Tracing Compositional Process:
Software synthesis code as documentary evidence

Christopher Burns

Center for Computer Research in Music and Acoustics (CCRMA), Stanford University
email: cburns@ccrma.stanford.edu

Abstract

Composers working with text-based software synthesis languages (such as the "Music N" languages) tend to produce rich documentation of their compositional process as a byproduct of their work. This evidence, in the form of custom programs and synthesis specifications, documents the technical means of a work's realization; insights into the composer's creative process may be extended if multiple drafts, versions and revisions are available, or when other contextualizing documents (for example, paper sketches and source soundfiles) are also preserved. In many instances, synthesis language scores will be valuable as historical documents, with implications for the reception and criticism of the associated works.

1 Introduction

Joseph Kerman, surveying the field of sketch studies, suggests that it encompasses "all kinds of research... on a broad range of documents: sketches, drafts, working autographs, reject sheets, *collettes* (or stickovers) – everything, in fact, that fulfills two conditions: (1) it has survived, and (2) it was in the composer's mind superseded. One would not in principle exclude the floor sweepings of electronic music studios" (Kerman 1982). While optimistic about the historical and critical value of sketches, Kerman confines his examples to documents on paper, produced by composers of traditionally notated instrumental and vocal music. What of those floor sweepings?

Composers of computer music – and any composers who use computers in their work – may create a wide variety of materials during the realization of a new piece of music. Perhaps they will write custom software, either from scratch or using a software synthesis language. They may record, generate or appropriate data, audio, MIDI, and/or music notation, each in a wide variety of types and formats, and stored using a number of different physical media. They may also produce paper sketches, encompassing prose, notation, or graphics.

The value of these various types of sketches and other compositional byproducts will depend on the specific context. Data may be of interest if its application or origin is evident. Audio recordings may be useful if they are transformed beyond recognition in the final work, or if unused recordings suggest the process by which segments were selected for use. Paper sketches are a more familiar category, but may not be decipherable or relevant to the completed music. The genre of music, the specifics of the work, and the investigator's interests (analytical, critical, and/or historical) are also influential: we are likely to ask very different questions of a "soundscape" work than of a piece in the "elektronische musik" tradition.

Compositional records for two works, Fernando Lopez-Lezcano's *IiceScCrReEaAmM* for four-channel tape, and Christopher Jones' *Matragn*, for clarinet and CD, are examined here in order to suggest the potential value of computer music sketches, and particularly of those records resulting from the use of sound synthesis languages. These works were selected because they used a familiar language, Bill Schottstaedt's Common Lisp Music (Schottstaedt 1994), and because the composers were willing to lend their sketch materials.

2 Synthesizer design as composition

Most of the records documenting the first version of Fernando Lopez-Lezcano's *IiceScCrReEaAmM* (premiered July 1998, and later revised) concern the composer's development of custom synthesis software for the piece. There are twenty-nine discrete versions of the composer's "grani" granular synthesis program and twenty-three versions of his "dlocsig" dynamic spatialization software dating from the period of composition. Unfortunately, only three closely related versions of the synthesis "score" are extant: the version numbering suggests that there were at least nineteen earlier stages which are now missing. Additionally, there are fifteen source soundfiles (mostly short, monophonic recordings, of percussion instruments, kitchen utensils, and children screaming and laughing), among a large number of related soundfiles not used in the completed piece. When asked if there were any paper sketches available, the composer jokingly replied, "'Paper?' What is, 'paper'?" – only electronic records were produced during the making of the work.

The revisions to the grani synthesis instrument offer some insight into the composer's creative process. The first version, co-written by Lopez-Lezcano and Juan Pampin, and dated November 6, 1996, was explicitly designed as a teaching tool for soundfile granulation. This version is of a simple design, with only the most basic musical and synthesis parameters available (the only time-varying control is for amplitude). A second iteration in March 1997 added time-varying control of the grains' duration and of the reading position in the source soundfile.

Lopez-Lezcano began serious development of grani (presumably side by side with the development of sound materials for *IiceScCrReEaAmM*) on January 19, 1998. Four new versions of the instrument were saved on that date, the last including sampling rate change by grain (providing more pitch and timbral flexibility), and optional controlled randomizations for some soundfile reading parameters. Two additional versions from the first two days of February document a number of additional changes. Major new features included the ability to specify the density of grains independently of their duration, and the availability of exponential envelopes. The user interface was also updated, with most parameters designed to accept either constant values or time-varying envelopes, and pitch control specified in semitones instead of as percentages of the original sampling rate. There were also a number of changes "under the hood," with a new method for managing the grains' sampling rate and some modularization of the code. Finally, these versions include the first available instances of calls to the instrument which operate on the "scream" soundfiles, prominent in the finished composition.

