Experiment I. On the other hand, HE detected 5% more unique potential problems than CW in Experiment II.

8.2.4 Summary of Experiments I and II

The results from Experiments I and II have shown so far, are summarized below:

Overall Performance

- In overall performance, TIW detected more than double the existing problems (compared to the existing techniques) in Experiments I and II. The results were found to be statistically significant.
- Based on the number and percentage of total potential problems detected by both inspection techniques, HE performed higher than the CW. Therefore, even though the statistical tests failed to show the differences between HE and CW as significant, HE was still used as the control experiment for Experiments III and IV.

Order of inspection technique

- The 'order of inspection technique' was found to be not significant. This suggests that the inspectors could not be influenced by their previous knowledge of inspection technique as long as they followed the instructions.

Inspector's background

- In terms of inspector's background, inspectors with a background in software engineering performed better than inspectors with a background in music technology. This was statistically significant only for TIW and HE, but not for CW. The reason could be due to software engineering students having a better knowledge of usability.
- However, even the inspectors from music technology who used TIW performed better than software engineering inspectors using existing techniques (HE and CW). This suggests that TIW is able to encourage inspectors who have no formal knowledge about usability, but at least having knowledge of the application domain (sound), to detect more potential problems.
- 'Background' significantly affects the overall effectiveness of TIW – therefore, by combing both backgrounds and inspecting a design at the same time, we could ensure a better performance than for a single inspector who has formal knowledge in usability. This will be further investigated in Experiments III and IV, which compare the performance between pair-inspectors (knowledge in sound and usability) and single inspectors (knowledge in usability alone).

The importance level of detected potential problems

- TIW was able to detect more potential problems at a higher severity level compared to HE and CW.
- The level of importance (using severity level) is more reliable if it was rated by a more experienced inspector.

Performance in each perspective

- From the graphs of Experiments I and II, TIW was able to detect more potential problems across all the perspectives (data, acoustic parameters and final sound) compared to the existing techniques (Heuristic Evaluation and Cognitive Walkthrough).
- TIW performed more or less the same as HE in the Data Perspective as the statistical tests appear to be non-significant in both experiments.

- In terms of severity levels, TIW was also able to detect more important potential problems across all three perspectives compared to the existing techniques.
- TIW was also able to detect uniquely more potential problems as well as most of the problems detected by the existing techniques.

Next, we will look at the analysis results of Experiments III and IV.

8.3 Analysis of Experiments III and IV

This section explains the second analysis group – Experiments III and IV. The two experiments used different sonification application designs as explained in Chapter 7. The applications were a 'Diagnosis Tool for Analysis of the Motion and Usage of Patient's Arm' for Experiment III; and an 'Audio-Visual Analysis Tool for Cervical Sample Slides (AVATCSS)' for Experiment IV. Experiments III and IV were comparing only two inspection techniques – our novel Task Interpretation Walkthrough (TIW) and Heuristic Evaluation (HE). HE was used as this technique outperformed CW in Experiments I and II. The following sections are also arranged and described based on the three supporting hypotheses. A summary of overall results for Experiments III and IV is also given.

8.3.1 Supporting Hypothesis 1 (H_1)

This section evaluates the first supporting hypothesis (H_1) for both Experiment III and IV. As for Experiments I and II, by giving the same setting of experiments, the hypothesis H_1 would be supported if TIW significantly improves the detection of potential problems over the existing inspection technique. The overall performances and statistical tests for both experiments are discussed in this section.

8.3.1.1 Individual Detection Effectiveness for All Potential Problems

Table 8-16 shows the number and percentage of detected potential problems by each technique as well as the mean of potential problems detected by inspectors. The total numbers of **unique** potential problems that were detected by both inspection techniques are 36 in Experiment III and 50 in Experiment IV. On average in Experiment III, TIW detected 89% of the overall unique potential problems, which is 28% more than HE. The means also show the inspectors with TIW was able to detect more problem with scores average of 14.8 compared to HE with scores average of 8.83. In Experiment IV, TIW detected up to 32% more than HE of the total unique potential problems. The means also show that inspector with TIW was able to detect more problems with average of 15.7 potential problems compared to HE with average only 7.67 as shown in table below. Generally, these results show that TIW is able to help inspectors to detect more potential problems than the existing technique (HE).

Table 8-16: Overall Performance for Experiment III and Experiment IV

Inspection Techniques	**Experiment III** *(N=6 for each technique)*		**Experiment IV** *(N=6 for each technique)*	
	Number and Percentage of Potential Problems	**Mean**	Number and Percentage of Potential Problems	**Mean**
Tasks Interpretation Walkthrough (TIW)	32 (89%)	14.8	42 (84%)	15.7
Heuristic Evaluation (HE)	22 (61%)	8.83	26 (52%)	7.67
Total Problem	**36 (100%)**		**50 (100%)**	

Were these results statistically significant? As in experiment I and II, this can be done by investigating the probability of the results to be happened by chance, or known as p value. All the p values are referring to and based on the means above unless otherwise stated.

Independent t-test and Mann-Whitney tests were conducted to test the significance of differences in performance between TIW and HE. The statistical test results of both experiments are shown in Table 8-17. The tests found that the differences in individual detection scores in both experiments to be significant at the traditional significance level of 0.05.

Table 8-17: Significant Differences in Detecting Potential Problems between TIW and HE for Experiment III and IV

N=6 in each technique	**Experiment III**		**Experiment IV**	
	Independent t-test	**Mann-Whitney test**	**Independent t-test**	**Mann-Whitney test**
Inspection Technique (TIW and HE)	$p<0.05$	$p<0.05$	$p<0.05$	$p<0.05$

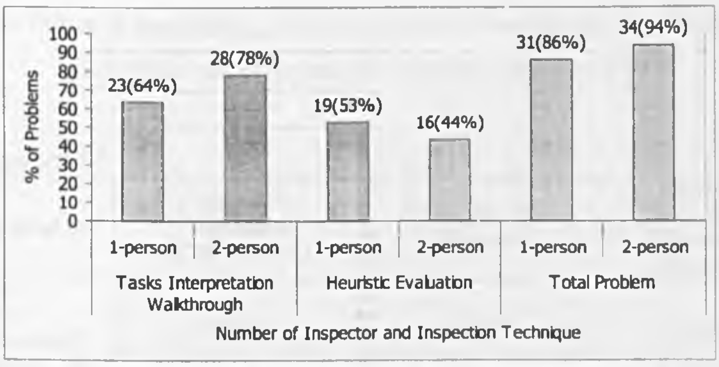
In summary, it is evident that TIW has significantly improved the inspector's detection effectiveness for potential problems of sonification applications compared to HE. The next section investigates the influence of the number of inspectors towards the detection performance.

8.3.1.2 Effects of number of Inspector on the Inspection Technique

As mentioned in the experimental design of Experiments III and IV (Chapter 7), the inspectors were split into two groupings– single and pair inspectors. All single inspectors have formal knowledge in software engineering, while all pair inspectors consisted of two subjects – one inspector with a formal knowledge in software engineering and one inspector with a formal knowledge in music technology.

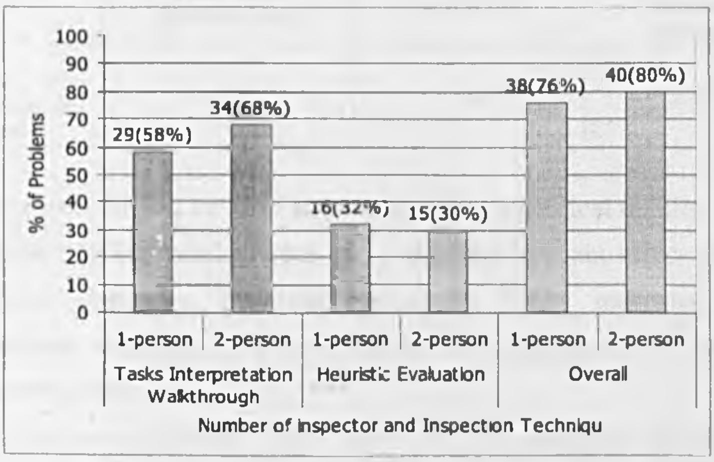
Figure 8-9 a) and b) shows the performance results for Experiments III and IV respectively. The bar graph shows the number and percentage of detected unique potential problems. For overall performance, the pair inspectors have performed slightly better than single inspectors. In Experiment III, TIW has also shown this same trend of results, where the 2-person and 1-person inspectors detected 78% and 64% of overall unique potential problems. The means also show that the '2-person' detected more problems than the '1-person' with average of 17 and 12.7 respectively. HE performed the opposite way, where the single inspector detected more unique potential problems than the pair. However, based on the means, the pair inspector detected slightly more problems than the single. In Experiment IV, the same trend of results was also occurred. The pair inspector and single inspectors detected 68% and 58% of the overall unique potential problems for TIW. The means also show that the pair detected more problems than the single with average of 18 and 13.3 respectively. As in Experiment III, HE also performed the opposite way, where the single inspector detected more unique potential problems than the pair. However, based on the means, the pair inspector detected slightly more problems than the single. **This suggests that TIW could encourage discussions in the inspection process. And HE could also encourage discussion to detect more potential problems but the type of problems could be limited.**

a)



MEAN	1-person	2-person
TIW	12.7	17.0
HE	8.67	9.0
Overall	10.7	13.0

b)



MEAN	1-person	2-person
TIW	13.3	18.0
HE	7.33	8.0
Overall	10.3	13.0

Figure 8-9: Percentage of Detected Potential Problems by Number of Inspectors for a) Experiment III and b) Experiment IV

Factorial ANOVA was run to test the effects of the two factors – 'inspection technique' and 'number of inspectors'. The 'inspection technique' factor consists of TIW and HE; while the 'number of inspectors' factor includes single and pair. The results are shown in Table 8-18. The p values are referring to the mean values above unless otherwise stated.

Table 8-18: Effect of Inspection Technique and Number of Inspectors on overall detection effectiveness in Experiments III and IV

(a) Experiment III

Experiment III	Inspection Technique (TIW and HE)	Number of Inspector	Inspection Technique x Number of Inspector
Factorial ANOVA (N=10 for each technique)	$p<0.05$	$p>0.05$	$p>0.05$

(b) Experiment IV

Experiment IV	Inspection Technique (TIW and HE)	Number of Inspector	Inspection Technique x Number of Inspector
Factorial ANOVA (N=10 for each technique)	$p<0.05$	$p>0.05$	$p>0.05$

Both Experiments III and IV have shown the same statistical results as follows:

1. The inspection technique factor has a significant main effect towards the individual detecting potential problems. This indicates that the performance between TIW and HE in detecting potential problems is significantly different.
2. There was no significant main effect for the number of inspectors in detecting potential problems. This shows that the number of inspectors was not affecting the potential problems detection performances.
3. There was no significant interaction between the 'inspection techniques' and 'number of inspectors' factors. This shows that all techniques have the same effects upon the number of inspectors.

Based on the statistical results above, even though the graphs show that the pair inspectors outperformed the single inspectors, however, the statistical tests failed to reveal the number of inspectors as having a significant main effect. Table 8-19 shows the significant differences of the number of inspectors in detecting potential problems in each inspection technique. The statistical results were also insufficient to justify the differences in detecting potential problems by different numbers of inspectors as significant.

Table 8-19: Significant Differences of Number of Inspectors on Detected Potential Problems for Experiments III and IV

(a) Experiment III

Experiment III Inspection Technique	**p-values (for different number of inspectors, N=3)**	
	Dependent T-test	**Mann-Whitney test**
Task Interpretation Walkthrough (TIW)	$p>0.05$	$p>0.05$
Heuristic Evaluation (HE)	$p>0.05$	$P>0.05$

(b) Experiment IV

Experiment IV Inspection Technique	**p-values (for different number of inspectors, N=3)**	
	Dependent T-test	**Mann-Whitney test**
Task Interpretation Walkthrough (TIW)	$p>0.05$	$p>0.05$
Heuristic Evaluation (HE)	$p>0.05$	$p>0.05$

In summary, the overall results support the first supporting hypothesis H_1, where TIW was significantly able to help inspectors to detect more and unique potential problems in overall performance. However, even though the graphs have shown differences in detecting potential problems between single and pair inspectors, the statistical tests have not provided enough evidence to support it. Both inspection techniques were not influenced by the number of inspectors.

Next, we look at the second supporting hypothesis (H_2).

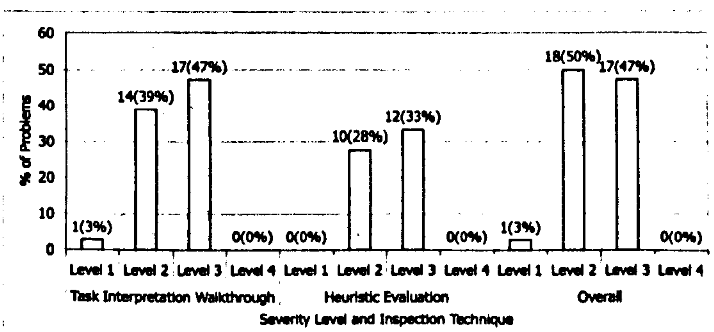
8.3.2 Supporting Hypothesis 2 (H_2)

This section analyses the results to determine the validity of the second supporting hypothesis, H_2. In H_2, TIW should significantly detect *more important* potential usability problems. As in Experiments I and II, Experiments III and IV also used the Neilsen (1993) severity level.

8.3.2.1 Effects of Number of Inspectors towards the Severity Level of the Problems

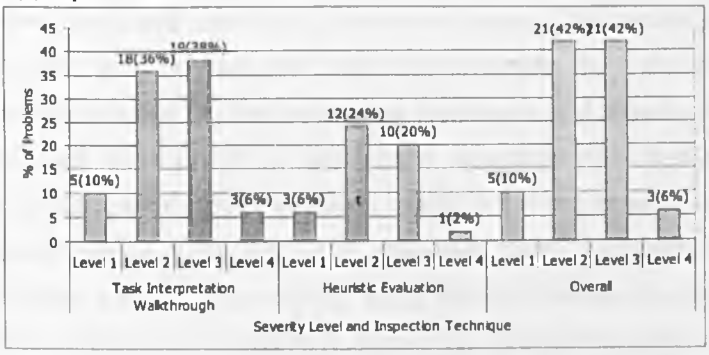
Figure 8-10 below shows the distribution of potential problems and means across the four severity levels. Based on overall performance, most of the problems were grouped as level 2 and 3. TIW detected more potential problems at each level compared to HE. The means value also show that inspectors with TIW detected more problems across all the severity levels than the inspectors with HE. In general, the graphs and means suggest that TIW was able to help inspectors to detect more 'important potential problems'.

(a) Experiment III



MEAN	Level 1	Level 2	Level 3	Level 4
TIW	0.33	6.33	8.17	0.00
HE	0.00	4.00	4.83	0.00
Overall	0.17	5.17	6.50	0.00

(b) Experiment IV



MEAN	Level 1	Level 2	Level 3	Level 4
TIW	0.33	7.67	6.17	1.5
HE	0.5	4.17	2.67	0.33
Overall	0.42	5.92	4.42	0.92

Figure 8-10: Detected Potential Problems by Severity Levels for (a) Experiment III and (b) Experiment IV

ANOVA was run to test the effect and interaction of two factors – inspection technique and severity level. The results are shown in Table 8-20. The p values are referring to the mean values above unless otherwise stated.

1. Inspection technique and severity level factors were found to have significant main effects on individual detection of potential problems. This indicates that the detection of potential problem performances of each technique as well as each in severity level are significantly different.
2. There was no significant interaction between technique and severity level factors. This shows that TIW and HE could detect potential problems at any severity level.

Table 8-20: The Effects of Technique and Severity Level in Detection of Potential Problems for Experiment III

ANOVA Testing Technique x number of inspectors x (Severity Level)	p-value
Technique	$p<0.05$
Severity Level	$p<0.05$
Technique x Severity Level	$p>0.05$

The same ANOVA test was also performed to the results of Experiment IV to evaluate the effects and interaction of the two factors. The results are shown in Table 8-21. The p values are also referring to means from the above Figure 8-10. As in Experiment III, the inspection technique and severity level factors have significant main effects on individual detection of potential problems. Referring to the mean values of Level 1 and 4 for the respective inspection technique in Figure 8-10 (b), TIW detected more problems in Level 4 (mean=1.5) than Level 1 (mean=0.33), while the HE detected more problems in Level 1 (mean=0.5) than Level 4 (mean=0.33). These differences caused the interaction between inspection technique and severity level to become significant in Experiment IV.

Table 8-21: The Effects of Technique and Severity Level in Detection of Potential Problems for Experiment IV

ANOVA Testing **Technique x number of inspectors x (Severity Level)**	**p-value**
Technique	$p<0.05$
Severity Level	$p<0.05$
Technique x Severity Level	$p<0.05$

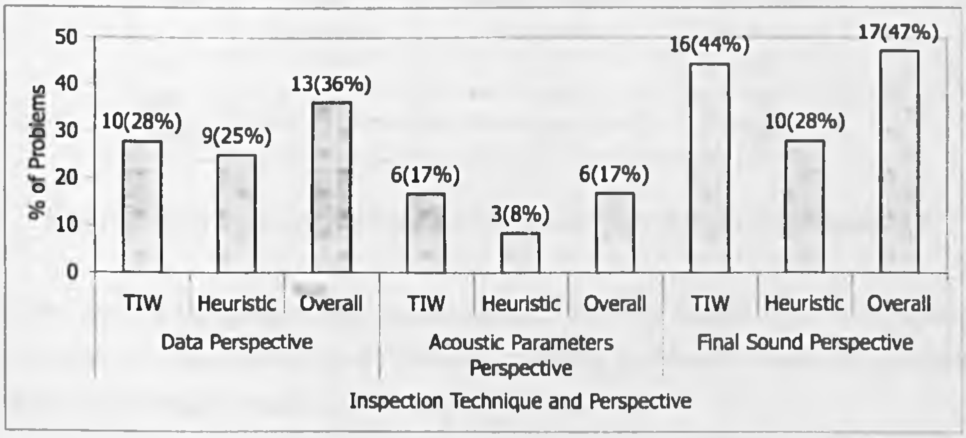
In summary, based on the graphs and statistical results, Experiment III and IV above suggest that **TIW was significantly able help inspectors to detect more and unique 'important potential problems' in overall performance compared to HE, which supports the second supporting hypothesis, H_2.**

8.3.3 Supporting Hypothesis 3 (H_3)

This section analyses the results of Experiments III and IV in supporting the third supporting hypothesis (H_3). As mentioned in previous section, by focusing on the data, acoustic parameters and final sound perspectives, the proposed technique should significantly improve the detection of potential problems in sonification application designs.

8.3.3.1 Performance by Perspective

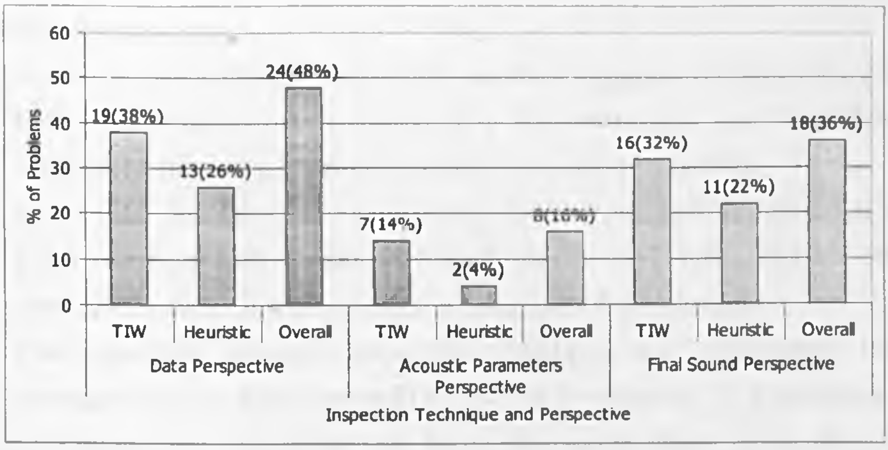
Figure 8-11 shows the summary of detected potential problems in each perspective for Experiment III, where the bar represents the percentage of potential problems. The graph shows that most of the potential problems were found in the Final Sound Perspective. It is followed by the Data Perspective and the Acoustic Parameters Perspective. **TIW was able to detect the highest number of potential problems across all the three perspectives.** However, the means show that HE detected slightly higher number of potential problems compared to TIW only in Data Perspective with 4.83 and 4.5 respectively.



MEAN	Data Perspective	Acoustic Parameter Perspective	Final Sound Perspective
TIW	4.5	1.83	7.67
HE	4.83	0.67	3.33
Overall	4.67	1.25	5.5

Figure 8-11: Summary of Detected Potential Problems for Experiment III

Figure 8-12 shows the summary of potential problems for Experiment IV. The results differed from Experiment III where more data was found in the Data Perspective than the in the Final Sound Perspective. **In this experiment also, TIW has managed to detect more potential problems in all perspectives.** TIW also shows the highest mean values across all the three perspectives with 7, 2.5 and 6.17 for Data, Acoustic Parameters and Final Sound perspectives respectively.



MEAN	Data Perspective	Acoustic Parameters Perspective	Final Sound Perspective
TIW	7	2.5	6.17
HE	3.83	0.33	3.5
Overall	5.42	1.42	4.83

Figure 8-12: Summary of Detected Potential Problems for Experiment IV

From both of the graphs and means above, TIW was found to be able to help inspectors to detect more and different potential problems in each perspective thus supporting the third supporting hypothesis, H_3.

Were these results significant? (All the p values are referring and based on the means above unless otherwise stated)

Factorial ANOVA was performed to evaluate the effect and interaction between the two factors (inspection technique and perspectives) and the results are shown in Table 8-22.

Table 8-22: The effect of Independent Variables (Technique and Perspective) on the Detection of Potential Problems for Experiment III and IV (Two-Factor Mixed Factorial ANOVA)

	Technique	Perspective	Technique x (Perspective)
Experiment III	$p<0.05$	$p<0.05$	$p<0.05$
Experiment IV	$p<0.05$	$p<0.05$	$p>0.05$

The results show that:

1. There was a significant main effect for the inspection technique factor for both experiments, which means that the techniques were significantly different in detecting potential problems in each perspective.
2. There was also significant main effect for the perspective factor for both experiments, which means that each perspective has different effects towards the detection of potential problems by the technique.
3. The interaction between 'inspection technique' and 'perspective' factors was significant in Experiment III but not in Experiment IV. This shows that techniques in Experiment IV have the same effect upon the three perspectives. However, this was not the case in Experiment III, where the techniques have different effects upon the three perspectives.

