Creating Functional and Livable Soundscapes for Peripheral Monitoring of Dynamic Data

Bradley S. Mauney and Bruce N. Walker

Sonification Lab, School of Psychology
Georgia Institute of Technology,
654 Cherry Street
Atlanta, GA, USA 30332-0170
maunbrad@yahoo.com, bruce.walker@psych.gatech.edu

ABSTRACT

Sonifications must be studied in order to match listener expectancies about data representation in the form of sound. In this study, a system was designed and implemented for dynamically rendering sonifications of simulated real-time data from the stock market. The system read and parsed the stock data then operated unit generators and mixers through a predefined sound mapping to create a 'soundscape' of complementary ecological sounds. The sound mapping consisted of a threshold-based model in which a percentage change in price value was mapped to an ecological sound to be played whenever that threshold or gradient had been reached. The system also provided a generic mechanism for fading and transitioning between gradients.

The prototype system was presented to stock trader test subjects in their work-listening environment for evaluation as a stand-alone system and in comparison to their preferred tools.

1. INTRODUCTION

As a result of the current state of computing technology, there is an ever-increasing amount of data that needs to be interpreted by researchers, analysts, and society as a whole. As the interpretation of large amounts of data becomes more and more a part of our everyday activities, our technology must evolve to support such activities by providing mechanisms for 'displaying' this data such that we can interpret its meaning [1].

Historically, large data sets have been effectively displayed visually through charts and graphs as a result of our powerful sense of sight. However it has been shown that in some situations, especially when a person is performing a visually demanding task such as a surgeon performing an operation, auditory graphs alone are more effective than visual graphs or a bimodal visual and auditory display [2]. This developing field of research into conveying data through auditory means is called sonification.

The sonification approach facilitates the interpretation of data by using nonspeech audio to convey data relations into perceived relations in sound [1]. Such successful and highly valuable sonifications include the Geiger-counter, used to detect radiation, and the pulse-oximeter, used to monitor heart rate and oxygen levels. In both cases, these sonifications were successful because they allowed the user to visually attend to other tasks while simultaneously monitoring the auditory data. In general, sonifications are used successfully when either the user cannot attend to a visual display due to physical limitations or the user cannot afford to attend because they are already involved in a visually demanding activity [1].

Research has demonstrated not only that sonification is better for conveying data in certain situations but also that large, multidimensional datasets can be sonified without overloading the user, as would be the case with a visual display [1]. Furthermore, Smith and Walker found that "the addition of useful information [such as tick-marks, axes, and labels] enhances the perceivability of auditory graphs" [3]. This topic of contextual information in the form of overlaid gradient information is the basis for this study.

Sonifications have already successfully been developed for various applications in science and medicine. Could sonifications also be implemented successfully in a well known yet challenging task environment such as stock trading? Furthermore, could this sonification be feasibly presented in an unobtrusive way such as a peripheral soundscape where the data is mapped to sounds of a natural environment such as a waterfall or beach?

The answers to these questions lie in creating an intelligent mapping between the complex, continuous movements of stock and index prices into a multimodal system of sonifications that accurately conveys these movements. The soundscape must be rich enough to convey data about price fluctuations but still simple enough that it can be easily parsed and interpreted into something meaningful by the user, even when only used on the periphery.

1. MONITORING THE STOCK MARKET

There are currently a great number of tools for monitoring a particular stock price or market indexes/averages ranging from simple web-based interfaces that refresh every 15 minutes to dedicated quotation and trading platforms such as TradeStation [4] used by professional traders and money managers. Among the more simple web-based graphing tools such as BigCharts [5] data is purely visual. Even among the professional grade, subscription platforms, there are only modest alert sounds that can be triggered by user-defined events.

In this primarily visual world, traders rely heavily on visually attending to 'tickers' or 'streamers' for the latest price movements. However, with so much information being displayed visually through these tools, there is a strong chance of visual overload or at least difficulty in finding a certain type of information.

This points to the likely effectiveness of a more peripheral, auditory system that conveys this price data with natural sounds. Understandably, several efforts have been made to sonify stock market data. These range from simply audifying the data [6] to much more complex multimodal displays created by Nesbitt and Barrass [7]. In many cases (somewhere in between these extremes), the simple price of a stock is mapped to a basic sound attribute such as pitch [8,3]. In other cases multiple data streams such as the stock price and trading volume are considered simultaneously and affect both the pitch and loudness of a sound [9]. Regardless, changes to a basic sound are driven by the data. An interesting alternative, and relevant to the work presented here, is the use of sampled audio segments as units or "grains" to build up a display through the cumulative sum of many concurrent grains. Nesbitt and Barrass [7] used recorded male and female voice segments in this manner. In that case the pitch and location of the speech clips was still driven by the data. It is important to note that the majority of these approaches have been aimed at the auditory display being either the sole information display, or a significant partner in a multimodal display [e.g., 7]. The auditory display often requires considerable attention in order to function as intended.

