Another Take on Renovating Dated Technology for Concert Performance

Richard Dudas
CREAMA
Hanyang University
Seoul, Korea
dudas@hanyang.ac.kr

ABSTRACT

In a decade of collaborating closely with performers on concerts of electro-instrumental music, not as a composer but as a technological co-performer of other composers' works, it has been regularly necessary to update – and sometimes improve – the technology used. This need often stems from the rapid obsolescence of both the hardware and software that has been used throughout the years. Indeed, many compositions that were created as recently as a decade ago need to have their technical component re-evaluated before a performance: all too often hastily designed and poorly maintained software necessitates some degree of overhaul in order to be used reliably in rehearsal and concert. Although this is not a new topic in computer music, many of the initial writings about this re-design of obsolescent technology have concerned themselves with a "historical performance practice" approach. Many performers and composers, however, concur that it is necessary to rethink and improve pieces' technological component in order to be able to re-interpret them afresh. Just as performers desire to make the music they perform come alive for their audience through the use of creative interpretive decisions within the confines of the musical style and limits prescribed by the score, so, too, should the technological performer endeavor to make the electronic and computer music repertoire sound its best, from both technical and musical standpoints, in order to entice and enchant today's increasingly discerning, technology-savvy audiences.

1. INTRODUCTION

An important part of my role as a computer musician (and also a significant part of my work within the concert activities of the CREAMA studios at Hanyang University) has been to work together with musicians to promote concerts of electroacoustic repertoire in the concert hall. This implies breathing new life into performances of old (and sometimes not-so-old) electroacoustic and computer music. In the context of a concert performance, a piece of music ideally should be made to sound its absolute best, regardless of whether it be a "classic" piece from the 1950's or a world première using the latest computing tools. Today, audiences are used to being dazzled with high-tech computer-controlled sound and lighting in a variety of domains both inside and outside the realm of the performing arts (e.g., rock concerts, film, theater, aquaria, firework displays and myriad other public events), and this certainly informs their expectations when they attend "classical" concerts involving technology. On the other side of the stage, those performers who work frequently with electronics are generally interested in creating a rich, memorable, captivating "show" (the vast majority of those with whom I have worked certainly are), and therefore the quality of the electronics should ideally be perceived by the audience as matching the caliber of the performer's showmanship, and, more generally speaking, it should also broadly fit into today's audiences' prevailing quality expectations.

The viewpoint taken in this paper is from my role as a *performer* of electronic and computer music, not as a composer or software programmer – though these certainly all inform each other to some degree. When I am "on stage" (which, for me, in reality is actually not on the stage but somewhere out in the hall [1]), I want to make sure that my role as a co-performer in a concert of electroacoustic music – usually sitting behind the computer with a finger poised over the spacebar [2] – allows compositions, composers and performers to sound the absolute best they can. This is obviously a mindset shared by others in the field. Consequently, part of the impetus for writing this paper is to continue to promote the wider dissemination of acquired knowledge in our discipline (as such, some of the material contained in this article expands upon some earlier writings [1]). By providing information about many of the updates which I have made – out of necessity for the purposes of concert performance – to pieces involving technology, others might hopefully also benefit from these experiences, and the techniques used.

[1] Although setting up the mixing board in the audience somewhere close to the "sweet spot" in the hall may be theoretically desirable, this is not always practical due to space limitations within the hall itself. The rise of tablet computers as control devices is certainly helping ease these limitations, since this allows the technological co-performer to control the electronics and/or mixer from anywhere in the hall.

[2] Although this provides a nice mental image to the reader, I actually always recommend ***not*** using the spacebar (nor mouse button) for triggering musical events, but rather another, smaller, key which makes less noise, such as one of the arrow keys or the *enter* key on an extended keyboard.

2. DATED TECHNOLOGY

The desire to assist performers in putting on a good "show" often entails having an additional, unsolicited, part-time job as "data archaeologist" in the studio in the weeks and months leading up to the concert. Many musicians who have been regularly involved in computer music performance have written at length about the need to maintain and update software and to rediscover and preserve obsolete hardware [2, 3, 4, 5, 6, 7]. Some of these same people (along with many others in the field) have been working on designing comprehensive software systems and repositories to consolidate repertoire into one technology. These include, but are not limited to Miller Puckette's Pd Repertory Project [8], Jamie Bullock's Integra Live [9], and David Brooke Wetzel's Interactive Event Manager (IEM) [10]. The Max/MSP and Pd environments themselves have become a veritable computer music lingua franca [11], and most of the live electronic music which I, myself, have performed has made use of the Max/MSP environment for its electronic component, so any examples shown here will use that environment without the use of any particular third party framework.

