Voice and Code

Josephine Bosma

Hidden Voices Still Speak

The purpose of this text is to provoke thought about the connections between our oral expressions of language (speech and song) and programming languages. I have no intention to deliver a scientific paper. It predominantly is a reflection on what voice is, and how it seems to relate to code. This paper is written based on the assumption that all human creations "say" something about the human condition. There is no technological world separate from our human world, our bodies, and our minds. We have created machines and languages as we have created the crops in our fields. As Gunther Kress puts it, "Any landscape, the communicational included, is the result of human work. 'Human' and therefore full of affect and desire; 'human' and therefore always cultural and social." [1] The things we have created are as much part of us as our thoughts are. The question is, however, how can we stay engaged with our externalized, materialized ideas and not be controlled by them? If we recognize ourselves in our creations, we are less likely to lose control over them. Adrian MacKenzie writes, "The problem is how to engage with a collective's embodiment of technical mediations without repudiating, or over-identifying with, technology." [2] I believe that reflecting on the human aspect in the languages that control our machines, networks, and systems is one way to achieve this, and I chose the most physical representative of language as an entry point: voice. In art, much of this reflection is performed in a playful manner, by toying with code as if it were some exotic tongue or by simply appropriating it, taking possession of it physically, vocalizing and externalizing what was initially presumed to act as an inner voice of the machine world only.

Voice, Resonance, and Transcription

Many belief systems in the world have been based on the assumption that the course of history could be influenced by ritual, magic, and words. Words are considered to have great power, the power of creation even, which is understandable if one just thinks of the impact words and messages can have on our personal lives. Words move us, and through moving us they change the world, by changing our perception. Naming things makes them come alive in our minds.

In the beginning there was just the body, its inner voice, and its audible voice. Audible voice is the most special of our physical extensions. It has no shape and no easily visible material presence, yet it can have great impact, even when used in a whisper.

Even if it is invisible, it can still be measured. Voice is produced by (and at the same time produces) a physical resonance. Inside us, breath, air, moves the vocal chords, which makes our voice resonate through our head as well as through the air around our bodies. Outside us, the resonance of our voice moves our audience's eardrums, and can even be powerful enough to make glass break. Voice can be like our arms and hands: a part of us that reaches out and touches something or someone.

The sound and impact of our voice depends on our position and our role within our class, clan, or family. It also depends on our mood, our emotional, and our physical state. Voice, in short, is never simply sound. Voice is part of who, what, and where we are.

Through its resonance, voice has the ability to remind us of our physicality. And it not only literally moves us. Its sound and content remind us of our humanity. The sound of a baby crying, the sound of a soothing voice, the roar of a bully: we all have experienced them and registered their meaning. The use of our voice is therefore not a simple gesture. The sound and impact of our voice depends on our position and our role within our class, clan, or family. It also depends on our mood, our emotional, and our physical state. Voice, in short, is never simply sound. Voice is part of who, what, and where we are.

Almost the same could be said about written text. Written text is a transcription of our audible, but also of our inner, cognitive "vocalization." Although we occasionally transcribe speech directly, written text is first of all a transcription of our inner voice rather than our oral expression. Nevertheless, this inner voice is still deeply connected to speech. Written text is an extension of, or a tool for, our voice. It would seem that our voice is not just the part of our expression that we register with our ears and hear. [3]

To explore the connection between voice and code, between our physical vocal expression and programming languages, means that we first have to look at the connection between our vocal expression, our expression in written text and other notation systems (such as notation systems for music and song) and the meanings we have ascribed to these expressions. Given that our voice has been "notated" in other ways as well since the advent of recording media, we have to also consider the influence of recorded and artificial voices on how we perceive and express ourselves. Douglas Kahn puts it poetically: "Writing had silenced the words from one's voice; phonography wrenched the voice from the throat and out of time." [4] Artificial voices and the subliminal languages of machines have changed our perception of voice even further. Our voice has become a complex of oral, written, and machinic expression. What is our relationship to the languages we created to perform calculations or run machines? Are these not also part of our voice?

