

## *Code As Language*

Loss Pequeño Glazier

<http://epc.buffalo.edu/authors/glazier>

### **Keywords**

poetry, language, programming, code

### **Abstract**

To consider how code can be language one must realize that inscription is not simply about recording ideas but about inscribing language in a specific medium. Writing occurs in space and space is itself part of the process through which writing produces meaning. In digital media, textuality is equally a function of the meaning of space. The materiality of the digital written object differs from that of the print object, but is no less material. There are specific categories that define the writerly features of writing code as mark-making. Ultimately, dynamic text offers the most interesting way to view how code is language. A poetics of dynamic text seeks to engage that delicate edge where language apparatuses meet, slip, and engage, to further the possibilities of the poetic text. Ultimately, a poetics of programming raises the question, where is the writing — in the code or in the displayed language? If language is defined as written symbols organized into combinations and patterns to express and communicate thoughts and feelings — language that executes — then coding is language.

### **Introduction**

Gaps

shocks through

absorbing

-- Bruce Andrews

To consider how code can be language, I would like to concentrate on language in its written sense, as an agency of mark-making to produce meaning in a given medium. Here, two observations should be made. First, the nature of such agency is not limited to simply recording pre-existing, definitive, or idealized thought, rather the act of inscription is itself a process of thinking through thought. Second, the process of inscription is not just about meaning being placed on a given material. Instead, meaning is made through the act of inscribing on the specific material. Further, the act of inscribing engages the visual, spatial, and material modes of, not mark-making, but of meaning making through making marks. (And, one would hope, with a consciousness of Marx). Finally, the recorded text is not an ideal or definitive one but is merely one articulation of many possible ones; echoing Jerome McGann, Johanna Drucker, and Cary Nelson, an exemplar of the specific material and social factors that condition mark-making at that moment. Writing is the registering of an individual iteration of the always dynamic multiple material possibilities of textuality.

### *How is the E-text Spatial?*

The book had some pages

Wet it becomes a brick  
to play with then discard

-- Michael Palmer

Writing occurs in space and space is itself part of the process through which writing produces meaning. In print, there are many ways to view the interrelated nature of space and writing. As Johanna Drucker has argued and Brian Kim Stefans has parodied, the New York Times, for example, draws its content not only from the text that occurs on the page, but from the masthead, the layout, the columns, the subheads on the page. Simply by looking at a New York Times page one knows not to expect a poem by Mallarmé, a treatise by Mao, or even reporting of any actual news. Ironic! Thus meaning is made not just by what the text says but equally by how space is used in the scene of writing. Such a dynamic has been explored to great effect by a large number of poets including Charles Olson, Susan Howe, and countless Concrete Poets.

Accordingly, there is no such thing as negative space. The book itself is dependent upon space. This includes the margins, top and side, the gutter, the space of the title page and verso, endnotes or footnotes, the bibliography and appendices, pages and bindings, and, more tangibly, the space between words, around illustrations, within individual characters, the kerning, indents, tabs, font. Space is everywhere! The book is a spatial construct. Without its given space a book would be much different in layout than what we know as a book (e.g. there's no reason a book couldn't be a dark blob of ink on a single scroll of paper). Even further, the space of the book is an articulation of the material presence of paper, the production means of printing, and social processes such as bookselling, reading, and remediation. In the work of many contemporary poets, white space can itself constitute the content of the literary work as much as the text itself, as William Carlos Williams argued in *Imaginations*. It's not just the artists that can be spaced out, but the texts themselves. In this context, it is interesting to look at white space in the development of the work of robotics pioneer Norman White. White's earlier geometric paintings expressed the substance of "empty" space and lead to strikingly similar patterns in his later circuit designs. These are works of circuit art that make meaning as much through the physicality of empty space as through the patterns that depend on that space for definition.

In digital media, textuality is equally a function of the meaning of space. This occurs at various levels:

- There is the space of the network, the fact that texts can exist across nodes and that physical space, though dramatically compressed, is inherent in the process of reading net-based literature. There is the space of the screen, its two dimensionality and luminescence, the way it can be navigated through scrolling, linking, and/or paging. There is the space of the hard disk, the way data is stored in scattered fragments on the disk and only appears to be coterminous on the screen.
- There are the spatial metaphors of specific interfaces. The Microsoft screen reality, for example, a monopoly paradigm sanctioned by the corporate U.S. government and complicit U.S. court system, is by no means an authoritative, useful, or even efficient model for effecting the potentials of digital writing (Indeed, the space of such an interface is designed to be a consumer item, as generic as the controls on a microwave or a television remote). There are many other interface paradigms, map-based and visual relational models for example, that have been overlooked for Microsoft's aggressive consumerist and lowest common denominator conceptualizations.

