Introduction

The new Cellmusic system has been developed to enable interactive performance with a number of players using domestic mobile devices (such as mobile phones). Each node consists of a mobile device running the Cellmusic software, and (optionally) a variety of other performance and controller devices. Those using the software assume the dual role of both composer and performer. There are a number of developments which arise from such an environment. Different methods of composition evolve, due to the distributed nature of the nodes. New software instruments and new methods of performing might arise. New working practices and methods of group interaction might also be produced.

This paper is the first publication of the Cellmusic software system. It will describe the design considerations which led to the development of the software, the motivation for producing the system, a description of the technology and device network.

1. BACKGROUND

Traditionally electro-acoustic composers might produce music with consideration to one or more of the following:

- High quality recordings of the natural environment
- Diffusion of sounds over a network of multiple speakers
- High quality sonic transformations between 2 or more sound sources
- Novelty in the composition or performance process, for example by using algorithms for music event generation

The digital domain of sound generation offers the composer a practically unlimited palette of sounds. Often, however, interesting pieces arise from the limitations which the composer might impose upon him or herself. For example, the range of sound sources might be limited according to a geographical location, or to a particular time or theme.

Some software (such as MAX MSP [1]) allows music generation services to be represented hierarchically in a network across 1 or more computers. The internet also facilitates network-based, collaborative works [2]. With the event of accessible wireless technology, the appeal of using it in performance is increasing (e.g. the Princeton Laptop Orchestra [3]). Small, portable devices have also been used for sound sequencing [4].

Cellmusic was initiated to build on these concepts, and to explore challenges in composing and performing for devices not originally intended for music performance.

2. THE CELLMUSIC SERVER

The Cellmusic server system has the role of distributing the software to remote devices and to pre-load devices with instructions for performance. This is facilitated using Bluetooth discovery services. Any java-enabled, discoverable Bluetooth device could potentially be invited to join a performance.

The server is a Bluetooth-enabled PC running Java. Since Cellmusic is a distributed composition system, there are often conflicts when several devices (nodes) make requests. For example, one device might be requesting control of a musical part at the same time as another. In this case, the server's role will have to resolve disputes based on pre-programmed rules which it loads into the mobile devices.

Bluetooth is designed to run at distances of up to 10 metres, however high powered devices will function up to 100 metres. A Bluetooth device can be discoverable. It may respond to an enquiry from another device with the following information:

- Device name
- Device class
- List of services
- Technical information (relating to the device features, manufacturer, Bluetooth specification implemented etc)

Some devices are limited to the number of simultaneous connections they can achieve (typically 3). In some cases, one device may be required to pair with another in order to access its services. A device may have to break a pairing in order to make itself discoverable to others. Every device has a 48-bit address which is unique. Generally these are hidden, with user-defined names appearing in response to scans.

During a performance, a message may be sent from one phone to another requesting that it plays a sequence at a certain time (determined by the phone's internal clock or timer). This is quite a crude mechanism for synchronization and further testing is required to determine latency times in performance.

3. COMPOSITION HIERARCHY

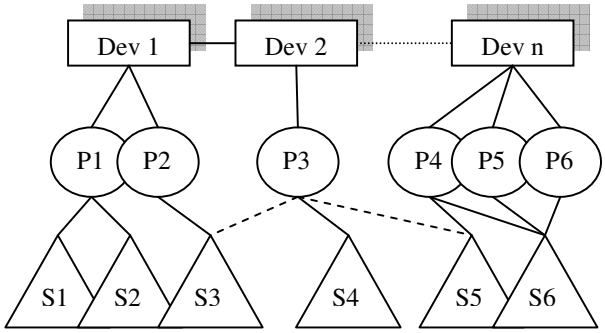

Figure 1. An overview of Cellmusic showing the devices, processes (p1 to p6) and sound objects (s1 to s6).

The Cellmusic system consists of a several devices with each device programmed with a number of processes (See figure 1). A process might be, for example, to play a sequence of notes or trigger the playback of an individual sound.
Processes on one device may refer to sound objects stored on the same client or a different one. A sound object might be a sound produced using MIDI using the phone's internal sound synthesis system or a sound sample (e.g. mp3 file).
Software and initial performance data are programmed using the handset. In some cases, devices are passive and will perform according to the pre-programmed performance data (e.g. a sequence). In other cases, devices will enable performers to interact with, guide and direct the piece by influencing events on other devices.

Although a fundamental motivation behind Cellmusic was to allow a truly distributed system, it is envisaged that the composer will have the option of broadcasting sound over a number of devices outside of the general control of the network, for example by using high quality stereo speakers. In this situation, the server will function as a gateway to route audio (which might still be controlled by the various nodes) to other devices.

Mobile devices are connected in series. Each device may send a message along the chain to another device. Messages may fall into the following categories:

- Initiation of a sonic event
- Transformation of a sonic event already in progress
- Selection of the sonic event being played
- Cueing of another device to start playing

The performer is presented with the following options displayed on the mobile device:

- Change the current sonic event
- Enter a sequence of events (which may either be initiated on the current device or another device)
- Rest (or pause)

- Play a sonic event (either immediately or after receiving a cue message from another device)

4. THE CELLMUSIC CLIENT

The Cellmusic client runs on a mobile phone or device compatible with Java. The function of the client is to collect sonic data, process it, and perform with it. Clients synchronize and communicate with each other using the Bluetooth communication protocol.