Now, we will look at the differences of individual detection of potential problems in each perspective. The statistical results for Experiment III are shown in Table 8-23. It shows that there were significant differences in the Acoustic Parameters and Final Sound Perspectives. However, the statistical tests were insufficient to justify the difference as significant in the Data Perspective.

Table 8-23: Significant Differences between Technique in Each Perspective for Experiment III

Perspective\technique	Task Interpretation Walkthrough (TIW) and Heuristic Evaluation (HE)	
	Independent t-test	Mann-Whitney test
Data	$p>0.05$	$p>0.05$
Acoustic Parameters	$p<0.05$	$p<0.05$
Final Sound	$P<0.05$	$P<0.05$

The same statistical tests were also performed on the results of Experiment IV to test the differences of individual detection of potential problems in each perspective, which is shown in Table 8-24. The Independent t-test and Mann-Whitney tests have shown significant differences in all perspectives for individual detection of potential problems.

Table 8-24: Significant Differences between Technique in Each Perspective for Experiment IV

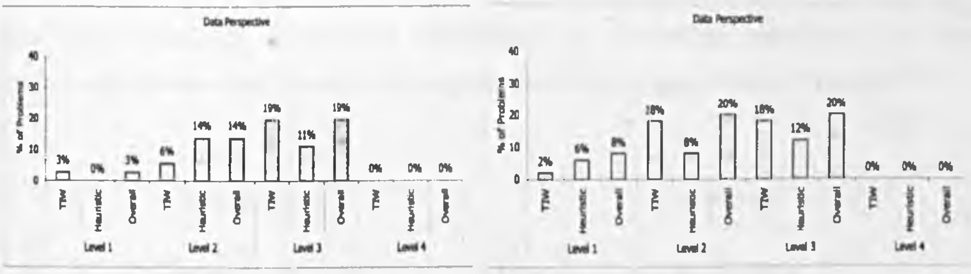
Perspective\technique	Task Interpretation Walkthrough (TIW) and Heuristic Evaluation (HE)	
	Independent t-test	Mann-Whitney test
Data	$p<0.05$	$p<0.05$
Acoustic Parameters	$p<0.05$	$p<0.05$
Final Sound	$p<0.05$	$p<0.05$

In summary, the statistical results show that the techniques and the three perspectives have significant main effects towards the detection performance of potential problems. Both experiments have shown statistically significance in both perspectives – Acoustic Parameters and Final Sound. However, the differences in the Data perspective were significant only in Experiment IV but not in Experiment III. This result suggests that TIW has significantly improved the detection performances of potential problems in the Acoustic Parameters and Final Sound perspectives; but could perform 'better or equivalent' in the Data perspective compared to the existing inspection technique (i.e., Heuristic Evaluation). In general, the graphs and statistical results above support the third supporting hypothesis, H_3.

Next, we will look at the severity level of potential problems to see how important the problems are in each perspective.

8.3.3.2 Performance by Perspective and Severity Level

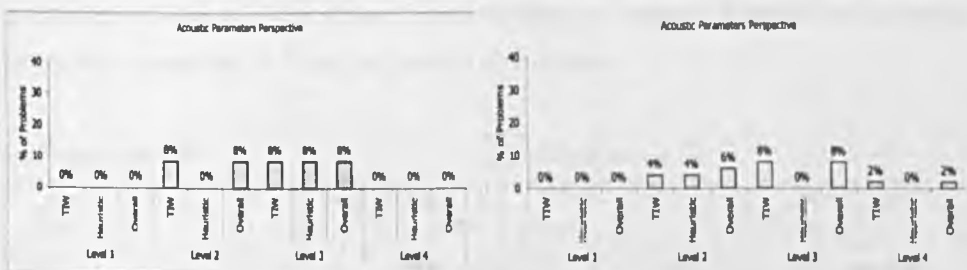
This section analyses the performance of TIW and HE in each perspective by severity level. Figure 8-13 shows the percentage of detected potential problems by severity level for the Data perspective. The graphs show that HE has detected more problems at Level 2 (for Experiment III) and Level 1 (for Experiment IV) compared to TIW. However, TIW has detected more problems at a higher level – Level 3 (for Experiment III); and Level 2 and Level 3 (for Experiment IV). Even though the differences in data perspective were found to be not statistically significant for Experiment III (but statistically significant in Experiment IV), TIW has managed to detect a higher percentage at the higher severity levels.



(a) Experiment III (b) Experiment IV

Figure 8-13: Detected Potential Problems in Severity Levels for Data Perspective

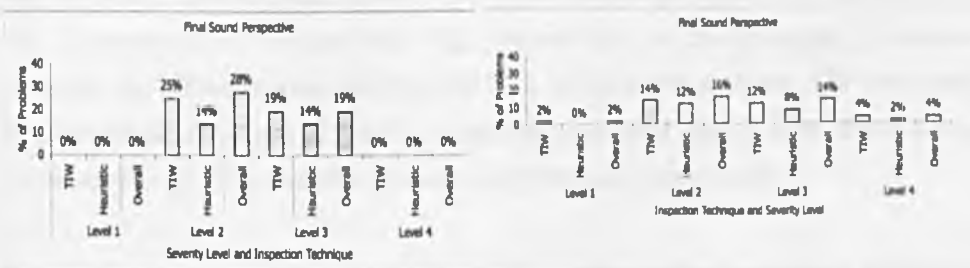
Figure 8-14 shows that not so many potential problems can be detected in the Acoustic Parameters perspective. However, TIW has detected more potential problems in Level 2, but equivalent in Level 3 for Experiment III. TIW has also detected more potential problems at Levels 3 and 4 for Experiment IV. From previous statistical analysis, the individual detection of potential problems in Acoustic Parameters perspective was found to be statistically significant for both Experiment III and IV. This statistical result and the percentage of severity level in Figure 8-14 indicate that TIW is able to significantly detect more higher severity level of potential problems down to the details of acoustic parameters compared to existing technique.



(a) Experiment III (b) Experiment IV

Figure 8-14: Detected Potential Problems in Severity Levels for Acoustic Parameters Perspective

Figure 8-15 shows that TIW has detected more potential problems across all the severity levels in the Final Sound perspective. From the previous analysis, the differences in individual detections of potential problems in this perspective were also found to be significant in both Experiments III and IV.



(a) Experiment III (b) Experiment IV

Figure 8-15: Detected Potential Problems in Severity Levels for Final Sound Perspective

In summary, TIW was found to be able to help inspectors to detect more and unique potential problems at a higher severity level in each perspective.

8.3.3.3 Distribution of potential problems on different perspectives

This section analyses the distribution of potential problems across each perspective. Figure 8-16 shows the numbers of potential problems detected uniquely as well as by both inspection techniques.

(a) Experiment III (b) Experiment IV

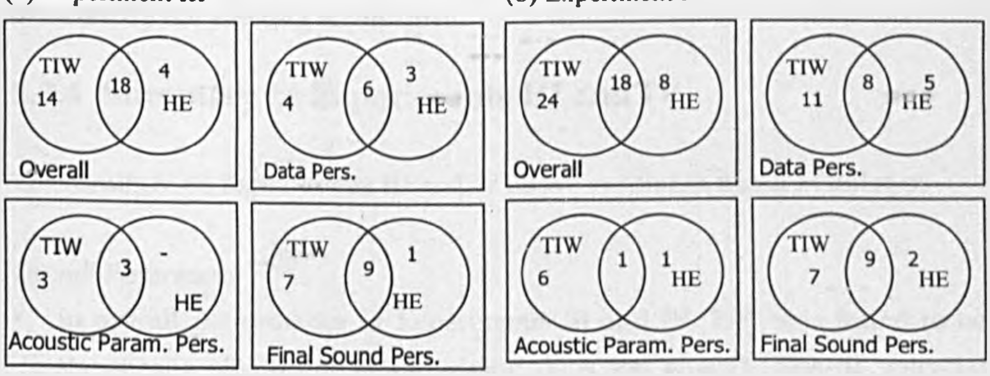


Figure 8-16: Overlapping of Potential Problems by Inspection Technique for Experiment III and IV

Referring to Figure 8-16(a) for Experiment III, around 82% of the potential problems detected by HE were also detected by TIW. In each perspective, TIW was able to detect 67%, 100% and 90% of the potential problems detected by HE in Data, Acoustic Parameters and Final Sound perspectives respectively. For Experiment IV [Figure 8-16 (b)], around 69% of the potential problems detected by HE were also detected by TIW. In each perspective, TIW was able to detect 62%, 50% and 82% of the potential problems detected by HE in Data, Acoustic Parameters and Final Sound perspectives respectively.

From the above results, we can see that most of the potential problems detected by HE can also be detected by TIW. Table 8-25 shows the number of problems uniquely detected by each inspection technique. The analysis shows that 39% and 48% of potential problems were detected uniquely by TIW in Experiments III and IV respectively. HE managed to detect uniquely about 11% and 16% of the overall detected potential problems.

Table 8-25: Potential problems uniquely detected by only one technique for Experiment III

Technique	Number/Percentage of problems	
	Experiment III	Experiment IV
Task Interpretation Walkthrough	14 (39%)	24 (48%)
Heuristic Evaluation	4 (11%)	8 (16%)

In summary, TIW has managed to detect uniquely more potential problems than the existing technique; and cover most of the potential problems detected by the existing techniques.

8.3.4 Summary of Experiments III and IV

The results from Experiments III and IV above are summarized as follows:

Overall Performance

- In overall performance of Experiments III and IV, TIW was found to be statistically significant at improving detection effectiveness of potential problems compared to HE.

Number of inspectors

- For TIW, the inspector pairs performed better than the single inspector in both experiments. The results were different for HE, where the single inspector performed better than the pair inspector. However, statistical testing was insufficient to support the results in both experiments; and found the number of inspectors has no overall significant effect on the performance of problem detection.

The importance level of detected potential problems

- TIW was able to detect more potential problems at a higher severity level compared to HE.

Performance in each perspective

- From the graphs of Experiments III and IV, TIW was able to detect higher numbers of potential problem across all perspectives (data, acoustic parameters and final sound) compared to HE.
- TIW performed more or less the same as HE in the Data perspective as the statistical tests appear to be non-significant in experiments III. The same result was also occurred in Experiments I and II.
- In terms of severity levels, TIW was also able to detect more important potential problems across all three perspectives compared to Heuristic Evaluation.
- TIW was also able to detect uniquely more potential problems as well as most of the problems detected by the existing technique.

8.4 Conclusions

This chapter discussed the analysis of the results of four experiments – I, II, III and IV. The results are presented based on the three supporting hypothesis, H_1, H_2 and H_3. The summary of the analysis can be referred to in Table 8-26 below. **In overall performance, all three supporting hypothesis were supported and shown to be statistically significant.** All experiments found that TIW significantly detected more potential problems in overall performance. All experiments also found that TIW was able to detect significantly more potential problems at a higher severity level compared to the existing techniques (Heuristics Evaluation and Cognitive Walkthrough). Finally, all experiments have shown that the proposed technique was able to detect significantly more important potential usability problems in each perspective.

In conclusion, based on the results of the four experiments, the stated hypothesis is supported: that the designers of sonification applications are able to detect significantly more important potential problems (prior to the implementation phase) by using our novel TIW technique i.e., analysing the task through different views (user, application and interaction) and paying attention to different perspectives (data, acoustic parameters and final sound) in the data state transformations. The overall conclusions, contributions, research limitation and future works of this thesis will be further discussed in the final Chapter 9.

Table 8-26: Summary of Results Analysis

Factor	Experiment I	Experiment II	Experiment III	Experiment IV
Team size	Individuals	Individuals	Individuals and Pairs	Individuals and Pairs
Subject	• 10 students of Masters Degree in Software Engineering (SWE). • 10 students of Masters Degree in Music Technology (MT).	6 Experienced Researchers in Sound	• 6 Students of Masters Degree in Software Engineering • 6 pairs (combination of SWE and MT)	• 6 Students of Masters Degree in Software Engineering • 6 pairs (combination of SWE and MT)
Inspection Technique	• Task Interpretation Walkthrough • Heuristics Evaluation • Cognitive Walkthrough		• Task Interpretation Walkthrough • Heuristics Evaluation	
Sonification Application Design	Mobile Phone Joystick Text-Entry with Sound.		Diagnosis Tool for Analysis of the Motion and Usage of Patient's Arm.	Audio-Visual Analysis Tool of Cervical Sample Slides (AVATCSS).
Hypothesis H_1	H_1 was supported where TIW was able to detect significantly more potential problems in overall performance compared to HE and CW.		H_1 was supported, as TIW was statistically significant in detecting more potential problems in overall performance compared to HE.	
	The performance between HE and CW was not statistically significant.		There were no significant effects for the number of inspectors in detection potential problems performances.	
	• The inspection order has no significant main effects on detection potential problems performances. • Inspectors' background factor has a significant main effect on detection performance for TIW and HE, but no evidence for the CW.		There were also no significant differences between single and pair inspectors in both techniques – TIW and HE.	

<table>
<tr><td></td><td>H_1 was supported, as in all experiments; TIW was able to detect significantly more potential problems in overall performance compared to existing technique(s).</td></tr>
</table>

<table>
<tr><td rowspan="3">Hypothesis H_2</td><td colspan="4">Severity level has significant main effects on individual classification of potential problems.</td></tr>
<tr><td>Each technique has different effects on severity levels.</td><td>Each technique has the same effects at all severity levels.</td><td>Each technique has the same effects at all severity levels.</td><td>Each technique has different effects on severity levels.</td></tr>
<tr><td colspan="4">More problems were found at the highest severity levels compared to existing technique, therefore, H_2 was supported where TIW was able to detect significantly more important potential usability problems.</td></tr>
</table>

<table>
<tr><td rowspan="7">Hypothesis H_3</td><td colspan="4">Perspective factor has significant main effects on individual detection of potential problems.</td></tr>
<tr><td>The existing techniques (HE and CW) have the same effects in each perspective.</td><td>The existing techniques (HE and CW) have different effects in each perspective.</td><td></td><td></td></tr>
<tr><td colspan="2">Results are all significant in all three perspectives between TIW and CW.</td><td></td><td></td></tr>
<tr><td colspan="3">Results are significant in Acoustic Parameters and Final Sound perspectives, but not in Data perspective – between TIW and HE</td><td>Results are all significant in all three perspectives between TIW and HE.</td></tr>
<tr><td colspan="2">All inspection techniques have different effects on detecting potential problems at different perspective.</td><td>All inspection techniques have the same effects on detecting potential problems at different perspective.</td><td>All inspection techniques have different effects on detecting potential problems at different perspective.</td></tr>
<tr><td colspan="2">Results are not significant in all perspective between the two existing techniques – HE and CW.</td><td></td><td></td></tr>
<tr><td colspan="4">TIW was able to detect more potential problems at the higher severity level.</td></tr>
</table>

	Most of the potential problems that detected by the existing techniques can also be detected by TIW.
	The H3 was supported, as TIW was able to detect significantly more important potential problems in each perspective (data, acoustic parameters and final sound).

CHAPTER 9 : SUMMARY OF DISSERTATION WORK

This chapter begins with the contributions of this research, which also includes a revisit of the research objectives, results analysis and problem statements. It is followed by a discussion of the scope and any limitations of this research. Thereafter, ideas for potential future research are discussed.

9.1 Research Contributions

This research:

1. Introduced a Human Computer Interaction (HCI) model for sonification applications, which consists of two models: namely **Sonification Application (SA) model** and **User Interpretation Construction (UIC) model** *[explained in Chapter 4]*.
2. Developed a diagrammatic way to describe sonification application called the **Task-Data State (TDS) diagram,** which is a combination of ConquerTaskTree (CTT) diagram and Data State model *[see Chapter 5]*.
3. Developed a novel inspection technique called **Task Interpretation Walkthrough (TIW),** which is based on the HCI model and uses the TDS diagram *[explained in Chapter 6]*.
4. Conducted four experiments to study the feasibility and effectiveness of Task Interpretation Walkthrough against some existing usability inspection techniques (Cognitive Walkthrough and Heuristic Evaluation) *[explained in Chapters 7 and 8]*.

9.1.1 Research Agenda

This section revisits the research objectives as explained in Chapter 1. Each objective (*in italics*) is discussed in the light of what has subsequently been done in this research and reported in previous chapters.

1. *Look at the possibility of inspecting usability aspects of sonification applications in the early stages of the design process;*

The emphasis of this objective is 'early stages', which means that potential problems should be detected *before* the software goes to the development process. A review of sonification was done to look at the general elements that could influence the design of sonification applications, which is reported in Chapter 2. Three elements were recognised to be important in shaping and influencing a sonification application – namely:

- the goals and objectives of application;
- the inputs and outputs to be processed; and
- the sonification technique (data-sound transformation technique).

Several existing auditory display techniques were also identified, which in general are also differentiable based on the above three elements. Examples of such techniques are Audification, Earcons, Auditory Icons, Parameter Mapping and Model-Based Sonification. Several design approaches were also analysed in order to understand the steps and focuses involved in the development processes. Such approaches included Syntactic, Semantic, Pragmatic, Task-Oriented and Data-State approaches. Finally, several existing tools and applications of sonification were presented to show the potential of where this kind of application can be useful; and therefore, that a proper and systematic design approach should be incorporated into its development process.

In summary, based on 'existing techniques', 'design approaches' and 'existing tools and applications', the design of sonification applications can be emphasised on the three elements (goals/objectives, input/output and sonification technique). Therefore, it was suggested that by focusing on these three elements, usability inspection to detect problems at the earlier design stage is indeed possible.

2. *Review the issues, capabilities and limitations of current sonification applications in terms of usability;*

Several existing usability evaluation methods were explored in order to understand the available techniques and to appreciate their suitability for use with sonification applications. Evaluation methods can be categorised based on the phase in which they occur in the standard software engineering lifecycle i.e., gathering information phase (Inquiry methods), design phase (Inspection) and post-development phase (Testing). These different kinds of evaluations are discussed in Chapter 3. Previous research has shown that most sonification applications are evaluated based on end-user testing with at least a working prototype. In this type of testing, users are required to carry out tasks such as matching, comparing, classifying, memorising, identification etc. The users' performances are recorded and analysed, forming a record of the application's capabilities, useful for further design improvements. Several issues and limitations which impede the usability testing of sonification applications are discussed. Examples of such issues include data reduction, data scaling, mental image of data, musical knowledge requirements, number of acoustic parameters, sound aesthetics etc.

This objective has been carried out, giving an idea of existing usability evaluations, current practise in evaluation of sonification applications, and consideration of issues that potentially cause problems to the final application.

3. *Propose a technique for analysing the tasks in sonification applications, and develop a model which allows us to understand how users interpret the sound output of sonification applications.*

A Human Computer Interaction (HCI) model was introduced (explained in Chapter 4) to help us understand the design of sonification applications. The model is inspired by Norman's HCI Model (1988) as well as the three elements as explained in objective 1 above. Our HCI model for sonification applications consisting of two parts: the Sonification Application (SA) model and User Interpretation Construction (UIC) model.

The SA model is used to describe the design of sonification application:

- from three *perspectives* (Data, Acoustic Parameters and Final Sound);
- using three *task types* (user, application and interaction);
- and considering four *data states* (raw data, processed data, acoustically-ready data and final sound).

The UIC model is used to describe how the user might perceive and interpret the output of the applications, which is described in three levels i.e., Selection, Reasoning and Hypothesising.

In summary, the HCI model was used to describe how the data is transformed into sound (SA model) and how the user could interpret this sound (UIC model).

4. *Propose a systematic usability inspection technique for sonification applications*

A systematic usability inspection technique was introduced to tailor with the characteristics of sonification applications. The technique is called Task Interpretation Walkthrough (TIW) as explained in Chapter 6. This technique involves two parts – inspection preparation and inspection activities. In the preparation phase, several materials and documents required for the inspections are prepared by a chief inspector together with the designer i.e., Task-Data State (TDS) diagram of sonification application design, description of interpretation, description of context of use and inspection method (instruction).

The Task-Data State diagram is based on the SA model, and is a diagrammatic way to describe the design of sonification applications (explained in Chapter 5). The diagram is based on a novel combination of a Data-State Diagram and the ConcurTaskTree. In the TDS diagram, tasks are decomposed into three perspectives (Data, Acoustic Parameters and Final Sound) and views (user, application and interaction) as well as the task inputs and outputs to portray the different states of data.

The list of interpretation is based on the UIC model, which produces a list of conditions (from the Selection Stage); a list of premises (from the Reasoning Stage); and a list of hypotheses (Hypothesis Stage).

The Context of use describes for the inspector the scope of the application so that it is inspected reasonably. The context include the 'users and user tasks'; 'application tasks, equipment and input/output; Interface and Interaction'; and 'Environment'. All these materials are used within TIW to inspect sonification application designs. The full procedure of the TIW can be found in Chapter 6. In general, the core idea of TIW is to understand the rationale behind the design through critical analysis of the tasks (SA model) as well as potential interpretation (UIC model). Thus this objective has been satisfied as we have proposed a new systematic usability inspection technique for sonification applications.

5. *Provide recommendations for usability inspection or evaluation of sonification applications.*

Four experiments have been conducted to evaluate the effectiveness of the Task Interpretation Walkthrough (TIW) by comparing it with two existing inspection techniques: Heuristics Evaluation (HE) and Cognitive Walkthrough (CW). The TIW was found to be statistically significant in detecting more potential problems in overall performance. The discussion and analysis of experimental results can be found in Chapter 8. **Based on the overall results analysis, the hypothesis and the supporting hypotheses of this research were statistically supported.**

As the TIW was developed based on the hypothesis of this research, it is supported that more potential problems can be found in the design of sonification applications by analysing the task through TDS, specifically through its different views (*users, application and interactions*) and paying attention to different perspectives (*data, acoustic parameters and final sound*) in the data state (*raw data, processed data, acoustically ready data and final sound*) transformations.

