QUESTIONS ABOUT THE SECOND MOVE

Jim Rosenberg

The first move being what it has always been – hearing the words. Or seeing the words; whatever way they first come to mind. Or perhaps they are not even words but fragments of words, or even letters. They come to mind and then *they must be put somewhere:* The Second Move. Of course the traditional answer to the question of where the words go in the second move is: into a notebook. Or even perhaps right into place in the poem itself. The second move is so habitual – subliminal almost – that most poets would react quizzically to even giving it a name, or thinking of it as a move at all: it is as natural as breathing. (Perhaps it *is* breathing (Ginsberg 1971; Olson 1950.)) But for the cybertext poet, this is suddenly an issue, not simple at all. What is the cybertextual equivalent of a notebook? Where do the words go when they are still only *a scrap?* Must the cybertext poet give up scrap collecting?

Philippe Bootz, in "The Functional Point of View" (1997), describes a vastly different approach. The cybertext author creates a *texte-auteur*, a kind of scenario by which the cybertext is assembled. The cybertext author becomes like a filmmaker, having to extrapolate in the mind how the final piece will work. There can be scraps of *texte-auteur* material – but these are not pieces of cybertext, they are pieces of *directions* for assembly of cybertext. The poet collects not scraps but "metascraps." Despite all of our vaunted rhetoric about the interchangeability of reading and writing (Joyce 1995), despite all of our emphasis on the interactivity of the cybertext, the writing process is in fact not interactive at all. There is no immediate feedback between the author and what replaces "the page" at the level of scraps – at the point of the Second Move – no ability to react instantaneously to those nascent bits of word behavior and say, no, that's not quite right. The *texte-auteur* is a one-way street.[1]

The question is not how to go from the Second Move to the Third Move, but how to make a Second Move which is (instantly, on the spot) *returnable* to the First Move. A Second Move which is already cybertext. A Second Move which is already interactive. How can we say we know what interactive writing is until we have done this?

So where is the code in this second move? Where is the algorithm? And can we say we have cybertext with no algorithm? Surely there is no code at all in the First Move. The poet "hears words." The only code is the code of the neurons. (We do still have the term "natural language" – as opposed to "formal language" – for a reason.) Must the code be written on the fly as part of the Second Move? How do we put the code into the cybertext poet's *fingers?* For the Second Move is a matter of fingers. If the First Move is a matter of ears, of eyes, of breath, it is at the Second Move where something becomes made, some object which exists some-where for the first time outside of the poet's head. A *thing* has to happen. Never mind that it might be a digital thing. It is still an object, something *preservable*. It may be the Second Move but it is the First Object. The note. The scribble. *The scrap.* Something tangible. Something the poet can forget about without losing it, and have the scrap to bring back to mind. But a cybertext scrap. A word object which (already) behaves.

The question is one of arena. Where does the word object go when it is fresh, instantaneous, just born? The usual answer to this question is that it goes into some kind of cybertext authoring system. Like Flash (Macrome-dia Inc. 1995), for instance. But will it *play* at the Second Move? Perhaps not. Perhaps it needs a whole support structure that has to be provided along with the words. Perhaps code needs to be written first. Or back-grounds need to be made: other layers. The word at the Second Move is a vagabond with no camping equipment: it has no place to go. So it fills out a form, sits on the sidewalk, waits for the builders to arrive. This is what the *texte-auteur* is: a requisition. A request for service. A request for builders, with a list of things they need to do. In my own case, not even a word object at all, but a notation in a kind of intermediate language, e.g. "Build a word cluster here with the following layers. ..." The word has to wait, perhaps weeks, before the word object gets built, according to requisition.

So then, what is to be done, to allow a Second Move that will play, that doesn't need the requisition form? The poet needs something to act as a notebook. As the first wet-paint home for scraps. Not just word scraps, but word scraps that behave. That can move or respond or be layered –

whatever kind of *behavior* the poet's aesthetic calls for. Behaviors that are pre-built, receptacles waiting for the words. No code needed at the Second Move because the code is already there. A *construction kit* for word objects.

