

December Variations (on a Theme by Earle Brown)

Richard Hoadley

Anglia Ruskin University

richard.hoadley@anglia.ac.uk

ABSTRACT

Earle Brown's *December 1952* is a score characterised by the use of 31 abstract graphical elements. Brown later reimagined it as a Calderesque orrery in which "elements would actually physically be moving in front of the pianist" [1]. Although there are many more recent examples of graphic, open and animated scores, for the purposes of this practice-led research the simplicity and grace of Brown's score makes it a pragmatic choice as it is significantly easier to follow the "translations" being applied. This composition involves research into the construction of a software system allowing multiple automatic 'variations' of the piece, live and in real-time, using common practice notations.

Each variation is created by mapping a uniquely generated version of Brown's original score according to a series of settings - the size and shape of the elements, the 'route' taken through the score: right to left, top to bottom or vice versa, etc. In its current form there is no interaction between performer and score.

The notation provided, although detailed, is intended to be used as a foundation for performance rather than as precise instructions. In this way the project also helps explore the nature or intuition and improvisation through technology and notation.

1. INTRODUCTION

Music performed live has been considered highly visual by many: musical instruments are physical, visual entities; references to and metaphors of music and musical instruments are commonly used in visual and graphic arts, for instance Paul Klee's 1932 abstract painting *Polyphony*.

Music scores are themselves intriguing graphically (or intriguingly graphic) and many practitioners, for instance Erik Satie and Wassily Kandinsky, have exploited this. There has also been a strong tradition since the 1950s of composers including significant graphic elements in their scores. In the case of a work such as Cornelius Cardew's *Treatise* these have become paramount.

Copyright: ©2014 Richard Hoadley et al. This is an open-access article distributed under the terms of the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

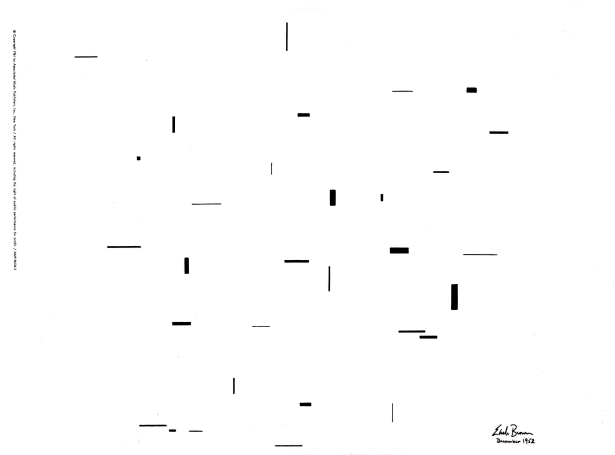


Figure 1. Earle Brown *December 1952*, excerpt from FOLIO (1952/53) and 4 SYSTEMS (1954), H 306mm x W 419 mm, 1961 by Associated Music Publishers, Image courtesy The Earle Brown Music Foundation

2. DECEMBER 1952 - COMPOSING AND IMPROVISING

As one of the most quoted works of the twentieth-century, I do not intend discuss the compositional or performance history of *December 1952* here. David Ryan, John Yaffé and Amy Beal have provided elegant and comprehensive accounts elsewhere [2, 3, 4].

Some years ago I witnessed a performance of *December 1952* (see Figure 1 in conjunction with the essential accompanying prefatory note printed in *Folio and Four Systems*) arranged for a group of about ten musicians. 'Traditional' physical parts had been prepared for each instrumentalist by presenting the original score to a grid; each rectangle was then translated into a note or chord and transcribed onto standard manuscript paper.

Evidently, a lot of effort had gone into the creation of the hand-written score, but if that process was so important, why hadn't Brown undertaken this work himself (for an indeterminate number of players)? Practically, the transcription was necessary in order to coordinate a number of individual parts, but isn't the lack of coordination between multiple instruments a part of the score, if it is interpreted in that way? According to Brown, he used this notation to help him 'improvise' on paper. It was a part of his search for "a new notation...an attempt at correlating my own conception with an extremely rapid way of 'composing', which was, I have said, almost like improvising myself" [1].