- There is the space of programming. In programming, space has functional, representational, and symbolic levels of meaning. Declaring space is fundamental to writing a program. The instantiation of variable, for example, is specific down to the level of the data type. Indeed, one declares data types as specific as integers, strings, characters, boolean, byte, short, long, float, and double. This is making space in the program! As William Carlos Williams has noted, "When we name it, life exists" (*Imaginations* p. 115). A statement such as:

```
life = new Life ();
```

seems to breathe vitality into naming unimagined at the times Williams theorized poetry in his book *Imaginations*.

There are other spatialities key to writerly programming. Decisions such as whether copies of objects are used in new contexts or existing objects are overwritten, recursion, the articulate realities of arrays, and the energized antics of loops within loops, are a decisive part of the logic of programs. When writing programs to be displayed in a GUI, there are many ways that space plays a role: the number, obstinacy, and design of individual pop-up windows, navigation bars, text fields, and screen layouts. There is the fact that a program is a linear file but one dependent on structures of space within that file. These include the expressive practice of indenting code, the aligning of related elements, the spatial interleaving of descriptive comments. And finally, there is the relative substance of space in different contexts. How it varies, for example, in lines of code versus within a string, and other cases.

### ***What is the Material?***

Not words but  
wing-sounds

there is no *nothing*

-- Keith Waldrop

One hears the occasional argument for the immateriality of the e-text. This includes that made by Johanna Drucker in her *Language as Writing: Intimations of Immateriality* essay. There have been many arguments that assert that the printed text is more permanent than the digital text. This, I would argue, is a matter of historical perspective. In the way that, from a Buddhist perspective, a particular personal problem looks so much less significant when viewed from the perspective of 6,000 lifetimes than from the one we usually desperately cling to, the difference in the durability of paper and digital media looks much less drastically different if viewed in terms of thousands of years instead of hundreds. I do understand the issue raised by those who ask, as Susan Howe did in Buffalo in 2002, how can you touch a digital file? This question raised by Howe, a poet noted for her innovative explorations of print lineation, genuinely evokes an issue in media transition. One can begin to understand digital materiality if one seeks, rather than a tactile grasp of the medium, an understanding based on an examination of qualities and issues the medium presents.

- The digital art object consists of marks in magnetic media. Marks are not usually made directly and must be made through an intermediary instrument of mark-marking, such as a keyboard, mouse, touch pad, or other device.

- Like alphabetic characters, marks are representational. Sequences of marks convey ideas through grammar, syntax, and expressive logic. Such ideas are evoked by the marks but are not independent of them.
- Unlike print, such marks are fungible and highly transmissible.
- The material is a spectral one. It gives the impression of being impermanent because it requires display or projection to be viewed. Unlike film or photo negatives however, it has indiscernible tactile qualities, nor even on a microscopic level.
- The material naturally provides for a dynamic, rather than a fixed literary object; this is in stark contradistinction to conventional notions of the literary object.
- The material allows for algorithmic thinking. One should, however, be specific about the use of the term algorithmic. That is, its use should be closer to being a synonym for logical pseudocode rather than a synonym for a concept, trope, or other such general popularized extraction.
- The digital literary object is one that is highly specific in its historic and material circumstances. One only need think of cgi-bin programs at their height, of php, of the effects of pixelation, as striking as any romantic language by Rimbaud, of the Mario Brothers generation of video games, or of present efforts in php.

### ***How It Is/Not Writing***

To understand the words as so liberated is to understand poetry.

-- Williams

The fact that marks are representational immediately suggests that encoding is writing. Of course, by this definition, one could also say that other forms of mark-making, sculpture, painting, playing an instrument are also writing. I would accept that these other examples are also "writing" if we define writing as an act of engaging a material to explore ideas through the process of working through that given material, physically, socially, and ideologically. The case of encoding is even more closely located to what we might conventionally think of as writing. What are some of the writerly features of code?

