Does Writing Have a Future?

Vilém Flusser

Introduction by Mark Poster
Translated by Nancy Ann Roth

Electronic Mediations
VOLUME 33

University of Minnesota Press
MINNEAPOLIS · LONDON

Originally published as *Die Schrift. Hat Schreiben Zukunft?* Copyright 1987 European Photography, Andreas Müller-Pohle, P. O. Box 08 02 27, D-10002 Berlin, Germany, www.equivalence.com. Edition Flusser, Volume V (2002[5]).

Published by the University of Minnesota Press
111 Third Avenue South, Suite 290
Minneapolis, MN 55401-2520
http://www.upress.umn.edu

Library of Congress Cataloging-in-Publication Data

Flusser, Vilém, 1920–1991.
[Schrift. English]
Does writing have a future? / Vilém Flusser ; introduction by Mark Poster ; translated by Nancy Ann Roth.
p. cm. — (Electronic mediations ; v. 33)
Includes bibliographical references and index.
ISBN 978-0-8166-7022-2 (hc : alk. paper)
ISBN 978-0-8166-7023-9 (pb : alk. paper)
1. Writing—Philosophy. 2. Written communication. I. Poster, Mark. II. Title.
P211.F68 2011
302.2'244—dc22
2010030719

Printed in the United States of America on acid-free paper

The University of Minnesota is an equal-opportunity educator and employer.

17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

For Abraham Moles,
who discovered and began to research writing
after writing.

Introduction

Writing, in the sense of placing letters and other marks one after another, appears to have little or no future. Information is now more effectively transmitted by codes other than those of written signs. What was once written can now be conveyed more effectively on tapes, records, films, videotapes, videodisks, or computer disks, and a great deal that could not be written until now can be noted down in these new codes. Information coded by these means is easier to produce, to transmit, to receive, and to store than written texts. Future correspondence, science, politics, poetry, and philosophy will be pursued more effectively through the use of these codes than through the alphabet or Arabic numerals. It really looks as though written codes will be set aside, like Egyptian hieroglyphs or Indian knots. Only historians and other specialists will be obliged to learn reading and writing in the future.

Many people deny this, mainly out of laziness. They have already learned to write, and they are too old to learn the new codes. We surround this, our laziness, with an aura of grandeur and nobility. If we were to lose writing, we say, we would lose everything we owe to such people as Homer, Aristotle, and Goethe, to say nothing of the Holy Bible. Only how do we really know that these great writers, including the Author of the Bible, would not have preferred to speak into a microphone or to film?

But laziness doesn't explain everything. There are people, and I count myself among them, who believe that they could not live without writing. And this is not because they want to imitate Homer, for they know that no one can write as he did anymore, even a second

Homer; rather they believe that writing is a necessity because their being is expressed in, and only in, the gesture of writing.

Of course, they could be wrong. But even assuming that they are right and that the production of video clips does not suit their being, their *forma mentis,* it would not prove that their form of being has become obsolete, that such people have become dinosaurs. It's true that not everything obsolete is necessarily expendable. What is called progress is not necessarily the same thing as improvement. Dinosaurs were very nice animals in their way, after all. And yet the insistence on writing is becoming questionable today.

The question is, What is distinctive about writing? What sets it apart from comparable gestures of the past and future—from painting, from pressing on computer keys? Is there anything specific at all that is shared by all kinds of gestures of writing—from the chiseling of Latin letters in marble to the brushing of Chinese ideograms on silk, the scratching of equations on boards, or the pounding on the keys of typewriters? What sort of life did people have before they began to write? And how would their lives look if they abandoned writing? All these and many more questions would obviously concern not only writing itself but also the reading of what is written.

These are simple questions only at first glance. A comprehensive book would be required just to grasp them all. But the crux of the matter is that such a book would be a book. Instead of what? That is the question.

Instructions

One way to anticipate the kind of thinking that characterized the informatic revolution is to observe those who manipulate the apparatus, setting the new signs into electromagnetic fields. The word *program* is the Greek equivalent of the Latin *praescriptio* and the German *Vorschrift.* Are these people continuing to write or starting again? Are contemporary reactionaries on the mark when they assert that nothing has changed fundamentally, that the essential always stays the same? To whom are these people writing? For they are not writing past a conclusion to another human being. Rather they write with and for apparatuses. Didn't the earlier discussion show that writing to other people was the essential thing about writing? So the essence of writing has changed for these people; it is another writing, in need of another name: programming. For reactionaries, this is not just uncomfortable; it is terrifying.