This will resonate with many other composers (including the author): the effort of notating can itself interfere with the process of composing. While composing with technology does not immediately help with this - it slows down the process for many - once a satisfactory computer aided system has been constructed (composed?), improvising using other media such as physical movement or drawing can be a very effective compositional method. More traditional paper-based systems are often used in this way - quickly drawn sketches outline complex imaginings, preparing the way for the real work of getting the dots right.

In another statement by Brown also supported by the sketch in the prefatory note, he felt that in a possible physical version of *December 1952* “there would be a possibility of the performer playing very spontaneously, but still very closely connected to the physical movement of these objects” [1], revealing what would appear to be an ambivalence for the territory between improvisation and composition.

3. AUTOMATIC NOTATION

Although T.R. Green in his HCI focused analysis of notations has described music notation as principally a graphic notation it also includes many non-graphic elements enabling effective manipulation on an algorithmic level (the general MIDI format is an example of this) [5].

This division between the graphic and the semiotic is reflected in another dichotomy in music: between ‘signal’ and ‘music’ processing, succinctly observed by Carola Boehm [6]. While the author’s own primary compositional inspiration lies in patterns of and relationships between the discrete packets of information called ‘notes’, the problems and advantages of electroacoustic signs and sounds are equally fascinating, if different. There is now more research than ever into the development of software tools for the understanding, analysing and representation of electroacoustic music. For example, see Patton [7], Blackburn [8, 9], Couprie’s *EAnalysis* [10] and Clarke and Manning’s *Tools for Interactive Aural Analysis* [11].

Conversely, since Brown’s experiments in the 1950s there has been significant interest in graphic and, when technology has allowed, interactive scores. This use of technology has in some cases enabled a fuller understanding of the nature of notation [12].

3.1 Other Automatic Notators

Dominique Fober, the developer of INSCORE, provides an account of other automatic notators [13] alongside descriptions of the abilities of those pieces of software to generate convincing and flexible common practice notation alongside text, graphics and other forms of image manipulation. This complements Harris Wulfson’s 2007 and Jason Freeman’s 2010 papers introducing automatic and real-time notators [14, 15]. Fober also places INSCORE among other current and historical paradigms of score generation such as Guido and Music XML. *Quantum Canticorum*, a composition by the author uses physical movement to influence

music notation as well as audio and so the ability to format this notation live is of central importance [16].

In terms of tools for live notation, related work includes *eScore* [17, 18]: a system exploring composer/performer interactions through real-time notation developed for others to use, but presented with particular compositions as examples.

Animated notation is another related area, about which Ryan Ross Smith has established an interesting collection of work [19]. Although the practice of animated notation includes a variety of methods which do not include the live generation of material, as new software is developed it is clear that the latter will play an increasingly important role.

Collins [20] provides an overview of algorithmic and generative composition without music notation, but Michael Edwards’ *Slippery Chicken* [21] is a computer aided composition (CAC) system featuring the ability to generate sophisticated common practice notation based scores.

Didkovsky and Hajdu [22], Hajdu et al. [23], Agostini and Ghisi [24] describe systems which include methods for defining and projecting notation live. MaxScore/JSML and the Bach Project use live notation as a part of more general CAC systems rather than as dedicated live notators.

3.2 Why Compose Automatically?

Bearing in mind that it is in most cases appreciably more time-consuming to construct methods for composing rather than just composing, what are the reasons for pursuing this activity? Analysing what composers and commentators have said about why algorithmic processes have been used reveals divergent practices.

3.2.1 Algorithms as control

Complementing the introduction to the subject (see section 3.1), Collins elsewhere suggests creativity, understanding and a certain type of control as a prime motivator: “to cite Gregory Chaitin, computer programs are frozen thought; they stand as beautiful (human), artistic, creative, intellectual objects. Algorithmic music is compositional design at a meta-level, human creativity in musical representations, examination of particular rule sets in a space of multiple music theories, with the composer/designer/musician becoming a composer-pilot through musical modelling space. Composers model composition itself, and such systems give us valuable insight into the relations of music theory, musical design and aural instantiation” [25].