A different approach, which may be appropriate in somewhat different task settings, is to focus on a lightweight, peripheral auditory display that would not require the trader to be sitting in one place (they tend to move around a lot) or wear a head tracker or other devices. This lightweight approach would be unlikely to overload the user because by its very design it would be peripheral, supplementary, and unobtrusive, and would fit in with the existing workflow of intraday traders. The idea would be that the user could switch attention to the soundscape at will, perhaps when curious about the day's overall market movement, and the rest of the time either ignore it or simply appreciate its aesthetic quality. The system would not aim to replace the complex trading platforms of professional traders but rather supplement them, possibly aiding performance. Finally, an additional goal was to develop a simple but powerful platform that could be used to further study the effectiveness and design of such a peripheral auditory display in a variety of situations where a continuous flow of data may at different times need to be attended to or ignored, depending on other primary tasks.

2. SOUND DESIGN

Again, while numerous studies have shown that sonifications can be beneficial, the best implementation for a certain situation is still not always clear [2]. Studies have also found that natural sounds are more easily recognized in an office environment than artificial tones [10]. For this reason, we sought to design a system that would use natural sounds such as birdcalls, insect songs, rain, and thunder to create an immersive soundscape to sonify the stock market. This approach should lead to a display that can be easily distinguished from the background when necessary, but can also be allowed to fade out of attention, and not be tiring or "intrusive" when not desired.

In an ad hoc survey of existing stock data analysis tools and interviews with professional traders, we found that in monitoring a certain stock or index's movements, intraday traders are concerned with *percentage change* in price as much as or even more than the actual point or dollar value of that change. Additionally, many traders and technical analysts often use a 30-day *simple moving average* (SMA) as a baseline when inspecting the recent price movement of a stock. To our knowledge, these are stock market characteristics that have not been the focus of any previous sonifications.

In designing a sonification to address these variables that intraday traders use, we decided to employ natural sounds to serve as user-defined gradients or threshold values around the SMA. The user would define a series of thresholds as a percentage change above or below the SMA. When running, the system would calculate the current change in price and turn on the sounds that represent each threshold that had been met or exceeded. The effect is cumulative in that a lower threshold would continue to be active as higher thresholds are reached, resulting in a more complex sound the further the price moves from the SMA. The sound representing a threshold would continue to play until the current price moved back beneath the threshold. When a threshold is reached, the corresponding sound clip could either be looped continually, such as a bubbling brook, or could be played randomly a certain number of times per minute, as specified by the user.

In creating a realistic, aesthetically pleasing soundscape, we found sound quality to be an important factor. We used sound clips close to the audio quality of a CD (44.1kHz). For clips to be looped continuously, an audio editor was often necessary to crop the clip such that the endpoints were not distinguishable.

By this design, the soundscape is generic enough to be applied to any stock or market index or any continuous data source, with minor re-configuration. By adding the random factor to clip timing within some interval, the soundscape becomes more natural and less mechanical. In fact, the soundscape should never sound exactly the same twice, even when sonifying the same data. Table 1 contains an example soundscape configuration with mappings for each threshold, sound clip, and frequency property.

3. SYSTEM IMPLEMENTATION

With a sound design in mind we set out to find a technology that would support the development of such an application.

3.1. Choosing a Technology

In order to meet the requirements of flexibility, rapid-prototyping, and platform-independence, Kees van den Doel's JASS (Java Audio Synthesis System) [11] was chosen

Threshold	Type	Sound Generated
+1.60%	Random/ Hit	Large cricket calling at 2 samples per minute
+1.50%	Random/ Hit	Roadrunner calling at 1 sample per minute
+1.00%	Random/ Hit	Cicada singing at 1 sample per minute
+0.75%	Random/ Hit	Cuckoo calling at 1 samples per minute
+0.50%	Random/ Hit	Small cricket singing at 2 samples per minute
+0.25%	Random/ Hit	Bullfrog croaking at moderate tempo – 1 sample per minute
0.00%	Loop	River at normal gain, speed, and tempo
-0.50%	Loop	Light rain begins
-1.00%	Loop	Heavy rain (multiple overlapping samples, increased gain)
-1.50%	Random/ Hit	Thunder crashes at 1 sample per minute
< -1.60%	Random/ Hit	More violent thunder at 1 sample per minute

