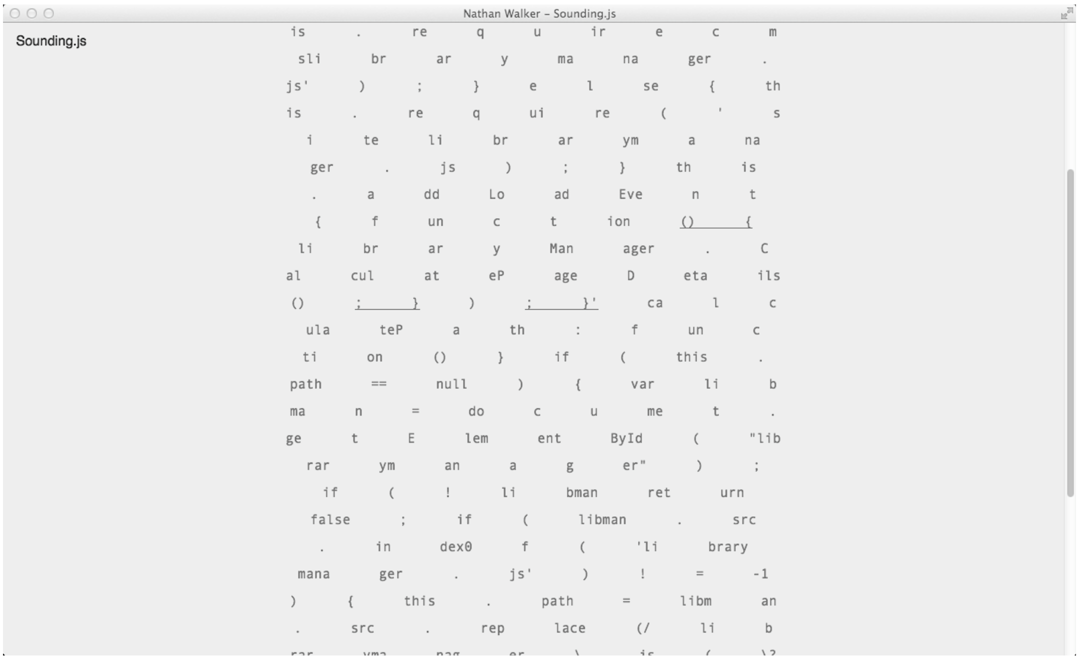
Transitional Materialities and the Performance of JavaScript

Nathan Walker

This is an Author's Accepted Manuscript of an article published in Performance Research Journal 18:5 2013
available online: http://www.tandfonline.com/doi/full/10.1080/13528165.2013.828942

I am going to approach questions around materiality in digital writing practice in terms of performance, sound, and sound-poetry. I'm using Alan Golding's term 'Transitional Materiality' as a kind of model which connects my own practice in this project 'Sounding.js' to the practice of the Language Poets and their exploration of materiality, performance and the sound of language. I will also consider some propositions of the Italian futurist Filippo Tommaso Marinetti.



```
is      .      re      q      u      ir      e      c      m
 sli      br      ar      y      ma      na      ger      .
js'      )      ;      }      e      l      se      {      th
is      .      re      q      ui      re      (      '      s
  i      te      li      br      ar      ym      a      na
  ger      .      js      )      ;      }      th      is
  .      a      dd      Lo      ad      Eve      n      t
  {      f      un      c      t      ion      ()      {
 li      br      ar      y      Man      ager      .      C
al      cul      at      eP      age      D      eta      ils
()      ;      }      )      ;      }'      ca      l      c
  ula      teP      a      th      :      f      un      c
  ti      on      ()      }      if      (      this      .
path      ==      null      )      {      var      li      b
ma      n      =      do      c      u      me      t      .
ge      t      E      lem      ent      ById      (      "lib
  rar      ym      an      a      g      er"      )      ;
   if      (      !      li      bman      ret      urn
  false      ;      if      (      libman      .      src
  .      in      dex0      f      (      'li      brary
 mana      ger      .      js'      )      !      =      -1
)      {      this      .      path      =      libm      an
.      src      .      rep      lace      (/      li      b
```