3.2.2 Algorithms as external agents

In contrast, another reason often given is that algorithmic processes reveal variations, details and perspectives that wouldn’t have been considered using conscious methods. The British composer Harrison Birtwistle, an early adopter of these techniques, has been particularly prominent in this amongst notation based composers [26]. In electronic music generated in real-time the process can happen each time the music is rendered, arguably providing a level of ‘interpretation’ not usually available (and often not desirable) in fixed-media pieces.

3.2.3 Algorithms as a part of composition

David Cope has discussed his perhaps unsurprising view that the act of composition is by definition algorithmic: “Every composition is a finite sequence of steps. It is logical, then, to assume that...the act of composing is as an algorithmic process. Algorithms emerge...as the most appropriate tool for the creation and study of music” [27], and elsewhere: “I do not believe that one can compose without using at least some algorithms.” [28]

3.3 Why Notate Automatically?

3.3.1 Synchronisation and improvisation

In compositions by the author where the movement of dancers is acquired and the data used [29, 16, 30], live notation is crucial in order to ensure synchronisation of both tempo and frequency/pitch between the generated audio, notation (and therefore instrumental performance) and the dancer’s movements directly influencing the algorithmic processes.

As an aside, these methods also allow non-experts (such as children) to express themselves through movement and hear the result in real-time - this is very much enhanced if it happens in real-time.

3.3.2 Mapping and translation

One of the key hypotheses of this work is that it is possible to ‘translate’ expressive gestures from the graphic domain into the musical domain and that any such translation will enhance the musical experience. To the extent that music notation is already a graphical language this shouldn’t be too great a cognitive leap. The idea of mapping and translation has been increasingly (and justifiably) criticised over recent years [31, 32], but these need not be one-to-one mappings or ‘mickey mousing’ and may involve many-to-many or mappings between ambiguously related functions.

3.3.3 Auditioning

While not reliant on real-time generation of notation, implementing methods of synchronous audition of generated material enables immediate testing of compositional decisions involving algorithms. This is more akin to practices in electroacoustic or acousmatic music where it is normal to be able to hear versions of the music as it is composed.

3.4 Live Notation and Improvisation

This research includes investigation of the middle-ground between composition and tool or instrument - it is therefore important that there is sufficient time during rehearsals to discuss these translations with performers. A hypothesis is that the use of live notation performed at the moment of creation by a human musician (in addition to algorithmically generated audio triggered and modulated by the same movements) will gain an advantage through utilisation of the musician’s training enabling levels of expression, tonal quality, interaction and feedback unobtainable in other ways.

The relationship that exists between compositions which use these methods and improvisation also needs investigation. Figures 2 and 3 (also see Figure 4) show contrasting



Figure 4. The ‘graphic’ of Variation 8

phrases which have been generated live during rehearsal or performance. The instrumentalist, in this case a pianist, is presented with very specific pitches and durations to play. These are crucial to the identity of the music: two ‘types’ are deliberately contrasted with each other for aesthetic purposes. In each case, the tessitura is informed by the vertical position of the particular graphic.

These phrases are purely ephemeral. They are generated and then a few moments later deleted or replaced. While both instrumentalist and composer are aware of the type of material that is likely to occur, any detail remains unknown. While the instrumentalist is encouraged to follow the score as closely as possible, there can be no wrong or right notes. The process lies somewhere between performance and improvisation: a position that may take a little time for some performers to become fully familiar with, but not one that is fundamentally problematic (the reactions of performers to some of these phenomena are described elsewhere [29]: after an initial period of adjustment, most performers are enthusiastic about the possibilities of this type of performance). Unlike some other examples of cross-domain mapping, the techniques used in these compositions rely on the performing expertise of the instrumentalist to take advantage of the live notation. Performers who are most used to contemporary music practice with limited rehearsal time will be best placed to work with the system.

These processes and a performer’s reactions to them reveal differences and similarities between improvisation and performing-composing with notation. The results might be considered a particular balance between improvisation and composition, or ‘comprovisation’ as has been suggested by Sandeep Bhagwati [33].