From a certain angle, such terror in the face of the new appears harmless. So people don't write alphabetically anymore but rather use other, so-called binary codes. Artificial intelligences are too stupid (perhaps only for the time being) to be able to decode letters. The new computer codes are in fact extraordinarily simple (as simple as artificial intelligences), but it is not simple to use them. They are structurally simple and functionally complex systems. Most of us have not mastered them; on the other hand, we have all learned the alphabet, and print has resulted in a comprehensive, democratizing literacy. The new computer codes have made us all illiterate again. A new literate caste has arisen. For most of us, the new writing (computer programs) is suffused with that kind of

mystery that surrounded alphabetic writing before the invention of print. What cannot be decoded is a frightening secret. People fall to their knees *(supplex turba)* and try to appease it (the Golden Calf before the two tablets). Of course, nothing could be easier than to penetrate the mystery. One has only to learn the secret codes (in the case of the Romans and Jews, the alphabet; in our case, computer codes). But that is exactly what our fear of the new makes impossible. Learning it is child's play only for our fearless children. We have to try other things. We have to try to use a typographic way of thinking to get to grips with post-typographic "writing." Since this essay is literal, it will try to dispel a terror of programs.

If a *program* is to be understood as writing directed not toward human beings but toward apparatuses, then people have been programming since writing was invented—before there were any apparatuses. For one wrote to human beings as though they were apparatuses. One prescribed models of human behavior, and these instructions constitute a prominent thread in the advancing discursive mesh we call Western literature. Using this thread to guide us in a survey of Western history, the development can be represented as follows: at the beginning, since the Stele of Hammurabi, these instructions were called "commandments"; then, with the Twelve Tablets, they became "laws," which later branched out into decrees, regulations, and other forms of instruction; during the Industrial Revolution, instructions were added that pertained to people's behavior toward machines, or "user's manuals"; until finally, since the informatic revolutions, the program discussed earlier—namely, instructions to machines—completed this development. Programs are not only a completely new way of writing, they are also the culmination of a pattern established when writing began.

The thread of instructions just described (and with it the history of the West that is articulated in it) can be understood in various ways, for example, as a tendency toward desacralization. The commandments (say, the Ten Commandments) were holy. They had a heavenly author. It was a superhuman authority that made human

beings into marionettes (apparatuses). The laws (say, constitutional law) had, if not a heavenly, at least a mythical author (e.g., the people), and this mythical authority manipulated the people's behavior. It became more and more clear subsequently that instructions were made by people manipulating other people. User's manuals revealed that all instruction seeks machine-like, automatic human behavior. This is why user's guides are shorter the more automated the machine, until, with fully automatic machines, they become superfluous. In their place are the programs. Here no human beings require instruction. Instructions can instead be issued to apparatuses. In this way, it becomes clear that the goal of instructions (and of Western history) has been completely profane behavior and that, when this goal is achieved, it is superfluous to instruct people at all or to manipulate them. They behave as they should automatically.

The thread of instructions can be read equally well as a tendency to devalue behavior, to reduce it to an object of scientific study (should "science" be understood as value-free thinking and action). The commandments prescribe behavior according to eternal values, the laws, behavior consistent with high values. Subsequent instructions tend to become value-free, until finally, user's manuals apply to functional behavior only. So it concerns a depoliticization and functionalization of behavior, which can be read from the syntactical construction of instructions. They change from imperative propositions ("thou shalt") to functional *if–then* propositions. The commandment "thou shalt honor thy father and mother" becomes advice for use: "If you want to eat chicken soup, do this and this with the tin of chicken soup." This steady devaluing of behavior concludes with programs. In logically constructed computer programs, there is no symbol for *should.* Accordingly, it becomes clear that the tendency of instructions (and of Western history as a whole) is toward a complete depoliticization of all behavior and that when this goal is achieved, human beings and their society will steer themselves automatically, like a cybernetic system.

These two readings of the tendency inherent in instructions convey some sense of the rising functional way of thinking. It is a profane, value-free thought. It can no longer be grasped in historical, political, or ethical categories. Other cybernetic, computable, functional categories must be applied to it. For this reason, programming cannot actually be called writing. It is a gesture that expresses a different kind of thought.

The question remains whether the effort undertaken earlier to demystify programming has dispelled the terror. Of course, the matter can be approached from an optimistic point of view. Because programs instruct apparatuses, the burden of instruction shifts from human beings to inanimate objects, and human beings become free to behave as they like. From this standpoint, the tendency inherent in instructions and culminating in programs is aimed at freedom. Apparatuses behave better and faster than human beings: they assemble automobiles better, they sew better, dig better, and soon will be able to do their cherry-picking more efficiently. And they think better too: they calculate, draw, and make decisions faster. (They are, curiously, better at calculation than they are at cherry-picking.) From now on, people can concentrate on programming apparatuses. Could that not be the freedom we have sought since history began?