Table 1: *Example Soundscape Mapping. The threshold value, as a percentage change from the average, is mapped to a sound clip that is either looped continuously or played randomly a certain number of times per minute.*

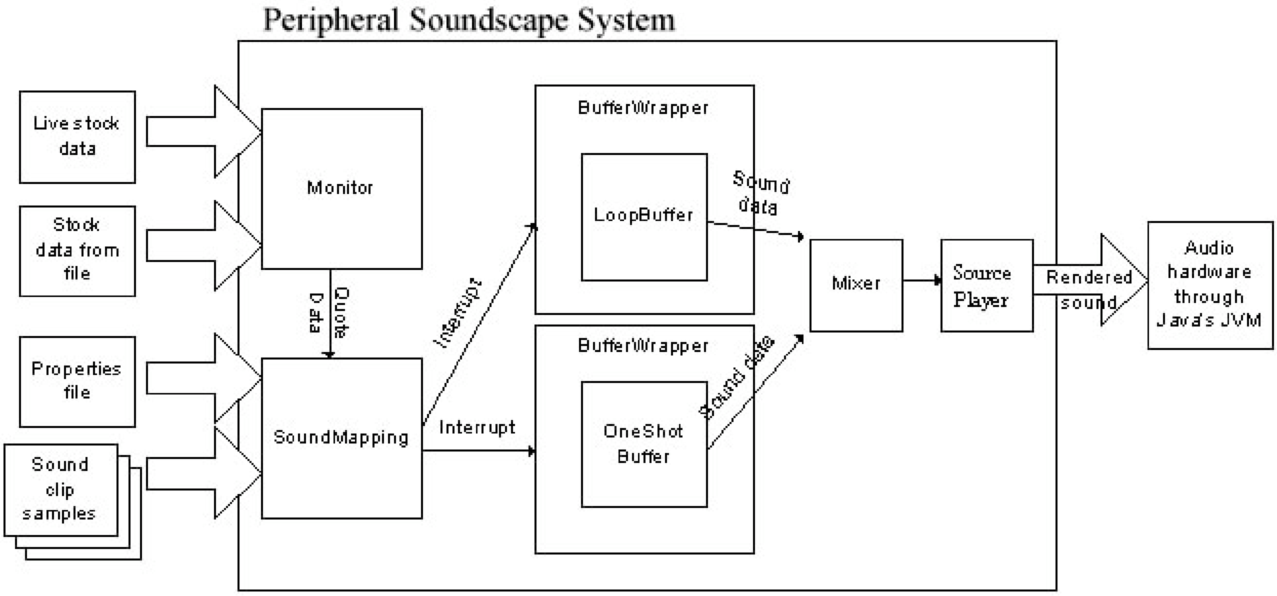

Figure 1: *System Architecture. Stock data is parsed and queued by the Monitor module. The SoundMapping module initializes LoopBuffers and OneShotBuffers encapsulated by spawned BufferWrapper threads as specified by the system's properties file. The Mixer module controls relative gain of each buffer and the SourcePlayer renders sound data to the audio hardware via Java standard libraries and the Java Virtual Machine.*

as the primary application programming interface on top of Java 1.4 standard libraries. The JASS package provided all necessary support for the real-time synthesis, filtering, mixing, and rendering of sound while at the same time keeping latency and processing load low.

Another key feature of JASS is that by using a unit generator approach, the somewhat cumbersome underlying JavaSound API is conveniently abstracted providing a simple, fully extensible foundation for the application. JASS also omits complicated music support such as MIDI, which is not needed in this case. In creating our livable soundscapes, JASS proved to be the most convenient tool available.

3.2. System Architecture

The final system architecture (see Figure 1) uses a multithreaded, unit generator based approach. Each threshold mapping sound clip is managed as a JASS unit generator or buffer of audio information that can be 'hit' (played once) or looped continuously once started. Each unit generator and its respective type, threshold, fade-in duration, fade-out duration, gain, and cycles per minute are defined in the properties file that is loaded by the SoundMapping module on startup (see Figure 1). These properties are then used to initialize each buffer and spawn each BufferWrapper thread.

The output of each unit generator (sound buffer) is pushed to a mixer module that controls the relative gain, automatic gain control, and pan for each channel. Each unit generator is encapsulated by a BufferWrapper module, each running in a separate thread, which is responsible for controlling the internal buffer. The final output from the Mixer is passed to the SourcePlayer (child thread) module, which then renders the sound to the audio hardware through Java standard libraries and the virtual machine.