4. TECHNICAL ENVIRONMENT

Algorithmic processes are constructed within *sclang*, the language part of the SuperCollider (SC) audio environment [34, 35]. The algorithms generate time and pitch values which are then sent to either the SCsynth or, via OSC (implemented as an SC class by the author), to the programme



Figure 2. A rendering of the opening of *December Variations*



Figure 3. A rendering of *Variation 8*

INSCORE [13] which is able to generate a variety of notations, including standard music notation. While, for both technical and musical reasons I am currently concentrating on the latter, I am involved in collaborative projects using generative graphics and original, algorithmically generated text. (As an aid to the composition and rehearsal processes, it is technically possible, if not musically desirable, to use a ‘synthesised’ player instead of a human musician.)

5. THE GENERATIVE PROCESS

All musical material is derived from analysis of the original Earle Brown score. The graphic elements used here were reproduced with a graphics programme and the measurements stored - this process will be automated in future. This object, for instance:

```
dec[0] = [ 15.1694, 1.0583, 0, 45.8611, 26.8111 ]
```

is 15.1694 units wide, 1.0583 units tall, has a rotation of zero degrees, is 45.8611 units from the left-hand side of the page and 26.8111 units from the top of the page. Each element is defined in this way¹ enabling easy re-ordering. By default, the score is read from left to right and top to bottom, but it is very simple to reverse either or both of these presets. It would be quite straightforward to construct generative readings which ‘wander’ through the picture (as suggested by Nicolas Collins [36]), but this would make any relationship to the original score quite impenetrable and so less practical while the composition is still in development.

¹ These are relative measurements only. The data are scaled between SuperCollider, INScore and the relevant computer screen; the scale of the original image is less relevant.

Having declared the dimensions of all elements, a number of functions are used to manipulate, reinvent, interpret and playback the data.

The following section describes the main variables currently available to generate the theme and its variations. As will be noted, these were implemented progressively in order to generate the type of piece that is now the result; it will be interesting to test the same processes on different graphic inputs - in particular possibly dynamic ones.

5.1 rNum

A global control, `rNum` determines the level of randomisation required in a given variation (see Figure 4). This rather crude determinant will be specified in increasing detail in subsequent developments.

On generation, a 2d array is created comprising the relevant notes (see section 5.6) and a duration. For illustration here is one such member out of the original 31:

```
[ [ 59, 64, 67 ], 0.5 ]
```

5.2 ~tempo and ~rel

`~tempo` is used to determine the actual playback duration of this item. The associated control `~rel` is used to determine the release time of a given event.

5.3 ~rubato and ~quantise

Rubato (`~rubato` and `rubatoOffset`) are used to colour synthesised playback for auditioning. Quantisation (`~quantise`) controls notational granularity. Rhythm, particularly when using tuplets algorithmically, can be difficult to notate, especially when dealing with this *prima vista* material.

5.4 startStop

Most variations use all graphic elements in a single variation, but in some cases contrasting textures were used. `startStop` enables the ability to choose where a passage should begin and end. It is described in terms of the sequence of graphic elements (in this piece, 0-30 is the full range). `newDecember` specifies whether a new ‘page’ of notation begins (`true`) or not. This is the case in variation 4, where the first and last 15 elements are defined in highly contrasting ways.

5.5 lohi

`lohi` determines the pitch range of a given variation. This is given in MIDI values, for instance: [50, 90].

5.6 chordHeight

`chordHeight`, for instance [1, 6], will scale the relevant dimension of an object (from within its own appropriate range) to a value in this case between 1 and 6. This value represents the number of notes generated in the chord.

5.7 chordIntervals and intervalWeighting

`chordIntervals` determines the range of intervals permitted in the entity while `intervalWeighting` prescribes the likelihood of these intervals being chosen.

5.8 Performing

The above outlines the level of ‘interpretation’ currently available in the composition. Each variation is a function with its own set of global and local variables and arguments. In order to provide control over the tempo and timing of each variation, the performer must evaluate a function on the computer before playing each variation. This is achieved by evaluating a single line of code in SuperCollider (pressing the enter key or equivalent) - this could also be achieved via foot-switch, etc.

6. CONCLUSIONS AND FUTURE WORK

This research provides insights into and answers for questions concerning composition, notation and performance: how does notation provided in real-time (a type of sight-reading) effect the performance? How is it different from ‘completely’ improvised performances? How might it work when coordinating multiple instruments?