Signs, Letters, Numbers

We have already been familiar with transcription systems for our human reflections and communications for centuries. These transcription systems have many shapes or appearances. Hieroglyphs, characters, cuneiform script, and Braille are just some examples. The way we transcribe our "voice," inner and outer, is not bound to sound or phonetics, even if we tend to think of the alphabet that way. Just think of how we need phonetics to describe the sound of a word in a dictionary. Methods for transcribing voice depend on culture and circumstances, which are in turn embedded in transcription or in written text. In his book *Literacy in the New Media Age* Gunther Kress explains how it is strange to many non-western cultures, for instance, to have only one way to write a word. These cultures sometimes have three transcription systems to do so. He writes, "transcription systems are not meaning neutral: social meanings attach to them." [5] The same could be said about mathematics and programming languages.

Mathematics is not just calculation. The numerical system may be just a way to represent quantities, but mathematics is more than that; it is also an extreme form of logical reasoning. Mathematics is said to be a (maybe even the) universal language. Even if this is to some extent true, this assumption also tends to give way to an almost religious belief in the power and superiority of calculative reasoning. The overt logic and calculation of the mathematical language are just as seductive as are the emotional lures of gut reasoning. Both seem to represent sincerity and truth, and can therefore easily be misinterpreted or overestimated.

Programming languages, much like mathematics, are designed to be practical and logical. According to the Wikipedia explanation of programming languages, they "differ from most other forms of human expression in that they require a greater degree of precision and completeness." [6] The mention of "human expression" in this quote is interesting. It reminds us that in most copyright battles, those in favor of open source or free software will claim their right to the freedom of speech on the basis of the First Amendment of the US constitution. Much like the text of the First Amendment connects voice to written text, anti-copyright activists connect voice to all computer code.

The Wikipedia explanation or definition of programming languages unfortunately fails to mention cultural or social subjectivity in code. It only states, "the need for diverse computer languages arises from the diversity of contexts in which languages are used," [7] and talks about, among other things, how programmers differ in taste and how novices will write simpler code then "experts." However, subjectivity in code is inscribed in the same way as it is in mathematics, which is through choices of target, application, or negation. Like written text, which can bear invisible negations of unwanted topics or results, code is never truly innocent. As the famous quote from the artist group I/O/D – "Software is mind control – get some" [8]– provocatively reveals, there is awareness of the cultural traces in code and software, but most information on programming and code does not move beyond the realm of the "purely" technical. As mathematics, the technology of organizing, processing, and steering information is perceived as neutral and aimed at truth or perfection. A relationship to human endeavor and desire is mostly ignored.

Magic, Music, and Voice

The temptation to see the universal language of mathematics as (key to) universal truth is very old. Around 500 BC, philosopher and mathematician Pythagoras already connected mathematics to questions about the origins of life and culture. "Pythagorean thought [...] first coined and systematically expressed the idea that a symbolic-mathematical source code underlies the universe and describes nature and culture alike." [9] It was and still is tempting to take this thought one step further, and presume that there is a way to manipulate this underlying source code. Various systems involving a combination of letters, words, and mathematics were conceived with the goal to recreate or change the world. Kaballa, for instance, "effectively combined the Pythagorean idea of the world being composed of numbers so that everything can be described in numerical terms and proportions, with the magical concept of language as an agent that affects matter." [10] Our faith in and awe for new technologies seems to reflect a residue of this belief in the powers of the codes we design. We have developed compositions of text and numbers that actually "do something," even if the end product is often no more than a super-calculator, or only partly and imperfectly

accomplishes what it is supposed to do. Much like a few hundred years ago, the real magic (or the magic of reality) is still unfolding in the combination of our creations and the way we use or understand them.

The presumed magical power of words was and is of course not always dependent on mathematics and calculation. Words have had far greater impact on their own, as naming, as vocalization, as becoming, and have been used as such in rituals and rites. The greatest magic achieved by words is the stimulation and sharing of memory, and through that, the generation and preservation of history. The power of words lies in the naming of situations, events, and people, and in the ability to enable a becoming of history. The creation of history is an act that is loaded with cultural and social complexities, but it is in many ways based on some simple technical facts. These facts all turn on one thing: the human body and its possibilities (or limitations).