On the input side, the Monitor module is responsible for parsing quote data from either a real-time source feed or a flat file. These quotes are then assembled into a quotes queue that serves as an input buffer. The SoundMapping running in the master thread then receives quotes from this buffer at regular intervals and notifies all BufferWrappers (each a child thread of SoundMapping) of the update through an interrupt call. Upon interrupt, each BufferWrapper updates itself with the current threshold value by computing the deviation from the simple moving average, checking this against its own threshold, and activating or deactivating appropriately. If activated, the buffer provides its sound data to the mixer, which is then run on the SourcePlayer. If deactivated, the buffer simply provides the default zero value.

4. INTERFACE

Since the application is primarily an auditory display, no graphical user interface was developed for the current version, although the application could be extended to include such an interface in a future version. Currently the system runs from a standard, Java-enabled console and is configured with command line arguments and properties files.

The system reads in a plain-text properties file that defines all system properties as well as each threshold value and the corresponding attributes of that threshold (see Figure 2). The system also reads a specified data file from which simulated real-time data can be generated. Based on the properties file, the system will attempt to load a number of .wav format audio files as specified in the threshold definitions. Since all properties are defined in plain-text files, the system's configuration and threshold properties could be easily edited from a standard text editor or, as indicated, from a future graphical user interface.

```
…
grad0=1.6 //threshold
grad0clip=lcricket.wav //sound clip
grad0type=hit //loop or hit randomly
grad0hpm=2 //samples per minute
grad0gain=1 //relative gain 0-1
grad0fis=1 //fade in seconds
grad0fos=1 // fade out seconds
…
```

Figure 2: *An example entry from the properties file corresponding to the first entry of Table 1. This entry specifies that while the current data value is greater than the threshold of 1.6% above the average, a large cricket will call twice per minute with a maximum volume relative to the rest of the soundscape. The clip will both fade in and fade out over one second.*

5. EVALUATION OF COMPLETE SYSTEM

5.1. Evaluating the Sound Design

To evaluate the sound design, the system was presented to stock traders of a private equity fund that manages tens of millions of dollars of investor funds. These traders place several large trades per week averaging $20 million in volume per month. Even though they only place a few trades per week, they nearly constantly monitor the stock, currency, commodity, and futures markets to identify trading opportunities. Each trader typically sits at a desk and uses a computer based trading platform [4] with extensive visual graphing tools but nothing particularly auditory related. The traders interviewed each had their own private office, but this is an exception to the industry norm–generally a trader works from a 'bullpen' with several other traders.

5.1.1. Method

The soundscape was evaluated using a semi-structured interview and progressive training (described below). The soundscape was first presented with the subject only having been told that the sounds represented the price movement of the S&P 500 index. The subject was then asked to describe what he had heard. Next, the concept of threshold mapping was explained and a description was provided of each corresponding sound clip. The soundscape was then played a second time, and the subject was asked to describe what he had heard. The subject was then shown the underlying price quote data and given a brief description of the system's threshold activation and events for which they should listen. After a third listening, the subject was asked to describe the trends in the data from what he had heard.

After the listening task, each subject was then interviewed as to his perception of the system and its usefulness in a stock-trading environment.

5.1.2. Findings

On the first listening the users were generally confused as to the nature of the soundscape. They could distinguish the sound clips and recognize them as events of some type, but none perceived the threshold mapping that was taking place.

After the second listening, each subject reported he could roughly follow what was being displayed. With the thresholds in mind the listeners could identify when the price was moving up and down. Upon the third listening, the subjects reported they could better distinguish the trends while knowing specifically the events to listen for and could identify when a threshold had been reached and when a local maximum or minimum value had been passed.

The general reaction was that the system is a novel approach to conveying stock market data but that many sound mappings would be too 'busy' to be used in an office or trading floor environment citing that these environments require considerable concentration and are naturally noisy. It was also noted that traders are primarily dependent on visual graphs, are quite accustomed to using visual tools, and could be resistant to the adoption to such an auditory tool because of its auditory (and thus foreign) nature. This may indicate the need for further research, such as a longitudinal study.

5.2. Evaluating the Technology

The system, which was conceived as a stock market specific sonification tool, has evolved into a powerful, extensible, and generic sonification tool for threshold based soundscapes. The system could be adapted to take in some other data feed and easily configured to present this data as a soundscape. The cross-platform nature of the underlying Java components and the processing efficiency of the JASS package allow the application to be run on even a modest computer with minimal latency and overhead. In fact much of the success of the system could be attributed to the robustness of the underlying JASS components. For example, a 550MHz/256MB RAM/Integrated Sound machine had little trouble running four parallel instances of the entire system.