Of course environments like Flash already provide an assortment of pre-built objects. So what exactly is the problem? There are several. At the moment of the Second Move, the final home of the word object scrap is: *Don't Know*. Just as for centuries the final home of a line jotted down in a notebook is: don't know. If writers have been doing this for centuries, why does software make it so difficult? The corollary: at some perhaps much later move, the poet does know: the word object does get a home. So it should be very simple to just move it, to plop it into place. With all of its interactivity or animation or behavior of whatever kind coming right along. Just as the syntax comes along with the words when you paste in a sentence somewhere. (Syntax *is* a kind of behavior. It tells how the supposedly linear word stream folds and unfolds, how parts of the word stream attach at a distance. How pieces of the text attract or repel. Places where the sockets are, other places that go into the sockets. Almost a kind of animation as the connection happens.) Cut-and-paste is so simple to use in an ordinary word processor – at some later move the results of the Second Move must be pasteable – somewhere else.

The set of kinds of objects available in Flash at the time of the Second Move is *closed:* you can't extend it. You can create new objects for *the reader:* objects that work at run-time. But what about new objects for the writer? New kinds of objects that can be used with the poet's fingers. Just after hearing the words. With no sound of code to drive away the words. Just like drawing: a swoop, a hand gesture, a motion. Clay being layered up, oozing out from the fingers as the wheel spins. The tactile feel of the words being molded into place.

And then: you play it. Immediately, on the spot. The Third Move is to play the word object fresh and instantaneous, cock your head, wonder: How do I like how this plays? Is it right? Not a question you have to wait weeks to ask, but a question you can ask on the spot. Just after having heard the words. A Third Move that happens so soon after the Second Move, perhaps the Second Move took so little time you aren't even aware of having made it. A Second Move that is disappearable, out of mind, like breathing. From the First Move to the Third Move as if there never was a Second Move, as if the Second Move is *built in* to the First Move. Second Move: go away! Recede, become subliminal.

The requisition still happens – but this requisition is for the programmer to create new kinds of objects in advance. In advance of even the First Move. This is not a *texte-auteur* for a specific piece, but a requisition for enlarging the construction kit.

So how do we do this? How do we specify the kinds of word objects we will need? We need to become almost more like sculptors than poets: we need to send an order to the *foundry* for fabrication of new kinds of word objects. But here is exactly the difficulty: There is no foundry. We have to do it ourselves.

There must be a system of *objects*. Of movable objects – movable with your fingers. Objects that allow the poet to make new kinds of objects and plug them in. Objects like an *attractor* that would pull in words or phrases or letters from a surrounding text space. Or a *repeller* that would send them away. *Kinds* of objects. (Programmers call these "classes".) A useful variety of kinds need to be available in advance, but also the poet should be able to add more, with reasonable effort. You will say my arithmetic is wrong: If the poet adds a new kind of object to the system, what move number is this? It happens prior to the First Move. Is it a negative-numbered move? It is a move from a *precomposition layer*, a layer that may only need to be done once to support a whole series of pieces: like building a desk. Not the writing but the carpentry. Trans-writing. A producing, like the producing the programmers did who wrote the Flash development environment except that this one is *open*, allows us to add our own kinds of production alongside the ones we were handed. Open to new objects not when the piece is played but when it is written: at the point of stage and scene and frame. The poet must be able to change the way the stage works – to be one's own stagehand.

This is not a pipe dream. There is such a system. Squeak (Ingalls et al. 1997; Guzdial 2001; see also www.squeak.org), for instance. A system that can support words as graphical objects, animations, new kinds of objects invented by the poet, objects that "carry" other objects around with them. A system that lets you move an object where you want it, drop it into place, and all the behavior (by itself and by its components) comes right along with it. A system that is available for free (including source code). A place to start.

An arena. An *object arena*. A place where small active objects (e.g. animations) can be put into a sentence as if they were words. A form of interactive writing that lets the Third Move be playing the results of the

Second Move – immediately, on the spot. The way the poet has always been able to read the line just after the Second Move – while the ink is still wet. Wetness: our cybertext composition process needs to be come wetter, like wet clay. More liquid. Or like a gel, spreadable with the fingers. The place the breath breathes into. The potter's wheel. The clay tablet, a still flexible digital surface.

NOTES

1. Editor's note: Please see Philippe Bootz's paper above and my translator's note. I translate *texte-auteur* as "text-of-composition." Readers will have to make up their own minds about whether Rosenberg's remarks here should also be taken to refer to Bootz's *texte-écrit*, or "text-of-inscription." (JHC)