I do not want to delve too deeply into the aesthetics and ethics of reworking old technology – that would take up an entire book – but I do need to touch upon the subject briefly because there are quite a few different schools of thought on this topic. Although I am presenting the re-working of dated technology via my personal viewpoint, it is just one of many perspectives. Some have discussed the idea of a "historical performance practice" approach to electronic music [12, 13], preserving (or even rebuilding) and using the original technology, while others believe in thoroughly analyzing the often obsolete technology used in the original version of a piece in order to be able to recreate it as closely as possible using modern technology. Although I personally believe these are both important first steps toward understanding that technology, I would nonetheless go one leap farther: I believe that in many cases the obsolete hardware and software technology can be updated and *improved* (or rather *streamlined*, if the reader considers the word "improved" to be too presumptuous in this context) without altering the composer's musical and conceptual intentions. I also believe it is the performer(s), who should make those calls, as a consequence of informed artistic evaluation of the music, just as they would do when interpreting traditionally notated music from the more distant past.

Of course, there are many cases where the imperfections of a given technology have been exploited by composers for both musical and extra-musical purposes; this use of the technology should certainly be taken into consideration in any updated version. To cite one example of this creative use of faulty technology, the inaccurate pitch detection of the early IVL Pitch to MIDI Converter from 1985 was used by composer and saxophonist Bruno Spoerri as a variation-generating feature in his improvisatory piece "Controlled Risk" [14]. Using a more precise modern pitch tracker in this scenario would undermine the musical intent of the original piece. However, on the flip side of the record, so to speak, in a scenario where one is updating a piece that has made use of an early hardware effects box, such as the Yamaha SPX-90, for simple pitch-shifting to create a "duo for solo instrument" effect, replacing that technology with a cleaner-sounding, more up-to-date and realistic transposition effect would probably be preferable. For example, would anyone really want to add artificial background hiss to a modern performance of a work such as Stockhausen's *Solo for Melody Instrument with Feedback* in a scenario where computer-based digital delay lines are used in place of the original analogue tape delay?! [3] Although analog noise was a unavoidable component of early electronic music – and a case could definitely be made for retaining it [5] – many composers at the time considered it an unwanted, though unavoidable, artifact and some took steps to minimize it. In the case of the original tape-delay version of *Solo*, Stockhausen recommended using a lowpass filter to mitigate the noise buildup on the tape [15], and was favorable to the cleaner sound of modern computer-based implementations [16].

Of course there will always be those fine-line cases where one would have to carefully consider whether the juddery-sounding effect of the original hardware unit might have been used for compositional purposes, or whether a more transparent and sonically cleaner version of the effect would be more musically effective or appropriate. Certainly, modern ensembles and orchestras, using instruments whose design was heavily modified (some would say greatly *improved* – or at least *streamlined*) throughout the 19th and 20th centuries, are able to perform older works with a fresh interpretation and modern "sound," while simultaneously remaining faithful to the composer's original intentions. Analogously, there is no reason why modern technology couldn't be used to replace the original technology of dated electroacoustic pieces, and similarly retain the composer's intentions in spite of yielding a slightly different sound quality, compared to the original. There are no set rules, however, and those involved in updating outdated electronic music for performance will certainly be confronted with having to make their own artistic decisions (just as performers do when confronted with interpreting the nuances in a traditionally notated score) based on their understanding of the composer's musical intent. Simon Emmerson rightly points out that there is no single "correct" way to re-perform electronic music and that it is often the piece which dictates the appropriateness of one re-interpretative choice over another [5].

3. ORAL TRADITION

Interestingly, there is also an important history of oral tradition in computer music – something which seems rather

[3] There would definitely be a case for answering "yes" to this rhetorical question – for example, the addition of artificial vinyl surface noise has become a desirable aesthetic in some digital-sourced musical genres. Certainly, digital audio workstation plugins have been created for both the purposes of noise removal/reduction as well as retro-sounding noise emulation/introduction.

anachronistic or paradoxical – and this is something that has not been written about to any great degree. On the technical front, the need for having an oral tradition often stems from a general lack of documentation where the operation of technology is concerned. The published scores for many pieces either do not indicate clearly indicate how to operate the hardware or software used (although some make a valiant attempt), or they often discuss an earlier, outdated version of the technology. The consequence of this is that performance instructions for many pieces are verbally conveyed from generation to generation of computer musician. [4]

One personal example of this was my preparation for a performance of Jonathan Harvey's *String Quartet No. 4 with live electronics* together with the Quatuor Diotima, where I was fortunate enough to have been able to spend several hours in a teleconference with Arshia Cont at IRCAM, who had performed the piece and was able to go through both the details of the Max/MSP patch and the score itself – *bar by bar* – to teach me how to perform it. Alas, it is not always possible to have this kind of luxury. Fortunately, many people have recently been working on analyzing and documenting "classic" pieces of electroacoustic and electro-instrumental repertoire, and are able to communicate with the composers, technicians and performers who created and initially performed the works. One such example would be the recent software emulation of Truax's *Riverrun*[17].