- Grammar. There are rules to be followed and expression is articulated through the use of syntax. Sometimes you can break rules and get away with it. Sometimes breaking such rules ain't bad for expressing what you wish to express. However, a consciousness of the rules is fundamental to literary production.
- Semantics. Just because you follow grammatical rules (your program compiles) doesn't mean your object will function (For example, just because you're married, it doesn't necessarily mean you're happy).
- In terms of digital language art, language is a means to make language.
- Encoding is a process of mark-making. As such, it is a form of writing.
- Encoding is a means to an end but it is most expressive when the means is itself a focus of attention.
- Meaning emerges through the process of engaging the medium.

- Errors are part of the process of making meaning through encoding. Errors sometimes mould the production of the literary work.
- The encoded work has inherent unpredictability, often surprising even its maker.
- Encoding is making.

### *Poetics of Dynamic Text*

The apparent

present.

-- Rae Armantrout

Of course, one might say that dynamic text, or text that is different on each reading, is a mere dramatization of the unarguable fact that even *Wuthering Heights* is different each time you read it, depending on the characteristics of the last Heathcliff or Catherine in your own personal life and whether you are reading it on vacation in Yorkshire or in the depths of the Yucatan jungle. Each reading of such a novel is different, of course, because differences in context, setting, and personal circumstances foster different interpretations, cause different words, images, etc. to jump out.

However, dynamic text, text that is *physically* different each time you encounter it offers interesting possibilities.

- It forces the literary work away from the idea of a final form presented on a fixed surface "of record". This is why conventional link-node hypertext offers little in the way of innovation and why, when link-node hypertext declares itself to represent new technology, it is actually quite a heinous misrepresentation.
- When the artistic work is forced away from fixed form, one must look deeper for a sense of meaning. This means looking to the concept, mechanism, or operation that underlies the work, querying the core stability underlying the work, that which remains constant beneath the litigious, shifting illusion of its surface. One must find what is solid beneath the transitory — much like meditation!
- This also means literally looking deeper — to the code. A work of programmed literature, and here I would emphasize works that are hand coded as opposed to interface assembled, present a complex of writing, that is, textuality superimposed on textuality. In this environment, one move can affect elements on other planes of activity. As in 3-D chess, one must think on several levels before making a mark!
- Programmability may be a defining characteristic of what might really be called New Media writing. That is, not writing that has been remediated to sit on the screen like a colorized stuffed Iowa pheasant on the mantelpiece, but writing that engages a complex of language (im)possibilities. Most importantly, the reader should approach the work keenly aware of the writing within writing that makes such a work happen. Indeed, thinking again of Norman White's robotic artworks and White's tendency to use semi-transparent cases in their construction, one should think of looking at the surface while bearing in mind the "writing" beneath. It is useful to approach programmatological works with such a mindset.

- An example of this might be in processes that engage writing code structures, displaying and sounding them, then altering the code to alter their soundings. Such an interplay offers interesting new ways to conceive of a poetics of coding. This is reminiscent of an interview with a Hispanic writer I heard, where he mentioned writing a piece in English then translating to Spanish then revising by translating back to English. It was a way of using two language processes to sound against each other to build ... un interlinguistic objeto of arte.
- The dynamic qualities of such works are dependent on specific and varying notions of seed and on supporting randomization and selection algorithms. These are issues in the work also worthy of attention and debate.

"Dynamic" is not here meant to simply mean text that moves. Neither is it meant to mean text that merely has computational origins. The object that is at the center of this inquiry is one that does not just sit there (or sit there and move). Rather a poetics of dynamic text seeks to engage that delicate edge where language apparatuses meet, slip and engage, to further the possibilities of the poetic text. Indeed, one could look at some dynamic texts at this point to consider this, but we will continue.

### *Where's the Writing?*

The bright tongues of two  
languages

dance in the one light.

-- Robert Duncan

Ultimately, a poetics of programming raises the question, where is the writing? It is a perspective that looks into the coded work of digital text art, taking into account the complex and interconnected layers of expression that constitute the work. A digital poet who writes merely on the surface or a programmer who does not see the writing the code makes, is simply not exploring the potentials of the medium. What are the poetics at work in these layers?

