Random Sound and Vision

Ian Andrews

I have been working with video and electronic music since the early 1980s. From the mid 1990s, I had maintained a website which documented my activities (see http://ian-andrews.org). Because this site was very kindly hosted without charge on a friend's server, I considered the inclusion of rich media content, such as MP3 files, as too much of an imposition on my host's bandwidth—with its high cost at the time. But I really wanted sound on my site because it was such an important part of my practice.

In 2000, I began working with Macromedia Flash to explore the possibilities of sound, and low-bandwidth vector graphics and animation. I was soon creating low-bandwidth sound compositions by using short (less than one second) segments of sound sequenced in a Shockwave Flash (SWF) file. In the late 1980s, module file formats were developed so that musicians using Amiga computers could transfer music files over networks. These MOD files consisted of instrument (audio samples) information, and sequence or note pattern-timing information. Applications used to create and play the files back on the Amiga were known as *tracker programs*.

Later in the 1990s, a similar technology was developed with MPEG-4 Structured Audio (SA), where the generation of sound, using Structured Audio Orchestra Language (SAOL), and timing information coded in Structured Audio Score Language (SASL), are combined for the transmission of low bit-rate, high quality audio over networks. I was sure that a similar methodology could be employed with Flash, but with the added attraction that all the visitor needed to play back the music was the Flash plug-in, which was already, by that time, quite ubiquitous (although MPEG-4 Structured Audio greatly exceeds the audio possibilities of Flash, it has not been implemented in popular MPEG-4 players such as Apple Quick-Time).

Because most musical compositions contain a large number of repeated elements—which often are quite short—these elements need only be downloaded once, and then referenced by the timing information in the SWF file to play back on the end user's computer. This drastically reduces file size. The aesthetic direction my music had taken had tended toward minimal structures of clicks, buzzes, and drones, all of which I could easily achieve by looping and layering these short audio samples. In many cases, I used compression and bit-depth reduction to my advantage on some of the sounds to create distortion effects. I found that I could create rich textures and timbres using several compression types and levels. I could even use some short sounds at full bandwidth without contributing greatly to the final file size.

A nice feature of Flash is its deployment of multiple timelines (or movie clips) within one project file. Combining looping movie clips of different lengths, and nesting timelines inside other timelines, can quickly create complex sound compositions. This timing complexity opens up some interesting composition methods that are beyond the capabilities of standard sequencing applications. But the real power of Flash lies in its scripting language, ActionScript. I soon found that I could program arrangements of sequences (movie clips) to play at various times throughout a track, but the one feature that really excited me was the random function (a pseudo random number generator). I was fascinated by the idea that I could make compositions that could play for hours, rarely (if ever) repeating the same phrase more than once.

My first exercises in this area were the 21 small pieces that make up *Ether-1* (2002). Each piece utilises random generation of vector graphics, text, and sound. The project was inspired by my fascination with the sounds of short-wave radio, and my research into early 20th Century Italian and Russian Futurist poetry and its obsession with the technology of wireless telegraphy.

I considered *Ether-1* as an online, experimental, electronic music album rather than a piece of animation or Web art. My next project was a compilation album consisting of artists working in a similar medium. I was curious to find if others were working in similar ways, and thought that it would be a good way to establish some contacts. For *Ether-2,* I asked artists to submit non-interactive SWF files in which sound featured as a major component. File size was not to exceed 100 Kbytes. *Ether-2* features work from artists from four continents.

In 2003, I considered the idea of creating a sound-only Flash online album. *Radiohack* has nine tracks, the largest of which is only 26 Kbytes. The pieces all are randomly generated, but unlike the earlier pieces, which were of infinite duration, these pieces play for a fixed time. It's possible to listen to this music on the slowest of dial-up connections, but on broadband the response is almost instantaneous.