Nonetheless, on the performance front, a lack of a clearly defined and universal performance practice in electroacoustic music where the operation of technology is concerned certainly contributes to the continued need for an oral tradition. Canazza and Viodlin have suggested the creation and preservation of a multimedia archive for each piece to be used in lieu of actual oral communication, which can be distributed alongside the written performance materials, as a means of preserving the important details of each composition's specific performance practice from both the technical and musical perspectives[4]. This would definitely be a step forward, but in the past decade and a half, few have followed this advice.

4. CASE STUDIES

The following case studies are used to show some practical and useful examples of techniques that I have used for the re-design or re-implementation of dated technology. Some of them are basic sound editing work, but most are implementations created with Max/MSP. Some of the techniques shown are simple, others more complex. I have tried to make a selection of real-world examples that demonstrate commonly-used effects, instead of things that may be specific to the technology for the particular piece being discussed. These are not just common improvements that can be made to older patches, but can be taken into consideration when designing new projects as well.

4.1 Davidovsky's *Synchronisms No. 3*

Mario Davidovsky's *Synchronisms No. 3 for Cello and Electronic Sounds* (1964) is not as widely performed as *Synchronisms No. 1* for flute, but is an equally masterful early piece for instrument and tape that deserves a more prominent place in the repertoire. In re-performing this work, I ran into the problem that the tape part which was provided to us by the publisher for performance had been poorly and unprofessionally de-noised: the digital processing which had been effectuated on it not only caused the usual intermittent noise pumping, but also accidentally altered and garbled a few of the sounds in some places. Fortunately I was able to locate a cellist who had an older "straight dub" of the tape part, so I was able to create my own updated version. My alterations to the original were only a few: I decided to leave the original tape part mostly as-is – complete with the original background noise and all (which the listener quickly gets acclimated to) – however, I opted to create sharp fade-ins on cues that began with a fast attack [5], and more natural fade-outs of the background noise at the end of cues. The final tape cue starts with a long crescendo that is supposed to seem to have been transferred to the electronics from the sound of the onstage cello. A sudden, abrupt introduction of analog noise at the start of this cue rather ruins this intended effect (even though the cello is playing fortissimo), so this cue was de-noised and the de-noised version cross-faded carefully into the original version by the end of the crescendo in order to preserve the musical intentions of the cue. The cues were triggered with a simple Max/MSP patch that displays cue duration and playback location, so that in rehearsal, the cellist could easily start from the middle of a given cue, if desired.

4.2 Malec's *Attacca*

Whereas Davidovsky intentionally made use of multiple tape cues in many of the pieces in the *Synchronisms* series, there are a large number of pieces for instrument(s) and tape which originally used a continuously-running tape with built-in silences. These pieces often containing corresponding flexibility in the instrumental part consisting of material which can be omitted, repeated or improvised in case the performer plays the material faster or slower than the silence provided in the tape part. Ivo Malec's *Attacca – concerto pour percussion solo et partie électroacoustique* (1985-86) was given the same minimal clean-up treatment as the previous example. But more importantly, the tape part was edited into several individual cues, allowing the performer some freedom during long stretches of silence in the electronics. This is something that many older instrument and tape pieces can benefit from, as it gives greater interpretive freedom to the performer and allows the re-entry of the electronics to be placed at a

[4] This is also becoming increasingly true in the realm of acoustic compositions, where the performance of extended techniques and timbral effects often goes beyond what can be easily expressed with current standard musical notation.

[5] Actually, an exponential fade-in was used just *before* the electronic sounds begin, so the background noise does not seem to enter before the sound itself. In some cases, I resorted to de-noising the very beginning of the attack, and mixing or cross-fading this back into the tape part.

known point in the score in musical time – something which can often be dependent on physical parameters, such as the acoustics of the hall, or emotional parameters, such as the performer's "feeling" and reaction to the audience. Dividing the tape part into multiple cues when there are long silences is effective for musical reliability and can be used to great effect – not to mention ease for the instrumentalist in both rehearsal and performance – in pieces such as Davidovsky's *Synchronisms No. 6* and Harvey's *Tombeau de Messiaen*.

4.3 Musgrave's *Narcissus*

Thea Musgrave's *Narcissus* (1987), which exists in the composer's own versions for both for flute and clarinet with digital delay, has been a widely performed work in both original versions as well as a few arrangements for other instruments by individual performers. The piece, whose electronic component was originally designed for the rare and quickly obsolete Vesta Koza DIG 411 digital delay rack unit, has been the subject of many software re-creations, using PureData, Max/MSP and other software [18, 19, 20]. The delay effect is not particularly complex from a technical standpoint, though it is musically effective for the piece. In spite of the fact that there were initial concerns that the piece would not outlive the hardware used [21], the popularity of the piece combined with the fairly straightforward technical setup has kept it in the repertoire. It is therefore not necessary to describe the original hardware and the task of porting old technology to new. The Max/MSP implementation by David Brooke Wetzel [20] is an accurate re-implementation of the features (and shortcomings) of the Vesta Koza DIG 411 delay unit, has often served as the groundwork for other software re-implementations, including the concert patch described here. Our updated concert Max/MSP patch, however, includes some important modifications, improvements and techniques which merit discussion.