There have not always been pen and ink, a tape recorder, or a hard disc to preserve information. Furthermore, some information was never thought of as worthy of transcription and preservation. In both cases, memory had to be preserved differently. What better way to inscribe the slow and forgetful human body with memories of its culture and people than through storytelling, repetitive incantations, and song? The endless repetition of chants and incantations in ritual is a form of self-hypnosis and learning, as much as it is a creation of new inner vistas.

In addition, the evolution of the personal computer created the possibility to let written text be closer to voice than it had ever been before, both in code and in human communication. We are no longer dealing with printed matter, with a silenced or frozen voice, but with forms of writing that are fleeting, time-based, or self-expressive.

It might seem a far leap from song to code, but both are vocal ways of enhancing memory and sharing information. They are in many ways extremes of the vocal spectrum, song being closer to primal vocal expression and code being an expression and extension of calculation and logical reasoning. However, both are also tools, or vehicles, for enhancing and preserving cultural and social expression. Their creation is rooted in the materiality of our bodies and the desire for enabling not just physical procreation, but eternal life through preserving our immaterial selves. To put it in a very blunt and short way: song and code are both means to overcome death. They both represent and prolong our immaterial selves. It seems we have designed machine languages in accordance with the needs of our own fallible, slow, physical memory.

Forbidden Fruit

If we understand programming languages as related to vocal expression, as related to our bodies, and especially as related to song, it might explain why copyright issues are particularly difficult in this area. Copyright developed from the 16th century onwards, first of all for print publications. [11] This which means that it most strongly developed in the area of written text. Copyright for music (sheet music and recorded music) developed much later, especially copyright for popular and folk music. The history of popular and folk music is, even more than that of classical music, positively filled with copies, covers, and quotations. It is largely an oral history, and as I mentioned earlier, oral history works with the limitations and possibilities of the human physical memory by using repetition and quotation. Oral history was a kind of copy-pasting avant la lettre.

The development of programming languages and code has been mostly dependent on group efforts, on shared authorship, and quotation. As such, it has more in common with song than with printed text. Copyrights mostly lean on the definition of authorship. Authorship in code is a complex matter, and it often is impossible to trace code or software back to individual authors. Personal computers and networks have only emphasized and expanded this phenomenon since the 1980s.

In addition, the evolution of the personal computer created the possibility to let written text be closer to voice than it had ever been before, both in code and in human communication. We are no longer dealing with printed matter, with a silenced or frozen voice, but with forms of writing that are fleeting, time-based, or self-expressive. It is therefore not surprising that people want to resist copyrights in the area of digital media networks: we are looking at forms of communication that are much closer to oral, social, or cultural expression and communication than printed text.

Copyright is developing in such a way that programming languages, and specifically the code underlying many pieces of software, are not only turning into illegal or elite knowledge, but start to represent the oppressed, silenced voice. Copyright is not just making words silent, but also forbidden, untouchable, unusable. This is happening at a time when popular culture is becoming infested with high-tech research and development tools and everybody, no matter what level of expertise in code, more or less has access to networks, tools and products of programmers and crackers. The combination of oppression and access produces great tempta-

tions. The turmoil surrounding new media has created new excitement, myths, and mysteries. These are two of the things that make toying with code outside of the machine (in performance, fiction, design or song, for instance) so appealing.

Pop, Explorers, and Privacy

Since the mid-90s, urban clothing has been adorned with printed text referring to cyber culture, especially clothes for men and boys. T-shirts, jackets, coats, and accessories with the words "download," "access," "network patrol," or "worldwideweb" printed on them were all over the place at the end of the 90s, at least inthe Netherlands. There was a techno-fetish boom in pop culture that is only now slowly fading away. The fetishization of new media is of course part of consumer culture, and therefore a product of the advertising frenzy surrounding the Internet and mobile telephony, that began in the early nineties. But it goes beyond that. Most of these clothes and accessories did not have the glossy look of brilliant techno futures to come, but were closer to military or "outdoor" clothing: outfits you wear for conquering the world, a street block, a mountain or when hiking in the woods. They were, so to speak, typical "men's clothes." Baggy shirts and pants with crooked looking prints seem to point more in the direction of hacker culture than the bright new clean future presented in advertising. The message in pop culture seems to be: new media turn us into explorers and warriors. For women, the clothes and accessories market created a far more glossy cyber future, at the height of which was a perfume called "Internet." Nevertheless the strong underlying message in pop culture was: new media are the New Frontier.