The most intensive development of grani took place on March 1-10. During that time the composer produced thirteen new versions of the instrument, with increased modularity in design, substantial documentation, and default settings for every parameter. New features included a more complete implementation of the grain density parameter, interpolated transitions between different grain envelopes, additional controlled randomizations, a modular filterbank construct, and the ability to spatialize the sound on a per-grain basis. Most of these features are used throughout the score: virtually every call to the "grani" instrument includes a time-variant specification of grain density, and the sampling rate randomization is also near-ubiquitous. The filters and randomizing "grain-density-spread" parameter are also used consistently though less frequently. Further additions over the next month (six versions between March 21 and April 6) added raised cosine grain envelopes (again, used throughout the finished score) and a generalized "modulator" construct, providing modulation synthesis just as the filterbank made subtractive synthesis available.

A final change to grani (while technically still a single instrument, now essentially a suite of tools) took place towards the end of the composition process. In mid-June Lopez-Lezcano translated Tim Stilson's model of the Moog voltage-controlled filter to Common Lisp Music (Stilson and Smith 1996), integrating it into the filterbank construct. This change, made late in the process, was most likely a response to a specific compositional need. However, it is difficult to isolate a particular issue in the score: all fifty instances of lowpass filtering in the finished work use the Moog model. Presumably, existing instances of the original Butterworth lowpass filter were replaced by the new design.

Shortly thereafter, the composer wrote an additional construct to integrate his dlocsig spatialization software (developed separately) with grani. The implication here is that attention to spatialization was also a later part of the compositional process; most likely a number of sonic materials were developed prior to their dynamic distribution in space. (The "move-sound" construct is ubiquitous in the final score; almost every sound in the piece is in motion). These were the last changes to grani during the period of *IiceScCrReEaAmM*'s composition, although the instrument has had a continuing life, and Lopez-Lezcano has provided a number updates and bugfixes, as well as translating the instrument for a newer version of Common Lisp Music, CLM-2 (Lopez-Lezcano and Pampin 1999).

While this narrative of software development tells us little about the pitches, rhythms, or spatial trajectories used in *IiceScCrReEaAmM*, it does provide a number of insights into the composer's intentions and creative process. For Lopez-Lezcano as for many computer music composers, development of software tools is itself an aspect of composition. While grani was clearly designed for release to an audience of composers, technicians, and students of computer music – consider the effort put into documenting both the user interface and the code itself – virtually all of the parameters and features built into the instrument are used at some point in the score of *IiceScCrReEaAmM*. (The exceptions are interface options included for completeness – for instance the ability to specify sampling rate change linearly, instead of by semitones). New features and capabilities were suggested by compositional desires.

One example is the instrument's ability to spatialize individual grains independently. Lopez-Lezcano added this feature to grani immediately after finishing work on his dlocsig dynamic spatialization unit generator. The precise localization and convincing trajectories produced by dlocsig are a signature feature of *IiceScCrReEaAmM*, while the more diffuse results of spatialization by grain are only used in eight instances. The effect is subtle, but provides one of the meaningful levels of contrast in the work. Technical and compositional work are tightly intertwined.

3 From paper sketch to synthesis score

If *IiceScCrReEaAmM* was conceived and produced entirely at the computer, Christopher Jones' *Matragn* was composed via a process oriented towards pencil and paper. This surely reflects the composer's personal preference; the presence of the clarinet part, and the decision to compose its music first, were additional factors. Paper sketches include prose commentary on the work; numerical and graphical representations of the form scheme; notated sketches of pitch material, and the autograph of the clarinet part. Electronic records include soundfiles (recordings of clarinetist Matt Ingalls improvising) and Common Lisp Music scorefiles. *Matragn* was composed during the Spring of 1999, and premiered on July 22nd of that year.

There is a strong correspondence between the paper sketches and the finished piece; most of the compositional decisions documented in the prose and diagrammatic sketches are carried out in the completed work. (There are many signs of erasures in the sketches, however; some of them may have been altered or corrected retrospectively). For instance, a sketch dated May 1st, 1999 describes the formal scheme and duration plan of the piece (Figure 1). The sketches specify seven sections with precise timing. Additionally, the modes of interaction between the clarinet and electronics are described in prose, as are their individual materials and behaviors. Most of these prose descriptions apply to the final piece, and the form and duration scheme is realized to the letter.