Some of the technical imperfections inherent in the original delay unit – largely due to its being amongst the first generation of digital hardware – which were also reflected in Wetzel's original MSP implementation are: audible clicks when the delay time is changed or when using the delay's "hold" feature, slight zipper-noise when rapidly changing modulation, volume and feedback parameters, and a limited delay length. Certainly some of these limitations in the original hardware unit informed compositional decisions, such as keeping a fixed delay length for large spans of the piece, as well as ensuring that large parameter changes occur during long rests. Nonetheless, among those deficiencies in the original delay unit (and basic Max patch), the most basic one to correct was to add smooth gain control to all parts of the signal chain using the *line∼* object, as shown in figure 1.

This is a basic improvement I recommend implementing in *all* older patches, many of which may have originally avoided using an extra signal object for concerns of exceeding the processing power of older computers. This update can be done by simply adding a message box describing a short ramp connected to a *line∼* object in-between the number box and the signal multiply. Even though Musgrave "composed around" this technical shortcoming of the original hardware delay unit, adding a smooth delay change gives the instrumentalist some "wiggle-room" with the timing of the pedal. Furthermore, it allows the loop/hold effect to produce more natural-sounding sustained tones.

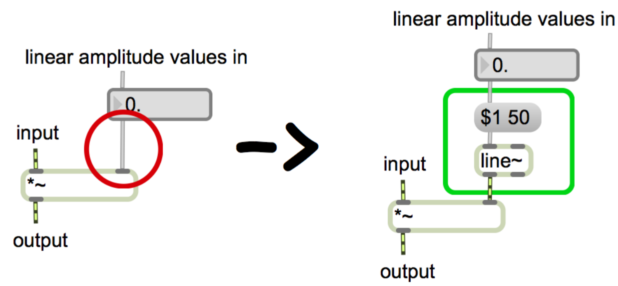

Figure 1. Smooth gain change.

Similarly, delay lines in older patches are often implemented in the simplest way possible (using a single pair of *tapin∼* and *tapout∼* objects in MSP). Changes to the delay time with this simple setup often causes audible and disconcerting discontinuity in the output. Updating and improving this in order to eliminate unwanted clicks is a little more complex than simply controlling the delay time with a *line∼* object, which would cause a quick glissando. The trick is to use a pair of delay lines in parallel and crossfade between them. [6] Only one delay line is used at any given moment, except for the brief instant when the output gains of the two delay lines are changed from off to on, and on to off, respectively. This technique, sometimes referred to as a "*smooth tapout∼*", is shown in figure 2. [7] Note that, in this implementation, the incoming delay time changes are speed limited to an amount of time slightly greater than the crossfade time.

Finally , the output gain of the entire delay effect has also been updated to take perceptual factors into consideration. When instrumentalists perform a crescendo or diminuendo, their timbre changes alongside the change in volume. A simple gain change does not sound very convincing in the context of this piece, so it was decided to implement a more complex volume control so the entrances and exits of the delayed instrumental sound appear to more closely match the change

[6] Note that this kind of crossfade is already built-in to the standard MSP *delay∼* object. You can activate it by sending the *ramp* message followed by a crossfade time in milliseconds. Unfortunately this feature has not yet been integrated into the *tapin∼/tapout∼* object pair, which facilitates the creation of delay lines with feedback, thereby requiring us, for the moment, to use some variation of the technique shown in figure 1.

[7] Throughout the years, several people have made abstractions to create such "clickless" delay changes. One such abstraction – *m4l.vdelay∼* – is included in the Max distribution. It employs a slightly different technique (granular overlap-add, instead of a simple crossfade) and may be better suited to situations where rapid, continual delay changes are required. The version shown here was primarily designed for efficiency in patches where infrequent changes in delay time are needed, and presupposes that the user will ensure that modulation depth values do not yield delay times less than the signal vector size being used.