- At the code level, poetic writing can consist of economical, tight, expressive, and meticulously annotated works. In addition, the inventive use of method overwriting, crafted declarations, and other tropes are strategies of writing. These strategies likewise can influence the object being produced. Writing with such an interplay relies on code that itself has a literary/poetic sensibility.
- At the surface level, a consciousness of 20th century innovative literary and artistic practice can do much to advance the potentials of the digital medium. Unfortunately, the general tendency of digital poetry seems to take a step backwards in terms of surface content as a kind of knee jerk off-reaction to the newness of the medium (As if, the more conventional the writing, the more likely it is the new medium will be accepted by practitioners of the old medium). Most early hypertexts, and most of those that are award-winning, are clear examples of how unadventurous such surface writing can be. In interface art, this problem can sometimes be even worse, as if Macromedia is the undercover FBI operative infiltrating the gatherings of digital Weatherman. This phenomenon is present today in interface works that, comparing digital media to the medium of video, have no more aesthetic depth than commercial music videos (There is digital poetry that offers crucial thinking-throughs of these questions, but one must seek it out).

- At both levels, a consciousness of writing as a thinking — through the specific qualities of the material qualities of the medium. The engagement with the material, between the coder and the code, is a real one. Like cha cha, there has to be pressure between the bodies for it to work.
- At both levels, a consciousness of writing practice as ideological expression. For example, how some simple programmatological objects require a lot of processor speed to support the overhead of the interface itself. Or how some programs require huge amounts of additional memory to produce certain fonts or how e-mail programs will keep reattaching binary code to a message when an image cannot be displayed, endlessly increasing the drain on available system virtual memory. What are the ideological implications of such a bloatware ethos? In these cases there is no incentive for precision or economy. No system eco-sensitivity. This is the same principle we find in the corporate auto industry-oil company matrix (just keep making more cars and placing residential areas further from work locations), a set of assumptions that have had a devastating effect on the environment, the lifestyle of the average worker, and even policies of war.
- The writing can probe the interface or coding issues but the result should be interesting. Let's not produce more television commercial jingles nor regurgitate the *Wired* aesthetic, itself a declaration of corporate identity.
- Not just text art that uses programming but code as poetic practice. The code or the text may be interesting but most interesting is their interrelation.

If language is defined as written symbols organized into combinations and patterns to express and communicate thoughts and feelings, then coding is language (With the exception of, that is, the language I use when I receive compile errors!). One may, however, extend this definition to suggest that language is play. Language is where one plays and it is the play in language as a medium — the slippage and double denotations — that can create delight enough to lighten a dreary Buffalo even in November. How is encoding language? At both the code level and on the surface, relevant are indicators such as humor, innovation, irony, double meanings, and a concentration on the play of language. It is the making of marks with a sense of marksmanship, the bull's-eye being that specific point where language doubles, allows multiple meaning, and launches the executable onto its unpredictable, though guided, path through the artful potentials of inscription.

### ***References and Notes***

- Andrews, Bruce, *Wobbling* (New York: Roof, 1981).
- Armantrout, Rae, *Made To Seem* (Los Angeles: Sun & Moon, 1995).
- Drucker, Johanna, *Figuring the Word* (New York: Granary, 1998).
- Duncan, Robert, *Bending the Bow* (New York: New Directions, 1964).
- Palmer, Michael, *Codes Appearing* (New York: New Directions, 1981).
- Waldrop, Keith, *The House Seen from Nowhere* (Brooklyn, NY: Litmus, 2002).
- White, Norman. See <http://www.normill.ca/> (21 January 2006).

Williams, William Carlos, *Imaginations* (New York: New Directions, 1970).

### ***Author Biography***

Loss Pequeño Glazier is a poet, professor of Media Study, and founder and director of the Electronic Poetry Center (<http://epc.buffalo.edu>), the world's most extensive web-based digital poetry resource, housed in the Department of Media Study, State University of New York, Buffalo. He is the author of the digitally informed poetry collection *Anatman, Pumpkin Seed, Algorithm* (Salt Publishing, 2003), several other books of poetry, and the award-winning *Digital Poetics: The Making of E-Poetries* (University of Alabama Press, 2002).