9.1.2 Experiment Results

This section gives a summary of the hypothesis, supporting hypotheses and the results analysis of this research, which is also explained in detail in Chapter 8. The hypothesis implies that by using Task Interpretation Walkthrough, the designers of sonification applications should be able to detect significantly important potential problems before the implementation phase. Four experiments have been carried out in order to support the hypothesis through three supporting hypotheses, which are summarised below.

Detection of overall potential problems [first supporting hypothesis]

The first supporting hypothesis investigated the overall performance of TIW in detecting potential problems. Table 9-1 shows the percentage of potential problems in each experiment and technique.

Table 9-1: Overall Performance for All Experiments

	Experiment I			Experiment II			Experiment III		Experiment IV	
	TIW	HE	CW	TIW	HE	CW	TIW	HE	TIW	HE
Number and % of overall potential problems by experiment	40 (100%)			36 (100%)			36 (100%)		50 (100%)	
% of potential problems by technique	95%	40%	38%	83%	36%	25%	89%	61%	84%	52%
% of potential problems detected uniquely	50%	0%	5%	56%	8%	3%	39%	11%	48%	16%

On average in all four experiments, Task Interpretation Walkthrough detected significantly more potential problems compared with Heuristic Evaluation and Cognitive Walkthrough[1]. Most potential problems detected by existing techniques were also detected by TIW in all four experiments. TIW managed to detect uniquely 50%, 56%, 39% and 48% percent of potential problems in Experiments I, II, III and IV respectively. In general terms this means that when several inspection techniques were evaluated together, roughly half of all the discovered problems were *only* detected by TIW, and *not* by the existing techniques. This is a great improvement over the status quo.

Detection of major anomalies [second supporting hypothesis]

The second supporting hypothesis investigated the performance of each technique in detecting important potential problems through their severity rates. Table 9-2 shows the overall performance by severity level for each technique in the four experiments. On average, TIW was able to detect more problems at the higher levels of severity (shown in ***bold***).

[1] Cognitive Walkthrough was used only in Experiments I and II.

Severity Level	Experiment I			Experiment II			Experiment III		Experiment IV	
	TIW	HE	CW	TIW	HE	CW	TIW	HE	TIW	HE
% overall potential problems	40 (100%)			36 (100%)			36 (100%)		50 (100%)	
Level 1	5%	0%	0%	3%	3%	3%	3%	0%	10%	6%
Level 2	85%	35%	33%	22%	11%	11%	39%	28%	36%	24%
Level 3	5%	5%	5%	50%	17%	6%	47%	33%	38%	20%
Level 4	0%	0%	0%	8%	6%	6%	0%	0%	6%	2%

Table 9-2: Overall Performance by Severity Level

Detection of different types of potential problems [third supporting hypothesis]

The third supporting hypothesis investigated the overall detection performances of potential problems by three perspectives – data, acoustic parameters and final sound. All potential problems were categorised based on whether they were related to data transformation; acoustic parameters transformation; or final sound transformation. Table 9-3 shows the overall performance by perspective in the four experiments. On average, TIW detected more potential problems across all perspectives. However, the performance comparison of TIW and HE was not statistically significant in the Data perspective, even though visual inspection of the percentages– in Experiments I, II and III (*in bold*) – would appear to show a performance advantage for TIW. This is due to the fact that inspectors using HE mostly detected the same problems, which cause the total percentage become lower.

Table 9-3: Overall Performance by Perspectives

Perspective	Experiment I			Experiment II			Experiment III		Experiment IV	
	TIW	HE	CW	TIW	HE	CW	TIW	HE	TIW	HE
% problems detected by all techniques	40 (100%)			36 (100%)			36 (100%)		50 (100%)	
Data	**40%**	**25%**	18%	**31%**	**19%**	8%	**28%**	**25%**	38%	26%
Acoustic Parameters	17%	0%	0%	19%	0%	0%	17%	8%	14%	4%
Final Sound	38%	15%	20%	33%	17%	17%	44%	28%	32%	22%

Severity level in three perspectives

Table 9-4 shows the overall performance of each inspection technique by severity level in the Data, Acoustic Parameters and Final Sound perspectives respectively. On average, TIW managed to detect more potential problems at higher severity levels compared with existing inspection techniques, but especially in the Acoustic Parameters and Final Sound perspectives. However, in the Data perspectives, there were a few levels where HE outperformed TIW e.g. at Level 4 in Experiment II; at Level 2 in Experiment III; and at Level 1 in Experiment IV. But overall, TIW is able to detect more potential problems at almost all severity levels and in all three perspectives.

Table 9-4: Overall Performance by Severity Level in three Perspectives

(a) Data perspective

Severity Level	Experiment I			Experiment II			Experiment III		Experiment IV	
	TIW	HE	CW	TIW	HE	CW	TIW	HE	TIW	HE
Level 1	0%	0%	0%	0%	0%	0%	3%	0%	2%	6%
Level 2	35%	20%	13%	14%	11%	6%	6%	14%	18%	8%
Level 3	5%	5%	5%	17%	6%	0%	19%	11%	18%	12%
Level 4	0%	0%	0%	0%	3%	3%	0%	0%	0%	0%

(b) Acoustic Parameters perspective

Severity Level	Experiment I			Experiment II			Experiment III		Experiment IV	
	TIW	HE	CW	TIW	HE	CW	TIW	HE	TIW	HE
Level 1	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
Level 2	18%	0%	0%	3%	0%	0%	8%	0%	4%	4%
Level 3	0%	0%	0%	14%	0%	0%	8%	8%	8%	0%
Level 4	0%	0%	0%	3%	0%	0%	0%	0%	2%	0%

(c) Final Sound perspective

Severity Level	Experiment I			Experiment II			Experiment III		Experiment IV	
	TIW	HE	CW	TIW	HE	CW	TIW	HE	TIW	HE
Level 1	5%	0%	0%	3%	3%	3%	0%	0%	2%	0%
Level 2	33%	15%	20%	6%	0%	6%	25%	14%	14%	12%
Level 3	0%	0%	0%	19%	11%	6%	19%	14%	12%	8%
Level 4	0%	0%	0%	6%	3%	3%	0%	0%	4%	2%

Effect of knowledge and number of inspectors on the performance of potential problems detection

Two extra independent variables (that were potentially important in having an effect on the performance of detection potential problems) were tested for each inspection technique. Besides 'inspection technique', the independent variables included the 'number of inspectors' who simultaneously carries out the inspection; and the 'background of inspectors', which represents inspectors' prior knowledge.

Table 9-5 shows the performance between the two knowledge backgrounds of inspectors in Experiment I. The results show that knowledge of usability (Software Engineering) helped the inspectors to detect more potential problems. TIW also encouraged inspectors who have knowledge in application domain (audio) to detect more potential problems. These results were all statistically significant. Combining these two backgrounds has also shown some improvements, which was done in Experiments III and IV.

Table 9-5: Effect of Inspectc r's Background on Potential Problem Detection Performance

Inspectors' Background	Experiment I		
	TIW	HE	CW
Software Engineering	73%	35%	25%
Music Technology	63%	28%	25%

The results of combining the two backgrounds are shown in Table 9-6. The results show that TIW performed better by pair-inspectors compared to single inspectors (with SWE background) in both experiments. The results were opposite for HE where single inspectors performed better than pair-inspectors. Therefore, even though these results were not statistically significant, it would be worthwhile recommending that TIW should be implemented by more than one inspector at the same time to encourage discussion.

Table 9-6: Effect of Number of Inspectors on Potential Problem Detection Performance

Number of inspectors	Inspectors' Background	Experiment III		Experiment IV	
		TIW	HE	TIW	HE
One inspector	Software Engineering (SWE)	64%	53%	58%	32%
Two inspectors	Music Technology (MT) + Software Engineering (SWE)	78%	44%	68%	30%

9.1.3 Research Implications

The implications of this research are discussed based on the problem statements as discussed in Chapter 1 (section 1.2), which are rewritten and discussed below.

1. *Existing usability evaluations for sonification applications are mostly based on empirical user testing with a working system or at least a working prototype. The time and cost to develop the system or prototype could be a waste if the design was found to have high severity problems that require it to be completely redesigned.*

   This research has addressed this problem by introducing a novel usability inspection technique called Task Interpretation Walkthrough, which can be used to detect potential usability problems in a sonification application design. It can be carried out during the design phase of sonification applications. This gives designers an option to detect any potential problems that might occur for the end users. The feedback can be used to further improve the design before it goes to the implementation phase. The inspection can be carried out without a working prototype, which will reduce cost and time.

2. *Evaluations are often implemented in an ad-hoc way by individual designers or researchers, which leads to them being ineffective.*

   This research suggests that the inspection should be headed by a 'chief' inspector co-operating with the designer in the process of preparing the inspection materials. This could avoid any bias due to the design being inspected by the designer. The use of a chief inspector also gives a more objective view of the application, which is likely to produce more potential problems than if the designers do the inspection themselves.

3. *Empirical user testing often focuses on concrete tasks that can be measured and quantified, such as whether an object can be detected, or the speed of response when searching for data. More abstract and perceptual tasks are harder to deduce and quantify, such as how the user understands and analyzes the data.*

Empirical user testing focuses on detecting the 'effect' of a problem and post-analysing the results to imply the 'cause' of the problem. The effects are normally based on measurable testing performances e.g., if the user performed slowly in giving a response during data searching, then the 'cause' might be that the display might not be clear enough for the user.

With inspection techniques, this process is turned around, so that the focus is on identifying the 'cause' and trying to *predict* the 'effect' of the potential problems. TIW encourages inspectors to focus on the interaction between users and the application at three different perspectives (data, acoustic parameters and final sound). During inspection, the inspectors predict which part of the application design may cause potential problems and then put themselves in the user's shoes to predict its potential effects. However, the detection performance also relies on the inspector's knowledge and expertise, which was found statistically significant in this research.

4. *Existing usability inspection techniques are not suitable for inspecting sonification applications because they:*
   a. *do not consider sonification techniques, data and sounds.*
   b. *only focus on WIMP (Windows, Icons, Menus and Pointing devices) and GUI (Graphical User Interface).*

Task Interpretation Walkthrough was designed purposely to consider sonification techniques as well as their inputs (data) and outputs (sounds). Besides 'data' perspective, TIW also gives detailed attention to the sound outpus through two other perspectives called 'acoustic parameters' and 'final sound'. Based on the results analysis in Chapter 9, TIW was found to detect significantly more potential problems related to sounds compared with existing inspection techniques (Heuristics Evaluation and Cognitive Walkthrough)

5. *Certain measurements are often not suitable for sonification applications. For example, in monitoring applications (where sound is used simply to alert the user), a memorizing criterion probably is not very important.*

As mentioned above, inspection techniques focus on identifying the 'cause' of problems and predicting their 'effects'. Therefore, the measurement of performance is no longer such an issue and thus does need to be taken into consideration at the inspection phase of the sonification application design process. However, later in the process, when user tests may still be carried out, such tests can be designed in a more informed manner, based on the outcomes of the usability inspection procedure. Measurements can be designed to be more focussed on spotting certain problems that were predicted by TIW.

6. *Since a sonification application can use different sonification techniques, it is important to evaluate each technique for its effectiveness before proceeding to other criteria such as efficiency and satisfaction.*
   Task Interpretation Walkthrough gives a better insight into sonification applications by giving more attention to the transformation processes involved in converting data into sound. The transformations processes are inspected by dividing them into three general perspectives – data, acoustic parameters and final sound. Each transformation is then further decomposed into three tasks: users, application and interaction. By describing sonification applications at this greater level of detail, we allow inspectors to look deeper into the techniques and scrutinise them to find any potential problems that may hinder the goal of the sonification application.

9.2 Research Limitations

This section considers the potential limitations of this work in terms of study design and the proposed inspection technique itself.

9.2.1 Study Design

This research is, by necessity, limited in terms of the environment in which the experiments were conducted. The limitations include the time to run experiment; the learning process needed for the inspection technique; and the number of subjects (inspectors) for each technique. However, these limits did not prejudice the statistical significance of the experiments.

For all four experiments, subjects were given a fixed period of time to do the inspection. The inspection time given was enough in all experiments as all subjects managed to finish each inspection session within the given period. However, the subjects in control group were required to use two existing techniques (Heuristic Evaluation and Cognitive Walkthrough) in sequence during Experiments I and II. Therefore, even though the time given was enough for the control group, this could also influence and force them to finish each technique as soon as they can. However, the statistical test found that there was no significant effect due to order of inspection, which indicates that the inspection results of the second existing inspection technique for the control group were not influenced by the first inspection technique.

All inspectors were given reading materials, which explained the technique that they would use. The subjects were required to learn about the technique themselves and this made the learning process of the inspection technique non-standard. It could have been better if a proper class to be conducted to teach them on how to conduct the inspection. However, all inspectors received an explanation of the respective technique in each inspection session before they started the inspection process. At the same time, the observer was always available during the inspection process for the inspectors to ask any questions regarding the technique.

For Experiment I, there were 10 subjects for each group – the experiment group (using TIW) and the control group (using HE and CW). The results of Experiments I and II were also important in deciding which of the existing inspection techniques (Heuristics Evaluation or Cognitive Walkthrough) would be used in Experiments III and IV. Due to the limitation in the number of inspectors, the control group were required to use both existing techniques in sequence. Therefore, it was important to know the effect of inspection order towards the detection results of both existing techniques. The 10 available inspectors consisted of two different backgrounds – 5 inspectors with a background in Software Engineering (SWE) and 5 with a background in Music Technology (MT). Therefore, to balance up the inspectors' background in each inspection order, only 8 inspectors were considered. This includes two inspectors with a background in SWE and two inspectors with a background in MT for each inspection order (HE-then-CW and CW-then-HE).

Experiment II had only 6 subjects as it was difficult to get inspectors who have research experience in the application domain (sound). The intention was to have 5 subjects for each control group. However, 2 subjects for the proposed technique were not able to participate, and therefore, the best 3 from the results of the existing techniques were chosen to compare with 3 results of TIW.

In Experiments III and IV, the same subjects were used, which consisted of 3 pair-inspectors and 3 single-inspectors for each technique. Single inspectors only involved subjects with a background in SWE. It would be interesting if single inspectors could also involve subjects with a background in MT for comparison purposes.

9.2.2 Inspection preparation

Task Interpretation Walkthrough requires several documents called inspection materials to be used by inspectors. The inspection materials include the Task-Data State diagram of the sonification application design; Description of Interpretation; Description of Context of Use; Inspection method (instructions) and other information relating to the sonification application design. Since the inspection materials were prepared from scratch only for the purpose of usability inspection, the preparations of these documents require the chief inspector to spend a lot of time with designer to gather all required information.

The Task-Data State diagram of the sonification application to be inspected is the document that requires the longest time than the other documents in the overall inspection preparation phase. However, this could be avoided if the designers were able to use the Task-Data State diagram at the *design* stage to describe their application design so that it could be used straightaway in the inspection materials. As for now, the diagram is created manually which also makes it difficult and slow to be produced.

9.2.3 Results validity

One may ask, how many of the potential usability problems identified by the proposed technique are true usability problems? As mentioned earlier in this thesis the problems detected by the proposed usability inspection technique are always referred to as *potential* usability problems. The word potential is used to indicate that the problem found may not be the actual end user problems. They are only considered as 'real' if an end-user experiences them and they have a negative impact on their performance. The problem of detecting 'unreal' usability problems is actually one of the generic problems with usability inspection. For instance, Jeffries, R. & Desurvire (1992) and Desurvire et al. (1992) reported that only 44% and 28% of potential problems found during inspection turned out to be real in HE and CW respectively.

Therefore, the results that are presented in this thesis also face the same issue of potentially not being the true end user problems. As a reminder, this thesis only compares performances between the proposed technique (TIW) and the two existing inspection techniques (HE and CW). Thus, it is not in the remit of this thesis to validate and confirm that the problems found are genuine. Further research could be done to validate the results of this thesis by comparing the problems found by the proposed technique with the problems that the users actually have with the same real working applications. This is suggested for future research.

9.3 Future Research

The overall research work in this dissertation provides some initial evidence for the effectiveness of the novel Task Interpretation Walkthrough usability inspection technique. Four experiments have been carried out with different conditions as well as to evaluate results replications. Below are some suggestions and guidelines for future work of the Task Interpretation Walkthrough.

Simplifying inspection materials

The preparation of inspection materials was found to be difficult and complicated. If the preparation process and materials could be simplified to make it fast and easy, this could increase the efficiency of this technique, and increase the likelihood of its widespread uptake by researchers and developers of sonification applications.

Comparison with end user testing
It would also be interesting to see a comparison of results between the performance of Task Interpretation Walkthrough and end user testing. This would include the type and number of problems detected by TIW which really do occur in the application with end users. The results can be used to validate the potential problems detected in this research.

Develop diagrammatic software
It could be very helpful if the diagram could be produced by specific diagrammatic software. It could make the analysis of application and inspection materials processes faster than if it was done manually.

Implementing Task Interpretation Walkthrough with visualization application
Many of the transformation processes and perspectives in sonification were inspired from the field of visualization. Therefore, it would be interesting to see how TIW can be generalized and used also for visualization applications, and how it would compare in this domain alongside existing techniques such as HE and CW.

9.4 Conclusion

This chapter has summarised the dissertation work by outlining its contributions, limitations and future research. Overall, this research has introduced an HCI model for sonification applications, which consists of the Sonification Application (SA) model and the User Interpretation Construction (UIC) model. These are used to prepare inspection materials for the novel proposed usability inspection technique for sonification applications called Task Interpretation Walkthrough (TIW).

TIW uses the SA model to describe the design of a sonification application in a new diagrammatic form called a Task-Data State (TDS) diagram. UIC is used to understand the way that users might interpret the output of the sonification application.

A series of experiments has been carried out to evaluate the performance of TIW and support the hypothesis of this dissertation. The results have shown that the research hypothesis was supported, where the significantly important usability problems were able to be detected before the implementation phase by analysing the user, application and interaction tasks and paying attention to the data, acoustic parameters and final sound perspectives.

However, this research encountered a few limitations in its experimental design as well as during the inspection preparation phase. It was difficult to find subjects to become the inspectors especially for the professional and experienced people with a background in audio design. Preparation of inspection materials was also found to take a long time. Finally, several future researches are suggested in order to expend this area. However, without the existence and co-operation of researchers with similar interests, this would not be feasible.

It is hoped that by sharing experimental materials, research findings, experience and knowledge, the community can help this area of research to grow and mature.

APPENDIX A:

EXPERIMENT 1 AND 2

Available only on attached CD

APPENDIX B:

EXPERIMENT 3

Available only on attached CD

APPENDIX C

- HEURSITIC EVALUATION
- TASK INTERPRETATION WALKTHROUGH
- ATTACHMENTS

HEURISTICS EVALUATION

Summary

This document contains the materials for you to do a usability inspection for an application called *'Audio-visual Analysis tool of Cervical Sample Slides'*. This document is divided into FOUR sections explaining about the application and the inspection procedure that you need to follow. Please read all the sections carefully before you start the inspection activity. The summaries of all sections are as follows:

(A) INTRODUCTION	: *Gives you a brief introduction about this document.*
(B) DESCRIPTION OF APPLICATION	: *Gives you a brief description about the application to be inspected.*
(C) INSPECTION PROCEDURE	: *Explains what you should do for the inspection including a description of the tasks that the user will perform on the system and a complete written list of actions needed to complete the tasks.*
(D) INSPECTION RESULT	: *How and where you need to write the problems*

(A) INTRODUCTION

You are going to inspect a '*Audio-Visual Analysis Tool of Cervical Sample Slides*' application to find usability problems or anomalies. A usability 'problem' or 'anomaly' is considered to be any aspect of the design, which if changed, would lead to an improved system. They are issues that may hinder the effective, efficient and satisfying use of the system. To find the possible problems or anomalies of the application, you are required to follow the inspection procedure as explained in Section C below. You need to write all the problems in Section D.

You are to conduct the inspection using heuristics evaluation. A list of usability heuristics will be provided. The provided heuristics will help you to identify usability problems. But you may also report usability problems which may not be covered by these heuristics.

Before you start inspecting the application, this technique requires you in the first place to understand the users and the application. The descriptions about the users and the application are given in Section B.

(B) DESCRIPTION OF APPLICATION

*[Please refer to **Attachment 4**]*

(C) INSPECTION PROCEDURE

In general, this technique provides a list of usability heuristics as your guideline during the inspection. The provided heuristics will help you to identify usability problems, but you may also report usability problems, which are not covered by these heuristics.

Before you start inspecting the application, this technique requires you in the first place to understand the users and the application. Descriptions of the users and the application will be given.

Heuristics inspection is to examine the application and see if any of the usability heuristics is violated. If so, violation must be reported by describing what the application has done wrong.

What you should do;

STEP 1: You need to understand the application through the Descriptions of application, Description of User and Example of tasks to be accomplished.

STEP 2: Step through the example of tasks: use the list of usability heuristics below to identify any kind of anomalies or problems with the application.

List of Usability Heuristics

1. Visibility of system status
The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

2. Match between system and the real world
The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

3. User control and freedom
Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Supports undo and redo.

4. Consistency and standards
Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

5. Error prevention
Even better than good error messages is a careful design, which prevents a problem from occurring in the first place.

6. Recognition rather than recall
Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

7. Flexibility and efficiency of use
Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user so that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

8. Aesthetic and minimalist design
Dialogues should not contair information, which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

9. Error recovery
Help users recognize, diagnose, & recover from errors. Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

10. Help and documentation
Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

STEP 3:

For every problem you have found, rate its 'level of severity'

The Levels include:-

Level 1 = cosmetic problem - only needs to be fixed if extra time is available on project
Level 2 = minor usability problem – fixing this should be given low priority
Level 3 = major usability problem – important to fix, so should be given high priority
Level 4 = usability catastrophe – imperative to fix this before it goes for development

(D) INSPECTION RESULTS

Use the inspection procedure in (C).

1. Please write down the potential problems or anomalies you have found and indicate its severity level - Level 1, 2, 3 or 4.
2. You can explain further the problems you have listed above verbally through voice recording. Please state/say the reference number of the problem you are referring to.
3. If you found that the problem was so difficult to express by writing on paper, you can also explain it verbally through voice recording.
   - On paper, write the reference number of problem and mention '*please refer to verbal explanation*'
   - On recording, state/say the reference number of the problem you are referring to.