There are also interesting challenges remaining regarding how and when to present notation. In 1970 Earle Brown revealed “I have a sketch for a physical object, a three-dimensional box in which there would be motorized elements... as the elements in December are on the paper. ...[It] would...sit on top of the piano and...the vertical and horizontal elements would...physically be moving in front of elements as they approached each other... the performer playing very spontaneously, but still very closely connected to the physical movement of these objects.” [1]

It is clear that far from being a metaphorical, abstract collection of graphics, *December 1952* is a visualisation

of something more concrete (although Brown rather spoils this a moment later by saying that he was “not really ...all that interested in constructing it”!) Brown’s schemes are far from impossible to implement in either software or hardware (although each has different implications) and one of the main future directions for this research points toward the implementation of Brown’s vision in software. This is a graphic exercise, but the thought of implementing a notated version is compelling.

This challenge carries with it more intriguing problems, such as ensuring that notation remains usable through the animated process: schemes which uses opacity as an expressive quality such as that described by David Kim-Boyle [37] may be useful here.

There is often a tension in systems such as these: are they compositions in themselves or tools for composition? This is a difficulty that David Cope has faced when discussing the lineage of his works generated using ‘Emmy’ [28]. Above, I have considered the system described as a form of composition, where the controls are chosen in order to create what is intended to be a rounded, satisfying creation according to my own creative criteria. Another issue to monitor here, then, is how well the ‘system’ might adapt to other graphics.

Finally, and possibly most importantly, this work has been encouraged by two Brown experts, David Ryan and John Yaffé, a personal friend of Brown. A lingering concern is that the ‘translation’ of such graphics into the precision of common practice notation somehow contradicts the freedom expressed by the score. However, both David and John assure me that Brown would have approved of these practice-led experiments because they were using creative methods appropriate to their time [38].

7. REFERENCES

- [1] E. Brown, “On December 1952,” *American Music*, vol. 26, no. 1, pp. 1–12, 2008.
- [2] D. Ryan, “Earle Brown: A sketch,” CD liner note for ‘Earle Brown: Selected Works 1952-1965’, Sony Music, New York, 2006.
- [3] J. Yaffe, “An interview with composer Earle Brown,” *Contemporary Music Review*, vol. 26, no. 3-4, pp. 289–310, 2007.
- [4] A. C. Beal, “An interview with Earle Brown,” *Contemporary Music Review*, vol. 26, no. 3, pp. 341–356, 2007.
- [5] A. Blackwell, T. Green, and D. Nunn, “Cognitive dimensions and musical notation systems,” in *Proceedings of International Computer Music Conference*, Berlin, 2000.
- [6] C. Boehm, “Book review,” *Organised Sound*, vol. 7, no. 1, pp. 79–82, 2002.
- [7] K. Patton, “Morphological notation for interactive electroacoustic music,” *Organised Sound*, vol. 12, no. 2, pp. 123–128, 2007.

