Virtual Muse

Charles O. Hartman

Published by Wesleyan University Press

For additional information about this book
http://muse.jhu.edu/books/9780819572578

Access Provided by New York University at 03/02/13 4:40PM GMT

> > > > / 9 /

UNCONCLUSION

I don't think computer poetry teaches us much about computers, so far. Or at least mine doesn't; as I've said several times, these programs don't push programming into new territory. What I find interesting is that these experiments, which are so simple from a computer science point of view, can help remind us in a new way of things we already knew about poetry and about language.

Maybe this isn't surprising. The originators of the cabalistic system I mentioned in the preceding chapter were almost certainly responding to a new *technology* of language—writing—when they began to treat words as hidden numbers. (Preliterate people, for whom language is exclusively speech, can't think of words as composed of letters that might have numerical equivalents.) Early in the twentieth century, poets, especially Williams and Pound and Charles Olson, discovered new expressive resources in such techniques as the precise two-dimensional arrangement of words on the page, which made poetry a little more like painting. These discoveries depended on a technological innovation: the typewriter. Typewriters took the capabilities that printers had enjoyed for centuries and made them available directly to authors. Williams's extremely careful use of lineation—now part of the toolkit of hundreds of poets—depended on his ability to produce quickly, in his own study, dozens of variations on the "printed" artifact of his poem. Some people argue that computers, simply as word-processing machines, have begun to transform writing too. Probably any technology newly applied to language will suggest a sudden new slant on the words by and among which we live.

Thinking up ways to "do computer poetry" makes us look at language and poetry from an unfamiliar angle. This seems appropriate if one of poetry's functions is to make us aware, with a fresh intensity, of our relation to the language that constitutes so much of human life—or if you like, of how language constitutes so much of our relation to the world. How do words mean when we put them into new contexts? Under what conditions does the meaning web tear apart? What meanings can words make (or can we make of them) when we disturb their normal relation to each other? These are questions that any poet can be seen as asking; and from this point of view, someone experimenting with computer poetry is continuing an age-old project.

One of the first things I relearned from working with programs like Prose was how essential the reader is to the poem. Part of this can be put in simple economic terms: the reader is a consumer, and any manufacturer who ignores the consumer's needs goes out of business. But the point is not, God help us, that poets should anxiously adjust the poem-product to current demand. (As readers we need far more then we may know we need or can think to ask for.) Instead, we're bound to recognize that writing a poem is entering into an elaborate, subtle, unspoken contract with the possible readers of that poem—the kind of contract that binds a family or a society together, not once and for all, but over and over again every day.

So those diagrams I showed in the preceding chapter are seriously incomplete. Whether the computer generates or manipulates text, we need to see it in context:

PROGRAM ———> TEXT ———> READER

or

TEXT ———> PROGRAM ———> TEXT ———> READER

This correction is likely to make us think about poetry in what M. H. Abrams calls "pragmatic" rather than "expressive" terms. That is, rather than dwelling on how the poet has said what the poet wanted to say, we'll probably concentrate on how the reader is affected by the poem. This means asking what the poem *does* and looking at the devices and choices and parts of a poem as the ways it does it. The

reader's role isn't just to be a spectator but to be the arena in which the poem acts itself out.

If the reader is so important, we could also think of moving him or her away from that end-of-the-road box in the diagrams and back into the process somewhere. We could try to make the reader's constructive role in the poem more conscious, for instance by giving the reader explicit choices to make. The catchword here is "interactive poetry," and poets have begun exploring several possibilities in this direction. In 1989, a Canadian poet, Rod Willmot, wrote "Everglade," which he calls "the first hypertext poem ever created." Hypertext, a very hot topic these days, is essentially an attempt to make text (which we normally think of as a speechlike, linear string of words) multidimensional. In a way, texts of all kinds have always done this. We remember, we read back and forth, we make connections that couldn't be diagrammed in one dimension. The Talmud, as my colleague Roger Brooks argues, is a hypertext without computer technology. The basic text of the Mishnah is surrounded on the page by the most important links to other texts, and the traditional Talmudic scholar is the living embodiment of a vastly larger network of such links. (The computer, especially the CD-ROM that can place the vast literature of the Talmud within the computer's instantaneous reach, has been a blessing to scholars.)

Willmot's poem is several dozen computer-screen-sized poems. With a click of the mouse, the reader can see one or more words highlighted on the screen. Each of these, with another click, turns out to be a link to another poem. The associative links aren't just coincidences of words; they're part of the poet's meaning—the meaning of the *whole* poem, a meaning *behind* the individual screen of poetry.

Though hypertext is advertised as a kind of apotheosis, as what text always wanted to be when it grew up, in a sense this approach reduces the role of text. The poem itself becomes a *path* through the network of texts. Since there are many paths and since the order in which we read different pieces of a poem affects our sense of what the whole poem means, "Everglade" is many poems. Here again is our old friend the arbitrary, but now it's the reader who exercises it

and in that sense participates in the making of the poem. The French poet Raymond Queneau made a work with a similar effect in the precomputer middle of the century. His "Hundred Billion Sonnets" looks like ten perfectly normal sonnets—except that each line in each sonnet can be replaced by the corresponding line in any of the others. His title points out that this allows for 10^{14} different fourteen-line sonnets (a hundred billion). Clearly the reader is expected to construct and read some selection of these (just a selection; if each one takes a minute to read, the complete work would occupy us all day every day for 190 million years).