-end-

TASK INTERPRETATION WALKTHROUGH

Summary

This document contains the materials for you to do a usability inspection for an application called *'Audio-visual Analysis tool of Cervical Sample Slides'*. This document is divided into FOUR sections explaining about the application and the inspection procedure that you need to follow. Please read all the sections carefully before you start the inspection activity. The summaries of all sections are as follows:

(A) INTRODUCTION
[gives you a brief introduction to what you are going to do]

(B) DESCRIPTION OF APPLICATION AND USERS
[Gives you a brief description of the application to be inspected]

(C) INSPECTION MATERIALS
[List of documents that should be with you during the usability inspection]

(D) INSPECTION REQUIREMENTS
[Important things that you need before you start the inspection]

(E) INSPECTION PROCEDURES
[Explains what you should do for the inspection]

(A) INTRODUCTION

In the experiment, you are going to inspect an application to find usability problems or anomalies. A usability 'problem' or 'anomaly' is considered to be any aspect of the design which, if changed, would lead to an improved system. They are issues that may hinder the effective, efficient and satisfying use of the system. To find the possible problems or anomalies of the application, you are required to follow the inspection procedure of Task Interpretation Walkthrough technique.

In this technique, you will be given a 'list of abstract tasks goal' that the application should achieve and a 'list of contexts in which this application will be used' – as your usability inspection guideline.

Before you start inspecting the application, this technique requires you in the first place to understand the application, the users and the context where this application might be used (*the materials will only be given during the experiment*).

(B) DESCRIPTION OF APPLICATION AND USERS

Please refer to Attachment 4.

(C) INSPECTION MATERIALS

For this inspection, you should have the following documents (attachments):-

(1) Task-Data State Diagram – Attachment 1
(2) Context of Use Descriptions – Attachment 2
(3) Output of Users' Interpretation process - Attachment 3

(D) INSPECTION REQUIREMENTS

You need to understand the application design through all the attached documents (*Inspection Materials*).

[NOTE: YOU NEED TO UNDERSTAND THE DESIGN OF THE OVERALL APPLICATION BEFORE YOU START THE INSPECTION]

(1) Task-Data State Diagram

This diagram is important as you will be using it most of the time during the inspection. This diagram explains the design of the application to be evaluated. It describes how the data is transformed into sound. Please refer to Attachment 1.

(2) Context of Use Descriptions

'Context of use' provides you with information about where this application will be used and operated, e.g. For whom the application is designed, What the application is used for, In what condition it will be used etc.

(3) Output of Users' Interpretation Process

Goals of user's interpretation explain 'what the user should achieve and interpret from the output of the application'. The user's goals of interpretation are divided into three – namely ***selection, reasoning*** and ***hypothesising***. The goal for selection focuses on filtering out and attending only to significant output without involving users cognitive or thinking processes. The goal for reasoning involves the activity where the users construct, arrange or put together the output and try to relate it with their knowledge and experience to make a logical judgement, called premises. The goal of hypothesizing is for the user to be able to make sense, conceptualise or conceive the significance of the premises by also relating them with their knowledge, previous experience or even using their instinct.

(E) INSPECTION PROCEDURES

(I) You need to inspect THREE (3) Sections of the application, as follows:

SECTION 1: INSPECTION OF DATA TRANSFORMATION
SECTION 2: INSPECTION OF ACOUSTIC PARAMETERS TRANSFORMATION
SECTION 3: INSPECTION OF FINAL SOUND TRANSFORMATION

You are advised to inspect one section at one time. Please step through and pay full attention, particularly to the tasks and data states of the specific transformation process in the diagram that you are currently inspecting.

(II) To do the inspection;

[Please remember these steps, as you need to do the same thing for all of the questions]

Below are the procedures for you to follow in this inspection session. The inspection is divided into two steps –

Step 1: General Inspection (looking at the goals and tasks on the TDS diagram)

Step 2: Thorough Inspection (a more detailed look at all the tasks, interpretation stages and context of use).

Step 1 [General Inspection]

In this step, you are required to go through the abstract tasks using the Task Data State (TDS) diagram of the application to be inspected. The materials required are simply the 'TDS diagram' and the 'list of user requirements'.

Inspection Step 1.1

This step becomes the checklist to ensure the application will fulfill the user requirements. For each of the user requirements, you need to step through the abstract tasks of the TDS diagram and make sure that all the requirements are offered by the application. In the case that any requirements have not been offered, this should be reported.

Inspection Step 1.2

You need to step through the macro and detail levels of the TDS diagram. You need to check whether the goal of Abstract Process tasks is important and necessary to be performed. Should you find any potential problems, you need to report them.

Do you think the goal of abstract process task is important and necessary for this application?

1. If NO, explain why it is not important? Describe the problem(s) if any.

2. If YES, do you think 'each sub-task' (user, application, interaction) of the abstract process task above is important?

For unimportant sub-tasks (user, application, interaction), why it is not important? Describe the problem if any.

Step 2 [Thorough Inspection]

In this step, you will examine the application in more detail. The inspection materials to be used include the TDS diagram, Contexts of Use, User Interpretation and other necessary materials (e.g., graphical user interfaces).

You need to walk through the Abstract Process task of the TDS diagram. In each Abstract Process task, you will need to create several 'questions' based on the contexts of use and user interpretation, which will be explained in the next step. Based on the questions, you are required to do the following:

a. Identify and detect any possible <u>cause</u> of the potential problems that may hinder the effective, efficient and satisfying use of the application.
b. From the 'cause' above, look for any <u>potential failure stories</u> (effects) that may influence the usability of the application;

The 'cause' is the potential error of the design that brings up the potential failure stories. The 'potential failure stories' refer to the effects of the potential problems in the design that might happen to the user if they use the application later on.

For each Abstract Process task, the inspector needs to do the following:

<u>Inspection Step 2.1</u>

This step focuses on the 'USER TASKS' of the abstract process task. It requires you to concentrate on the criteria from the 'Context of Use: Users and Users Tasks'. For each 'abstract process task', its 'user task goal' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions for you to examine the design and detect any potential problems (could be the 'effect', 'cause' or both). The questions are:

From the criteria in the context of 'Users and Users Tasks', do you foresee any problem;

1. For the USER to achieve successfully the **<user tasks issues>**?

2. For the user to recognize, filter or attend only to the significant **<conditions>**?
3. For the user to construct, arrange or put together several conditions to form the **<premise>**?
4. For the user to make sense or conceptualize the **<hypothesis>**?

Inspection Step 2.2

This step focuses on the 'APPLICATION TASKS' of the abstract process task. It requires you to concentrate on the criteria from the 'Context of Use: Application Task, Equipments and Input/Output'. For each 'abstract process tasks', its 'application task goal' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions for you to examine the design and detect any potential problems (could be the 'effect', 'cause' or both). The questions are:

From the criteria in the context of 'Application Task, Equipments and Input/Output', do you foresee any problem:

1. *For the APPLICATION to achieve successfully the **<application tasks issues>**?*
2. *That could influence the user to recognize, filter or attend only to the significant **<conditions>**?*
3. *That could influence the user to construct, arrange or put together several conditions to form the **<premise>**?*
4. *That could influence the user to make sense or conceptualize the **<hypothesis>**?*

Inspection Step 2.3

This step focuses on the 'INTERACTION TASKS' of the abstract process task. It requires you to concentrate on the criteria from the 'Context of Use: Interface and Interaction'. For each 'abstract process task', its 'interaction task goal' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions for you to examine the design and detect any potential problems (could be the 'effect', 'cause' or both). The questions are:

From the criteria in the context of 'Interface and Interaction', do you foresee any problem;

1. *For the INTERACTION Design to achieve successfully the **<interaction tasks issues>**?*
2. *That could influence the user to recognise, filter or attend only to the significant **<conditions>**?*
3. *That could influence the user to construct, arrange or put together several conditions to form the **<premise>**?*
4. *That could influence the user to make sense or conceptualize the **<hypothesis>**?*

<u>Inspection Step 2.4</u>

This step focuses on the 'USER, APPLICATION AND INTERACTION TASKS' of the abstract process task. It requires you to concentrate on the criteria from the 'Context of Use: Environment. For each 'abstract process task', its 'user , application and interaction task goals' and its 'user interpretation list (condition, premise and hypothesis)' will be used to generate several questions for you to examine the design and detect any potential problems (could be the 'effect', 'cause' or both). The questions are:

From the criteria in the context of 'Environment', do you foresee any problem;

1. *For all the [USER, APPLICATION AND INTERACTION Design] to achieve successful **<their issues respectively>**?*
2. *That could influence the user to recognise, filter or attend only to the significant **<conditions>**?*
3. *That could influence the user to construct arrange or put together several conditions to form the **<premise>**?*
4. *That could influence the user to make sense or conceptualize the **<hypothesis>**?*

All the problems found will be rated by inspector using a severity level (1 to 4) [adapted from Nielsen (1990)]. This rating is applied to prioritize the problems encountered

1 = cosmetic problem - only needs to be fixed if extra time is available on project

2 = minor usability problem – fixing this should be given low priority

3 = major usability problem – important to fix, so should be given high priority

4 = usability catastrophe – imperative to fix this before application can be released

need to repeat Step 2 (consisting of Inspection Steps 2.1, 2.2, 2.3 and 2.4) for every 'abstract process task' from the TDS diagram. Each potential problem found by inspectors needs to be written down along with its severity level.

All potential problems need to be recorded as follows:

List of Problems:	
Problems [describe the cause of the problem(s)]	**Failure Stories [describe the effects of the problem(s)]**
[list the problems here]	*[write the story here]*

<u>**Example:**</u>

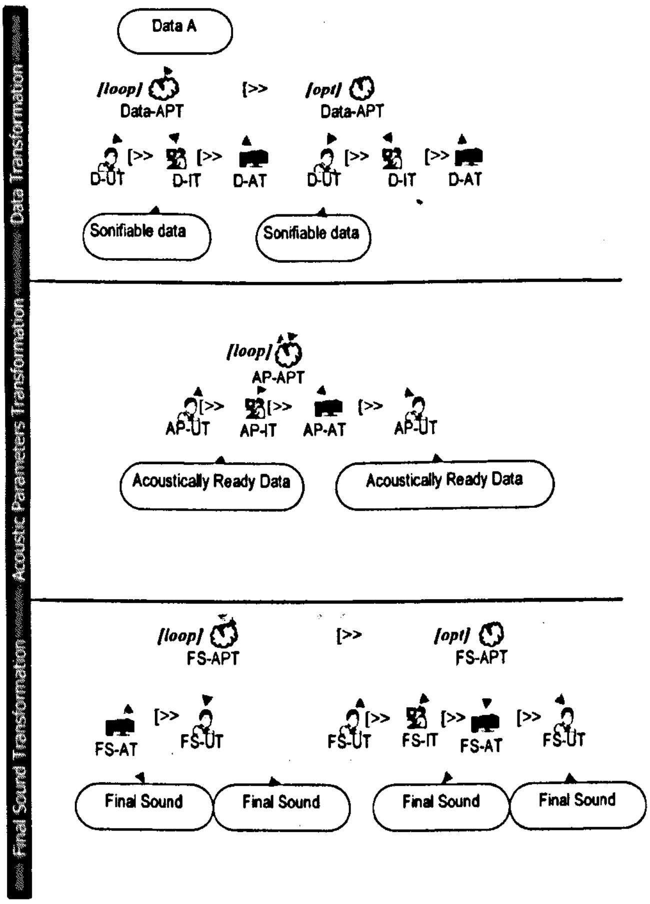
The application uses two different levels of pitch to represent two categories of information.

Problem : *Level 3: pitch cannot be differentiated distinctly, especially by the users who have no knowledge of music or sound; and you may need to consider using two different timbres.*

Failure story : *For the user who has no knowledge of music or sound, they might not be able to perceive two distinct pitches but rather to perceive them as the increase in sound volume. Therefore, they might not interpret it as two different categories but rather as an increase in volume of the category.*

Attachment 1: Task-Data State Diagram

Below are the basic notations and explanations about the diagram:-



(i)

Transformation Processes

'Transformation' is a process of changing 'original data' into a 'new state of data'. In this diagram, we describe Sonification Applications in three transformation processes that change raw data into sound. Each transformation is described in a row of the diagram (please see the diagram).

The transformations include:-

√ *Data Transformation* –a process of transforming raw data into a new form which is more suitable to be transformed into sound.

- √ ***Acoustic Parameters Transformation*** –a process of converting the data (*or the output of the data transformation*) into some intermediate acoustic parameters such as amplitude, pitch, timbre etc.
- √ ***Final Sound Transformation*** –a process of mixing all the acoustic parameters (*or the output of the acoustic parameters transformation*) and converting them into sound to be listened to by the user.

(ii) Types of task:-

Notations	Descriptions
 [Abstract Process Task]	An **Abstract Process Task** represents a sub-transformation process in each perspective. It is described in general, and can be further decomposed into user, interaction and application tasks. It is performed to achieve and accomplish the goal of the three main transformation processes (data, acoustic parameters and final sound)
 [User Task]	A **User Task** is performed by the user independent of the application in order to achieve and accomplish the goal of the sub-transformation process (Abstract Process Task) it belongs to.
 [Interaction task]	An **Interaction Task** is performed by the user through interactions with the application in order to achieve and accomplish the goal of the sub-transformation process (Abstract Process Task) it belongs to.
 [Application Task]	An **Application Task** is performed by the system to process, control, manipulate and transform input (data) into output without user interruption in order to achieve and accomplish the goal of the sub-transformation process (Abstract Process Task) it belongs to.
DATA STATE	**Data state:** Describe the input and output of the tasks representing different states of the data.
	*The flow of data state.*

Temporal Relationship:

||| *:interleaving – task can be done in any order*
>> : *enabling*
[ ]>> : *enabling with info passing (output is required)*
|[ ]| : *synchronization*
[> : *deactivation*
[Task name] or *[opt]* : *Optional task*
[loop] : *iteration*
[loop n] : *iteration 'n' times*
[once] : *task is done only once*

(iii) Input and Output

Referring to the diagram above, each abstract process task has input and output. The output of a task normally becomes the input of another task. These inputs and outputs represen the different states of data. Each input and output is denoted by node. It displays the changes and intermediate results of the transformation and sub-transformation processes. One abstract task could have one or more input and output.

In general, there are 4 main data states, namely – data, sonifiable data, acoustically ready data and final sound. The data might need to go through several processes before it becomes the final output of the next transformation process, e.g. sonifiable data (*output of data transformation*) or acoustically ready data (*output of acoustic parameters transformation*).

(iv) Abstract tasks description *[will be given during the experiment]*

To ensure the diagram is more readable and understandable, the goal of each abstract task is further described based on its 'purpose' and 'issues'.

- *Purpose* – explains the objective or 'what the application wants to achieve';
- *Issues* – explains the 'quality' of what the application wants to achieve;

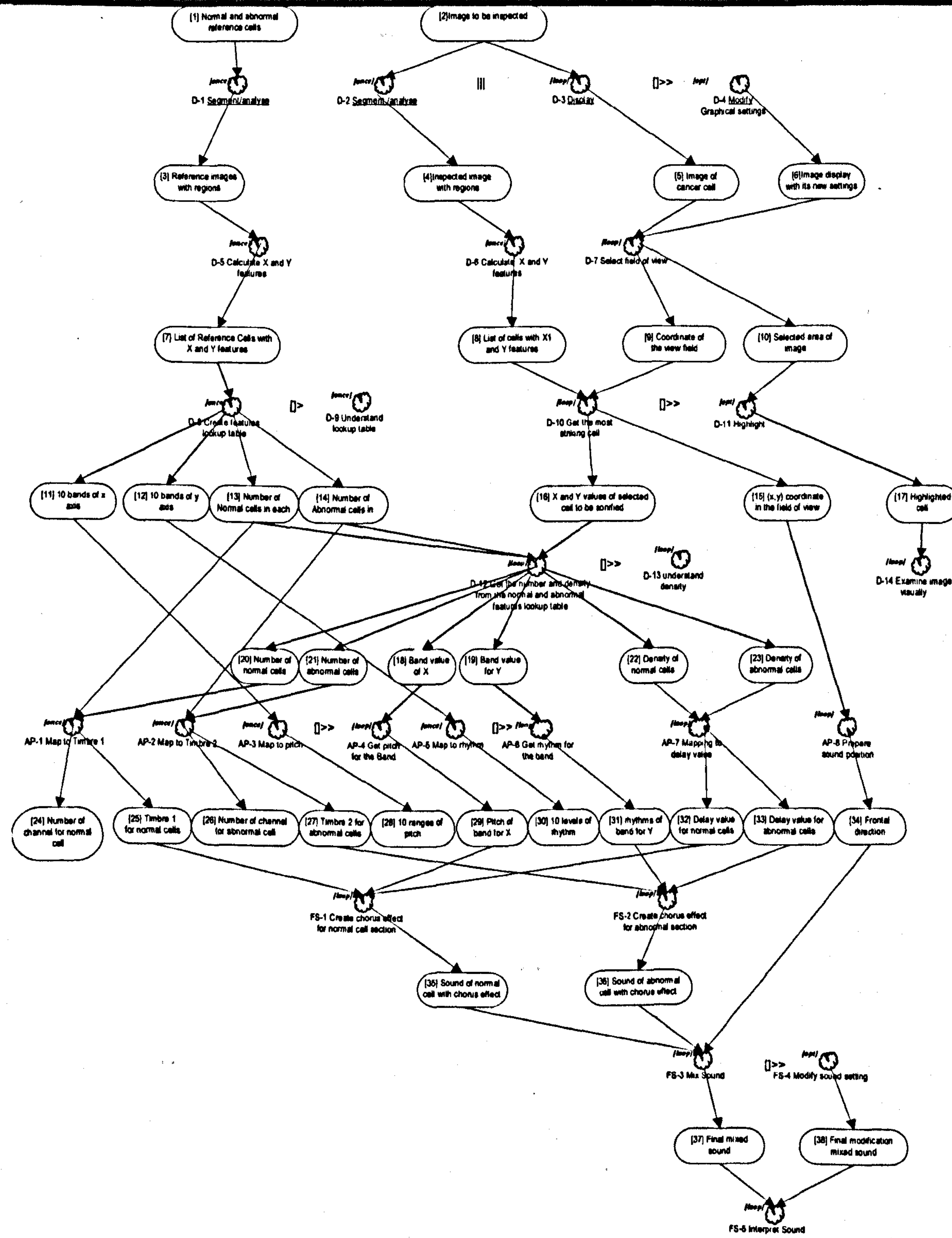
The 'overall abstract tasks' explains the idea from the designer's point of view about what should be done during each transformation/sub transformation process.

In Summary, the diagram shows the Task-Data-State Diagram, which can be used to model signification applications. Starting from the top, we can see how the data is transformed into sonifiable data, followed by the acoustically ready data and finally into the final sound. The processes involved in changing the data states are explained in task form. Starting from the abstract process task, it is then decomposed into user, interaction and application tasks. By doing this, we can see in a bit detail the tasks and processes involved in each transformation.

