Book reviews

Richard Boulanger and Victor Lazzarini, eds, *The Audio Programming Book*. Cambridge, MA: The MIT Press, 2010. ISBN 978-0-262-01446-5

Scott Wilson, David Cottle and Nick Collins, eds, *The SuperCollider Book*. Cambridge, MA: The MIT Press, 2011. ISBN 978-0-262-23269-2

In his Preface, Boulanger describes a '40-year journey to *The Audio Programming Book*', his pilgrimage populated by a variety of founding fathers and fellow travellers including, Max Mathews, Barry Vercoe, F. Richard Moore, Gareth Loy and Bruce Pennycook. Much of what he describes resonates well with my own experience in seeking proficiency with the art and science of musical sound synthesis and signal processing. Boulanger reports synthesising the tune *Happy Birthday* with MUSIC 10 taking 'days back then'. By the time I was at the University of Texas studying with Russell Pinkston in the early 1990s, direct digital synthesis on an IBM RT workstation was down to significantly under an hour to render a Bach chorale. (Not the most musically useful application of a computer, but many lessons learned on my part; perhaps the most important being that Bach wasn't really writing for computer-generated sine-waves.) In contrast, my granular synthesis efforts (Anderson 1993) at the time took approximately three days to render a single texture of about 45 seconds. Boulanger's late-night sessions in the computer music studio with Pennycook ending 'with breakfast at a local diner' sound very familiar, too. My own nights in Studio IV tended to finish at Players, a burger joint open to the early hours near the UT campus. Matt Ingalls, mentioned by Boulanger in his Acknowledgements, regularly stood in as my own Pennycook. We might call these the days of 'classic computer music'. Code was written in C, or, more often, a synthesis-scripting language like Csound. The composer sat in front of a terminal or computer screen – and sound was rendered offline to be auditioned later. I can remember, at the time those of us working in the 'computer music' studio felt we were doing something much more profound, much more avant-garde, than those in the analogue studio. Rather than merely making sounds by 'playing' the instruments found in the analogue studio, we were 'inventing' our own, new instruments – to play a new music. And, we acted with the notion there were very few working with these obscure means. Perhaps a naïve view, but it gave enough of a sense of purpose to mitigate the time lags between when a sound (often needing some sort of improvement) could actually be auditioned. And of course, these time lags gave plenty of time for 'profound' discussions with fellow Studio IV users between render attempts. We're in a different world now. Among other things, if a composer wishes to work with granular synthesis, off-the-shelf software is now available for the task – there's not necessarily a particular need to write code. Now the artist working with the computer and digital synthesis isn't required to also be the instrument designer and builder, or to understand a chosen technique enough to realise it in code. We might also suggest that the computer studio, particularly in terms of user experience, has become much like the analogue studio: things are plugged together, knobs are twiddled, sounds result. And, we might say the term 'computer studio' is actually redundant, as most music production now centres around PCs running digital audio workstation software. Ultimately, with many artists working with synthesis software running on laptops, the word 'studio' itself may be passé.

MIT Press's *The Audio Programming Book* and *The SuperCollider Book* both come in the wake of Boulanger's successful *The Csound Book* (Boulanger 2000), which has become the de facto reference for anyone interested in working with the Csound programming language. While Matthews has endorsed this text as 'the most important book written in the last decade',[1] I have found *The Csound Book* to be both especially valuable and extremely frustrating. Both responses come from the same aspect: the text's 32 print and 45 electronic (on the two accompanying CD-ROMs) chapters. Some of these are very well written, others less so, but with the task of sifting through it all left to the reader – something I've always considered the editor's role. Boulanger and Lazzarini's new book follows a similar model. *The Audio Programming Book* is presented as 15 chapters

---

[1] *The Csound Book* – Comments. http://mitpress2.mit.edu/e-books/csound/fpage/pub/csbook/contents/comments.html (accessed 7 September 2012).