My work, 'Sounding.js', is both a live sound-poem and an online interactive website[i] that enables the user to 'activate' a recording of the performance by moving their cursor over the digital score. Composed using a complete and fully functional JavaScript, the code is presented as a phonic utterance; a vocal exploration of non-phonic programming language and the spatial and temporal possibilities of embodying the digital text in performance. I'm particularly interested in the materiality of writing in relation to the comparisons of page, screen and voice and to what N. Katherine Hayles means when she says it is crucial 'to reconceptualise materiality as the interplay between a text's physical characteristics and its signifying strategies' (Hayles 2002: 72). The aim of 'Sounding.js' is to consider the poetics of code, specifically JavaScript, as a score for performance, and how the writing practices of computer programming might exist beyond the screen. I am suggesting an embodiment of digital text within the domain of performance and sound-poetry.

As an artist I'm interested in the event of writing, whether this is on the page, on the screen or in the mouth. In my work I employ strategies of writing that are not limited to mark making but extend into sound, arrangement and movement. I'm interested in writing as a response to the materials *of* writing. In recent years, and independent to my practice, I began to teach myself how to build websites. In doing so I arrived at JavaScript and wanted to understand not only how to read and write it but also how I might explore this visually dynamic form of writing as a score for performance. I decided to explore the question, what could JavaScript sound like?

Invented in 1995 JavaScript is a programming language that executes the behaviour of websites. It doesn't control the content or the overall aesthetic of the content, these are controlled by html and css. Some people read and write in this language, but most web developers use online JavaScript libraries like JQUERY that provide useful fragments of prewritten code samples. This kind of patch-writing is what most developers engage in when writing code, it is rare that you would write from scratch. For the majority of web users this code is invisible, and when by chance or by mistake it becomes visible, it is ignored, considered as a non-natural language, nonsensical and visually confusing. JavaScript enables a behaviour; its content is its function. As Giselle Beiguelman (2006) says in her provocation 'Nomadic Poetry', 'Close the book. Open the text: the data-space text, diagram, abstract machine that makes no distinction between content and expression' (286). I want to alter the distinction between content and expression, where JavaScript becomes the content as well as the behavioural device. When JavaScript is used on a website the computer 'tokenizes' the code, removing all white space, translating linguistic signs into numerical data and fixing human error as it goes, altering the text into a series of 'tokens' or blocks of code. This process questions the materiality of the human-writing of code script as the computer translates whatever we write into something else, a something else that is more functional, more accurate, more efficient and, of course, one that we could not write. Some of us may be able to read the code but only intelligent machines can execute it.

It is my opinion that some of the forms of writing that exist in JavaScript code were predicted by the Italian futurist F.T. Marinetti. Canadian poet Christian Bök, suggests that Marinetti's ‘parole in libertá’ (words in freedom) '…gives voice not only to the ecstatic impulses of an organic anatomy but to the electric impulses of an operant machine' (Bök 2009: 131). Marinetti advocates abolishing what he calls the old syntax in order to verbalise a completely new way of living brought about by the advances of science and technology. One Hundred years after Marinetti's 'Zang Tumb Tumb' (1912-14) and 'Destruction of Syntax—Imagination without strings—Words-in-Freedom' (1913) the new syntax inhabits our daily lives through programmable and electronic code. Written codes that exist behind all of our digital engagements, enabling not only a fixed desktop interface but more recently a hand-held portable and social experience. This then, is an *itinerant* syntax, nomadic in its execution of spatial transitions.

Marinetti's argument for the destruction of syntax pre-dates and predicts the kinds of writing we see in web-based programming code.

For example:

- the removal of the adjective as a corrupter of the noun
- uninterrupted sequence of new images
- no punctuation or verse form
- no finite verbs
- adjectives replaced by compound nouns
- conjunctions replaced by mathematical symbols

(Adapted from Perloff 1986: 59)

Furthermore, in the 'Technical Manifesto of Futurist Literature' (Flint 1991: 92) Marinetti states that 'to accentuate certain movements and indicate their directions, mathematical symbols will be used' (Marinetti In Flint 1991: 93) (he also considers these symbols akin to musical notation, suggesting the parallel use of code as score). It is through Marinetti that I am attempting to discover what JavaScript sounds like, following his instruction to 'listen to the motors and reproduce their conversations' (Marinetti In Flint 1991: 96).