- [8] M. Blackburn, "The visual sound-shapes of spectromorphology: an illustrative guide to composition," *Organised Sound*, vol. 16, no. 1, pp. 5–13, February 2011.
- [9] S. Emmerson and L. Landy, "The analysis of electroacoustic music, the differing needs of its genres and categories," in *Proceedings of the Electroacoustic Music Studies Network Conference*, 2012.
- [10] P. Couprie, "EAnalysis : aide à l'analyse de la musique électroacoustique," in *Journées d'Informatique Musicale*, Numediart, Ed., Numediart. Numediart, 2012, pp. 183–189. [Online]. Available: <http://hal.archives-ouvertes.fr/hal-00823848>
- [11] P. Manning, "The TaCEM Project (technology and creativity in electroacoustic music)," in *Electronic Music Symposium at Anglia Ruskin*. Cambridge UK: Anglia Ruskin University, May 2013.
- [12] K. Lassfolk, "Music notation as objects," Ph.D. dissertation, Faculty of Arts, University of Helsinki, Helsinki, November 2004.
- [13] D. Fober, Y. Orlarey, and S. Letz, "INScore: An environment for the design of live music scores," in *Proceedings of New Interfaces for Musical Expression*, Oslo, Norway, 2011.
- [14] H. Wulfson, G. Barrett, and M. Winter, "Automatic notation generators," in *Proceedings of the New Interfaces for Musical Expression Conference*, New York, 2007.
- [15] J. Freeman and A. Colella, "Tools for real-time music notation," *Contemporary Music Review*, vol. 29, no. 1, 2010.
- [16] R. Hoadley, "Quantum Canticorum," in *Proceedings of the AISB Convention*, Goldsmiths, University of London, April 2014.
- [17] C. McClelland and M. Alcorn, "Exploring new composer/performer interactions using real-time notation," in *Proceedings of International Computer Music Conference*, 2008.
- [18] M. Alcorn, M. Ortiz-Perez, and J. Yang, "Composer-conductor-performer - realtime manipulation of through-composed musical notation," in *Proceedings of the Notation in Contemporary Music Symposium*, Goldsmiths, University of London, 2013.
- [19] R. R. Smith, "Animated notation website: <http://ryanrosssmith.com>," February 2014. [Online]. Available: <http://ryanrosssmith.com>
- [20] N. Collins, "Musical form and algorithmic composition," *Contemporary Music Review*, vol. 28, no. 1, pp. 103–114; 103, 2009.
- [21] M. Edwards, "An introduction to Slippery Chicken," in *Proceedings of International Computer Music Conference*, Ljubljana, 2012, pp. 349–356.
- [22] N. Didkovsky and G. Hajdu, "MaxScore: music notation in Max/MSP," in *ICMC*, 2008.
- [23] G. Hajdu, K. Niggemann, A. Siska, and A. Szigetvari, "Notation in the context of quintet.net projects," *Contemporary Music Review*, vol. 29, no. 1, pp. 39–53, 2010.
- [24] A. Agostini and D. Ghisi, "Bach: an environment for computer-aided composition in Max," in *Proceedings of the International Computer Music Conference*, Ljubljana, 2012, pp. 373–377.
- [25] N. Collins, "The analysis of generative music programs," *Organised Sound*, vol. 13, no. 3, p. 237, 2008.
- [26] D. Beard, "The shadow of opera: Dramatic narrative and musical discourse in Gawain," *twentieth-century music*, vol. 2, pp. 159–195, 2005.
- [27] P. da Silva, *David Cope and Experiments in Musical Intelligence*. Spectrum Press, 2003.
- [28] K. Muscutt, "Composing with algorithms: An interview with David Cope," *Computer Music Journal*, vol. 31, no. 3, pp. 10–22, 2007.
- [29] R. Hoadley, "Calder's violin: Real-time notation and performance through musically expressive algorithms," in *Proceedings of International Computer Music Conference*, ICMA, Ed., 2012, pp. 188–193.
- [30] —, "The Fluxus Tree," in *Proceedings of the Live Interfaces: Performance, Art, Music Symposium*, Leeds University, 2012.
- [31] M. Baalman, D. MoodyGrigsby, and C. Salter, "Schwelle: Sensor augmented, adaptive sound design for live theatrical performance," in *Proceedings of the New interfaces for musical expression conference 2007*, 2007.
- [32] C. Salter, M. Baalman, and D. Moody-Grigsby, "Between mapping, sonification and composition: Responsive audio environments in live performance," *Computer Music Modeling and Retrieval. Sense of Sounds*, pp. 246–262, 2009.
- [33] S. Bhagwati, "Personal communication," in *INScore Workshop*, Lyon, France, April 2013.
- [34] J. McCartney, "Rethinking the computer music language: SuperCollider," *Computer Music Journal*, vol. 26, no. 4, 2002.
- [35] S. Wilson, D. Cottle, and N. Collins, Eds., *The Super-Collider Book*. Cambridge, MA: MIT Press, 2011.
- [36] N. Collins, "Personal communication," in *INTIME symposium*, Coventry UK, October 2013.
- [37] D. Kim-Boyle, "Real-time score generation for extensible open forms," *Contemporary Music Review*, vol. 29, no. 1, pp. 3–15, 2010.
- [38] J. Yaffe, "Personal communication," October 2013.