-end-

Attachments

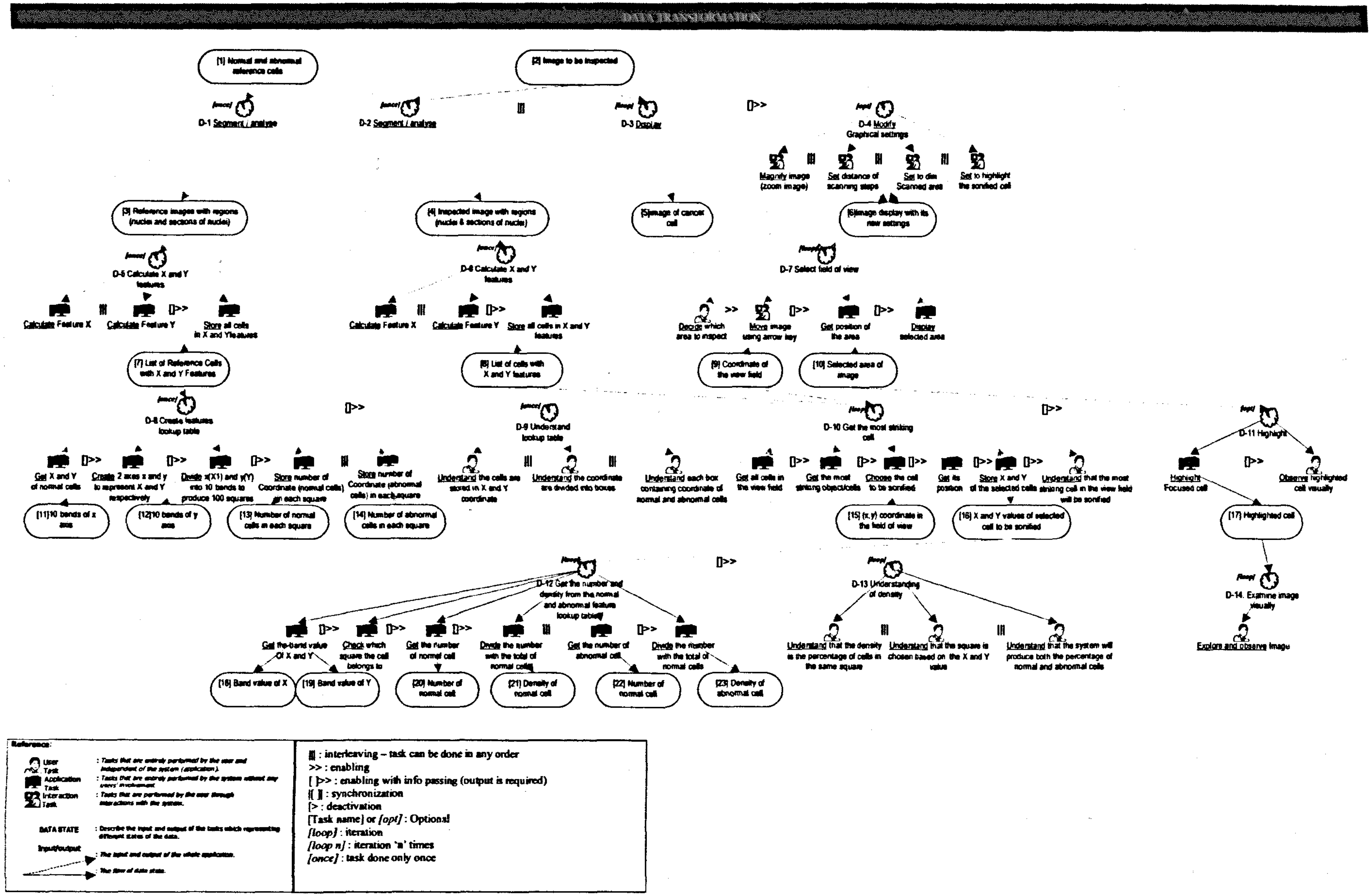
ATTACHMENT I: (ABSTRACT Task-Data-State DIAGRAM)



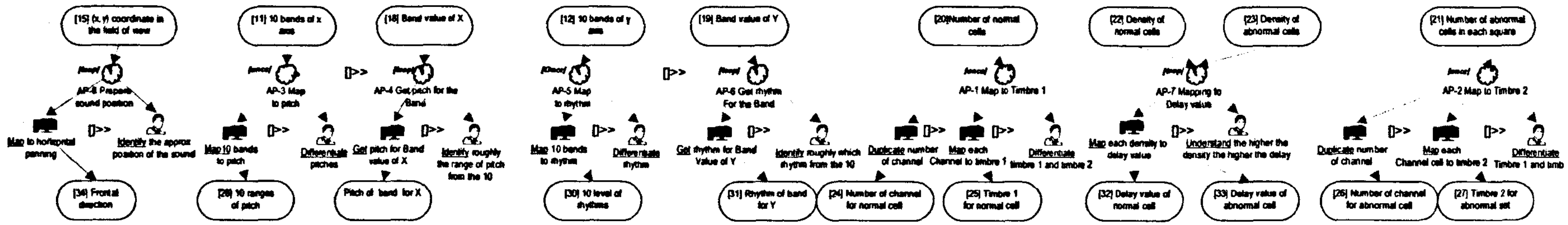
Legend:

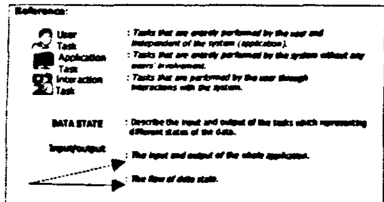
User Task	: Tasks that are entirely performed by the user and independent of the system (Application).
Application Task	: Tasks that are entirely performed by the system without any users' involvement.
Interaction Task	: Tasks that are performed by the user through interactions with the system.
DATA STATE	: Describe the input and output of the tasks which representing different states of the data.
Input/output	: The input and output of the whole application.
	: The flow of data state.

||| : interleaving – task can be done in any order
>> : enabling
[ ]>> : enabling with info passing (output is required)
|[ ]| : synchronization
[> : deactivation
[Task name] or *[opt]* : Optional
[loop] : iteration
[loop n] : iteration 'n' times
[once] : task done only once


DATA TRANSFORMATION
[1] Normal and abnormal reference cells
[2] Image to be inspected
[once] D-1 Segment / analyse
[once] D-2 Segment / analyse
|||
[loop] D-3 Display
[]>>
[opt] D-4 Modify Graphical settings
Magnify image (zoom image)
Set distance of scanning steps
Set to dim Scanned area
Set to highlight the sonified cell
[3] Reference images with regions (nuclei and sections of nuclei)
[4] Inspected image with regions (nuclei & sections of nuclei)
[5]image of cancer cell
[6]image display with its new settings
[once] D-5 Calculate X and Y features
[once] D-6 Calculate X and Y features
[loop] D-7 Select field of view
Calculate Feature X
Calculate Feature Y
Store all cells in X and Y features
Decide which area to inspect
Move image using arrow key
Get position of the area
Display selected area
[7] List of Reference Cells with X and Y Features
[8] List of cells with X and Y features
[9] Coordinate of the view field
[10] Selected area of image
[once] D-8 Create features lookup table
[once] D-9 Understand lookup table
[loop] D-10 Get the most striking cell
[opt] D-11 Highlight
Get X and Y of normal cells
Create 2 axes x and y to represent X and Y respectively
Divide x(X1) and y(Y) into 10 bands to produce 100 squares
Store number of Coordinate (normal cells) in each square
Store number of Coordinate (abnormal cells) in each square
Understand the cells are stored in X and Y coordinate
Understand the coordinate are divided into boxes
Understand each box containing coordinate of normal and abnormal cells
Get all cells in the view field
Get the most striking object/cells
Choose the cell to be sonified
Get its position
Store X and Y of the selected cells
Understand that the most striking cell in the view field will be sonified
Highlight Focused cell
Observe highlighted cell visually
[11]10 bands of x axis
[12]10 bands of y axis
[13] Number of normal cells in each square
[14] Number of abnormal cells in each square
[15] (x,y) coordinate in the field of view
[16] X and Y values of selected cell to be sonified
[17] Highlighted cell
[loop] D-12 Get the number and density from the normal and abnormal feature lookup table
[loop] D-13 Understanding of density
[loop] D-14. Examine image visually
Get the band value Of X and Y
Check which square the cell belongs to
Get the number of normal cell
Divide the number with the total of normal cells
Get the number of abnormal cell
Divide the number with the total of normal cells
Understand that the density is the percentage of cells in the same square
Understand that the square is chosen based on the X and Y value
Understand that the system will produce both the percentage of normal and abnormal cells
Explore and observe image
[18] Band value of X
[19] Band value of Y
[20] Number of normal cell
[21] Density of normal cell
[22] Number of normal cell
[23] Density of abnormal cell
Reference:
User Task : Tasks that are entirely performed by the user and independent of the system (application).
Application Task : Tasks that are entirely performed by the system without any users' involvement.
Interaction Task : Tasks that are performed by the user through interactions with the system.
DATA STATE : Describe the input and output of the tasks which representing different states of the data.
Input/output : The input and output of the whole application.
: The flow of data state.
||| : interleaving – task can be done in any order
>> : enabling
[ ]>> : enabling with info passing (output is required)
|[ ]| : synchronization
[> : deactivation
[Task name] or [opt] : Optional
[loop] : iteration
[loop n] : iteration 'n' times
[once] : task done only once

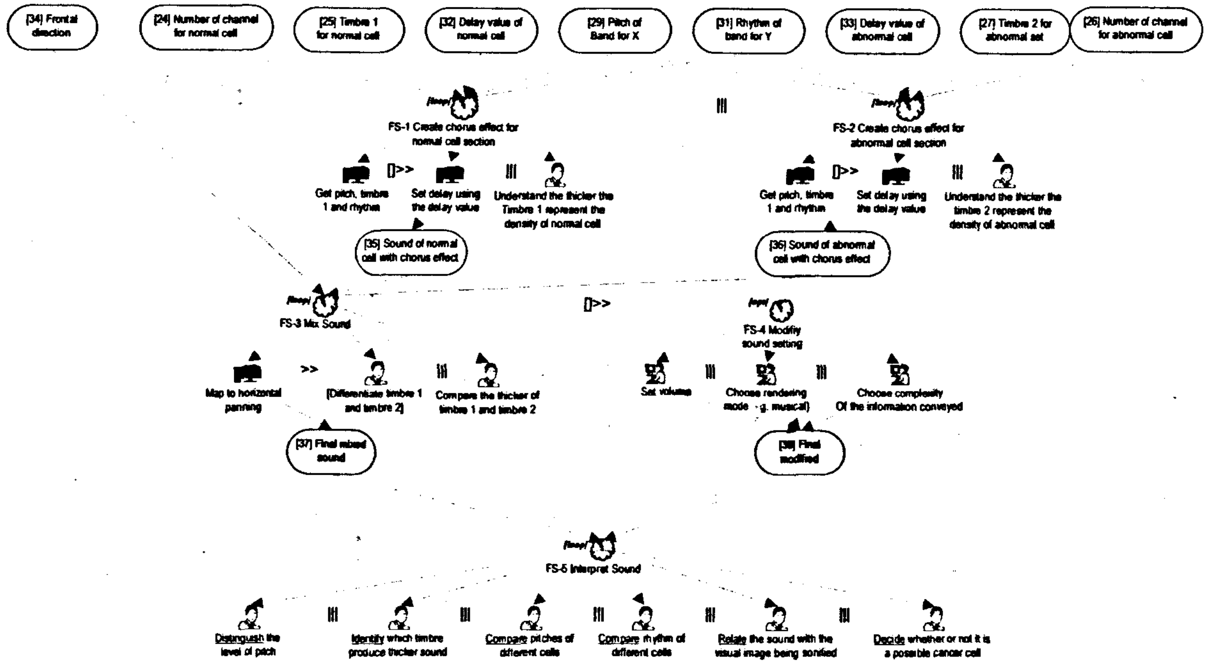
ACOUSTIC PARAMETERS TRANSFORMATION

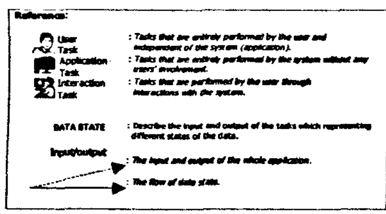




[] : interleaving – task can be done in any order
>> : enabling
[ ]>> : enabling with info passing (output is required)
|[ ]| : synchronization
[> : deactivation
[Task name] or *[opt]* : Optional
[loop] : iteration
[loop n] : iteration 'n' times
[once] : task done only once

FINAL SOUND TRANSFORMATION





||| : interleaving – task can be done in any order
>> : enabling
[ ]>> : enabling with info passing (output is required)
|[ ] : synchronization
[> : deactivation
[Task name] or *[opt]* : Optional
[loop] : iteration
[loop n] : iteration 'n' times
[once] : task done only once

Attachment 2: Context of Use Descriptions

	Criteria	Descriptions
1. USERS AND USERS TASKS	**1.1 Users Personal attributes** *[Age, gender ; Physical capabilities and limitations ; Attitude and motivation]*	(1) Cytoscreeners, advanced practitioners and pathologists (2) Users' age can be anything from mid-20's to retirement age. (3) The users are highly motivated towards the correct inspection of the slides. Their attitude is a priori highly skeptical as to the benefits of the audio system, but not hostile.
	1.2 Users' Practical Experiences *[Previous and related experiences]*	(1) The users are professionals trained to the visual inspection of cervical smears (2) User's experience in spotting visual patterns and using those as classification clues may be transposable to auditory clues (3) Users are experienced at navigating through a slide by means of microscope knobs (4) Users use computers for data entry (5) Some users enjoy listening to music or the radio during their work
	1.3 Users' Knowledge and cognitive system *[Previous and related knowledge]*	(1) The user might or might not have any formal or informal knowledge in music
	1.4 Users' Skill and Motor System *[Previous and related Skills]*	(1) The user has no problem using a keyboard, mouse, knobs and buttons of a reasonable size
	1.5 Users' Perceptual system	(1) The user has no hearing problem (2) The user has no problem in visual perception.
	1.6 Users' tasks	**Please refer to the user tasks in the Tasks-Data State Diagram.**
2.0 APPLICATION TASKS; EQUIPMENTS AND TECHNICAL; INPUT/OUTPUT	**2.1 Application tasks**	**Please refer to the application tasks in the Task-Data State Diagram.** Basically, (1) This application is to provide cytologist who is visually screening a smear slide with complementary information with which to make his/her diagnostic. This information is provided aurally simultaneously to the visual inspection. (2) Might require the user to get used to the metaphor used and to choose his/her favorite settings. (3) This application will have two states: a. The pre-processing state where the slide is processed and the images analyzed. b. The live stage during the slide screening when the audio is rendered.
	2.2 Equipments and Technical *[Hardware; Software; Network; Reference materials; other equipments]*	(1) In reality, the cells are fixed onto a glass slide, ready for inspection via a microscope, but here we will consider the situation where we dispose of digital images of sections of slides (fields of view). (2) A stand alone C++ program will do the preprocessing work (3) Computer interface will be used at this preliminary stage for the visualization/navigation of the images and the synchronization of the visual field with the audio field (4) In-ear headphones will deliver the audio (5) Mouse/keyboard/joystick etc might be used to navigate through the images
	2.3 Input/Output	**Please refer to the state of data in the Task-Data State Diagram**
3.0 ENVIRONMENT	**3.1 Physical environment** *[Workplace conditions Auditory environments Atmospheric conditions, Location, safety equipment, etc.]*	(1) The user will use the application while sitting at a desk (2) The user will be busy on an intense visual task while receiving the application audio output (3) The user will have a quiet environment .
	3.2Organization/social environment *[Structure; Group working; Work practices Assistance; Interruptions; Communication structure; Attitude and culture]*	(1) The user is unlikely to be interrupted but will take regular, short, visual breaks at his/er workstation and regular/longer breaks, doing other tasks (2) Some interesting slides are presented at group meetings. It could be conceived that the audio output could be valuable in such a setting too.
4.0 INTERFACE/ APPLICATION	**4.1 Users Interaction** *[objects; actions/steps; duration; constraint etc]*	**Please refer to the interaction tasks in the Task-Data State Diagram.** 1. Input is probably the mouse and a toolbox menu 2. outputs are cell images and sound 3. inspection of each field of view can be expected to last no longer than 3 second 4. the user needs to be able to mark and save the image when problem areas have identified
	4.2 Other Interfaces *[e.g. Graphical interfaces]*	**- refer to attachment 4**

Attachment 3: Output of Users' Interpretation Process

Data Transformation

Task Ref.	Goal for transformation	Goal for the user to achieve [based on the input/output of the transformation's goals]		
	Goal of Abstract Process Tasks for <u>Data Transformation</u>	**Goal of <u>Selection</u> For Data Transformation** (outputs » condition)	**Goal of <u>Reasoning</u> for Data Transformation** (conditions » premises)	**Goal of <u>Hypothesizing</u> for Data Transformation** (premises » hypothesis)
D-1 D-2	**Segment/Analyse** ***Purpose*** • To get rid most of cytoplasm and very small objects from the digital image of cervical LBC slide. • To segment/analyse several normal and abnormal cell images to be used as the data references. • To segment/analyse a slide to be inspected. ***Issues*** *User's tasks* • <u>Note: we think that it is not necessary for the user to know in detail about this background process.</u> *Interaction tasks* • - *Application tasks* • -To get rid successfully of most of the cytoplasm and very small objects from the digital image of cervical LBC slide for further image analysis.	**Input** • Normal and abnormal reference cells • Image to be inspected **Output** • Reference images with regions (nuclei and section of nuclei) • Inspected image with regions (nuclei and section of nuclei) **Conditions:** • -	**Premises:** • -	**Hypothesis:** • -
D-3	**Display** ***Purpose*** • To display the slide of image being inspected. ***Issues*** User's tasks • - Interaction tasks • - Application tasks • to display the whole image and prepare it to be explored and manipulated.	**Input** • Image to be inspected **Output** • Image of cancer cell (displayed) **Conditions:** • -	**Premises:** • -	**Hypothesis:** • -

D-4	**Modify Graphical Setting**
	Purpose
	• To change the graphical setting of the *image for exploration.*
	Issues
	User's tasks
	• -
	Interaction tasks
	• To help and allow flexibility to the user to explore and manipulate the image. These include:
	  • To zoom in and out of the image
	  • To set distance of scanning steps
	  • To set to dim scanned area
	  • To set to highlight the sonified cell
	
	Application tasks
	• -
		Input
		• Image of cancer cell (being displayed)
		Output
		• Image display with its new settings
		
		Conditions:
		• Image size
		• Dimmed area
		• Highlighted cell
			Premises
			• Highlighted cell is the current cell being sonified
			• Dimmed area shows that it has already been explored.
			• If the cell become bigger, then it is in zoom-in mode and some parts of the image are hidden (cannot be displayed through field of view).
				Hypothesis:
				• -

D-5	**Calculate X and Y features** ***Purpose*** • To represent a cell with two features – X and Y (which have already been proved to successfully differentiate between normal and abnormal cells) • To calculate the X and Y feature values of several reference cells (normal and abnormal cells) ***Issues*** User's tasks • - Interaction tasks • - Application tasks • to calculate the X and Y features as accurately as possible and produce a distinguishable normal and abnormal cells distribution as the cell reference.	**Input** • Reference images with regions (nuclei and section of nuclei) **Output** • List of reference cells with X and Y features **Conditions:** • -	**Premises:** -	**Hypothesis:** -
D-6	**Calculate X and Y features** ***Purpose*** • To calculate X and Y features of every single cell on the image being inspected ***Issues*** User's tasks • - Interaction tasks • - Application tasks • To ensure all cells in the image are detected and stored in its X and Y features	**Input** • Inspected image with regions (nuclei and section of nuclei) **Output** • List of cells with X and Y features **Conditions:** • -	**Premises:** • -	**Hypothesis:** • -

D-7	**Select field of view** ***Purpose*** • to display the image or some part of the image as selected by the user • to allow the user to explore a 'zoom in' image. ***Issues*** *User's tasks* • need to observe/decide where to explore the image and know the area that has not been scanned/inspected yet. *Interaction tasks* • to allow a smooth transition of image movement so that the user feels like *looking, zooming and moving the image* as they normally do with a microscope. *Application tasks* • the application should be able to identify the position area so that the area could be dimmed if the user sets to dim the scanned area.	**Input** • Image of cancer cell • Image display with its new settings **Output** • coordinate of view field • selected area of image **Conditions:** • Current image appears on field of view • Some parts of the image are dimmed • *moving image when the arrow* buttons are pressed. •	**Premises:.** • the image can be moved and scanned only in zoom-in mode. • Field of view will only display the zoomed image. • Cell in the current image appearing on screen is the one being inspected. • The image is only explorable in 'zoomed in' mode. • Scanned (checked) area should be able to be differentiated • Field of view displays only part of the image.	**Hypothesis:** -
D-8	**Create reference features lookup table** ***Purpose*** • To create a table to store the list of normal and abnormal cells with X and Y features as the reference ***Issues*** User's tasks • Interaction tasks • Application tasks • to produce a clear distribution of normal and abnormal cells • the 10 x 10 bands must be good enough to represent equally the distribution of the cells.	**Input** • list of cells with X and Y features **Output** • 10 bands of x axis • 10 bands of y axis • Number of normal cells in each square • Number of abnormal cells in each square **Conditions:** • -	**Premises:** -	**Hypothesis:** -
D-9	**Understand Lookup Table** ***Purpose*** • to understand how the normal and abnormal cells are grouped. ***Issues*** User's tasks • need to understand that the cells are distributed in 2D graphs based on X and Y *feature* values for x and y axis respectively. • Each axis is divided into 10 bands, which produces 100 square boxes. • To correctly represent the density of normal and abnormal cells through its number in any of the square boxes.	**Input** • 10 bands of x axis • 10 bands of y axis • Number of normal cells in each square • Number of abnormal cells in each square **Output** • - **Conditions:** • Feature X • Feature Y • axis x and Y • x for feature X	**Premises:** • Feature X is represented by x axis. • Feature Y is represented by y axis. • Cells will be stored and categorized based on their X and Y features. • x and y axes are divided into 10 bands to produce 100 squares. • Normal and abnormal cells are distributed among the 100 square boxes.	**Hypothesis:** • the density of normal and abnormal cells is dependent on the number of normal and abnormal sample/reference cells in a square box.