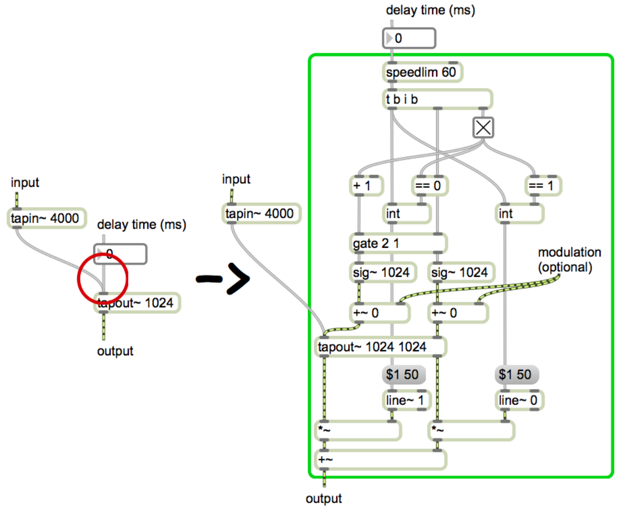

Figure 2. Smooth delay change using crossfade.

in timbral quality of instrumental crescendi and diminuendi. This modification to the piece amounts to a performance decision made for purely musical reasons. [8] Technically speaking, it was achieved by applying a high shelf filter to the attenuated output signal. The amplitude of the filter's shelf is controlled by the same linear amplitude value which controls the output gain, as shown in figure 3. Note that this additionally makes a more natural volume control than simply using linear amplitude – in terms of perceived volume, it is similar to squaring the linear amplitude value, a technique often used when converting MIDI velocity values to amplitude [22].

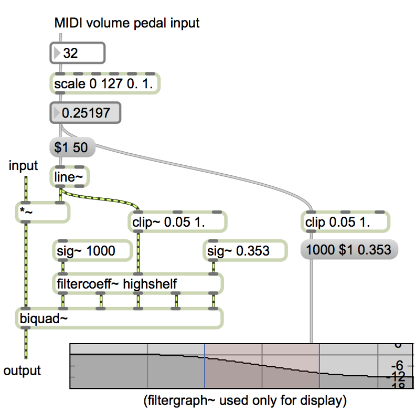

Figure 3. High shelf filter for musical volume change.

[8] A few other purely musical decisions were included as options to the updated patch, including the ability to set the delay time in each section of the piece to the performer's preferred tempo for that section, instead of leaving it fixed at 1024 ms, and to spatialize the electronic part.

4.4 Harvey's *Advaya*

The original version of Jonathan Harvey's *Advaya* (1994) for cello, keyboard and electronics, made use of an Akai S-1000 sampler (played by the keyboardist), a Yamaha SPX-100/1000 rack-mounted effect box (only for its harmonizer effect) and two CD players for soundfile playback. For the piece's premiere at IRCAM, the soundfile playback was implemented on the Ircam Signal Processing Workstation (ISPW), even though it had been designed with commercial CD players in mind for ease of performance outside IRCAM. By 2006, Frédéric Voisin and Denis Lorrain at IRCAM had rebuilt a Max/MSP Patch to replace the SPX and CD players. Although the Akai S-series samplers were popular and easily available throughout the 1990s, they are now becoming somewhat more difficult to obtain, but, more worryingly, it is becoming increasingly difficult to find a functioning SCSI CD drive which can be connected to the Akai in order to load the samples themselves. Therefore, when our center decided to program this piece for the second time in 2010, I realized it would be a good opportunity to implement the Sampler in the MSP patch, since this piece made use of only the most basic sample playback features of the Akai sampler.

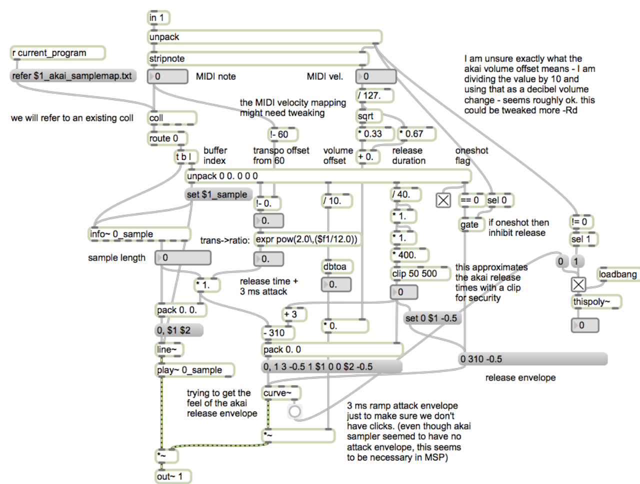

Figure 4. Imitation Akai sampler voice for use in *poly*∼.

The polyphonic patch to be used with the MSP *poly*∼ object, shown in figure 4, implements a basic sampler modeled on the Akai S-series samplers as used in Harvey's *Advaya*. It is not meant to be a comprehensive imitation of the Akai, but at least it might serve as a basis for emulating a similar bare-bones sampler in other pieces. The *coll* object in the patch contains a list of values taken directly from parameters in the Akai sample banks for *Advaya*:

- MIDI Note Number (*coll* index)
- MSP Buffer Number
- Number of Semitones to C3
- Akai Volume Offset
- Akai Release Time
- One-Shot Flag (play entire sample, ignore release)
- Sample Name (only for reference – not used)

Keeping the original values for these parameters as they appeared in the Akai (instead of putting them into a more Max-specific format) made it easier to cross-check them against the values displayed in the sampler in order to avoid errors while designing the patch. Furthermore, keeping them in their "original format" could allow the MSP sampler to be re-designed in the future to more closely imitate the Akai, should it be necessary. This does mean that the parameters need to be manipulated slightly in Max before being used – one example of this would be the seemingly complex conversion of the *Release Duration* parameter into millisecond values that reflect the actual release time used by the Akai (just to the right of the center in the figure).