Later that year, I conceived the idea of creating a very sparse music piece composed of a number of random modular pieces to be played simultaneously on a number of laptop computers. I submitted this project to the International Symposium of Electronic Arts (ISEA) 2004 and it was accepted, although not in its original form but as a two-channel video installation. It was displayed at the Kiasma Museum of Contemporary Art in Helsinki. The work, *In a Few Seconds Across the Ocean,* refers to 20th-century art that might broadly be described as a poetics of radio (and noise) ranging from the Futurist poetry of Marinetti or Klebnikov to Stockhausen's 1966 work *Hymnen: Anthems for Electronic and Concrete Sounds. In a Few Seconds Across the Ocean* uses several generative techniques—random number generation, permutative (elements in individual clips of different lengths that, when combined, form a finite number of different patterns each time they repeat) structures, and graphics and sound interaction, to produce aleatory music graphics and text (see Figure 1). The piece plays continuously in a loop that never repeats in the same way twice. The generative structure is designed to create a multi-channel sound environment where the randomisation of the process distributes the different sound elements throughout the space.
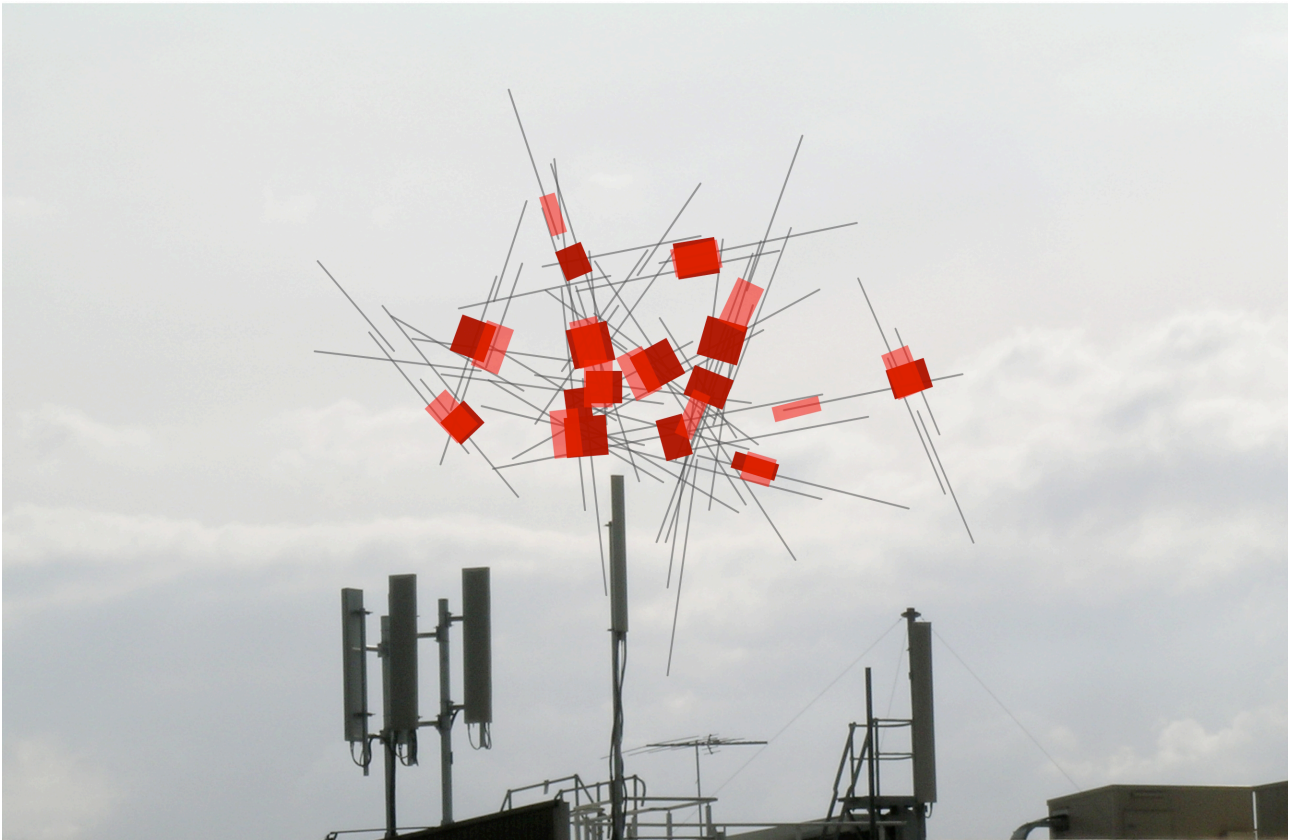
Figure 1. Still from In a Few Seconds Across the Ocean (*2004*).

In Helsinki I met Timo Kahlen, who was also exhibiting at Kiasma for ISEA 2004, and we broached the idea of a collaboration. *Sounddrift* (2005) combines Kahlen's beautifully constructed scale models of empty museum spaces with my random sound and graphic clusters. The piece is interactive and requires users to click on a small moving target in to progress through the different rooms. The idea of this difficult interface was to retard the progress of the visitors so that they would have a chance to listen without moving too quickly from one room to the next.

Up until 2007, much of my work with Flash involved the use of vector graphics and bitmap stills but not video. I began to think of how randomised video could work. It doesn't easily lend itself to the scripting strategies of graphics and sound, so I needed a quite different approach. I thought about cutting video into short sections and then randomising the playback order of those sections. I could also randomise the sound play back independently of the video. For my first piece, *Etiquette*, I used a documentary on teenage social-guidance, *Junior Prom* (1956) from the *Prelinger Archives* (see http://www.archive.org/details/prelinger). This worked well because the original film was structured in the form of a number of short three to five second 'tips' that I could easily isolate from the main narrative and then recombine in interesting ways. The second piece, *Satisfaction*, combines the images of two films from the *Prelinger Archives*: a Cold War propaganda film, *The Challenge of Ideas* (1961), and a film about consumption, *American Thrift* (1962). Because much of the original films contain music throughout their soundtracks, I liked the idea of repeating certain refrains more often than they occur, which gives the piece a certain banality.

Although Flash offers a wide range of possibilities and almost universal playback, I have to say that creating music in Flash is far from easy. It's virtually impossible to construct something with any idea of accuracy. It's more a process of playing around with different ideas until you get close to an approximation of what you want. After a while you get a feel for it so that you can quickly establish what is possible and what is not. Timing idiosyncrasies have to be, in a sense, built in to the composition. Glitches and flaws must be welcomed as aesthetic additions. Once the

inevitabilities of these errors are accepted into the composition process, the possibilities are intriguing.

All projects discussed can be seen at:
http://ian-andrews.org