Rooted in an experimental literary sensibility, his digital poetry consists of visual text-art that engages sound, photography, video, interactivity, and programming to explore new possibilities for time-mediated and web-based digital art. Recent performances of, *Baila*, his digital poem for dancers, have occurred in London and Buffalo. He is the author of acclaimed works such as *Io Sono At Swoons, White-Faced Bromeliads on 20 Hectares, Mouseover, Viz Études*, and his work-in-progress, *Territorio Libre*.

He is organizer and director of E-Poetry: An International Digital Poetry Festival, the first and one of the most celebrated digital poetry series in the field. His work has been shown at various museums and galleries, including the Kulturforum, Berlin, the Royal Festival Hall, London, and the Guggenheim, New York, and he has lectured and performed throughout the U.S. and in London, Paris, Berlin, Norway, Spain, Mexico, Canada, and other countries. Selected digital projects and other work are available on his EPC author page (<http://epc.buffalo.edu/authors/glazier>).

### ***Citation reference for this Leonardo Electronic Almanac Essay***

#### ***MLA Style***

Glazier, Loss Pequeño. "Code as Language." "New Media Poetry and Poetics" Special Issue, Leonardo Electronic Almanac Vol 14, No. 5 - 6 (2006). 25 Sep. 2006  
<[http://leoalmanac.org/journal/vol\\_14/lea\\_v14\\_n05-06/lpglazier.asp](http://leoalmanac.org/journal/vol_14/lea_v14_n05-06/lpglazier.asp)>.

#### ***APA Style***

Glazier, L. P. (Sep. 2006) "Code as Language," "New Media Poetry and Poetics" Special Issue, Leonardo Electronic Almanac Vol 14, No. 5 - 6 (2006). Retrieved 25 Sep. 2006 from  
<[http://leoalmanac.org/journal/vol\\_14/lea\\_v14\\_n05-06/lpglazier.asp](http://leoalmanac.org/journal/vol_14/lea_v14_n05-06/lpglazier.asp)>.

*“... the highbrow hot spot of cyberspace... a digital bible for the lowdown on the [hi-tech] art scene.” —Wired*

## Leonardo Electronic Almanac (ISSN: 1071 4391)

### Nisar Keshvani, Editor-in-Chief

Established in 1993, Leonardo Electronic Almanac is, jointly produced by Leonardo, the International Society for the Arts, Sciences and Technology (ISAST), and published by MIT Press, is an electronic journal dedicated to providing a forum for those who are interested in the realm where art, science and technology converge. For over a decade, LEA has thrived as an *international peer reviewed electronic journal* and web archive covering the interaction of the arts, sciences, and technology. LEA emphasizes rapid publication of recent work and critical discussion on topics of current excitement with a slant on shorter, less academic texts. Many contributors are younger scholars, artists, scientists, educators and developers of new technological resources in the media arts.

### New Media Poetics and Poetry Special

Guest edited by **Tim Peterson**

- :: Code as Language - *Loss Pequeño Glazier*
- :: Lens: The Practice and Poetics of Writing in Immersive VR (A Case Study with Maquette) - *John Cayley with Dimitri Lemmerman*
- :: Numbered Space and Topographic Writing - *Lori Emerson*
- :: Digital Poetry: From Cybertext to Programmed Forms - *Phillippe Bootz*
- :: Concrete and Digital Poetics - *Manuel Portela*
- :: Writing the Virtual: Eleven Dimensions of E-Poetry - *Stephanie Strickland*
- :: \_Net.Drenching -- Creating The Co[de][i]n.Text\_ - *Mez*
- :: Morphing Into New Modes of Writing: John Cayley's riverIsland - *Maria Engberg*
- :: 'Virtual Typography': Time Perception in Relation to Digital Communication - *Matthias Hillner*

### New Media Poetics and Poetry Gallery

Curated by **Tim Peterson**

- :: Hymns of the drowning swimmer - *Jason Nelson*
- :: open.ended - *Aya Karpinska and Daniel Canazon Howe*
  
- :: MotionText Ferment - *mIEKAL aND and Camille BacoS*
- :: The Trustfiles - *Nadine Hilbert and Gast Bouschet*

*Lens: The Practice and Poetics of Writing in Immersive VR*  
Copyright © John Cayley



art / science / technology

<http://leomanac.org> [lea@mitpress.mit.edu](mailto:lea@mitpress.mit.edu)