in the printed text, 41 in electronic form (14 as continuations of those in print), 8 printed appendices, and with significantly more supporting code examples, software applications and other extras found on the accompanying 4.67GB DVD. In his Foreword, Max Mathews suggests, 'This is not just a book; it is an encyclopaedia' and 'It is the essential library' (xi). One can take these comments as either a rousing endorsement, or a suggestion of how to use the text. The authors of a previous review (Trevino and Allen 2012: 85) have indicated, 'The book eludes any obvious manner of use', and also state 'the organization of the massive amount of information presented here has clearly challenged the editors'. I'd like to echo these sentiments.

When I reflect on how I know what I know about programming for audio and musical applications, a few books jump to mind: Charles Dodge and Thomas Jerse's *Computer Music* (Dodge and Jerse 1985), F. Richard Moore's *Elements of Computer Music* (Moore 1990) and Brian Kernighan and Dennis Ritchie's *The C Programming Language* (Kernighan and Ritchie 1988). These are my urtexts; all were good places to start. But admittedly there is much missing if one wishes to become proficient with writing code for audio. *The Audio Programming Book* does hold out the promise of becoming either the tutorial or reference foundation text on the topic. And here's where the frustration for me begins: I'm not convinced encyclopaedias make good tutorials or vice versa. The contents of *The Audio Programming Book* are laid out to intimate a progressive reading of the material, which is strongly implied by the amusing C language conceit of numbering the chapters from 0. (The C programming language indexes, or counts, from zero.) While the titling of the initial sections offer an eager promise to this end ('C Programming Basics', 'Audio Programming Basics' and 'Audio Programming Essentials'), a sequential reading of these first seven chapters by five different authors (Lazzarini, Richard Dobson, John Ffitch, Boulanger, and Gabriel Maldonado) does not feel rewarding. Each of the individual chapters is either good or suitable and on topic, but they don't particularly feel contiguous as one might expect. And while having the same idea repeated or discussed by different voices can be valuable, too much of this can lead to a feeling of treading water in the material. I wouldn't say this is a welcome outcome for a tutorial. I was left wondering about whether, within their limits, what I like about my own urtexts has to do with a perceived single voice of the author. Dodge and Jerse know they have told the reader something, so do not need to keep repeating themselves. With the editors Lazzarini and Boulanger contributing nearly half of the initial sections, one might expect tighter integration.

Lazzarini, Dobson, Ffitch and Boulanger make the largest contributions to the text with 11, 10, 8 and 6

chapters and appendices, respectively. The more time I spend with *The Audio Programming Book*, the more I'm convinced I would have preferred the material to be presented in a slightly different way. One option could be some sort of multi-volume text, or something like a series of monographs by the four principal contributors above. In teaching a course on audio programming, one could easily use just the chapters authored solely by Dobson as the primary text. Admittedly, presenting all 56 chapters in print could make for an expensive proposition. That said, Loy's two-volume *Musimathics* (Loy 2006; Loy 2007) from the same publisher has been well received, so there is a place for multi-volume works. Another option could be an ebook-only text. In relegating 41 of the chapters to the DVD, these are not presented with the same status as those in print. And with the first 11 of the DVD chapters being 'Continuations of the Printed Text Chapters', the reader gets the sense that the physical text is just too small to contain everything the authors would like to relate. (Or, with another view, the editors shrug and text just goes on the DVD.) As an electronic-only text, all 56 chapters would immediately gain equal status in their presentation and, I would presume, use.