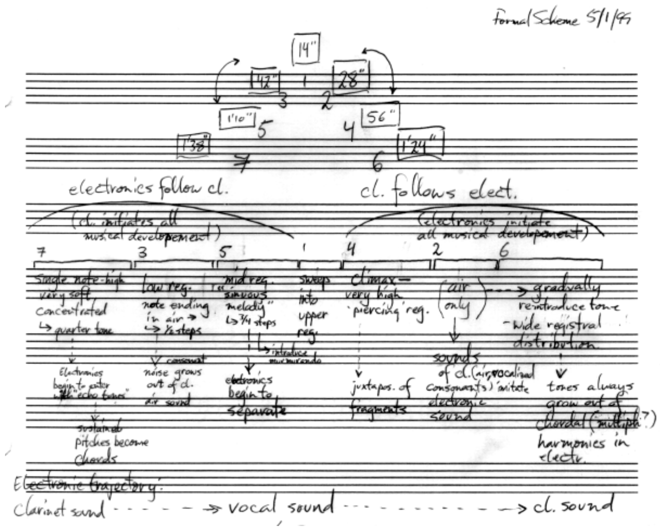

Figure 1. Sketch of formal scheme for *Matragn*

The principal exception to the relatively direct correspondence between sketches and finished work is in the derivation of pitch material. The composer appears to have tested a variety of potential pitch structures, and there are twenty-four different sketch pages which at least partially concern the elaboration and rotation of pitch sequences. Additionally, several pages document early drafts of clarinet material; while these sketches do contain an evolutionary relationship to the finished part, they do not appear intact in the completed work.

An important set of paper sketches document the clarinet part in its finished version, with graphic indications and hieroglyphics suggesting the composer's preliminary ideas for the continuity of the CD material. (Figure 2 is a sample page). Here the programming phase of the creative process, and thus the electronic records for the work, come into play. All the sound synthesis for the piece was realized in Common Lisp Music with Lopez-Lezcano's "grani" instrument (evidence of its continuing popularity at CCRMA).While there are only a few preserved revisions of the composer's CLM code, some sense of the evolution of the composition can be obtained by comparing the finished code for the piece with the initial ideas documented in the manuscript clarinet part.

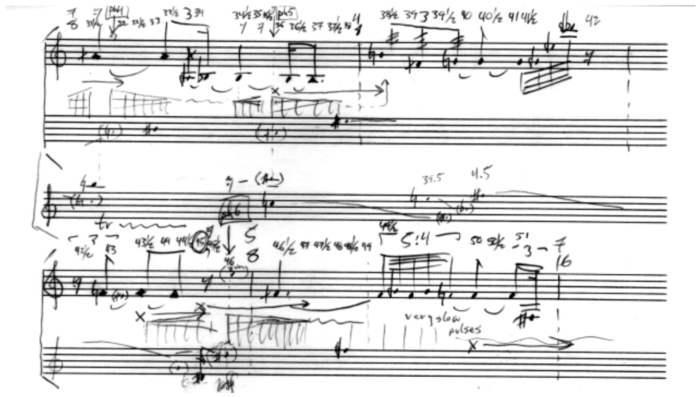

Figure 2. Clarinet part of *Matragn* with sketches for computer sound.

The paper sketches and 2 extant versions of the CLM code for the segment the composer referred to as "section 3" (mm. 30-44) evidence a process in which musical gestures were first roughly sketched and then more fully realized. A group of sounds which start the section (described by the composer in a CLM comment as "shadow sonorities") demonstrate this fleshing-out procedure. The paper sketch for this material suggests only a few pitches, with a single, wiggly line written below. In CLM, a number of new pitches are introduced (and in some cases, pitches from the sketch are changed), and each pitch is realized with two closely-spaced calls to the grani instrument.

There is also documentation of compositional revisions, where more specific plans were later changed. The timings for a set of short, low pitches (named the "bass framework" in a comment) were rethought, with all the onsets after the first shifted forward in time by fourteen to fifteen seconds. Similarly, a set of sonorities termed "foreground material" (depicted in Figure 2 by stems with feathered beams) were experimented with, and then largely deleted from the final version.

Finally, the CLM code includes some experiments, with evidence of the composer working to realize an imagined sound. In the drafts for section 3, there are comments like "here's a bass noise," with grani parameters similar to the specifications eventually used in the "bass framework." Similarly, there are a set of "possible shadow timbres," with some examples related to the finished "shadow sonorities" and others discarded. A more thorough analysis of both the sketches and the finished work might help to suggest the reasoning behind these selections, revisions, and other decisions; in the meantime, they stand as strong evidence of the composer's concerns and his working process.

4 Conclusions

Karlheinz Stockhausen described the rigors of recording his work on *Kontakte*: "the definitive formulation of the technical language... cost me more than a year of my life. A year locked up every morning for three or four hours consecutively in the electronic studio in Cologne... an exhausting experience that tested me terribly and which I wouldn't know how to repeat" (Stockhausen 1987). In spite of Stockhausen's heroic efforts, almost unique in the electroacoustic repertoire, there are still gaps in the record for *Kontakte*: Michael Clarke reports (and my own testing confirms) that there are "a number of ambiguities in Stockhausen's data" and that empirical alterations are necessary to produce a satisfying recreation (Clarke 1998).