The hacker (explorer-warrior) aesthetic is an intrinsic part of new media culture, no matter how much the media industry has tried to suppress it. This aesthetic represents not only the dark and mysterious side of new media, but also simply represents the hidden or inner world of new media culture. I see vocalization of code as a way of affirming or celebrating the power of this inner world, but also of defending the right to "speak" in it, or to criticize it.

Copyright is developing in such a way that programming languages, and specifically the code underlying many pieces of software, are not only turning into illegal or elite knowledge, but start to represent the oppressed, silenced voice.

At the same time, programming languages have created new perspectives. Reading code has created a new type of inner voice. When reading (code or other "text"), we always use the same parts of our brain that control the muscles for speech (which actually seem to start trembling ever so slightly, see footnote 3), but now this process combines with our visual memory and imagination in order to envision the virtual processes slumbering inside the code. Not only does a linear story unfold in text or the "poetry" of programming languages, but while we read, processes and activities also occur visually and imaginatively in our minds, in real time. Code is a new kind of grammar that allows us to represent, suggest, and "see" actions, futures, and "outcomes." This grammar is often hard to follow when one hears it read aloud, but we do not need to do that in order to voice it.

Vocalizations of code so far are mostly symbolic actions, whether artistic or activist, which have little to do with informal, everyday speech. I witnessed the first reading of code, or something very close to it, at "Hacking in Progress" [12] in 1997. One of the events there was a PGP (Pretty Good Privacy, email encryption software [13]) key signing meeting, in which holders of a PGP key were invited to come and identify themselves to a small circle of people. They handed in their individual keys on a sheet of paper to the person leading the "ceremony." When this person read out people's code, they were to raise a hand to identify their personal key. After each key reading, the individual key "holder" was asked to confirm whether the reading was correct. It was late evening and the big tent was empty, except for the PGP key signing party in the middle of it. The atmosphere was serene. It had an almost religious feel to it, and it was very impressive. What this gathering revealed was the core of networking: the people behind the machines and the purpose of those machines. Even if the goal of such a meeting is to ensure that the weakest link in encryption software (humans) is strengthened as much as possible (a technical part of the whole encryption process, one might say), the meetings and exchanges also reveal and empower the human connections served by this software.

Vocalized Code in Art and Activism

Code has been turned into vocals in various ways, and there seem to be four distinctive ways in which code is vocalized: through loose vocal interpretation, through the use of artificial voices, as political pamphlet (forbidden knowledge), or by revealing the human voice in written code. The latter is not at all recognizable as the sound of the physical human voice.

The Slovenian artist Igor Stromajer started singing the HTML code of his website at new media art symposia in 1999, as a reaction against the business-style artist presentation that has become the standard at these events. He printed out the code and just improvised a

melody as he sung it. Stromajer then got permission and funds to do a project in the opera house of Ljubljana, called *Oppera Theoretica Internetika*, a wild mixture of vocal improvisation, news recordings, and reading of code, which also took place in 1999. Ever since, Stromajer has done various operas and ballets in which he performs code. He seems to poke fun at his own ideas of making code physical through song and dance as much as he mocks the seriousness of most media art. The work is humorous as well as critical. As the references to cyber culture in street fashion that are fetishizing new media in a crude way, Stromajer's work is not concerned with a correct depiction of code. These projectsare all about pose, about gesture. They are theatrical and absurd. In his last performance of *Oppera Internetika*, Stromajer did not even use his own voice, but speech software to vocalize the code. The use of speech software in performances is not uncommon. The American poet and artist Alan Sondheim, for instance, has employed speech software for performing a form of duet with a computer, "using the old Mac voices for the code stuff, and paralleling / contradicting them with live voice." In both the projects of Stromajer and Sondheim, the code seems to represent the symbolic order of the computer; it represents the myth, the dark, and the mysterious. Code is not necessarily used in a meaningful way here, but in a manner very similar to the way popular culture deals with new media in general. This is code as glitter or gloom, or code as punk or rock and roll.