Electronic writing practices have been engaging with Marinetti's concept of reproducing (and representing) the conversations of their computers. Rita Raley mirrors Marinetti's instruction defining codework, the process of using code as text in new media poetics, as that which 'makes exterior the interior workings of the computer' (Raley 2002). For me this is informed by a performance practice that is post-Cagean and open to interpreting anything as a score for performance. When we take programming code out of its functional use as simulator, as behavioural determiner, we are not removing the materiality of the screen-text but enabling / highlighting a transitional materiality, by 'infiltrating the surface' (Raley 2002) of natural languages.

Alan Golding (2006) calls this 'transitional materiality', posing one way of considering the materiality present in all forms of poetic text as being part of a spectrum (2006: 277) and not in opposition to one another. In considering the work of the Language poets he see these texts as 'technotexts' a term used by N. Katherine Hayles to describe 'work that foregrounds its own materiality or inscription technology' (Golding 2006: 251). The technotext relates closely to Steve McCaffery's assertion that sound poetry is, 'the organization...of language around its own phonic substance, as a self-referring materiality, non-representational and escriptive rather than descriptive' (McCaffery 1998). Perhaps this is *the removal of the adjective as a corrupter of the noun* that Marinetti advocated.

I would like to relate that self-referring materiality to a writing machine of the 20th Century, the typewriter. Perhaps we might learn from the code-like visual poetics of the language poets what is to come in our current century. It is no coincidence that online code uses monospaced typefaces that mimic the typewriter font, or that cascading style sheets use line breaks and indentation as a way of allowing a clearly paced readability. As Charles Olson argued in his seminal work 'Projective Verse' (1950),

> It is the advantage of the typewriter that, due to its rigidity and its space precisions, it can, for a poet, indicate exactly the breath, the pause, the suspensions even of syllables, the juxtapositions even of parts of phrases, which he intends. For the first time the poet has the stave and the bar a musician has had. For the first time he can, without the convention of rime and meter, record the listening he has done to his own speech and by that one act indicate how he would want any reader, silently or otherwise, to voice his work. (Olson 1966: 22)

In this sense, the computer, the screen and the languages of digital domains develop further the performance of poetic texts in relation to their materiality. I'd like to draw our attention to the work of three poets/artists who have explored forms of code, post-Marinetti, and the materiality of language as sound.

Considering 'Lift Off' (1979) by Charles Bernstein, a transcription of everything lifted off a page with a correction tape from a manual typewriter. We are confronted by a string of letters and symbols:

> HH/ ie,s obVrsxr;atjrn dugh seineocpcy i iibalfmgmMw
> er,,me”ius ieigorcy¢jeuvine+pee.)a/nat” ihl”n,s
> ortnsihcldseløøpitemoBruce-oOiwvewaa39osoanfJ++,r”P
> (Bernstein 2010: 36)

Charles Bernstein says that 'when sound ceases to follow sense, when, that is, it makes sense of sound, then we touch on the matter of language' (Bernstein 1998: 21). When we consider 'Lift Off', visually it is not dissimilar to the appearance of JavaScript. To people who do not write or work with code, both 'Lift Off' and JavaScript can appear impenetrable. JavaScript has its own syntax, consisting of a constellation of symbols, half words, joined words, punctuation as interruption, parenthesis, empty parenthesis, repeated symbols, and equation, to name just a few formations.

In an information theory sense, the look of the JavaScript and 'Lift Off' presents us with noise; it is visually noisy but materially silent. The acoustic attribute 'noise' is what happens when we see non-word-like shapes. Confusingly, because we conclude that this noise is inarticulate, or that we can not articulate it, in actuality it remains silent. In her essay 'The Sound Shape of the Visual' (2009) Ming-Qian Ma explores these inarticulate signs saying 'they tend to be perceived, by virtue of … perceptual unfathomableness, as having reached a “silence”' (Ma 2009: 250-251). These 'silent icons of authority' (2009: 251) are 'extralinguistic', like code, they are engaged in cause and action. Using the acoustic attributes of the visual poem and its use as score for performance we can explore the possibilities of sound as material. It is through this idea that we can see materiality not only in terms of physical properties but as having a 'dynamic quality that emerges from the interplay between the text as a physical artefact, its conceptual content, and the interpretive activities of readers and writers.' (Hayles 2004: 72). Bernstein has said that this is the only poem he has never publicly performed. It was, however, performed by Kenneth Goldsmith[ii], author of 'Uncreative Writing' (2011) who notes 'Bernstein chooses to foreground the workings of a machine, rather than the sentiments of a human...Bernstein's poem is, in some sense, code posing as a poem' (Goldsmith 2011 17-18). It appears similar to the JavaScript code, although this is not a digital text but an analogue archive of writing *as writing*. The machine writing or, the writing machine.