If Bluetooth is enabled on a device, the user will receive a request to download the software. If this is accepted, and the user grants all permissions to the software, then the installation procedure is completed. The user may then select options to modify the performance data and to interact with other performers.

5. RESULTS

There have been several non-public performances of Cellmusic. This has enabled the system to be tested before being used in public performance.

5.1. The Rolling Wave

The rolling wave uses one sonic event and sequence event to trigger a rippling cascade of sound. Performers stand in a straight line. Sequential commands are loaded into each device to trigger the sound of the next device in the chain. The last device in the chain sends a command to the first device, which causes the triggering of 2 sounds in sequence with a slight delay between them. In the next round, 3 sounds are triggered from each device, and so on. This creates a rippling effect of ever-evolving densities. After a pre-determined number of rounds, sounds are dropped from the sequences 1 by 1 at the end of each round, until there is silence.

5.2. City Living

This piece is meant to evoke the fast pace of city living. The performers represent the path of a train. At the start of the piece, the train emulates the sound which might be heard near any point beside the rail track. As the piece progresses, the listener is taken to a sonic abstraction of a city dweller's state of mind as his life becomes more stressful and how the day-to-day routine of a 9 to 5 job becomes unbearable.

5.3. cell-o-phane

In this piece, performers stand in a circle with one node in the middle. Small, sonic events are used to surround the middle point, which, at the start of the piece emits a strong audio signal. The piece progresses, with granular textures surrounding the middle node. Eventually, the identity of the middle node is wrapped (and eventually smothered) under waves of sound.

6. EMERGING COMPOSITIONAL PROCESSES

The most inspiring use of Cellmusic comes from its potential for accessibility and portability to many different environments, including public spaces. Acting without a server, users can simply transmit the software to each other, set up their respective performance data, and initiate a performance. The potential of this will be explored further, but in the trials so far, it has become apparent that freedom from fixed performance environments and ease of use are appealing to composers and performers alike.

Using a laptop as a server, there is potential to create a more complex performance space in public. Again, the trials have shown that pre-loading performance data to a performer's phone inspires the new performer to investigate tweaking the options while gaining confidence in taking more control of the piece.

The preliminary private performances and tests of the system have allowed composers and performers to gather sonic material which might be used immediately and spontaneously in performance. Sonic material must be selected carefully, since audio reproduction is mainly dependent on devices' built-in speakers.

One behavioural element which emerges with the system is the concept of performers performing ‘subversive’ acts whereby they attempt to take control of a performance or to swamp the network with their events and performance data.

7. CONCLUSIONS AND FURTHER DEVELOPMENT

Currently there is a limitation of the number of devices which can be connected, before issues of latency start to manifest themselves. The speed of response of the network is dependent on the slowest device. Therefore the use of the server will be used to explore scheduling events.

Other devices could be incorporated into the network, for example, wireless speakers. Also, devices for lighting control could be explored. An installation might allow users to wander into a space, download the software to their mobile device, and take interact with an immersive environment.

The current user interface uses text and 2 dimensional representations of sound data and transformations. There are a host of 3 dimensional display features accessible through Java which could be utilized. A version for cellmusic is not yet available for iPhone, and this would be an ideal platform to explore given the speed of the processor and the touch screen interface. Indeed, the software might incorporate calls to any hardware device to establish the availability of features and services which it might want to use.

Creatively, cellmusic relies upon composers and performers gathering sonic data. This concept could be expanded to allow manipulation of video data from phones. Either individual users might view their own video performance, or a server might process incoming video messages to produce a shared visual experience.

Development of mechanisms to limit user participation might also be considered. Currently, a composer of a piece may wish individual performers to control only certain elements at certain moments.

Synchronization of devices might be explored even further. It may be desirable for certain pieces to be controlled from the server in terms of timing and event management. In order to achieve this, other protocols might be investigated.

Importantly, the development of a website (cellmusic.co.uk) will allow users to share experiences, interact, share pieces through downloadable data and provide useful suggestions and information for future versions of cellmusic.

References

[1] Puckette, M. “Combining Event and Signal Processing in the MAX Graphical Programming Environment” Computer Music Journal, Vol 15, No. 3, 1991.

[2] GIBSON I.S., DOVEY M. A web-based system for collaborative electro-acoustic composition. International Journal of Advanced Manufacturing Technology. 2006. London, UK: Springer.

[3] Trueman, D., Cook, P and Smallwood S. "PLOrk: The Princeton Laptop Orchestra, Year 1”, in *Proceedings of the International Computer Music Conference*, 2006. New Orleans.

[4] Bryan-Kinns, N., & Healey, P. G. T. “Daisyphone: Support for Remote Music Collaboration.”, in Proceedings of NIME 2004, Hamamatsu, Japan, pp. 27-30