On the other hand, the system has several drawbacks that were identified during testing and user evaluation. The system is currently only capable of sonifying one channel of data at a time, i.e. one stock or stock index. The decision was made not to allow multiple channels due to the complexity of the sound design and threshold mapping. Also, the system only supports the model of a percentage change from some standard value; it does not support threshold mapping of pure values. This decision was made to match the way our stock traders like to perceive their data, as a percentage change. Lastly, the system currently runs only from a console, making it difficult to use for the majority of the user population. A fully featured graphical user interface would be logical addition to the system.

6. CONCLUSIONS

In general, the users/test subjects felt that the threshold-based soundscape was a fairly intuitive way to perceive price fluctuations, especially with regard to percentage deviation from moving averages. They also felt that the sound for each threshold should be unique enough to distinguish it from the others when multiple samples are overlapped. They agreed that the overall distinguishable characteristic of nature sounds versus the sounds from an office environment aided in the perception and interpretation of the data and was 'relaxing' to listen to. This provides support for the use of sounds that are natural, but not necessarily part of the listener's everyday acoustic ecology. They also agreed that it would be more useful than visual graphs when they are away from their desks and moving around their offices, which supports our user-centered decisions to make the display fit into the existing work flow. They felt that the peripheral nature of the system would keep it from 'getting in the way' once they became more acclimated to it being there. They also felt that significant training would be needed in order to use the system with the accuracy and equivalency of their standard visual tool. However, it should be noted that the goal of this project was not the identification of a specific data value, as might be the case with a visual system, but rather the notification of threshold crossings. For this purpose the listeners felt it would be very effective and useful.

The user evaluation raised many new questions regarding performance, training, and visual versus auditory attention. Further research on the system would involve quantifying with test subjects many of the very qualitative results from the user interviews such as performance and equivalence versus or in combination with visual graphs.

One possible experiment would involve a user listening to the soundscape while performing some visually demanding task, as a test of the peripheral nature of the display. The user could be periodically asked to assign a value to the current state of the soundscape based on their perception of which thresholds had been crossed and then return to the visual task. As Smith and Walker have pointed out [3], training is an important element in performance with an auditory display. Thus, the users of the stock sounds system discussed here could also have various types and degrees of training with the soundscape prior to the combined task, ranging from a simple explanation of the threshold mapping to a full tutorial.

With regard to the software application, useful enhancements might include: a sophisticated graphical user interface to simplify the loading and sonification of data files; an internet module with live data feeds that would allow the user to select a particular stock and listen to the sonification in real-time; and a library of pre-assembled soundscape themes such as rain forest, city street, or sea shore that would help other researchers quickly build up a display that is coherent and balances aesthetics and performance from the very start.

7. REFERENCES

[1] Kramer, G., Walker, B., Bonebright, T., Cook, P., Flowers, J., Miner, N., et al. (1999). *The sonification report: Status of the field and research agenda*. Report prepared for the National Science Foundation by members of the International Community for Auditory Display. Santa Fe, NM: International Community for Auditory Display.

[2] Barrass, S., & Kramer, G. (1999). Using sonification. *Multimedia Systems, 7*, 23-31.

[3] Smith, D., & Walker, B. (2002). Tick-marks, axes, and labels: The effects of adding context to auditory graphs. *Proceedings of the 2002 International Conference on Auditory Diplay, Kyoto, Japan, July 2-5 2002.*

[4] TradeStation Securities. "http://www.tradestation.com"

[5] BigCharts by MarketWatch. "http://bigcharts.marketwatch.com/"

[6] Frysinger, S. P. (1990). Applied research in auditory data representation. In E. Farrell (Ed.) *Extracting meaning from complex data: processing, display, interaction*. Proceedings of the SPIE, Vol 1259, pp. 130-139.

[7] 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 Diplay, Kyoto, Japan, July 2-5 2002.*

[8] Brewster, S. A., & Murray, R. (2000). Presenting dynamic information on mobile computers. *Personal Technologies, 4(2)*, pp. 209-212.

[9] Neuhoff, J. G., Kramer, G., & Wayand, J. (2000). Sonification and the interaction of perceptual dimensions: Can the data get lost in the map? *Proceedings of the 2000 International Conference on Auditory Display, Atlanta, GA, USA.*

[10] Mynatt, E.D., Back, M., Want, R., Baer, M., & Ellis, J.B. (1998). Designing Audio Aura. *Proceedings of the CHI Conference, CHI 1998,* (pp. 1-8).

[11] van den Doel, K., & Pai D. (2001). JASS: An Audio Synthesis System for Programmers. Proceedings of the 2001 International Conference on Auditory Display, Espoo, Finland, July 28-August 1, 2001.