Another machine, a book-machine, Steve McCaffery's 'Carnival' (1967-70), explores the visual noise of the page as a score for his extreme live performances. Released not only as a book but also online as a digital facsimile, the work appears, like code, to be dense, un-readable, and incorporates non-phonic symbols, repetition and alphanumerical marks. McCaffery says 'Carnival' 'offers the reader the experience of non-narrative language' (McCaffery 2000: 446). Although created using a manual typewriter and other non-digital mark making techniques, the movement of reading in McCaffery's work, in both the online edition and the artist's bookwork, is a process of navigation. The text is presented as a network, recalling Roland Barthes' famous metaphor of 'text' from 1986. This networked reading, moving not simply from side to side but up and down, diagonally, paragrammatically, presents us with a spatial and temporal materiality that we are not used to in literature. It is a practice more closely associated with movement, mapping spaces and territories of landscapes and now online domains. The performance of 'Carnival' is a practical application of McCaffery's own theory of the 'Voice in Extremis' (1998) which he says is an 'attempt to emancipate voice from code'. He goes further, 'Language, signification, and code are certainly corporealized...yet voice, empowered by this embodiment, is still not freed from language…sound poetry's goal is 'the liberation and promotion of the phonetic and subphonetic features of language' (McCaffery 1998: 163).

Christian Bök's 'Cyborg Opera' is not, strictly speaking, code in the same way we have seen in the examples of Charles Bernstein and Steve McCaffery. This excerpt entitled 'Motorized Razors' attempts to remove the linguistic materiality of computers and computer games away from the screen and into the emergent materiality of live performance.

From 'Motorized Razors'

rotozaza whipsaw
zoom
zang
tu m' tu m' dada

roborama buzzsaw
buzz
bazooka buzzbomb
bang zang
zing zing zipgun
zulu
acid jazz
is zaum jiujitsu

xerox juggernaut
jaws of a jigsaw
ripsaw zumburook
blunderbuss boom

gyrozaza hacksaw
dire
doom
dynamite cadenza

razorama circsaw
merz
berserkers curse
each
law which causes
rust on scissors

chainsaw maniacs
hijack high-tech
tick-tock clocks
as klaxons klang

gangs of gung-ho
guys
with axes
battling gatling
guns
attack boomboxes
oxyacetylene sun
(Bök n.d)

Similar to the syntax of JavaScript, words are pushed together, spaces removed, but the mathematical symbols and inarticulate signs are not present. Instead we have a Futurist- inspired onomatopoeic text that simulates the sounds presented in Super Mario Bros and other computer game experiences. Bök explains that 'if poetry hopes to go beyond reason then it must disintegrate meaning, perhaps through primal shouts or ritual chants so that words may recapture a “primordial feeling” or “primordial concept”' (Bök 2009: 130). As N. Katherine Hayles states, 'materiality thus cannot be specified in advance; rather, it occupies a borderland—or better, performs as connective tissue—joining the physical and mental, the artefact and the user' (2004: 72).

Digital textuality is adaptable, fluid and open. My attempt to sound code aloud relies upon these understandings of electronic text, that Golding calls 'oscillatory…[the] matter of fluid signs, of signifiers in motion' (Golding 2006: 250). Can we embody the materiality of texts within the new and evolving domain of digital language? I'm arguing that through phonic experiments the non-natural languages of the computer can perform their own materiality live.