Straightforward code readings have a very different purpose. This type of vocalization is closer to a form of enlightenment: code is taken from darkness into the spotlight. There is no mystification here, on the contrary: the content of the code in question is of highest importance. Italian philosopher Franco Berardi, also known as Bifo, made a statement by reading out loud the source code of the "I love you" mail virus. [14] He dedicated himself to this task with great enthusiasm: I strongly recommend you listen to it. The video of this reading was an important part of an exhibition on computer viruses, also titled "I love you" At the Museum for Applied Arts in Frankfurt, Germany. i The reading sounds like poetry, and gains in strength if one considers that this particular virus managed to cause economic damage of about 10 billion dollars and maybe even more harm in personal data loss. The use of the term "virus" for a string of code suggests that natural illnesses can infect your computer, but like common desktop software, computer viruses were designed and written by people. They have been "crafted" to express someone's thoughts, intents, or desires. Viruses, decryption, and encryption software have the potential to destabilize society and are therefore often understood as "illegal" knowledge, as forbidden text. Therefore they are powerful even when they are vocalized through a human body.

The most subtle way of vocalizing code, however, is by letting it speak for itself. By applying the rigorously rational approach of programming and mechanization to emotional human realities, the British artist Graham Harwood managed to voice the unspeakable cruelty of both code and human history at the same time: his work *Lungs* [16] is a Perl poem that calculates the lung volume of people dying under inhuman circumstances. It does so in order to finally "play" the sum total of the lung volumes of all these dead people in one blast through an air raid alarm horn. This project condenses many of the qualities of our "new voice" in one piece: the voice manifests in the new grammar of code; in Harwood's magnificent manipulation of the ancient poem *London* on which *Lungs* is based; and last but not least in the blast of the horn that plays the collected last breaths of people. The human voice has never had so many dimensions.

References:

[1] Gunther Kress, *Literacy in the New Media Age* (Routledge: New York, 2003).
[2] Adrian MacKenzie, *Transductions, Bodies and Machines at Speed* (Continuum Press, 2002).
[3] This seems to be supported by brain research. Desmond Fearnly-Sander's review of Terrence W. Deacon's *The Symbolic Species* (WW Norton & Company: New York, NY, 1997) states: "as you formulate a phrase, one of your brain activities is the motor control of tongue and larynx needed to produce the sounds, and it appears that some shadow of these processes occurs even if you don't actually speak. Processes needed for motor control appear to be used for thinking." http://human-nature.com/nibbs/02/spr.html
[4] Douglas Kahn, *Noise Water Meat, A History of Sound in the Arts* (The MIT Press: Cambridge, MA,1999)
[5] Gunther Kress, *Literacy in the New Media Age* (Routledge: New York, 2003).
[6] http://en.wikipedia.org/wiki/Programming_language
[7] Ibid.
[8] http://bak.spc.org/iod/
[9] Florian Cramer, "Words Made Flesh," Piet Zwart Instituut website, http://pzwart.wdka.hro.nl/mdr/research/fcramer/words-madeflesh/
[10] Ibid.
[11] Nicolas Barker, *Form and Meaning in the History of the Book* (British Library: London, 2003).
[12] HIP97 was a hacker festival organized by people from the former hacker organization Hacktic and the Dutch provider xs4all. http://www.hip97.nl/
[13] http://www.gnupg.org/
[14] http://epidemic.ws/love.mp3
[15] This exhibition was first shown in Frankfurt am Main, Germany in 2002, and in Berlin in 2003. In 2006 it traveled to Novi Sad and Belgrade, Serbia; http://www.digitalcraft.org/iloveyou/
[16] http://databaseimaginary.banff.org/getBioDes.php?id=9&t=3&fc=9