	Interaction tasks • - Application tasks • -	• y for feature Y • 10 bands of x axis • 10 bands of y axis		
D-10	**Get the most striking cells(s)** ***Purpose*** • To choose which cells are going to be sonified (converted into sound) ***Issues*** User's tasks • Must be aware that the decision on which cell to be sonified is made by the application. • Understand that the sound is actually from the highlighted cell in the field of view. • The user should also be aware that the sound is produced only if they did not move the image within 2 seconds. Interaction tasks • - Application tasks • Need to use intelligent criteria to choose which cells are to be sonified within the field of view, especially if there are so many potential abnormal cells being displayed.	**Input** • List of cells with X and Y features • Coordinate of the view field **Output** • Coordinate of cell in the field of view • X and Y value of selected cell to be sonified. **Conditions:** • Several cells on the field of view	**Premises:** • The cell to be sonified is selected by the system, therefore, more attention is still required for the other cells. • A sound will be heard only if the user didn't move the image in 2 seconds	**Hypothesis:** -
D-11	**Highlight** ***Purpose*** • To highlight the cell being sonified ***Issues*** User's tasks • Understand the highlighted cell is currently being sonified. Interaction tasks • - Application tasks • -	**Input** • Selected area of image • Cell coordinate in the field of view • X and Y values of selected cell to be sonified **Output** • Highlighted cell **Conditions:** • Highlighted cell • Cell not highlighted but looks 'strange'	**Premises:** • Sonified cell will be highlighted by the system.	**Hypothesis:** -

D-12	**Get the density from the normal and abnormal feature lookup tables** ***Purpose*** • To get related information on the 'cell to be sonified' with the information from the 'reference features lookup table'. ***Issues*** User's tasks • - Interaction tasks • - Application tasks • to give impression of which square of the 'cell to be sonified' belongs to. • To get all the information from the 'reference features lookup table' and present them in the form of sound.	**Input** • Reference features lookup table • X and Y values of selected cell to be sonified **Output** • Band value of X • Band value of Y • Number of normal cells • Number of abnormal cells • Density of normal cells • Density of abnormal cells **Conditions:** • -	**Premises:** -	**Hypothesis:** -
D-13	**Understanding of density** ***Purpose*** • To understand what the 'density' which is used in representing the cell being sonified actually is. ***Issues*** User's tasks • To understand how to get this density so that the user can relate them with the sound output. Interaction tasks • - Application tasks • -	**Input** • Band value of X • Band value of Y • Number of normal cells • Number of abnormal cells • Density of normal cells • Density of abnormal cells **Output** • - **Conditions:** • number of normal cells in square • number of abnormal cells in square. • number of overall normal cell samples (N_{normal}) • number of overall abnormal cell samples ($N_{abnormal}$)	**Premises:** • Density of normal cells is the 'number of normal cells in square' divided by the 'number of overall normal cell samples (N_{normal})' • Density of abnormal cells is the 'number of abnormal cells in square' divided by the 'number of overall abnormal cell samples ($N_{abnormal}$)' •	**Hypothesis:** -
D-14	**Examine image visually** ***Purpose*** • To allow the user to use their expertise to examine and inspect the image of the cell ***Issues*** Users Tasks • The user should be able to use their previous knowledge and experience to explore and observe the image being inspected visually. Interaction tasks • - Application tasks • -	**Input** • Highlighted cell **Output** • - **Conditions:** • -	**Premises:** -	**Hypothesis:** -

Acoustic Parameters Transformation

Task Ref.	Goal for transformation	Goal for the user to achieve		
	Goal of Abstract Tasks for <u>Acoustic Parameters</u> Transformation	**Goal of <u>Selection</u> for Acoustic Parameters Transformation** (outputs » condition)	**Goal of <u>Reasoning</u> for Acoustic Parameters Transformation** (conditions » premises)	**Gal <u>Hypothesizing</u> for Acoustic Parameters Transformation** (premises » hypothesis)
AP-1	**Map to Timbre 1** ***Purpose*** • to duplicate the number of sound channels based on the number of normal cells found in the square and map them all with Timbre 1 ***Issues*** User's Tasks • must be able to differentiate the different timbres • observe and differentiate how thick the sound is, through the number of channels. Interaction Tasks • - Application Tasks • to produce a complex and identifiable sound.	**Input** • Number of normal cells in each square **Output** • Number of channel for normal cell • Timbre 1 for normal cell **Conditions:** • Timbre 1 • Sound thickness	**Premises** • -	**Hypothesis** • -
AP-2	**Map to timbre 2** ***Purpose*** • To duplicate the number of channels based on the number of abnormal cells found in the square and map them all with Timbre 2 ***Issues*** User's Tasks • Must be able to differentiate the different timbres • Predict and differentiate how thick the sound is through the number of channels. Interaction Tasks • - Application Tasks • to produce a complex and identifiable sound	**Input** • Number of abnormal cells in each square **Output** • Number of channels for abnormal cells • Timbre 2 for abnormal cells **Conditions:** • Timbre 2 • Sound thickness	**Premises:** •	**Hypothesis:** •
AP-3	**Mapping to pitch** ***Purpose*** • to represent the 10 bands of x axis with 10 ranges of pitch. ***Issues*** User's Tasks • the user should be able to detect the different pitches • the user should also be able to rank the	**Input** • 10 bands of x axis **Output** • 10 ranges of pitch **Conditions:** • 10 ranges of pitch • x axis	**Premises:** -	**Hypothesis:** -

	pitch by comparing it with the previous sound they heard. Interaction Tasks • - Application Tasks • to produce 10 'clearly differentiated' ranges of pitch			
AP-4	**Get Pitch for the Band** ***Purpose*** • map the current 'band for the X value of sonified cell' to the pitch based on the predefined 10 ranges of pitch. ***Issues*** User's Tasks • the user should be able to identify roughly the pitch of sonified cell from the 10 pitch ranges. Interaction Tasks • - Application Tasks •	**Input** • 10 ranges of pitch • band value of X **Output** • pitch of band for X **Conditions:** • 10 ranges of pitch • higher pitch • lower pitch	**Premises:** • The higher the pitch , the higher the value of X	**Hypothesis:** • -
AP-5	**Map to rhythm** ***Purpose*** • To represent the 10 bands of y axis with 10 different rhythms ***Issues*** User's Tasks • The user should be able to detect the different rhythms. Interaction Tasks • - Application Tasks • to produce 10 'clearly differentiated' types of rhythm.	**Input** • 10 bands of y axis • **Output** • 10 levels of rhythm **Conditions:** • 10 different levels of rhythm • y axis	**Premises:** • -	**Hypothesis:** • -
AP-6	**Get rhythm for the Band** ***Purpose*** • To map the current Y value of sonified cell to the rhythm based on the predefined 10 levels of rhythm • ***Issues*** User's Tasks • the user should be able to identify the level of rhythm and predict roughly the position of rhythm based on the previous sound played. Interaction Tasks • - Application Tasks - get rhythm for band value of Y	**Input** • Band value of Y • 10 levels of rhythm **Output** • Rhythm of band for Y **Conditions:** • 10 different rhythms • Fast beat rhythm • Slow beat rhythm	**Premises:** • The faster the beat of rhythm, the higher the value of Y	**Hypothesis:** • -

AP-7	**Mapping to delay value** ***Purpose*** • To create delay to the sound of normal and abnormal cells based on their cell density in the box respectively. ***Issues*** User's Tasks • The user should be able to observe and differentiate different delay Interaction Tasks • - Application Tasks • -	**Input** • Density of normal cells • Density of abnormal cells **Output** • Delay value of normal cells • Delay value of abnormal cells **Conditions:** • Slowness of the sound	**Premises:** • The slower the sound, the higher the density is.	**Hypothesis:** • -
AP-8	**Prepare sound position** ***Purpose*** • To position the sound based on the position of the cell being sonified in the field of view. ***Issues*** User's Tasks • Able to predict roughly the position of the sound in a situation where the sonified cell is not highlighted. Interaction Tasks • - Application Tasks -	**Input** • cell's coordinate in the field of view **Output** • Frontal direction **Conditions:** • Sound position	**Perceptions** • If the cell being sonified was on the left of the view field, the sound will be coming more from the left and vice versa.	**Hypothesis:** • -

Final Sound Transformation

Task Ref.	Goal for transformation	Goal for the user to achieve		
	Goal of Abstract Tasks for <u>Final Sound Transformation</u>	**Goal of <u>Selection</u> for Final Sound Transformation** (outputs » condition)	**Goal of <u>Reasoning</u> for Final Sound Transformation** (conditions » premises)	**Goal of <u>Hypothesizing</u> for Final Sound Transformation** (premises » hypothesis)
FS-1	**Create chorus effect for normal cell section** ***Purpose*** • To combine all acoustic parameters including Timbre 1, pitch of X, rhythm of Y, Delay and number of channels to produce a sound of normal cell section. ***Issues*** User's Tasks • The user should be able to recognize timbre 1, which represents a normal cell, and try to observe its other characteristics i.e. the delay and number of channel through the thickness of the sound, level of pitch as well as the type of rhythm. Interaction Tasks • - Application Tasks • to successfully produce a sound for normal cell with its chorus effect.	**Input** • Number of channel for normal cell • Timbre 1 for normal cell • Delay value of normal cell • Pitch of X • Rhythm of Y **Output** • Sound of normal cell with chorus effect **Conditions:** • Number of sounds • Delay of sounds • Type of Timbre • Level of pitch • Type of rhythm	**Premises:** • The sound will be thicker and complex if the number of channels is increased. • The delay of the sound will increase the complexity of the sound. • If the sound of timbre 1 is thicker and more complex than the other sound timbre, it is more likely the cell being sonified is a normal cell. • If the sound of timbre 1 has no chorus effect at all, therefore, there is no normal cell found in the square box. • If there is no chorus effect, the trend of pitch level and type of rhythm can be observed and compared with the previous sound – as the clue for the user.	**Hypothesis:** • The more complex and thicker the sound of timbre 1, the higher the possibility of the cell being sonified to be a normal cell. • If the sound of timbre 1 is not complex and thick, it does not mean that the cell being sonified is not a normal cell. • In a condition where there are no reference cells in the square box, the level of pitch and rhythm can be used as the clue to decide whether or not it is more towards a normal or abnormal cell.
FS-2	**Create chorus effect for abnormal cell section** ***Purpose*** • To combine all acoustic parameters including Timbre 2, pitch of X, rhythm of Y, Delay and number of channel to produce a sound of abnormal cell section. ***Issues*** User's Tasks • The user should be able to recognize timbre 2, which represents an abnormal cell, and try to observe its other characteristics i.e. the delay and number of channel through the thickness of the sound, level of pitch as well as the type of rhythm. Interaction Tasks • - Application Tasks • to successfully produce a sound for abnormal cell with its chorus effect.	**Input** • Number of channel for abnormal cell • Timbre 2 for abnormal set • Delay value of abnormal cell • Rhythm of Y **Output** • Sound of abnormal cell with chorus effect **Conditions:** • Number of sounds • Delay of sounds • Type of Timbre • Level of pitch • Type of rhythm	**Premises:** • The sound will be thicker and more complex if the number of channels was increased. • The delay of the sound will increase the complexity of the sound. • If the sound of timbre 2 is thicker and more complex than the other sound timbre, it is more likely the cell being sonified is an abnormal cell. • If the sound of timbre 2 has no chorus effect at all, therefore, there is no abnormal cell found in the square box. • If there is no chorus effect, the trend of pitch level and type of rhythm can be observed and compared with the previous sound – as the clue for the user.	**Hypothesis:** • The more complex and thicker the sound of timbre 2, the higher the possibility of the cell being sonified to be an abnormal cell. • If the sound of timbre 1 is not complex and thick, it does not mean that the cell being sonified is not a normal cell. • In a condition where there are no reference cells in the square box, the level of pitch and rhythm can be used as the clue to decide whether or not it is more towards a normal or abnormal cell.

FS-3	**Mix Sound** ***Purpose*** • To combine the sound of normal cell (FS-1) and abnormal cell (FS-2) sections with their chorus effects setting as well as sound position. ***Issues*** User's Tasks • The user must be able to listen to both sounds at the same time and try to identify the characteristics of both timbre 1 and 2. • The user should then be able to compare the characteristics of both timbres in terms of thickness and complexity. Interaction Tasks • - Application Tasks -	**Input** • Frontal direction • Sound of normal cell with chorus effect • Sound of abnormal cell with chorus effect **Output** • Final mixed sound **Conditions:** • Timbre 1 and 2 • Number of sounds for both timbres • Delay of sounds for both timbres • Level of pitch • Type of rhythm	**Premises:** • The same level of pitch and type of rhythm is used by both sounds – timbre 1 and timbre 2. • If the sound of timbre 1 is thicker and more complex than the sound of timbre 2, the sonified cell is more likely to be a normal cell. • If the sound of timbre 2 is thicker and more complex than the sound of timbre 1, the sonified cell is more likely to be an abnormal cell. • If there is no chorus effect to both timbre 1 and 2, the trend of pitch and rhythm can be used to predict whether or not the cell being sonified is more towards a normal or abnormal cell. • If both timbre 1 and 2 were producing equally thick and complex sounds, the user needs to use their knowledge and experience to decide whether or not the cell being sonified is potentially a normal or abnormal cell.	**Hypothesis:** • The thickness and complexity of a sound is influenced by its chorus effect, therefore, by observing these criteria, it could give the user information about the abnormality of the cell being sonified. • Since the number of reference cells might be limited, there is a possibility that NO chorus effect occurs. In this situation, the trend of pitch level and type of rhythm could be used as the clue about the abnormality of the cell being sonified. • In all situations, the user's expertise is still required to make the final decision on whether or not the cell being sonified/inspected has shown abnormality – a potential cancer cell.
FS-4	**Modification of sound setting** ***Purpose*** • To change the complexity of both normal and abnormal sound sections by enabling and disabling certain acoustic parameters ***Issues*** User's Tasks • To allow the user flexibility to reduce the complexity of the sound. • To allow the user to learn how to observe the acoustic parameters by enabling and disabling them while listening to the sound at the same time. Interaction Tasks • the user must be able easily to set either to enable or disable any acoustic parameters. Application Tasks • -	**Input** • Final mixed sound **Output** • Final Modified Mixed sound **Conditions:** • As above (FS-3)	**Premises:** • as above (FS-3)	**Hypothesis:** • As above (FS-3)

FS-5	**Interpret Sound** ***Purpose*** • To attend only to important sounds and interpret them based on the objective of the sound and the application, which is to detect cancer cells. ***Issues*** User's Tasks • The user should be able to distinguish each level of pitch of different timbres. • The user should be able to identify which timbre is producing a thicker sound. • The user must be able to compare the pitch level and type of rhythm among different cells. • The user should be able to relate the sound with the visual image being sonified and use their expertise to decide whether or not it is potentially a cancer cell. Interaction Tasks • - Application Tasks -	**Input** • Final mixed sound • Final modified mixed sound **Output** • - **Conditions:** • as above (FS-3)	**Premises:** • as above (FS-3)	**Hypothesis:** • as above (FS-3)

Attachment 4: Descriptions of Application

This document contains 4 parts - A, B, C and D.
Part A: Audio-Visual Analysis tool of Cervical Sample Slides
Part B: Objective of Application
Part C: Descriptions of Users
Part D: Goal, Tasks and Subtask to be accomplished

Part A: Audio-Visual Analysis Tool of Cervical Sample Slides

Figure 1: Cells image being inspected with view field

The proposed system is an audio-visual analysis tool of cervical sample slides. The main aim of this tool is to help users in analyzing cervical sample slides and detecting potential abnormal or cancer cell(s). The input is a digital image of cervical LBC slide as shown in Figure 1. It is stored in 'pnm' file format (image file format).

In the background process, the image being inspected is processed and segmented into 'regions of interest'. Through this process, most of the cytoplasm and very small objects are discarded. The remaining regions are mostly nuclei and sections of nuclei. The texture of these regions is analyzed by calculating two features, called X and Y. Feature X is referred to as the average area of the dark clumps in the nuclei and Y is the number of light clumps in the nuclei. Based on previous research, these features can be used to differentiate between normal and abnormal cells.

As a reference to the application, several normal and abnormal cells are segmented, analyzed and stored in X and Y values. Based on these values, a 2-d graph can be drawn as shown in Figure 2. This shows that the distribution of normal and abnormal cells can be separated.

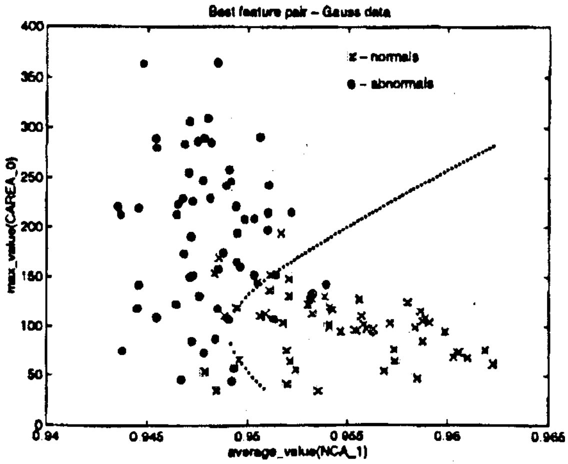


Figure 2: Normal and Abnormal Cells based on Feature X and Y

In this application, the graph in Figure 2 is divided into grids as shown in Figure 3. Assume that the range of the x-axis is between 0 and X_{max} and the range of the y-axis is between 0 and Y_{max}. Each range is then divided into 10 bands. This produces 100 squares that contain several cells in each square. The amount of normal and abnormal cells in the same square is used to represent the density of cells (density of normal cells and density of abnormal cells) that belong to that square. The density is calculated by adding the number of cells

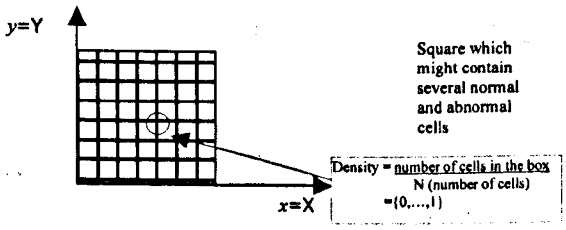


Figure 3: Grids of X and Y

in a square and dividing by the number of overall samples :

The number of normal and abnormal cells in each square is stored in a table called '**Reference X-Y Features Look-up Table**' as shown in Figure 5. The table is used to store the number of normal and abnormal cells, where the 10 bands of X-Y values are used as the reference. This table is used as a reference to predict whether or not the 'cell image being inspected' can be categorized as a normal or abnormal cell. For example, if there were more abnormal cells found in the table from the X and Y values of the 'inspected cell image', it can be hypothesized that the 'inspected cell' is more likely to be categorized as an abnormal cell and vice versa. However, in this application, 'what exactly the two features (X and Y features) are' is not important for the user to know and understand.

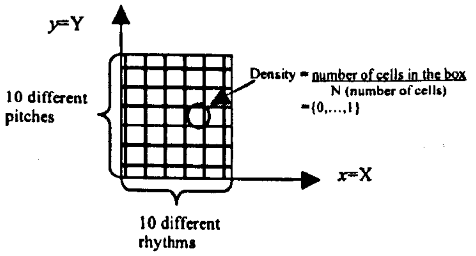


Figure 4: Sound mapping

Figure 4 shows the 10-band values of X and Y are mapped into rhythm and pitch respectively. 10 ranges of pitch are used to represent the 10 bands of y or 'values of feature Y' and 10 different rhythms are used to represent the other 10 bands of x or 'values of feature X'. These pitches and rhythms are used to represent the position of the square. The density level of normal and abnormal cells in each square is described by the chorusing effect. The chorusing effect makes a copy (as another sound channel) of the main signal and plays back the copy with a very slight delay and pitch shift. The delay is set based on the density value of normal and abnormal cells, while the number of copies (or sound channel) is based on the number of normal and abnormal cells in a particular square. The pitch of each channel will be slightly shifted to ensure that they are slightly different to each other. By doing this, it will make the sound 'fatter', thicker' and more complex. The sound will be more complex and sound busier with the increasing number of normal or abnormal cells. The densities of normal and abnormal cells are differentiated by mapping them with two different timbres (timbre 1 and timbre 2). The position of the cell being sonified is mapped into the degree of panning horizontally. The user should be able to recognize the direction of the sound source indicating a region of the field of view where the sonified object lies. However, the identification of which cell is being sonified can also be done through visual display as the sonified cell can be set so that it will be highlighted. In summary, a potential abnormal/cancer cell is represented by two sounds – the 'sound of normal cells' and the 'sound of abnormal cells'. Each sound is represented by the following acoustic parameters:

1. Level of pitch – based on the value of Y feature;
2. Type of rhythm – based on the value of X feature;
3. Either Timbre 1 for normal cells or Timbre 2 for abnormal cells;
4. Delay of chorusing effect – based on the density of cells e.g. density of normal cells for the 'sounds of normal cells' and so to abnormal cells;
5. Number of channels – based on the number of cells e.g. number of normal cells for the 'sound of normal cells' and so to abnormal cells;
6. Pitch shifting of every channel.

Figure 6 shows the only graphical user interface of this application. To inspect a slide of a cell image, the user needs to load the image by pressing the '*Load New Image*' button. The application displays the overall image on the 'Field of View' as shown in Figure 7.a). At the same time, the system will do the background process. The background process involves the analysis and segmentation of the image into 'regions of interest' exactly as with what the application does to the reference cells. Features of X and Y of the cells are calculated and stored in a table using the pixel position as the reference.

Current Image X-Y Features Lookup Table

Coordinate	*X Feature*	*Y Feature*

Reference X-Y Features Look-up Table

y \ x	*Band 1*	…	..	*Band 10*
Band 1		*Number and density of normal and abnormal cell*		
..				
..				
Band 10				

Figure 5: Look-up Tables in Pre-Processing

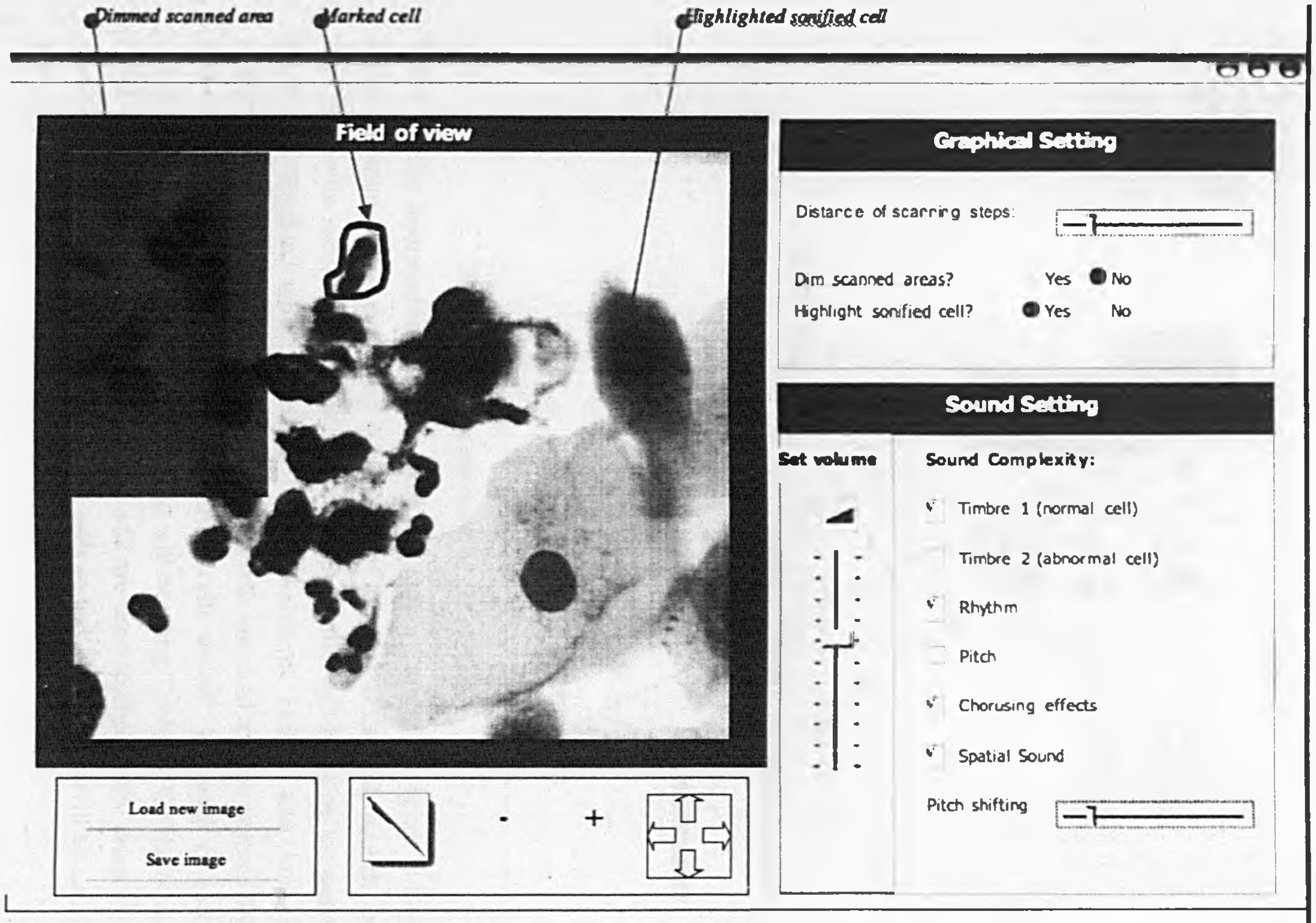
This pre-processing of the image is important to reduce latency (processing time) of the output processing as the application will allow the user to explore the image interactively. The table is called **'Current Image X-Y Features Lookup Table'**, containing - cell position (coordinate) and X-Y values as shown in Figure 5. The two tables are linked to each other so that the information forms the 'Reference X-Y Features lookup Table' i.e. 'band for X and Y'; and 'the number' and 'density' of normal as well as abnormal cells can be accessed through the coordinates of the cell being sonified.