On intended uses, the final five electronic chapters on the DVD, with the title 'An Audio Programming Course for Electronic Musicians' are clearly intended as a tutorial. As Boulanger is listed as an author for all of these, I think it is fair to conclude these are used in Boulanger's teaching at the Berklee College of Music. With the copy I have received for review, the first four chapters are represented only as a short abstract, followed by 'Download the latest version of the course materials and source code @ http://csounds.com/apb', where Boulanger confirms this use in teaching. Unfortunately at the time of this writing, all links from this page return 'Page not found' notifications. Boulanger does 'recommended that you check back several times a year for updates and revisions'. The broken links I am seeing now may be corrected in the future, but I am surprised that the DVD doesn't actually contain the chapters listed as contents – particularly ones authored by one of the editors. Again, I was left thinking that maybe some of the material could have been edited or presented in a different way. Perhaps a clear set of online tutorials (by multiple authors) would have been appropriate?

Of all the extras supplied under the heading 'DVD Special Content', perhaps the most curious to be found are the 'Classic Computer Music Software and Source Code', including cleaned up, compilable versions of Mathews' MUSIC V and Moore's cmusic.[2]

---

[2]MUSIC-N, *Wikipedia, the Free Encyclopedia*. http://en.wikipedia.org/wiki/MUSIC-N (accessed 1 August 2012; Cmusic, *Wikipedia, the Free Encyclopedia*. http://en.wikipedia.org/wiki/CMusic (accessed 5 September 2012).

While certainly useful for the historian, maybe less so for the contemporary practitioner, I do find the inclusion of these two extras to be strangely pleasing – I expect because it gives a strong sense of connection to earlier days of computer music, and may be something akin to reviewing a famous score in the composer's own pen. As documentation for MUSIC V the DVD contains Chapters 2 and 3 from Mathew's *The Technology of Computer Music* (Mathews 1969). Oddly, this is not directly acknowledged anywhere in the accompanying documentation. The astute reader will find it listed in the 'DVD References & Links' PDF – but I will say I am disappointed the inclusion of these two chapters from Mathews' out-of-print text are not clearly documented. And, as for supplying content for the curious, why not throw in the first chapter from this important text, too?

Curiously, *The Audio Programming Book* appears to present its readers with two separate glossaries. This first in the printed text, titled 'Appendix G: Glossary', lists the four core contributors (Ffitch, Dobson, Lazzarini and Boulanger) as authors, while the second on the DVD, titled 'DVD Glossary: An Annotated Audio Programming Glossary' only lists Ffitch and Dobson. On closer examination, both of these texts appear to be the same, but for some formatting issues – which does lead me to wonder, why the repetition? (And, who are the authors? Most likely Ffitch and Dobson?) Is this an editorial oversight? As with the chapter numbering, a few programmer in-jokes do appear in the glossary: 'Endless loop: See Loop, endless.' I am slightly surprised, but pleased to see a definition for 'Ambisonics' here, and a suitable one at that, as this is not a topic appearing in much detail elsewhere. (Dobson discusses his custom extension to Microsoft's WAVEFORMA-TEXTENSIBLE format to support Ambisonic soundfiles in Chapter 2.) Under the definition of 'Music V', a number of more modern computer music languages are listed; and whether a oversight or not, SuperCollider does not appear here. A peculiar example of wilful oversight?

It is worth noting that there is a fairly strong bias toward Csound with the examples and project applications found in *The Audio Programming Book*, which is not necessarily surprising considering this effort follows on from Boulanger's very successful earlier text. Five chapters have Csound in the title, and numerous others either work directly with Csound or build interfaces to do so. In reviewing the contributing authors, I identify many of these as part of what I call the 'Csound crowd'. And, with a closer look, six authors are found to be core developers of the Csound 5 architecture (Kozar 2006). In viewing the bias towards Csound, I did find Eric Lyon's chapter, 'A Modular Synthesizer Simulation Program' to be especially interesting and puzzling: interesting in