In Sounding.js the human to computer transference of data is no longer primary. Instead, the transition from one form of language to another is the focus. N. Katherine Hayles suggests that 'as the unconscious is to the conscious, so computer code is to language.' (Hayles 2006: 137). The spatial metaphors that prevail in the discussions surrounding digital writing allude to the practices of writing within the emerging poetics of the twenty first Century. The architectures that are constructed virtually indicate a geo-graphic space for the reader and the user; what Beiguelman calls a 'temporal interface' (2006: 286) where that which exists underneath is foregrounded. Hearing code, at the extremes of language provides another spatial arrangement - sound as measuring device - mapping the terrain of networked spaces through the articulatory engine of the mouth. By listening to the computers and reproducing their conversations, the possibilities for understanding the transitional materialities of contemporary writing emerges as fluid, oscillatory and open.

References

Bernstein, Charles (1998) Close Listening: Poetry and the Performed Word, Oxford: Oxford University Press.

Bernstein, Charles (2012) All the Whiskey in Heaven: Selected Poems, Norfolk: Salt Publishing.

Beiguelman, Giselle (2006) 'Nomadic Poetry' in Adalaide Morris and Thomas Swiss (eds) New Media Poetics: Contexts, Technotexts, and Theories, Cambridge:

Leonardo Books, The MIT Press, pp. 285—290.

Bök, Christian Motorised Razors from 'The Cyborg Opera', Unpublished.

Bök, Christian (2009) 'When Cyborgs Versify' in Marjorie Perloff and Craig Dworkin (eds) The Sound of Poetry, The Poetry of Sound, London: The University of Chicago Press, pp. 129--141.

Flint, R.W. (1991) Let's Murder the Moonshine: Selected Writings trans. R.W. Flint and Arthur A. Coppatelli, Los Angeles: Sun and Moon Press.

Golding, Alan (2006) 'Language Writing, Digital Poetics, and Transitional Materialities', in Adalaide Morris and Thomas Swiss (eds) New Media Poetics: Contexts, Technotexts, and Theories, Cambridge: Leonardo Books, The MIT Press, pp. 249--283.

Hayles, Katherine, N. (2004) 'Print is Flat, Code is Deep: The Importance of Media-Specific Analysis', Poetics Today 25(1): 67--90.

Hayles, Katherine, N. (2006) 'Traumas of Code', Critical Inquiry 33(1): 136--157.

Ma, Ming-Qian. (2009) 'The Sound Shape of the Visual: Toward a Phenomenology of an Interface' Marjorie Perloff and Craig Dworkin (eds) The Sound of Poetry, The Poetry of Sound, London: The University of Chicago Press, pp. 249--269.

McCaffery, Steve (1984) 'Sound Poetry' in Bruce Andrews and Charles Bernstein (eds) The L=A=N=G=U=A=G=E Book, Carbondale: Southern Illinois University Press, pp. 88--91.

McCaffery, Steve. (1998) 'Voice in Extremis' in Charles Bernstein (ed) Close Listening: Poetry and the Performed Word, Oxford: Oxford University Press, pp. 162--177.

McCaffery, Steve. (1998–2001) Carnival [electronic version]. Coach House Books. www.chbooks.com/online/carnival/ accessed 28 January 2013.

McCaffery, Steve. (2000) Seven Pages Missing. Volume One: Selected Texts 1969–1999. Toronto: Coach House Books.

Olson, Charles (1966 [1951]) Selected Writings of Charles Olson, Robert Creely (ed) New York: New Directions.

Perloff, Marjorie (1986) The Futurist Moment: Avant-Garde, Avant-Guerre, and the Language of Rupture, Chicago: University of Chicago Press.

Raley, Rita (2002) 'Interferences: [Net.Writing] and the Practice of Codework.' Electronic Book Review: Electro Poetics www.electronicbookreview.com/thread/electropoetics/net.writing accessed 10 April 2013.

[i] 'Sounding.js' is available online at http://www.nathan-walker.co.uk/sounding. The interactive score uses a recording of the work performed at E-Poetry Festival 2013 at Kingston University, London, on 18th June 2013.

[ii] An audio recording and annotated script of Kenneth Goldsmith reading 'Lift Off' at the launch of 'All the Whiskey in Heaven' (2010) is available at https://jacket2.org/commentary/kenneth-goldsmiths-script-lift