Enthusiasm for this realm of possibilities shouldn't make us forget, though, that *all* poems are interactive. The reader is always an essential collaborator, if not in the making of the text, still in the making of meaning. We act as careful readers by making choices; the most essential one we make is to read the poem *again*; and each time we do, we read a somewhat different poem. When Robert Frost begins "The Most of It" with the line "He thought he kept the universe alone," we can hear "thought" as emphasized (he was mistaken, maybe foolish, to think so) or not (he was framing a hypothesis, pending further data). We can hear "kept" as a synonym for "preserved" (guarded from intrusion) or for "maintained" (as in *housekeeping*). We can hear "alone" as belonging with the second "he" (he and only he kept the universe) or as an adverb describing his lonely state. Each of these possibilities changes our sense of everything that happens later in the poem. Yet the choices aren't exclusive—we don't settle on one and reject the others—but cumulative. What the poem "means" is the sum of these possibilities and the ways they modify each other. Most readers won't hear all of these on the first time through, so "The Most of It" grows with each rereading.

If the poet lets the reader know that a computer was involved in the making of the poem, then the reader's awareness of interacting with the poem can be oddly heightened. I've remarked before that the presence of computer peculiarities in a poem's language tends to enhance our sense of strangeness—our suspicion of the language, as

it were. We're pushed one step back from immediate saying-and-hearing; the language shifts from *use* toward *mention*.

Another way to put it is that the computer's intervention can make the poet and the reader aware (and what poet and reader are aware of constitutes the meaning of the poem) of the peculiar objectivity of language. Language has everything in the world to do with persons and personalities. To a very large extent our personalities, among our fellow humans, are created by and embodied in what we *say*. Furthermore, language is made by people. There's no one else who can do it. Schoolchildren come to think of language as authoritatively enshrined in the anonymous Dictionary; but dictionaries have authors (usually committees of authors), and language has a history. Every word we speak was once spoken for the first time by somebody, and it didn't exist until then. Yet it feels very odd to think of language as being made by people, because we almost never see it happening. Whom do you know who has invented a word, one that entered into the language people speak or write?

Language is what some scientists have taken to calling an "emergent" phenomenon, like a brain or an epidemic or a stock market crash or a culture. Partly this means that you can't get to the meaning of a sentence by adding up the meanings of its individual words. But it also says something about how language arises and grows. We don't see people at work inventing language, but countless unconscious acts of creative variation are going on all around us. A slight change in pronunciation, a slight carelessness in the use of a word that becomes habitual, a plausible stab at reforming an old word to fit a new situation—these things that happen all the time accumulate to change languages and ultimately to make new ones. (Some variations accumulate; most die out.) To repeat a metaphor I used in the preceding chapter to talk about poetic form, these acts are to language what mutations are to biology—the material for evolution; and the evolution's engine is a kind of natural selection that works on this material. This is why a historical family tree of languages looks exactly like the bushy growths that evolutionary biologists draw.

So the language belongs to *all* of us rather than to *any* of us. This rigorous commonality keeps us linguistically honest (more or less). Humpty Dumpty—"I pay words extra, and they mean what I want"—will ultimately cut himself off from language, because a language requires at least two speakers who agree on it. A more intensely focused version of the same commonality keeps us honest in the give-and-take of a poet and reader coconstructing a poem. Students often worry: "But won't everyone read the poem differently? Won't everybody have a different interpretation?" Yes, in one sense, but only in the sense that we're all different. We're also all the same; our genetic makeup, our anatomical construction, even our personal histories are identical to within vanishingly small (if all-important to us) degrees of variation. The language communities to which we belong heavily constrain the interpretations we'll produce for a given bit of text. A poet and a reader make up a tiny community that depends on those larger, containing communities.

It's not just interpretation that's constrained by all this history and circumstance; what we say is also constrained to a great extent. How much of what you said in the course of yesterday consisted of things like "How are you?" and "Pass me the salt, would you?" If that makes for a close and happy community, if it keeps us from circling each other warily like competing species at a water hole, it also threatens to make life dull. If our talk makes our personalities, we'd just as soon find something to say that nobody else would quite say. A job of poetry, again, is to keep refreshing the possibilities for things to be said and heard. Maybe the job of language is to say what the world gives us to say. But it's also true that language, discovering new ways of saying things, generates new things to say. Here the computer can help in some direct if crude ways.

Furthermore, whether the reader knows what was involved in the making of a poem or not, the use of computers objectifies some aspects of the writing of poetry. Most often it emphasizes something we knew anyway: that most writing is rewriting. To rewrite, what the writer must do is to become a reader, to put herself or himself in the place of a reader and ask, "Is this what I want to read? If I change

this word, will the whole thing become something I'd *rather* read?" In the diagrams I gave earlier, READER isn't a person but a role. The drama changes when we take each other's parts.

It's worth making a three-way distinction among the *random*, the *arbitrary*, and the *contingent*. If we think (peculiarly) of human beings as language-output devices, the output is very largely contingent, depending on various kinds of history: of the speaker, of the speaker's relation to the person spoken to, of the language created by generations of speakers, of the world in which speakers and listeners find themselves. One effect of computer poetry experiments is usually to release language from contingency. The randomness of Prose, the arbitrariness of Diastext are contingency breakers. Of course, the meaning of a bit of language depends heavily on the contingencies that have shaped it. If we get rid of contingency entirely, replacing it with purely random or arbitrary linguistic acts, we get genuine gibberish. The point, rather, is to introduce calculated bits of mechanized anarchy into the language, put the results back into the contingent world where language lives, and see how the dust settles.

It's not just in computer poems that randomness, arbitrariness, and contingency compete and combine kaleidoscopically. We see the same interplay in the news, in every conversation every day, in the muttering senate inside our own minds. In some moods we resign ourselves to watching the contingent workings of history. In others we knock on wood to fend off random accident or go out adventurously in search of it. Sometimes we try to impose our arbitrary will on events, and sometimes it works, for a while, until the random and the contingent reassert themselves. None of the three can take over for long, which is just as well. We live most when we live in flux.

This page intentionally left blank