Two quite different kinds of objections come to mind. The first one, close at hand, is fairly easy to dismiss. It is concerned that some behavior cannot be taken over by apparatuses and that the sort of behavior that cannot be automated is exactly the sort that constitutes human dignity, for example, the commandment to "honor thy father and mother." That is an error. All modes of behavior, of any sort, can be programmed and automated. It is a matter of breaking the behavior down into its constituent elements, into actemes, and then computing them back together again. Just such breaking down and recalculating is what programming is. The commandment mentioned earlier can be broken down into actemes

such as "feed your bedridden mother rice pudding." Apparatuses will obey this commandment better, more quickly, and more precisely than human beings do.

The second hesitation to be optimistic weighs more heavily. It is concerned that freeing people from the obligation to behave in particular ways will result in a complete lack of freedom. If there is no necessity to act in a particular way (to work, to walk, to sit, to calculate, to draw), then all behavior will revert to an *acte gratuit,* a meaningless, absurd gesture. This objection assumes that freedom can open up only in the struggle against necessity. Completely unconditioned behavior is no more free than completely conditioned behavior. At this point, an optimist might object that any human behavior, whether compliant with instructions, is absurd in the face of death (the inevitability of death) and that the underlying intention of all instruction was always to give this absurdity a meaning. When instructions are shifted from human beings to apparatuses, human beings are free to give meaning to the absurd behavior of apparatuses (and in so doing, to their own behavior as a function of the apparatuses). Accordingly, to program is to give meaning, and the intention behind programming is to free human beings to give the world meaning and to make their lives in it free.

Maintaining an optimism that dispels our fear of programming, one could claim that with the demise of writing through programming, the goal of history is achieved. All behavior has become profane, scientific, functional, apolitical, and people are free to give such behavior meaning. History, and the mode of thought that produces history, is over. A new, posthistorical mode of thought is arising that assigns meaning to absurdity. Let us leave aside the question whether this optimism actually satisfies all conditions. Even if we do accept it, the question whether programming will render all writing obsolete remains open. All instructions can be programmed, but things other than instructions will be written. Literature does not consist wholly of commandments, laws, and

user's manuals, after all. And these other threads in the literary mesh may well not be programmable. So writing will continue after all. And by means of this sustained writing, historical, political, ethical, and aesthetic modes of thought will be preserved.

This (reactionary) objection proves to be an error. It is true that literature does not consist exclusively of instruction, of models of behavior. There are also models of knowledge (e.g., scientific and philosophical texts) and models of experience (e.g., poetry and everything understood by belles lettres). Dividing literature into models for behavior, knowledge, and experience follows the classical division of ideals into good, true, and beautiful, a division that has been insupportable since the Industrial Revolution. Today we have a way of reducing models of knowledge and experience to models of behavior by tracing all propositions back to *if–then* propositions. Propositional calculus permits all statements of whatever kind to be translated into functions. All literature becomes programmable.

A programmed literature would take all texts back to instructions so as to then be computed by artificial intelligences. Even judging by the synthetic images that are already available now, it is clear that exceptionally effective models of knowledge and experience can be produced in this way. As binarily, digitally coded models of knowledge illuminate the screen, from simple statistical curves to complex representations of whole theories, they put all scientific, alphanumerically coded texts in the shade. So-called computer art is just beginning to generate models of experience (fantastic, impossible configurations) that are in fact images, but images that rely on digitally coded programs that are themselves transcodings of alphanumerically coded texts. These remarkably powerful models of experience should be seen in the first instance as programmed poetry and fiction, and only then as "visual art." In this way, an optimistic perspective on the programming of all writing seems justified: if alphabetic writing is to be replaced by digital programming, then all the messages, texts, behavior,

knowledge, and experience that were once mediated by texts will be transmitted more effectively and more creatively through the new informatic media.

But we should not let ourselves be swept up by this optimism. Much would be gained by the programming of everything that has been written alphanumerically until now, but the terror of reactionaries cannot be dismissed so lightly. For in the recoding from alphanumeric into digital codes, something would be lost that not only reactionaries may acknowledge as the critical value of writing. For spoken language would lose its position as mediator between thinking and writing. Digital codes are ideographic in the sense of making concepts (ideas) visible. They differ from the alphabet in signifying no spoken sounds. In programming what was formerly alphabetically written, thought will have detached itself from language. And that is terrifying.

Writing, as we learned it in school, is a gesture of historical consciousness. Programming, as our children are beginning to learn it, is a gesture of a different sort, a gesture better compared to a mathematical than to a literary consciousness. The codes it uses are as ideographic as numbers. Wittgenstein, in his remark on the meaninglessness of saying "two and two is four at six o'clock in the afternoon," showed that mathematical thought is unhistorical. But until now, mathematical thought has been organically immersed in alphanumeric code and swept along in the flow of historical thought. Now programming is rising up from alphanumeric code, becoming independent and separating itself from spoken language. That justifies a degree of pessimism.