that the programming task in C is creative and likely to have musically interesting results, puzzling in that the resulting C programming (an interface for Csound) does something that is fairly trivial to do in Super-Collider. So, while Lyon's exercise here may be a useful way to develop C programming skills, I would not describe it as a good choice for an artist – if Lyon's end musical outcome is the primary goal. Starting with SuperCollider for Lyon's musical problem would have led to quicker results. Which brings me to… I have seen SuperCollider described as one of the 'so-called "modern synthesis systems"' in the documentation for the Csound extension TclCsound, a description which strongly implies Csound isn't 'modern'.[3] If *The Audio Programming Book* proposes to be an 'encyclopedic' text, missing at least a mention of SuperCollider, a 'modern system', does seem to be something of an omission. The last three chapters of *The SuperCollider Book*, with their focus on developer topics and the C++ infrastructure of SuperCollider, could have easily found a home in Boulanger and Lazzarini's text – and would have been a welcome, 'modern' inclusion.

In the first sentence of his open call for abstracts Nick Collins wrote, 'We're going to make a formal proposal to MIT Press for a "SuperCollider Book" analogous to the "Csound Book"'. (Collins 2007) Without repeating my earlier hesitations regarding *The Csound Book*, I am of the view that the editors of *The SuperCollider Book* have exceeded their model. All twenty-seven chapters of Scott Wilson, David Cottle and Collins' new text are included as chapters in print – none are excluded to a nether region of CD-ROM or DVD off-siting. The editors have made a real attempt to tie the various chapters by different authors together; throughout, topics introduced or briefly reviewed in one chapter are followed up by a reference to another where the issue is further developed. They make their approach clear in the Introduction, stating their goal 'to avoid duplication of materials except where absolutely crucial for clarity' (xiv). As one might imagine, this gives a real sense of deliberate presentation, unlike my experience of repeatedly treading water through the same material in *The Audio Programming Book*. Rather than including a CD-ROM or DVD, the editors of *The SuperCollider Book* have chosen to make available additional materials via a supporting web page (http://supercolliderbook.net). And, in the age of free and open-source software, I find this an enlightened, if not only practical choice. The page even provides an errata link, a useful addition to keep the fixed, printed text, 'live'. In the introduction the editors also makes clear they 'couldn't possibly cover everything' and make an effort to point the reader to various external

---

[3] *A Scripting Environment*. www.csounds.com/manual/html/tclscripting.html (accessed 3 August 2012).

web resources, listing the main SuperCollider page (McCartney et al. 1996) as a gateway.

On their project, the editors state, 'We're delighted to present a collection of tutorials, essays, and projects that highlight one of the most exciting and powerful audio environments' (xiii). Wilson, Cottle and Collins are all authors of various tutorials for SuperCollider.[4] (Wilson's tutorial is bundled with the standard distribution, and is usually where most users start.) One should not be surprised, then, that the first eight chapters of this welcome text are categorised as 'Tutorials' and 'Advanced Tutorials'. Numerous other chapters also have a tutorial slant, but on more specialist topics. Worth a mention is Dan Stowell's well-written introduction, 'Writing Unit Generator Plug-ins', which guides the reader through coding up synthesis and signal processing extensions to SuperCollider in C++. This is one of the three chapters that could have worked well as part of Boulanger and Lazzarini's text – I can see using it in conjunction with lessons learned from *The Audio Programming Book*. Other chapters may also be regarded as tutorial, but these tend to be about approaching specific problems or compositional tasks with SuperCollider. Examples of these are Brian Willkie and Joshua Parmenter's chapter on non-real-time synthesis and Alberto de Campo's chapter, 'Microsound'. Also included is material under the heading 'Projects and Perspectives'. These are the least tutorial, but are intended to give an insight on real-world uses of SuperCollider – towards very specific ends. Stefan Kersten, Vincenzo Lombardo, Fabrizio Nunnari and Andrea Valle's 'A Binaural Simulation of Varèse's *Poème Électronique*' walks through the interesting Lombardo-directed *Virtual Electronic Poem* project (www.edu.vrmmp.it/vep) (Dobson et al. 2005). As a reader, I'll echo my own delight with the editors' efforts. While *The SuperCollider Book* isn't comprehensive (can we imagine the ox-stunning result if it were?!), I do find great value for both the newbie and the old hand. In any case, that's the goal of SuperCollider's online help system – which is constantly under development as the system improves and refines.

