Abstract

Here we introduce ixi software – a project that aims to explore new interactive patterns and virtual interfaces in computer music software.

1 Introduction

ixi software (Magnusson, 2005a & 2005b) is an ongoing research project focusing on creating abstract screen-based interfaces for musical performance on computers. These are graphical user interfaces (GUIs) that do not necessarily relate to established conventions in interface design, such as using buttons, knobs and sliders, nor do they necessarily refer to musical metaphors such as the score (timeline), the keyboard (rational/discrete pitch organisation) or linear sequencing (such as in step sequencers or arpeggiators). Instead we represent musical structures using abstract objects that move, rotate, blink/bang or interact. The musician controls those objects as if they were parts of an acoustic instrument, using the mouse, the keyboard or other control devices. We have created over 15 of these instruments – each exploring new modes of interactivity where some of the unique qualities of the computer are utilised in fun, inspirational and innovative ways. Qualities such as remembering the musician's actions, following paths, interaction between agents, generativity, randomness, algorithmic calculations and artificial intelligence: – all things that are lacking in our beloved acoustic instruments.

2 Semiotic Elements in ixi software

In a recent paper (Magnusson 2006), we explained our loose and informal language for these instruments – a semiotics that suggests to the musician what the functionality of each interface element is, and what it can signify in a musical context.

We divide the elements of the interfaces into actors, context and network. The actors are the functional elements of the interfaces – the active objects that send out OSC (Open Sound Control) information to the sound engine. The sound engine can be any software that receives OSC information. The context is the environment in which the actors live. The environment varies but can have gravity, obstacles, routers, or define the location dependency of the actors. The network is the general setting of the interface usage, i.e. which parameters the actors are mapped to, which control devices are used to drive and affect the actors of the interface. In the network we consider other external factors such as soundcard, co-players, the sound system, etc.

As the actors in the most recent ixi applications output OSC information and are independent from the sound engine, we see them as signs with signifiers that have a structural (musical) context, but lacking a defined signified, i.e. the parameter in the sound engine that is being controlled. The users of our software have reported this as a positive and inspiring feature of the applications.

Figure 1: SpinDrum. The wheels contain 1 to 10 pedals. They rotate in various speeds, and when a pedal hits top position it sends out OSC info to the sound engine.

3 Conclusion

This demo at ICMC will demonstrate how we see this semiotics in the sense of the mapping between interface actors and parameters in the sound engines. The relationship between composing and mapping will be explored in the context of these applications, and we will talk about the technologies we use (SuperCollider, Pure Data, Java, Python, OSC) that yielded this split between an interface and a sound engine.

References