In addition to being used to create a polyphonic sampler or synth, the *poly∼* object can also be useful for patches with multiple *sfplay∼* objects, that are used to play longer sound files. Very often Max patches (such as the patch for *Advaya*) contain triggers for lengthy sound files, and putting their playback inside a polyphonic patch can help not only to streamline the patch but also to serve to smoothly overlap or fade-out sound files should the performer play more quickly through the series of soundfile cues than expected. A modified version of the polyphonic soundfile playback patch used for *Advaya* is shown in figure 5.

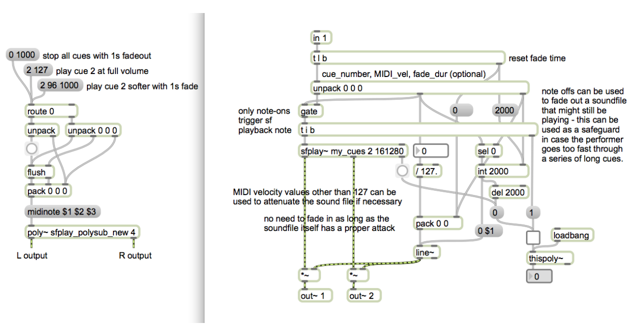

Figure 5. Using *sfplay∼* inside *poly∼*.

Naturally, this is just one possible implementation of such a patch; a simpler – or more complex – version could be used, depending on the context of the patch where it is needed. Notice that the note number sent to the *poly∼* is actually a cue number defined in an *sflist∼* object. Notice also the the *sfplay∼* object has been given a very large buffer size (161280). This is a safety measure to make sure the object does not stop playback if the computer's processor is overloaded (larger buffers mean that the processor has to read and convert chunks of sample values from the disk to memory less frequently). The *sflist∼* object should also be given the same buffer length argument! [9]

Similar updates (for both hardware sampler emulation and streamlining sound file playback) were made to the updated Max/MSP performance patches for Philippe Leroux' *M* (1994) and Magnus Lindberg's *Related Rocks* (1997).

4.5 Reich's *New York Counterpoint*

Many of the concerts of electro-instrumental music that our studios have been involved with have made use of a multi-channel speaker setup. In an era where many people have multi-channel home theaters, today's computer music concerts seem much richer if they go beyond having just a couple of speakers on stage. Steve Reich's *New York Counterpoint* (1985) is not always considered as an electroacoustic piece by the computer music community, however it does make use of recording technology at its core (after all, Reich was one of the early members of the San Francisco Tape Music Center). Most performances of this piece use a stereo tape recorded by the performer (or even the pre-recorded tape part available through the publisher). With 11 contrapuntal lines of music (10 pre-recorded parts and the live soloist), it is only natural that this piece has on occasion been performed by diffusing the ten soundfiles over a multi-speaker setup [23]. Working together with clarinetist Pete Furniss and composer Alex Harker in the UK, we prepared an updated version for the Minimal festival in Glasgow (screenshot shown in figure 6). This was based on an earlier multichannel version which we made in 2006. To spatialize the pre-recorded clarinet tracks, I used Ville Pulkki's *vbap* object suite for Max/MSP, a public-domain collection of external objects which implements vector based amplitude panning [24]. This is an excellent, easy, and efficient system to use for complex multichannel panning, and has been implemented within multiple computer music software platforms, including CSound, Max/MSP and Pd. Note the interface we made using the standard Max *lcd* object – this was useful to help us work out the location of the 10 sound sources in the 8-channel circular speaker array while creating and editing the panning cues.

4.6 Saariaho's *Noa Noa*

Kaija Saariaho's *Noa Noa* (1992) is another example of a piece which is often played and which exists in multiple versions for various technologies (fixed media, ISPW, Max, Pd). I am mentioning it here not in the context of updating software, but rather the possibility of updating its sound files. Many performers I have worked with have expressed an interest in re-recording instrumental sounds used within live electronic pieces in order to be able to include their own personal "sound" in the pre-recorded component of the pieces they are performing [25]. *Noa Noa* is an ideal candidate for this, since many of the sounds in this 20 year old piece are beginning to sound somewhat dated. As such, I have transcribed the soundfile cues which are composed of unprocessed flute

[9] If *sflist∼* and *sfplay∼* have different buffer sizes, it could cause some versions of Max/MSP to crash!