Because software synthesis languages like Common Lisp Music and its "Music N" forebears require extreme specificity and detail, composers using these tools generate rich and precise documentation as a byproduct of their work. This evidence is in the form of custom programs and synthesis specifications, readable by those initiated in the language. These documents, especially if accompanied by drafts and revisions, may be suggestive of composers' interests and intentions, which may in turn lead to areas of analytical or critical inquiry. Hopefully the brief studies presented here are indicative of the potential value of these materials.

This type of documentation is not necessarily produced by other computer music tools or working methods: for instance, a Pro Tools-oriented composer today concerned with technical documentation might wish to produce records in a fashion analogous to Stockhausen's labors for *Kontakte*. Nor will synthesis scores and similar "byproduct" documentation necessarily be preserved; by whatever accidents, many records for the works considered here are now lost. Composers are generally more interested in producing work than in documenting it. Sketches and drafts are often saved only if their continuing availability is necessary for the completion of a project, and mistakes and false starts are unlikely to be preserved.

Even meticulous record-keepers may not save sketch materials over time. For instance, Bill Schottstaedt described his documentation in a recent interview: "I used to keep elaborate records of every change I made in an on-going composition, so that I could back up if needed" (Schottstaedt 2001a). However, the intermediate stages are now lost: "Due to severe disk space limitations back then, I tended to clean up everything when a piece was declared finished, so in most cases, this stuff wasn' t saved" (Schottstaedt 2001b). In an era of inexpensive media, the economic dilemmas of storage are reduced. However, documents in any medium can only be viewed as more or less fragile, and the archival community has yet to reach consensus about best practices for the preservation of electronic records over the long term.

Composers who wish to take an active role in recording and preserving their compositional process might consider using revision tools like those used for collaborative software development. While composition is usually a solitary occupation, the automated timestamping and filing of revisions provided by these tools makes them attractive for archival purposes. A work composed with a synthesis language and documented with a versioning system would likely provide rich evidence of the creative process for a scholar patient enough to sift through every revision to the composer's code.

Electronic records may also be increasingly relevant for the study of composers concerned with acoustic music. As an increasing number of composers use computer-assisted composition software like Patchwork and OpenMusic (Assayag 1999), software will be an important part of the documentation they generate. For instance, Richard Toop suggests that sketch materials were indispensable in his efforts to elaborate and explain the compositional process of Brian Ferneyhough's *Lemma-Icon-Epigram* (Toop 1990); analytic work on Ferneyhough's more recent music may depend upon access to his PatchWork code. As with sound synthesis languages, programs written by composers in computer-assisted composition environments may prove to be a source of detailed technical information about the compositional process.

In the absence of significant scholarly, critical, or archival attention to an individual composer's work, the preservation and organization of sketches may not seem pressing. However, technical and creative documentation can play an important role in the reception history of a work: one of the many reasons for *Kontakte*'s canonical status is the ready availability of detailed information about the work's realization. Especially in a field where many composers produce works without scores, sketches and other documentation of the creative process can be crucial for study, teaching, and dissemination.

5 Acknowledgements

Thanks to Fernando Lopez-Lezcano and Christopher Jones for making their works and sketches available.

References

Assayag, G. *et. al.* 1999. "Computer-Assisted Composition at IRCAM: From PatchWork to OpenMusic." *Computer Music Journal* 23(3): 59-72.

Clarke, M. 1998. "Extending Contacts: the Concept of Unity in Computer Music." *Perspectives of New Music* 36(1): 221-246.

Kerman, J. 1982. "Sketch Studies." *Nineteenth-Century Music* 6(2): 174-180.

Lopez-Lezcano, F., and J. Pampin. 1999. "Common Lisp Music Update Report." *Proceedings of the International Computer Music Conference*. International Computer Music Association, pp. 399-402.

Schottstaedt, B. 1994. "CLM – Music V Meets Common Lisp." *Computer Music Journal* 18(2): 30-37.

Schottstaedt, B. 2001a. Untitled interview. http://mstation.org/schottstaedt.html

Schottstaedt, B. 2001b. Personal correspondence.

Stilson, T., and J. O. Smith. 1996. "Analyzing the Moog VCF with Considerations for Digital Implementation." *Proceedings of the International Computer Music Conference*. International Computer Music Association, pp. 398-401.

Tannenbaum, M. 1987. *Conversations with Stockhausen*. Oxford: Clarendon Press.

Toop, R. 1990. "Brian Ferneyhough's *Lemma-Icon-Epigram*." *Perspectives of New Music* 28(2): 52-100.