Before ending, I would also like to mention the Foreword to the text by James McCartney (SuperCollider's original author). In the early 1990s McCartney was employed in the Astronomy Department at the University of Texas, which overlaps my own time in Austin. Though, McCartney was in Astronomy rather than Music, I remember him mostly as an interesting lurker. A number of his engaging efforts were programmed on EARS (Electro-Acoustic Recital Series) concerts in the

McCullough Theatre – along with a few of my own. I can vividly recall McCartney's theatrical performance using the new Nintendo Power Glove (Anon 2012) as a musical interface. I was not necessarily convinced of the musical results, but it was something to see. In his Foreword McCartney asks, 'Why use a computer programming language for composing music?' I find it pleasing to know I was in Austin when he began to ask this question. He then goes on to describe the the history and origins of Super-Collider, beginning with a program written in 1990 named Synth-O-Matic. Of its functions, Synth-O-Matic included a variant of granular synthesis, and thinking back now, being more cozy with McCartney could have saved some time with my own granular synth efforts. We have since seen three versions of SuperCollider – and a huge growth in user-base. With SuperCollider3, the project became open-source and now includes a mechanism for third-party extensions. Both these factors suggest that Super-Collider will continue to be relevant into the future, and I expect *The SuperCollider Book* to have a lasting supporting role.

Joseph Anderson
josephlloydanderson@gmail.com

REFERENCES

Anderson, J. 1993. *in mosaic* (1992). On *Music from SEAMUS, vol. 3*. Los Angeles, CA: SEAMUS, EAM–9402 CD.

Anon, 2012. Power Glove, *Wikipedia, the Free Encyclopedia*. http://en.wikipedia.org/w/index.php?title=Power_Glove&oldid=508937176 (accessed 2 October 2012).

Boulanger, R. ed. 2000. *The Csound Book: Perspectives in Software Synthesis, Sound Design, Signal Processing, and Programming*. Cambridge, MA: The MIT Press.

Collins, N. 2007. [*sc-users*] *Open Call for Submissions for The SuperCollider Book*. https://www.listarc.bham.ac.uk/lists/sc-users-2007/msg01076.html.

Dobson, R. et al., 2005. Varèse's Poème Électronique Regained: Evidence from the VEP Project. *Proceedings of the 2005 International Computer Music Conference*. Barcelona/East Lansing: University of Michigan. http://hdl.handle.net/2027/spo.bbp2372.2005.164.

Dodge, C. and Jerse, T.A. 1985. *Computer Music: Synthesis, Composition, and Performance*. New York: Schirmer Books.

Kernighan, B.W. and Ritchie, D.M. 1988. *The C Programming Language*, 2nd edn. Englewood Cliffs, NJ: Prentice Hall.

Kozar, A. ed. 2006. Meet the Csound 5 Developers. *Csound Journal* 1(3), www.csounds.com/journal/2006spring/meetTheCsound5Developers.html (accessed 1 August 2012).

Loy, G. 2006. *Musimathics: The Mathematical Foundations of Music*. Vol. 1. Cambridge, MA: The MIT Press.

---

[4]*SuperCollider » Learning*. http://supercollider.sourceforge.net/learning (accessed 5 September 2012).

Loy, G. 2007. *Musimathics: The Mathematical Foundations of Music*. Vol. 2. Cambridge, MA: The MIT Press.

Mathews, M.V. 1969. *The Technology of Computer Music*. Cambridge, MA: The MIT Press.

McCartney, J. et al. 1996. SuperCollider. *SuperCollider » About*. http://supercollider.sourceforge.net (accessed 15 December 2010).

Moore, F.R. 1990. *Elements of Computer Music*. Englewood Cliffs, NJ: Prentice Hall.