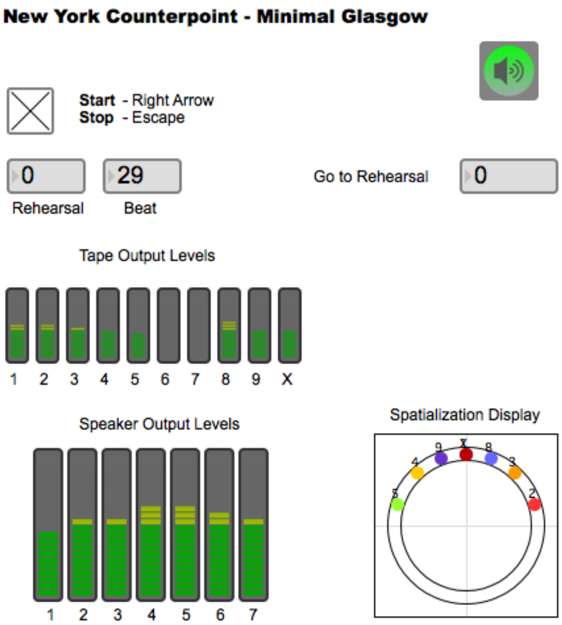

Figure 6. Multichannel panning for *New York Counterpoint*.

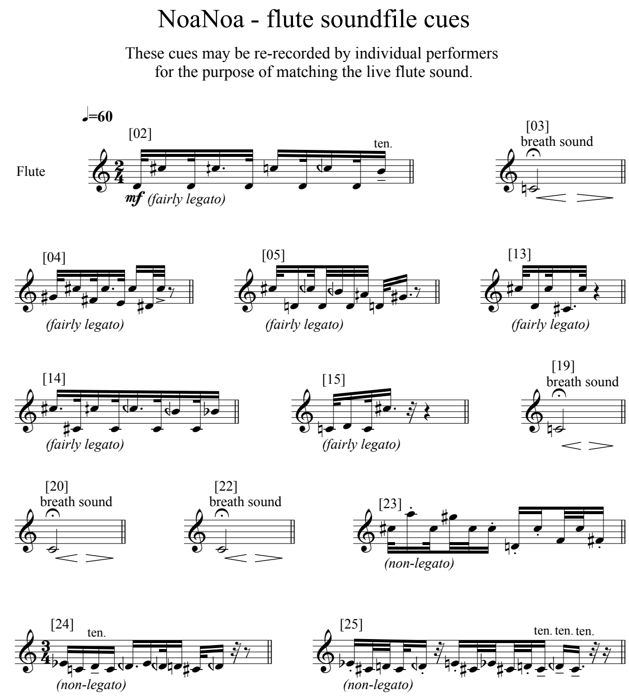

Figure 7. Transcribed cues for *Noa Noa*.

sounds, so performers I work with in the future will be able to personalize their performance. The first page of cues (those cues which can be recorded and replaced without additional sound file editing) is shown in figure 7.

While discussing this possibility via personal e-mail correspondence with Jean-Baptiste Barrière, who maintains Saariaho's Max/MSP patches, he admitted that some flautists (including Camilla Hoitenga, who premiered the work and to whom the piece is dedicated) have apparently expressed an interest in re-recording these cues. This is something the composer has been keen to do, but the right occasion (not to mention the time) to make the sound file update has never materialized.

5. CONCLUSION

In updating the technology for the many pieces I have co-performed with instrumentalists, one important circumstance has come to light: much of the electroacoustic and live electronic repertoire – even that which my be well-known from recordings – is seldom performed live outside its area of origin (usually Europe or the US). Part of the lack of a wider international dissemination of the electroacoustic repertoire among eager performing musicians certainly has a lot to do with its technological component: with its availability and condition of maintenance, with the apparent lack of ease in assembling the necessary technology or a concert, and with its seeming inflexibility. Keeping this repertoire alive by creating updated, vivid performing versions of mixed electro-instrumental pieces allows them to be performed more widely (at least until that technology also eventually becomes outdated). This wider dissemination has unfortunately been hampered to some extent by the difficulty in obtaining certain pieces of older electronic music hardware: a key first step in making an initial analysis of the technology, before designing an updated software replacement. The ideas and techniques shown here will hopefully aid others not only in updating older technology, but also in designing new technology for live musical works. Even in the face of technological obsolescence, interactive electroacoustic and computer music continues to develop forward alongside the rapid development in technology itself.

Acknowledgments

I would like to thank the many performers, sound engineers, computer musicians and composers I have worked with over the years. The list is too long to enumerate, but you all know who you are!

6. REFERENCES

[1] R. Dudas, "Expanding the Performance Possibilities of Real-Time Computer Music Repertoire through Re-Working Dated Technology," *Journal of the Japanese Society for Sonic Arts*, vol. 6, no. 1, pp. 6–11, 2014.

[2] S. Berweck, *It Worked Yesterday: On (Re-) Performing Electroacoustic Music*. Doctoral Dissertation, University of Huddersfield, UK, 2012.