The user could change and adjust the graphical setting such as zooming, setting the distance of scanning steps, dimming scanned areas and highlighting the sonified cell. The setting can be referred to the GUI in Figure 6.

1. Zooming can be done by pressing the lens icon – zoom in and zoom out. What actually happens in zooming is that it will enlarge the whole image and display only some part in the field of view as shown in Figure 7 b) and c).
2. The four arrows can be used to display (scan) different parts of the zoomed image by moving the image up, down, left or right. By doing this, different parts of the image can be displayed in the Field of View. The distance of scanning steps can be changed through the slider.
3. The scanned area can be dimmed to indicate that the area has been scanned and checked as shown in Figure 7 b). This feature can be disabled.
4. The image being sonified (currently producing sound) can be set to be highlighted by the system so that the user could pay more attention to that cell (as shown in Figure 6). This feature can be disabled.
5. The user is also allowed to mark the cell by pressing the 'pen' icon and draw on the image.
6. Any changes to this image can be saved by pressing the '*Save image*' button.

Besides exploring the image visually, the system also gives feedback in the form of sound. As explained above, the image is pre-processed where the information is stored in the '**Current Image X Y Features Lookup Table**'. Once the user stops moving or scanning the image for about 2 seconds, the system automatically produces the sound of the current cell in the *Field of View*. The system will check the most striking object/cell in the field of view and get its pixel coordinate. The X and Y values of the pixel coordinate are used to check its related information from the '**Reference X-Y Features Look-up Table**', which include – value of *band x and y*; and *density* and *number of* normal as well as abnormal cells. This information is used to produce the sound output.

Depending on the user setting, the cell being referred to and sonified can be highlighted (in a box) as shown in Figure 6.



Dimmed scanned area
Marked cell
Highlighted sonified cell
Field of view
Graphical Setting
Distance of scanning steps:
Dim scanned areas? Yes No
Highlight sonified cell? Yes No
Sound Setting
Set volume
Sound Complexity:
Timbre 1 (normal cell)
Timbre 2 (abnormal cell)
Rhythm
Pitch
Chorusing effects
Spatial Sound
Pitch shifting
Load new image
Save image
- +

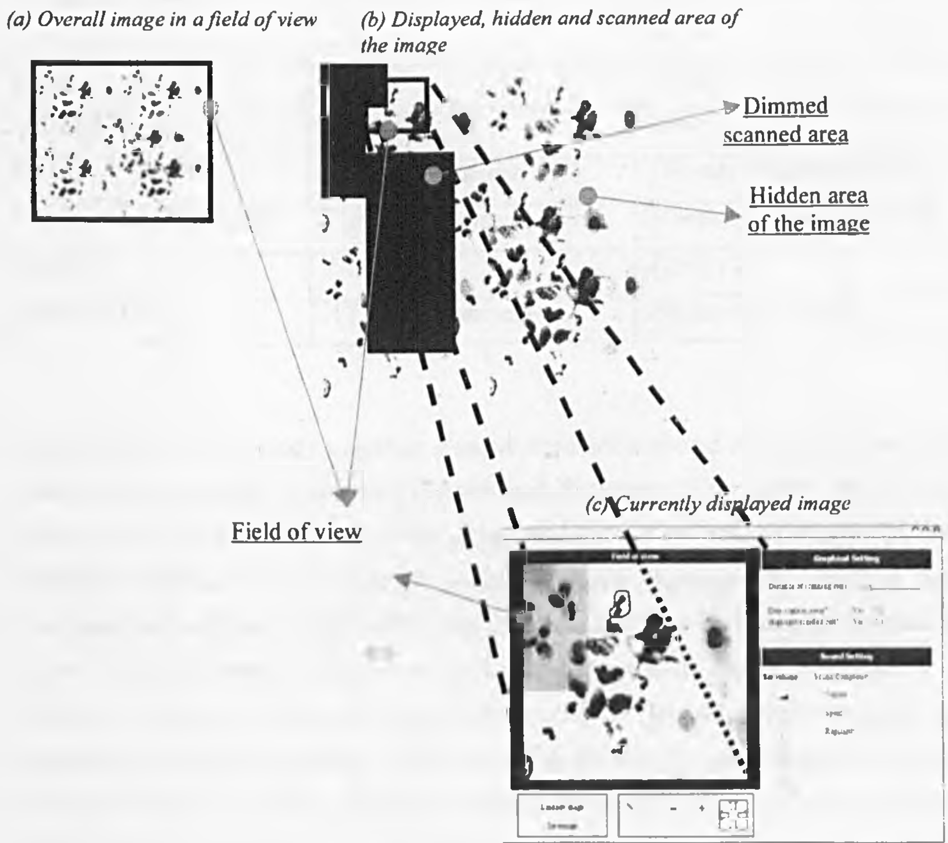


Figure 7: Zooming and Scanning of Cell Images

Based on the information retrieved from the look-up tables in Figure 5, two types of sounds are produced – sounds of normal cells and sounds of abnormal cells. Referring to Table 1 below, both sounds will be using the same pitch and rhythm, which is based on the value of *band x* and *band y* respectively. The *density* and *number of normal cells* are used to change the *delay* and *number of sound channels* of the 'normal cells sound'. The *pitch* of each channel will be slightly shifted. The sound will be using Timbre 1. On the other hand, the *density* and *number of abnormal cells* are used to change the *delay* and *number of sound channels* of the 'abnormal cells sound'. The sound will be using timbre 2 and the pitch of each channel will also be slightly shifted.

Table 1: Two types of sound output

Acoustic Parameters	Sounds of Normal Cells	Sounds of Abnormal Cells
Pitch	*Band x*	*Band x*
Rhythm	*Band y*	*Band y*
Delay	*Density of Normal Cells*	*Density of Abnormal Cells*
Number of Channels	*Number of Normal Cells*	*Number of Abnormal Cells*
Timbre	*Timbre 1*	*Timbre 2*
Pitch shifting	*All sound channels*	*All sound channels*

Mixing these two sounds together should produce a sound with the same pitch and rhythm, but with different timbre and thickness. Since these sounds are going to be played at the same time, the user should be able to detect the two different timbres and recognize which one is representing normal and abnormal cells. Besides differentiating timbres, the user should also be able to observe the thickness of the sound and recognize which timbre is thicker. The thickness of sound (chorus effect) is influenced by the delay value (density of cells) and number of channels. The higher the density and number of channels, the thicker the sound that will be produced. The thickness of the sound should be perceived by the user as the number (density) of normal and abnormal cells, which have more or less the same value of X and Y features of the cell being inspected or sonified. Depending on which timbre is thicker, the user should be able to get a clue about whether or not the cell being sonified is likely to be an abnormal or a normal cell. The users could also compare the sounds among different cells through their pitch and rhythms. For instance, if the trend of 'high density of abnormal cell' tends to produce a higher pitch, therefore, even though the chorus effect produces a 'thinner' sound if the pitch was higher or more or less the same as the sound of 'high density of abnormal cell', it is more likely to be a possible abnormal cell. To increase the flexibility of the application, the user is allowed to change the sound settings, such as volume and sound complexity, to suit their comfortable level. When the user changes the field of view, or changes the graphical settings, the system should re-select the object(s) to be sonified. The rendering is restarted and the transformation processes are repeated.

Part B: Objective of Application

Below are the objectives of this application.

1. To provide computed information on the textural aspects of certain cell nuclei
2. to aid in the diagnosis of abnormal cells
3. To analyze the texture of the nuclei present in the image
4. to alert the user to the presence of potential abnormal cells
5. to summarize the information relating to one cell
6. to be used by a cytologist as an aid to cervical LBC screening

Part C: Description of Users

Below is the description of the user of this application.

1. Cytoscreeners, advanced practitioners, pathologists
2. The users are professionals trained in the visual inspection of cervical smears
3. The users are highly motivated towards the correct inspection of the slides.

Part D: Goal, Task and Subtasks to be accomplished

Examples of tasks to be accomplished by the users

Goal 1: To produce optimum audio-visual settings

- Task 1.1: Load and save image to be inspected
- Task 1.2: Modify Sound Settings
  - Task 1.2.2: Set the volume for comfort
  - Task 1.2.4: Choose the complexity of the information conveyed
    - Task 1.2.4.1: Enable and Disable Timbre
    - Task 1.2.4.2: Enable and Disable Rhythm
    - Task 1.2.4.3: Enable and Disable Pitch
    - Task 1.2.4.4: Enable and Disable Chorusing effects
    - Task 1.2.4.5: Enable and Disable Spatial Sound
    - Task 1.2.4.6: Enable and Disable Pitch Shifting
- Task 1.3: Modify Graphical and Navigation Settings
  - Task 1.3.1: Modify the screening magnification in digital zoom
  - Task 1.3.2: Modify the distance of scanning steps
  - Task 1.3.3: Dim / do not dim the scanned area
  - Task 1.3.4: Highlight or do not highlight sonified cell

Goal 2: To inspect a number of cells on a screen while listening to the auditory display of

complex features of one or more interesting cells in the field of view

Task 2.1: Navigate graphical display and settle on a field of view

Task 2.1.1: Use arrow keys/mouse to move around the image

Task 2.1.2: settle display by not interfering with commands for a brief while

Task 2.2: Visually inspect cells on the field of view

Task 2.3: Listen to the audio and make a note of how normal the cell represented is

Task 2.3.1: Observe which timbre

Task 2.3.2: Observe chorus effets

Task 2.4: Relate the audio to the visual field

Task 2.4.1: Understand the approximate position of the sound source in a semi circle facing the user

Task 2.4.2: identify the cells located in the source direction and relate any of

those to the sounds heard

Task 2.4.3: alternatively, understand that the highlighted cell is the one being

played

Task 2.5: Pause to mark the problem cell/cells for future reference

Task 2.5.1: place an arrow/circle a problem area

APPENDIX D:

READING MATERIALS

Available only on attached CD

APPENDIX E:

TRANSCRIPTS AND SUMMARY DATA

Available only on attached CD

REFERENCES

Ag. Asri Ag. Ibrahim & Hunt, A. 2006a. *HCI Model for Usability of Sonification Applications*. Proceedings of TAMODIA'2006. Published in 'Task Models and Diagrams for Users Interface Design', Springer Lecture Notes in Computer Science, Springer-Verlag, ISBN 978-3-5540-70815-5, pp. 245 – 258.

Ag. Asri Ag. Ibrahim & Hunt, A. 2006b. *A General HCI Framework of Sonification Application*. Proceedings of The 8th International ACM SIGACCESS Conference on Computer & Accessibility.. Portland Oregon USA. October 2006. pp.221-222.

Ag. Asri Ag. Ibrahim & Hunt, A. 2007a. *Systematic Usability Inspection for Sonification Application*. Proceedings of The 13th International Conference on Auditory Display. Montreal, Canada, 26 – 29 June 2007. pp.453 – 460.

Ag. Asri Ag. Ibrahim & Hunt, A. 2007b. *Combining Data State and CCT Diagrams to Model Sonification Applications*. Proceedings of The 25th ACM SIGDOC 2007 Conference. El Paso, Texas. October 2007. pp.148-154.

Ag. Asri Ag. Ibrahim. 2006c. *Task Analysis for Sonification Applications Usability Evaluation*. Student Research Competition (SRC). Proceedings of The 8th International ACM SIGACCESS Conference on Computer & Accessibility. Portland Oregon USA. October 2006. pp.289-290.

Aires, R., Aluiso, S., Quaresma, P., Santos, D. and Silva, M.J. 2003. *An Initial proposal for Cooperative Evaluation on Information Retrieval in Portuguese.* Proceedings of the Computational Processing of Te Portuguese Language. Faro Portugal. pp.227-234.

Alty, J.L. & Rigas, D.I. 1998. *Communicating Graphical Information to Blind Users Using Music: The Role of Context.* CHI' 98. Los Angeles CA USA. 18-23 April 1998. pp.574-581.

Alty, J.L. & Vickers, P. 1997. *The CAITLIN Auralization System: Hierarchical Leitmotif Design as a Clue to Program Comprehension*. Proceedings of the

1997 International Conference on Auditory Display. Xerox PARC, Palo Alto, California. 2-5 November 1997. pp.89-96.

Ballas, J.A. 1994. *Effect of Event Variations and Sound Duration on Identification of Everyday Sound.* Proceedings of the Second International Conference on Auditory Display. New Mexico, USA. November 1994. p.147-155.

Barrass, S. 1997. *Auditory Information Design.* Phd Thesis. The Australian National University.

Begault, R.D., Wenzel, E.M., Shrum, R. and Miller, J. 1996. *A Virtual Audito Guidance and Alert System for Commercial Aircraft Operations.* Proceedings of the 1996 International Conference of Auditory Display. California, USA. November 1996. p.117-122.

Bevan, N. & Macleod, M. 1994. *Usability Measurement in Context.* Behaviour and Information Technology, 13, p.132-145.

Bevan, N. 1995. *Usability is Quality of Use.* Proceedingss of the 6th International Conference on Human Interaction, Yokohama. July 1995.

Bevan, N. 2001. *International Standard for HCI and Usability.* International Journal of Human Computer Studies, 55(4), pp.533-552.

Bias, R. 1994. *The Pluralistic Usability Walkthrough: Coordinated Empathies* in Nielsen, J. & Mack, R.L. (Eds.) *Usability Inspection Methods.* John Wiley & Sons, Inc. New York. pp.63 - 76.

Blattner, M.M., Sumikawa, D.A, and Greenberg, R.M. 1989. *Earcons and Icons: Their Structure and Common Design Principles.* Human Computer Interaction. 4(1), pp.11-44.

Bock, D.S. 1994. *ADSL: An Auditory Domain Specification Language for Program Auralization.* Proceedings of the 1994 International Conference on Auditory Display. Santa Fe, NM, 1994. pp.251-256.

Bonebright, T.L., Miner, N.E., Goldsmith, T.E. and Caudell, T.P. 1998. *Data Collection and Analysis Technique for Evaluating Perceptual Qualities of Auditory Stimuli.* Proceedings of the 1998 International Conference on Auditory Display. Glasgow, UK. November 1998. pp.505-516.

Bonebright, T.L., Nees, M.A., Connerley, T.T. and McCain, G.R. 2001. *Testing The Effectiveness of Sonified Graphs For Education: A Programmatic Research Project.* Proceedings of the 2001 International

Conference on Auditory Display. Finland. 29th of July- 1st of August 2001. pp.62-66.

Bovermann, T., Hermann, T., Ritter, H. 2005. *The Local Heat Exploration Model for Interactive Sonification*. Proceedings of the 2005 International Conference on Auditory Display. Limerick, Ireland. July 2005. pp.85-91.

Braun, N. & Dorner, R. 1998. *Using Sonic Hyperlinks in Web-TV*. Proceedings of the 1998 International Conference on Auditory Display. Glasgow, UK. November 1998. *www.icad.org/websiteV2.0/Conferences/ICAD98/papers/ BRAUN.PDF*. Downloaded: April 2005

Brewster, S.A. 1994. *Providing a Structured Method for Integrating Non-Speech Audio into Human-Computer Interface*. PhD Thesis, University of York. Website: *http://www.dcs.gla.ac.uk/7Estephen/ papers/ theses/ Brewster_* thesis.pdf. Downloaded: Jan 2005

Brewster, S.A., Wright, P.C. & Edwards, A.D.N. (1995). *Parallel Earcons: Reducing the Length Cf Audio Messages*. International Journal Of Human-Computer Studies; 43(2): pp 153-175.

Brewster, S.A.; Wright, P.C. and Edwards, A.D.N. 1992. *A Detailed Investigation into the Effectiveness of Earcons*. The Proceedings of The 1992 International Conference on Auditory Display. Santa Fe, New Mexico. October 1992. Addison-Wesley. pp.471-498.

Brewster, S.A., Wright, P.C. and Edwards, A.D.N. 1994. Guidelines for the Creation of Earcons. Website: *Http://www.dcs.gla.ac.uk/~stephen/earcons_ guidelines.shtml*. Downloaded: April 2005

Buxton, B., Gaver, B. & Bly, S. 1994. *Auditory Interfaces: The Use of Non-Speech Audio at the Interface*. Unpublished manuscript. Available at website: *http://www.billbuxton. com/Audio.TOC.html*. Downloaded: Feb 2005.

Chandler, D.C. 1997. *Semiotics for Beginners*. Website: *http://www.aber.ac.uk/media/ Documents/S4B/sem02.html*. Downloaded : January 2006

Chi, E,H. 1999. *A Framework for Information Visualization Spreadsheeets.* PhD Thesis, The University of Minnesota. Website: *http://www-users.cs.umn.edu/~echi/phd/chi-thesis.pdf.* Downloaded: May 2005

Chi, E.H. & Riedl, J. 1998. *An Operator Interaction Framework for Visualization System.* Proceedings of IEEE Symposium on Information Visualization (InfoVis98). Research Triangle, CA, USA. October 1998. pp.63-70.

Chi, E.H., 2002. Expressiveness of the Data Flow and Data State Models in Visualization systems. Proceedings of the International Working Conference on Advanced Visual Interfaces. Trento, Italy, May 2002. Website: *citeseer.ist.psu.edu/chi01expressiveness.html.* Downloaded: April 2005.

Coren, S., Ward, L.M. and Enns, J.T. 1999. Sensation and Perception, Fifth Edition. Harcourt Brace College Publishers. USA.

Cross, K. & Warmack, A. 2000. Contextual Inquiry: Quantification and Use in Videotaped Analysis. Adjunct Proceedings CHI 2000: Human Factors in Computing Systems 2000. The Hague, Netherlands. pp.317 – 318.

Daude, S. & Nigay, L. 2003. *Design Process for Auditory Interfaces.* International Conference on Auditory Display. Boston, USA. July 2003. pp.176-179.

Derthick, M., Kolojejchick, J. and Roth, S.F. 1997. *An Interactive Visualization Environment for Data Exploration.* Proceedings of Knowledge Discovery in Databases, AAAI Press, August 1997, pp.2-9. Website: *http://www.ri.cmu.edu/ pubs/pub_1115.html*

Desurvire, H., Kondziela, J. and Atwood, M.E. 1997. Usability Studeis of WWW sites: Heuristic evaluation vs Laboratory testing. Proceedings of the 15th Annual International Conference on Computer Documentation. Utah, USA. pp.153-160.

Dinh, H.Q, Walker, N., Song, C., Kobayashi, A. and Hodges, L.F. 1999. *Evaluating the Importance of Multi-sensory Input on Memory and the Sense of Presence in Virtual Environments.* IEEE Virtual Reality 1999, TX, USA. March 1999. p.222-228. Website: http://www.cs.stevens.edu/ ~quynh/smellorama. Html.

Dombois, F. 2001. *Using Audification in Planetary Seismology*. Proceedings of the 2001 International Conference on Auditory Display. Finland. 29th of July- 1st of August 2001. p.227-230.

Draper, Dec. 2006. *The Hawthrone Effect and Other Expectancy Effects: A Note.* Department of Psychology, University of Glasgow. Website: *http://www. psy.gla.ac.uk/~steve/hawth.html#Preface*. Downloaded: April 2005.

Dufresne, A., Martial, O., Ramstein, C. and Mabilleau, P. 1996. *Sound, Space and Metaphor: Multimodal Access to Windows for Blind Users.* Proceedings of the 1996 International Conference on Auditory Display. Palo Alto, California. November 1996. Website: *www.icad.org/websiteV2.0 /Conferences/ICAD96/proc96/dufresne1.htm.* Downloaded: April 2005.

Duursma, C. and Olsson, O. 1994. *Task Model Definition and Task Analysis Process.* Technical Report KADSII /M5/VUB/RR/004/2.0, Vrije University, Brussels. August 1994.

Edwards, A.D.N., Challis, B.P., Hankinson, J.C.K. and Pirie, F.L. 2000. *Development of a Standard Test of Musical Ability for Participants in Auditory Interface Testing.* Proceedings of the 2000 International Conference on Auditory Display. Altlanta, Gerogia, USA. April 2000. Website: *http://www-users.cs.york.ac.uk/~alistair/publications/pdf/MAT.pdf.* Downloaded: January 2001.

Edworthy, J. & Hellier, E. 2002. *Designing Urgency Into Sound.* Proceedings of Design Sonore Conference 2002, Societe francaise d'acoustique), Paris.

Fernstrom, M. & Graffith, N. 1998. *LiteFoot- Auditory Display of Footwork.* Proceedings of the 1998 International Conference on Auditory Display. Glasgow, UK. November 1998. Website: *www.icad.org/websiteV2.0 /Conferences/ICAD98/papers/FERNS000.PDF.* Downloaded: April 2005

Fernstrom, M. & McNamara, C. 2005 *After Direct Manipulation – Direct Sonification.* ACM Transactions on Applied Perception (TAP). Volume 2, Issue 4. October 2005. pp.495-499.

Fernstrom, M., Graffith, N. and Taylor, S. 2001a. *Bliain Le Baisteach – Sonifying a Year with Rain.* Proceedings of the 2001 International

Conference on Auditory Display. Finland. 29th of July- 1st of August 2001. pp.282-283.

Ferworn, A., Bodner, R. and Chignell, M.H. 2000. *Auditory WWW Search Tools.* Proceedings of the 2000 International Conference on Auditory Display. Alanta, Georgia, USA. April 2000. *http://www.icad.org/websiteV2.0/ Conferences/ICAD2000/PDFs/FerwornBodnerChignell.pdf,* Downloaded:April 2005

Finke, R.A. 1989. *Principles of Mental Imagery.* Cambridge, Mass: MIT Press.