Trevino, J. and Allen, D. 2012. Review of *The Audio Programming Book*, Edited by Richard Boulanger and Victor Lazzarini, *Computer Music Journal*, 36(2): 85–9.

Florence Feiereisen and Alexandra Merley Hill, *Germany in the Loud Twentieth Century: An Introduction*. New York: Oxford University Press, 2012. ISBN 978-0-19-975938-5

How is a nation's identity built? What are the characteristics of a particular country? What reasons do most people have for choosing their holiday destination? Usually the answers to such questions – somehow related to the abstract field of cultural representations – provoke associations with the field of the visual. However – apart from national anthems – what does sound have to do with cultural representations? From the perspective of cultural studies the field of sound is still under-represented as a research topic and an indicator of social phenomena. This is one of the points of departure of the essay collection *Germany in the Loud Twentieth Century*, edited by Florence Feiereisen and Alexandra Merley Hill. The editors view the discovery of sound as a serious research topic for cultural studies and thus offer a new kind of social approach to what sound studies calls sound ecology.

The book is introduced as a kind of case study of German sound, giving the impression or taking it as a presumption that there is something special which can be called 'German Sound'. Through the use of an anecdote – the well-known incident in Germany of Sarah Connor forgetting the text of the national anthem – it is explained that there are extreme differences between German and US-American sound culture. Identifying singing as an important function of the expression of national identity, in their introduction the editors make the so-called 'Hymnen-Streit' a much bigger topic than it ever was in the German media. The question of the national anthem can be considered as being another part of the field of sound as cultural representation. Referring later to R. Murray Schafer and his idea that 'the general acoustic environment of a society' can 'be read as an indicator of the social conditions which produce it and may tell us much about the trending and evolution of the society' (2), the editors attempt to use the perspective of cultural studies to examine an important sphere of the social world – the consciously as well as the unconsciously existing sphere of the aural – looking for approaches which are not at first glance psychoacoustic or musicological.

Under the primary perspective of German and cultural studies, some already established theories in the discipline of sound studies appear in new interpretations – in particular, McLuhan's idea of the social role of an acoustic space is given new importance. Thus, the editors propose that their work should be seen primarily as a plea for the establishment of transdisciplinary soundscape studies.

Although the phenomena presented in the book are quite diverse and not homogeneous, there is an attempt here to establish a kind of chronology to unify two articles dealing with different but somehow relative phenomena under the title of one section. Five such sections each open up a field in which sound appears as an interesting research object within the sphere of cultural studies. These fields include sound art, sound and politics, the role of sound in the divided Germany, the sound character of certain spaces, and the new forms of sound and noise appearance in the early twentieth century and social reactions to them.

The first section of the book takes it as a presumption that there was a new awakening of sound and noise at the beginning of the twentieth century due to factors such as industrialisation and a new kind of urbanisation. From there, also presuming that every soundscape consists of distinct sound events which had to be considered in a larger context, including the conditions of their reception, the question arises of how this world of 'increasing noise' and loudness was viewed. The phenomena of anti-noise movements and inventions are discussed, particularly that of Theodor Lessing's 'Antilärmverein' ('anti-noise association', under-represented until recently in serious research) in an article by John Goodyear. Goodyear also develops a summary of the long history of the earplug, illuminating in a certain way the philosophical concept of the antipodes silence and noise. The important role of the radio as a sound source determining the lives of Germans from the 1920s onwards is only touched on by Robert Ryder's analysis of Günter Eich's radio play *Dreams* from 1951, attributing a specific new role to the piece. The author employs the theories of Sigmund Freud and Walter Benjamin in a surprising manner in order to develop his concept of unconscious listening.

That a sound can help to define a certain space may appear at first glance to be a commonplace, but two very different perspectives on sound as characteristic of a certain space bring important aspects to the discussion. The interpenetration of sound and space is developed here as an interesting