[3] J. Bullock and L. Coccoli, "Modernising Musical Works Involving Yamaha DX-based synthesis: a case study," *Organised Sound*, vol. 11, no. 3, pp. 221–227, 2006.

[4] S. Canazza and A. Vidolin, "Introduction: Preserving Electroacoustic Music," *Journal of New Music Research*, vol. 30, no. 4, pp. 289–293, 2001.

[5] S. Emmerson, "In what form can 'live electronic music' live on?" *Organised Sound*, vol. 11, no. 3, pp. 209–219, 2006.

[6] R. Polfreman, D. Sheppard, and I. Dearden, "Time to Re-Wire? Problems and strategies for the maintenance of live electronics," *Organised Sound*, vol. 11, no. 3, pp. 229–242, 2006.

[7] D. B. Wetzel, "A Model for the Conservation of Interactive Electroacoustic Repertoire: Analysis, Reconstruction, and Performance in the Face of Technological Obsolescence," *Organised Sound*, vol. 11, no. 3, pp. 273–284, 2006.

[8] M. Puckette, "New Public Domain Realizations of Standard Pieces for Instruments and Live Electronics," in *Proceedings of the International Computer Music Conference*, Havana, 2001, pp. 377–380.

[9] J. Bullock and H. Frisk, "Libintegra: a System for Software-Independent Multimedia Module Description and Storage," in *Proceedings of the International Computer Music Conference*, Copenhagen, 2001, pp. 43–46.

[10] D. B. Wetzel, "The Interactive Event Manager (IEM): A Performer's Approach to Interactive System Design," in *Proceedings of the International Computer Music Conference*, Montreal, 2009, pp. 517–520.

[11] T. Place and T. Lossius, "Jamoma: a Modular Standard for Structuring Patches in Max," in *Proceedings of the International Computer Music Conference*, New Orleans, 2006, pp. 143–146.

[12] M. Akkerman, "Historical Performance Practice in Computer Music: Necessity or Outrage?" in *Proceedings of the Korean Electro-Acoustic Music Societys 2014 Annual Conference*, Seoul, 2014, pp. 93–96.

[13] G. G. Karman, *Closing the Gap between Sound and Score in the Performance of Electroacoustic Music*. Leuven University Press, 2013.

[14] B. Spoerri, "Brouillard – déebrouillé (vom Improvisieren mit der Maschine)," in *Audio-Design Symposium*, Elektronisches Studio Basel, Switzerland, 1997.

[15] K. Stockhausen, *SOLO für Melodieinstrument mit Rückkopplung*. Universal Edition A.G., Vienna, Austria, 1969.

[16] B. Sluchin, "A Computer-Assisted Version of Stockhausen's 'Solo for a Melody Instrument with Feedback'," *Computer Music Journal*, vol. 24, no. 2, pp. 39–46, 2000.

[17] M. Clarke, F. Dufeu, and P. Manning, "From Technological Investigation and Software Emulation to Music Analysis: An integrated approach to Barry Truax's 'Riverrun'," in *Proceedings of the Joint International Computer Music Conference and Sound and Music Computing Conference*, Athens, 2014, pp. 201–208.

[18] N. Bliton, *Thea Musgrave's Narcissus*. http://nateblitonmusic.com/?p=162, accessed 2014.7.24, 2011.

[19] J. Rudi and J. Bullock, "The Integra Project," in *Proceedings of the Electroacoustic Music Studies Conference*, New York, 2011.

[20] D. B. Wetzel, "Analysis and Reconstruction of Interactive Electroacoustic Works for Obsolete Technology: Thea Musgrave's Narcissus," in *Proceedings of the International Computer Music Conference*, Miami, 2004.

[21] R. Dick, "[Untitled Music Review]," *Notes: the Quarterly Journal of the Music Library Association, Second Series*, vol. 46, no. 1, pp. 236–237, 1989.

[22] R. B. Dannenberg, "The Interpretation of MIDI Velocity," in *Proceedings of the International Computer Music Conference*, New Orleans, 2006, pp. 193–196.

[23] A. Pras, F.-X. Féron, and K. Demers, "The Pre-Production Process of New York Counterpoint for Clarinet and Tape Written by Steve Reich," in *Proceedings of the International Computer Music Conference*, Montreal, 2009, pp. 493–500.

[24] V. Pulkki, "Compensating displacement of amplitude-panned virtual sources," in *Audio Engineering Society 22th Int. Conf. on Virtual, Synthetic and Entertainment Audio*, Espoo, 2002, pp. 186–195.

[25] P. Furniss and R. Dudas, "Transcription, Adaptation and Maintenance in Live Electronic Performance with Acoustic Instruments," in *Proceedings of the Joint International Computer Music Conference and Sound and Music Computing Conference*, Athens, 2014, pp. 456–462.