Frauenberger, C. & Noisternig, M. 2003. *3D Audio Interfaces for The Blind.* Proceedings of the 2003 International Conference on Auditory Display, Boston, USA. July 2003. pp.280-283.

Gardner, J.A., Lundquist, R. & Sahyun, S. 1996. *TRIANGLE: A Practical Application of Non-Speech Audio for Imparting Information.* Proceedings of the 1996 International Conference on Auditory Display. California, November 1996.

Gaver, W.W. 1992. *Using and Creating Auditory Icons.* In Kramer G (Ed.), Auditory Display, sonification, audification and auditory interfaces. The Proceedings of the First International Conference on Auditory Display. Santa Fe Institute, Addison-Wesley Publishing Company. October 1992. pp.417-446.

Gediga, G., Hamborg, K.C., and Düntsch, I., 2002. *Evaluation of software systems.* In *Encyclopedia of Computer Science and Technology,* volume 45, pp.127 - 153.

Grudin, J. 1993. *Obstacles to participatory design in large product development organizations.* In Schuler, D. & Namioka, A. 1993. *Participatory Design: Principles and Practices,* Lawrence Erlbaum Associates. pp.99-119.

Gunn, C. 1995. *An Example of Formal Usability Inspections in Practice at Hewlett-Packard Company.* Proceedings of the Conference Companion on Human Factors in Computing System. Colorado, USA. 1995. pp.103-104.

Hankinson, J.C.K. & Edwards, A.D.N. 1999. *Designing Earcons with Musical Grammars.* ACM SIGCAPH No.65. September 1999. pp.16-20.

Hermann, T. 2004. *Multiscale Auditory Display: Interacting With The Complexity level in Model-based Sonification.* Multiscale Geometry and

Analysis of High Dimensions, Oct 2004, UCLA, USA. Website: *http://www.ipam.ucla.edu/publications/mgaws3/mgaws3_5116.pdf.* Downloaded: April 2005

Hermann, T. 2002. *Sonification for Exploratory Data Analysis.* PhD Thesis, University of Bielefeld, Germany.

Hermann, T., Drees, J.M. and Ritter, H. 2003a. *Broadcasting Auditory Weather Reports – A Pilot Project.* Proceedings of the 2003 International Conference on Auditory Display. Boston, USA. July 2003. pp.208-211.

Hoffer, J.A., George, J.F. and Valacich, J.S. 1996. Modern Systems Analysis and Design. The Benjamin/Cummings Publishing Company, Inc. pp. 19

Humon, N., Thibault, B., Galloway, V., Willis, G. and Wing, J.G. 1998. *Sound Traffic Control: An Interactive 3-D Audio System for Live Musical Performance.* Proceedings of the 1998 International Conference on Auditory Display. Glasgow, UK. November 1998.

Hunt, A. & Hermann, T. 2004. *The Importance of Interaction in Sonification.* Proceedings of the 2004 International Conference on Auditory Display. Sydney, Australia. July 2004. Website: *www.icad.org/websiteV2.0/ Conferences/ICAD2004/papers/hunt_hermann.pdf.* Download: April 2005

Huopaniemi, J., Savioja, L., Takala, T. 1996. *DIVA Virtual Audio Reality System.* Proceedings of the 1996 International of Auditory Display. California, November 1996. pp.111-116.

Inman, D.P., Loge, K. And Cram, A. 2000. *Teaching Orientation and Mobility Skills to Blind Children Using Computer Generated 3-D Sound Environments.* Proceedings of the 2000 International Conference on Auditory Display. Alanta, Georgia, USA. April 2000. *http://www.icad.org/websiteV2.0/Conferences/ICAD2000/PDFs/LogeORI.pdf,* Downloaded: April 2005.

Jameson, D.H. 1994. *The Run-Time Components of Sonnets.* Proceedings of the 1994 International Conference on Auditory Display. Santa Fe, New Mexico. 1994. pp.241-247.

Janata, P. & Childs, E. 2004. *Marketbuzz: Sonification of Real-time Financial Data.* Proceedings of the 2004 International Conference on Auditory Display, Sydney, Australia. July 2004. Website: *www.icad.org/*

websiteV2.0/Conferences/ICAD2004/papers/janata_childs.pdf. Downloaded: April 2005

Jeffries, R. & Desurvire, H. 1992. *Usability Testing vs. Heuristic Evaluation: Was there a contest?* ACM SIGCHI Bulletin. Volume 24, Issue 4. October 1992. pp.39-41

John, B. E., & Kieras, D. E. (1996). Using GOMS for user interface design and evaluation: Which technique? ACM Transactions on Computer-Human Interaction, 3, pp.287-319.

Joseph, A.J. & Lodha, S.K. 2002. *MUSART: Musical Audio Transfer Function Real-Time Toolkit.* Proceedings of the 2002 International Conference on Auditory Display. Japan. July 2002. pp.64.

Kahn, M.K. & Prail, A. 1994. *Formal Usability Inspections* in Neilsen, J. & Mack, R.L. (Eds.) *Usability Inspection Methods.* John Wiley & Sons, Inc. New York. pp. 141-171.

Kantner, L. & Rosenbaum, S. 1997. *Usability Studies of WWW Sites: Heuristic Evaluation vs. Laboratory Testing.* Proceedings of the 15th annual International Conference on Computer Documentation. Utah, USA. 1997. pp.153-160.

Khaslavsky, J. & Shedroff, N. 1999. *Understanding the seductive experience.* Communications of the ACM 1999. vol.42, pp. 45 – 49.

Kramer, G., 1994. *Auditory Displays: Sonification, Audification and Auditory interfaces.* Proceedings Volume XVIII, Santa Fe Institute, Studies in the Science of Complexity, Addison-Wesley Publishing Company. 1994.

Kramer, G., Walker, B., Bonebright, T., Cook, P., Flowers, J., Miner, N., Neuhoff, J., Bargar, B., Barrass, S., Berger, J., Evreinov, G., Tecumseh, W.F., Grohn, M., Handel, S., Kaper, H., Levkowitz, H., Lodha, S., Shinn-Cunnigham, B., Simoni, M. and Tipei ,S. 1997. *Sonification Report: Status of the Field and Research Agenda.* Report prepared for the National Science Foundation by members of the International Community of Auditory Display, ICAD, Santa Fe, NM, 1999.

Krueger, M.W & Gilden, D. 1997. *KnowWhere^{TM}: An Audio/ Spatial Interface for Blind People.* Proceedings of the 1997 International Conference on Auditory Display. California. November 1997. Website: *http://www.icad.org/ websiteV2.0/Conferences/ICAD97/Kruger.PDF.* Downloaded: April 2005.

Laakso, S.A., Laakso, K.P. and Page, C.P. 2002. *DUO: A Discount Observation Method.* Website: *http://www.cs.helsinki.fi/u/salaakso/papers/DUO.pdf.* Downloaded: April 2005.

Leplatre, G. & McGregor, I. 2004. *How to Tackle Auditory Interface Aesthetics? Discussion and Case Study.* Proceedings of the 2004 International Conference on Auditory Display. Sydney, Australia. July 2004. Website: *www.icad.org/websiteV2.0/Conferences/ICAD2004/papers/lepastre_mcgregor.pdf.* Download: April 2005

Lodha, S.K., Beahan, J. Heppe, T., Joseph, A. & Zane-Ulman, B. 1997. *MUSE: A Musical Data Sonification Toolkit.* Proceedings of the 1997 International Conference on Auditory Display. California. November 1997. pp.36-40.

Lowgren, J. 1995. *Perspectives on Usability.* IDA Technical Report, LiTH-IDA-R-95-23, ISSN-0281-4250.

Lucas, P.A. 1994. *An Evaluation of the Communicative Ability of Auditory Icons and Earcons.* Proceedings of the 1994 International Conference on Auditory Display. Santa Fe,New Mexico, USA. November 1994. pp.121-128.

Macefield, R. 2007. *Usability Studies and the Hawthorne Effect.* Journal of Usability Studies. Vol.2, Issue 3. May 2007. pp.145-154.

Maffiolo, V., Chateau, N. and Mersiol, M. 2002. *The Impact of The Sonification of a Vocal Server on Its Usability and Its User-Friendliness.* Proceedings of the 2002 International Conference on Auditory Display. Japan. July 2002. pp.130.

Maguire, M., 2001. Context of Use within usability activities. International Journal of Human-Computer Studies (2001) 55, pp. 453-483.

Malandrino, D., Daniela, M., Negro, A., Palmieri, G. and Scarano, V. 2003. *NeMoS: Network Monitoring with Sound.* Proceedings of the 2003 International Conference on Auditory Display. Boston, USA. July 2003. pp.251-254.

Martini, J., Hermann, T., Amselmetti, D. and Ritter, H. 2004. *Interactive Sonification For Exploring Single Molecule Properties With AFM Based Force Spectroscopy.* Proceedings of the International Workshop on

Interactive Sonification, Ison2004, Bielefeld, Germany. January 2004. *Website: http://www.techfak.uni-bielefeld.de/ags/ni/projects/datamining/ datason/demo/ISON2004/DNASon.html*. Downloaded: April 2004

Martins, A.C.G. & Rangayyan, R.M. 1997. *Experimental Evaluation of Auditory Display and Sonification of Textured Images*. Proceedings of the Fourth International Conference of Auditory Display. CA, USA. pp.1-5.

McCormick, E.J. & Sanders, M.S. 1983. *Human Factors in Engineering and Design*. Mc Graw-Hill Book Co, Singapore.

McGookin, D.K. & Brewster, S.A. 2004. *Understanding Concurrent Earcons: Applying Auditory Scene Analysis Principles to Concurrent Earcon Recognition*. ACM Transactions on Applied Perceptions, Vol. 1, No.2, October 2004, pp.130-155.

McNamara, C. 1999. *General Guidelines for Conducting Interviews*. Website: http://www.mapnp.org/library/evaluatn/intrview.htm. Downloaded: April 2005.

Miele, J.A. 2003. Smith-Kettlewell Display tools: A Sonification Toolkit for Matlab. Proceedings of the 2003 International Conference on Auditory Display, Boston, USA. July 2003. pp.288-291.

Mirel, B. 1996. *Contextual Inquiry and the Representation of Tasks*. ACM SIGDOC Asterisk Journal of Computer Documentation, Volume 20 , Issue 1 (February 1996), pp: 14 – 21.

Mori, G., Paterno, F. and Santoro, C. 2002. *CTTE: Support for Developing and Analyzing Task Models for Interactive System Design*. IEEE Transactions of Software Engineering Vol.28. No.8 August 2002. pp.797-813.

Nagashima, Y. 2002. *Interactive Multi-media Performance with Bio-Sensing and Bio-Feedback*. Proceedings of the 2002 International Conference on Auditory Display. Japan. July 2002. pp.60.

Nesbitt, K.V. & Barrass, S. 2002. *Evaluation of A Multimodal Sonification and Visualisation of Depth of Market Stock Data*. Proceedings of the 2002 International Conference on Auditory Display. Japan. July 2002. pp.233.

Nielsen, J. & Mack, R.L. 1994. *Usability Inspection Methods*. John Wiley & Sons, Inc.

Nielsen, J. & Molich, R. 1990. *Heuristic Evaluation of User Interfaces.* Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering people. Washington, USA. 1990. pp.249-256.

Nielsen, J. 1993. *Usability Engineering.* Academic Press. USA.

Nielsen, J. 1995. *Usability Inspection Methods.* Proceedings of the Conference Companion on Human Factors in Computing Systems. Colorado, USA. 1995. pp.377-378.

Norman, D.A. 1988. *The Design of Everyday things.* Third printing 2000, First MIT Press edition, 1998. USA

Ohuchi, M., Iwaya, Y., Suzuki, Y. and Munekata, T. 2003. *A Game for Visually Impaired Children With A 3-D Virtual Auditory Display.* Proceedings of the 2003 International Conference on Auditory Display. Boston, USA. July 2003. pp.309.

Pair, J & Kooper, R. 1997. *COOLVR: Implementing Audio in a Virtual Environments Toolkit.* Proceedings of the 1997 International Conference on Auditory Display. pp.157-160.

Paterno', F., Mancini, C. and Meniconi, S, 1997. *ConcurTaskTrees: A Diagrammatic Notation for Specifying Task Models.* Proceedings of the IFIP TC13 International Conference on Human-Computer Interaction. Chapman & Hall, Ltd. London, UK. 1997. pp.362-369.

Patton, M. 1990. *Qualitative Evaluation and Research Methods.* Second Edition. Sage Publications, Inc. February 1990.

Pauletto, S. & Hunt, A. 2004a. *A Toolkit for Interactive Sonification.* Proceedings of The 2004 International Conference on Auditory Display. Sydney, Australia. July 2004. Website: *www.icad.org/websiteV2.0/Conferences/ ICAD2004/papers/pauletto_hunt.pdf.* Download: April 2005

Pauletto, S. 2004. Data-to-Sound Mapping Strategies: A Classification on The Basis of the Perceptual Characteristics of The Resulting Sound. PhD Transfer Report. Electronic Department, University of York.

Peres, S.C. & Lane, D.M. 2003. *Sonification of Statistical Graphs.* Proceedings of the 2003 International Conference on Auditory Display. Boston, USA. July 2003.

Petrucci, L.S., Harth, E., Roth, P., Assimacopoulos, A. and Pun, T. 2000. *WebSound: A Generic Web Sonification Tool, and its Application to an Auditory Web Browser for Blind and Visually Impaired Users.* Proceedings of the 2000 International Conference on Auditory Display. Alanta, Georgia, USA. April 2000. *http://www.icad.org/websiteV2.0/Conferences/ICAD2000 /PDFs/PetrucciPHRAP.pdf,* Downloaded: April 2005

Raven, M.E. & Flanders, A. 1996. *Using Contextual Inquiry to Learn About Your Audiences.* ACM SIGDOC Asterisk Journal of Computer Documentation Vol. 20, No.1. pp.1-13. 1996

Remenyi, D. & Smith, M.S. 1999. *Maximise Information Systems Value by Continuous Participative Evaluation.* Logistics Information Management, Volume 12, Numbers ½ 1999, pp. 14-31.

Revere, D., St.Anna, L.A, Ketchell, D.S, Kauff, D. Gaster, B. and Timberlake, D. 2001. *Using Contextual Inquiry to Inform Design of a Clinical Information Tool.* Submitted Proc AMIA, 2001. Website: *http://healthlinks. washington.edu/primeanswers/ presentations/revere.pdf.* Downloaded: April 2005.

Rieman, J., Franzke, M. and Redmiles, D. 1995. *Usability Evaluation with the Cognitive Walkthrough.* Proceedings of Conference companion on Human factors in computing system. Colorado, USA. 1995. Website: *http://www.acm.org/sigchi/chi95/proceedings/tutors/jr_bdy.htm.* Downloaded: April 2005. pp.387-388.

Rizzi, S.A, Sullivan, B.M. and Sandridge, C.A. 2003. A Three-Dimensional Virtual Simulator for Aircraft Flyover Presentation. Proceedings of the 2003 International Conference on Auditory Display. Boston, USA. July 2003. pp.87-90.

Rober, N & Masuch, M., 2004. *Interacting With Sound: An Interaction Paradigm For Virtual Auditory Worlds.* Proceedings of the 2004 International Conference on Auditory Display, Sydney Australia. Sydney, Australia. July 2004. Website: *www.icad.org/websiteV2.0/Conferences/ICAD2004/papers/ roeber_masuch.pdf.* Downloaded: April 2005

Root, R.W. & Draper, S. 1983. *Questionnaire as a software evaluation tool.* Proceedings of the SIGCHI conference on Human Factors in

Computing Systems. Boston, Massachusetts, USA. December 1983. pp.83-87.

Roth,, P., Petrucci, L., Assimacopoulos, A. and Pún, T. 1998. *AB – Web: Active Browser For Visually Impaired And Blind Users.* Proceedings of the 1998 International Conference on Auditory Display. Glasgow, UK. November 1998.

Royce, W. (1970). *Managing the Development of Large Software Systems.* Proceedings of IEEE WESCON 26 (August): August 1970 *Website:http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf.* Downloaded: April 2004. pp.328-338.

Saue, S. 2000. *A Model for Interaction in Exploratory Sonification Displays.* Proceedings of the 2000 International Conference on Auditory Display. Alanta, Georgia, USA. April 2000. *http://www.icad.org/websiteV2.0 /Conferences/ICAD2000/PDFs/saue.pdf,* Downloaded: April 2005

Savioja, L., Huopaniemi, J., Lokki, T. And Vaananen, R. 1997. *Virtual Environment Simulation - Advances in The Diva Project.* Proceedings of the 1997 International Conference of Auditory Display. California, USA. 1997. pp.43-46.

Scaletti, C. 1994. *Sound Synthesis Algorithms for Auditory Data Representations.* In: Kramer, G. (Ed.),. *Auditory display, sonification, audification and auditory interfaces.* Proceeding of the First International Conference on Auditory Display. Santa Fe Institute. 1994. Addison-Wesley Publishing Company. pp.223-251.

Scholtz, J. 2004. *Usability Evaluation.* Information Access Division (IAD) 2004 papers. National Institute of Standards and Technology. *http://www.itl.nist.gov/iad/IADpapers/2004/UsabilityEvaluation_rev1.pdf.* Downloaded: April 2005.

Sharp, H., Rogers, Y. and Preece, J., 2002. *Interaction Design: Beyond Human-Computer Interaction,* New York, NY:Wiley.

Someren, M.W.V, Barnard ,Y.F., Sandberg, J.A.C. 1994. *A Practical Guide to Modelling Cognitive Processes.* Academic Press, London. pp.29-37.

Stevens, C., Brennan, D. and Parker, S. 2004a. *Simultaneous Manipulation of Parameters of Auditory Icons to Convey Direction, Size and Distance: Effects on Recognition and Interpretation.* Proceedings of the 2004

International Conference on Auditory Display. Sydney, Australia. July 2004. Website: *www.icad.org/websiteV2.0/Conferences/ICAD2004/papers/ stevens_brenna_parker.pdf*. Downloaded: April 2005

Stevens, R. 1996. *Principles for The Design of Auditory Interfaces to Present Complex Information to Blind Computer Users*. PhD Thesis, University of York. Website: http://www.cs.man.ac.uk/~stevensr/papers/thesis-rds.pdf. Downloaded: Jan 2005.

Stevens, R.D., Brewster, S.A., Wright, P.C. and Edwards, A.D.N. 1994. *Design and Evaluation of an Auditory Glance at Algebra for Blind Readers*. In: Kramer, G. (Ed.),. *Auditory display, sonification, audification and auditory interfaces*. Proceeding of the First International Conference on Auditory Display. Santa Fe Institute. 1994. Addison-Wesley Publishing Company. pp.21-30.

Taylor-Powell, E. & Steele, S. 1996. *Collecting Evaluation Data: Direct Observation*. University of Wisconsin-Extension G3658-5. Website: *http://cecommerce. uwex.edu/pdfs/ G3658_5.PDF*. Downloaded: April 2005

Taylor-Powell, E. 1998. *Questionnaire Design: Asking Questions with a Purpose*. Program Development and Evaluation, University of Wisconsin-Extension G3658-2. Website: *http://cecommerce.uwex.edu/pdfs/G3658_2.PDF*. Downloaded: April 2005

Teppo, A. & Vuorimaa, P. 2001. *Speech Interface Implementation for XML Browser*. Proceedings of the 2001 International Conference on Auditory Display. Finland. 29th of July- 1st of August 2001. pp.272-275.

Upson, R. 2001. *Sonifications as Mathematics Teaching Tools*. In: Hiipakka, J., Zacharov, N., Takala, T. (Eds.), Proceedings of the 2001 International Community of Auditory Display, Helsinki University of Technology, Espoo, Finland. pp. 217-221.

Upson, R. 2002. *Educational Sonification Exercises: Pathways for Mathematics and Musical Achievement*. Proceedings of the 2002 International Conference on Auditory Display. Japan. July 2002. pp.70.

Vickers, P. & Alty, J.L. 1996. *CAITLIN: A Musical Program Auralization Tool to Assist Novice Programmers with Debugging*. Proceedings of the 1996 International Conference on Auditory Display. Xerox PARC, Palo Alto, CA94304, 1996. P.17-24.

Vickers, P. & Alty, J.L. 2000. *Musical Program Auralisation: Empirical Studies.* Proceedings of the 2000 International Conference on Auditory Display. Atlanta, GA. 2000. p.157-166.

Vickers, P. 1999. CAITLIN: Implementation of a Musical Program Auralization System to Study the Effects on Debugging Tasks as Performed by Novice Pascal Programmers. Phd Thesis, School of Computing & Math Sciences, Liverpool John Moores University.

Walker, B.N. & Cothran, J.T. 2003. *Sonification Sandbox: A Graphical Toolkit for Auditory Graphs.* Proceedings of the 2003 International Conference on Auditory Display. Boston, USA. July 2003. pp.161-163.

Wharton, C., Bradford, J., Jeffries, R. and Franzke, M. 1992. Applying Cognitive Walkthroughs to More Complex User Interfaces: Experiences, Issues and Recommendations. Proceedings of the SIGCHI conference on Human factors in computing systems. California, USA. 1992. pp.381-388.

Wilson, C.M. & Lodha, S.K. 1996. *Listen: A Data Sonification Toolkit.* Proceedings of the 1996 International Conference on Auditory Display. California, November 1996. pp.35-40.

Wixon, D., Jones ,S., Tse, L., Casaday, G. 1994. *Inspections and Design Reviews: Framework, History and Reflection - Usability Inspection Methods.* John Wiley & Sons, Inc.

Zhao ,H., Plaisant ,C., Shneiderman ,B. and Duraiswami ,R. 2004. *Sonification of Geo-Referenced Data For Auditory Information Seeking: Design Principle and Pilot Study.* Proceedings of the 2004 International Conference on Auditory Display. Sydney, Australia. July 2004. Website:
www.icad.org/websiteV2.0/Conferences/ICAD2004/papers/zhao_etal.pdf.
Downloaded: April 2005