

Wallace L. Chafe
University of California, Berkeley

Recently there has developed a great deal of interest in the differences between written and spoken language. I joined this trend a little more than a year ago, and have been exploring not only what the specific differences are, but also the reasons why they might exist. The approach I have taken has been to look for differences between the situations and processes involved in speaking on the one hand and writing on the other, and to speculate on how those differences might be responsible for the observable differences in the output. What happens when we write and what happens when we speak are different things, both psychologically and socially, and I have been trying to see how what we do in the two situations leads to the specific things that we find in writing and speaking.

I occasionally interact with the UNIX computer system at Berkeley, for various purposes. In the context of my concern about differences between writing and speaking, I have begun to wonder whether the kind of communication we are used to receiving from computers is more like writing or speaking. You may think that computers obviously write to us. They send us messages that we can read off of a cathode ray tube, or that get printed out for us on a piece of paper. In that respect what computers produce is written language. But it comes at us in a way that is very different from the way written language usually does. Usually we are faced with a printed page on which the writing is all there, and has been there for a long time. The temporal process by which the writing was put there has absolutely no relevance to us as we peruse the page at our leisure. The timing of our reading is in no way controlled by the timing by which the words were entered on the page. My computer terminal, on the other hand, is steadily chugging away, producing language before my eyes at the rate of 30 characters a second. Under some circumstances I could wait until it had produced a whole page before I began to read. But I don't usually do that. I eagerly follow the steady flow of letters as they appear, just as I would eagerly listen to the spoken sounds of someone who was telling me something I wanted to know. This processing in real time seems in that respect more like spoken language, although what is being produced is written. Furthermore, the computer system and I often, indeed characteristically, engage in quick exchanges, much like conversations, which is not what I am accustomed to doing with written language. So I want to suggest that when it is looked at from the point of view of the dichotomy between written and spoken language, the computer language we normally deal with is neither fish nor fowl. It is produced in written form, but on the other hand it is produced in real time, and we are able to respond and interact as we are not able to do with a printed page.

Recent work seems to have shown that there are a number of features which are characteristic of spoken language, and a number of other features characteristic of written. It is not that spoken language never contains any of the features of writtenness, or that written language never contains any of the features of spokenness. It is only that certain features tend to be associated with one or the other medium, and that the features become more polarized as one approaches the extremes of colloquialness on the one hand, or of literariness on the other. In between one finds various mixtures of literary talk and conversational writing.

In looking for reasons why these distinguishing features exist, I have found it useful to attribute some of them to the temporal differences between writing and speaking, and some of them to the interactional differences.

Temporally, writing as an activity is much slower than speaking. Speaking seems to be produced one "idea unit" at a time, each idea unit having a mean length of about 2 seconds, or 6 words. Every so often a sequence of idea units ends in a falling pitch intonation of the sort we identify with the ending of a sentence. Pauses usually occur between idea units, and longer pauses between sentences. The idea units within a spoken sentence tend to be strung together in a coordinate fashion, typically with the word "and" appearing as a link. There is little of the fancy syntax we find in written language, by which some idea units are subordinated to and embedded within others. It has been hypothesized that speakers' attention capacities are not great enough to allow them to engage in much elaborate syntax. The flow of idea units is enough to keep them occupied. Writing, on the other hand, is peculiar in that the process of writing itself occupies an inordinate amount of time, even though, once we get past the first grade, it doesn't require a great deal of attention. Thus, writers have a lot of extra time and attention available to them, and apparently they often use it to construct elaborate sentences. As a result, whereas the sentences of spoken language have a distinctly fragmented quality, those of written language tend to be more integrated, with much more attention paid to subordinating idea units within others in complex ways. This integration vs. fragmentation dimension seems to be at the root of a number of the features which distinguish writing from speaking.

The other dimension I have been interested in seems to result from the different relation writers and speakers have to their respective audiences. Whereas speakers can interact directly with their listeners, obtaining ongoing confirmation, contradiction, and feedback, writers cannot normally do so, but are constrained to pay more attention to producing something that will stand on its own feet when it is read by someone later on in a different place. We can speak of the greater involvement of speakers, as contrasted with the greater detachment of writers. Many of the specific features distinguishing speaking and writing can be lined up on this involvement vs. detachment dimension.

How can a computer produce language that is maximally congenial to us humans, given the familiarity we already have with the characteristics of spoken and written language? What kind of human language should a computer simulate, in order that we can process it most easily? And to what extent is a computer able to produce such a simulation?

Let's play with the assumption that we human users would feel most at home with a computer terminal with which we could converse in something resembling human conversation, as close as this can be approximated by a machine which (1) can't yet make satisfactory sounds, but has to write what it says; and (2) doesn't know how to experience involvement with a human being. Let's consider what this machine would need to do to make us feel that we were interacting in something like the way we interact when we use spoken language.

Timing is one of the important factors. Instead of steadily producing letters at the rate of 30 a second, this machine might try producing language as spoken language is produced in real time. That would mean doing it at half the speed, for one thing; 15 characters a second would be about normal for the way we assimilate spoken language, and perhaps the rate at

which we naturally take in information. But we would not want it spitting out one letter at a time at a steady rate, as it does now. That has little to do with the way we take in language, either spoken or written, under normal circumstances. Perhaps it should give us one word at a time, but I think it more likely that we would feel most comfortable with syllables: syllables timed to simulate the timing of syllables in normal English speech. Roughly speaking, stressed syllables would be longer and unstressed syllables shorter. A careful study of the timing of natural speech could introduce more sophistication here. At the end of each idea unit -- on the average after every 6 words -- there would be at least a brief pause, signaling the boundary of the idea unit and allowing time for processing. At the end of a sentence -- on the average after every 3 idea units -- the pause would be longer, and paragraph boundaries would be signaled by longer pauses. Idea units would be relatively fragmented. Many of them would be connected by "and," and there would be little of the elaborate syntax one tends to find in written language.

As for involvement, the computer would need to learn that humans are imperfect recipients of information, and that redundancy and requests for confirmation are among the important devices to be used frequently in communicating with them. Frequent direct reference to the addressee is another feature of involvement that the computer could easily learn to use.

My terminal recently told me the following, at 30 steady characters per second:

The "netlpr" command, when executed between computer center machines, now sets the ownership of net queue files correctly so that "netrm" will remove them and they are listed by the "netq" command.

While this is reasonably good written language, and comprehensible as such, I am asking whether meaningful linguistic interaction in real time might not better proceed something as follows, where you can imagine syllables being timed as they are timed in spoken English, brief pauses at the ends of lines, and longer pauses where I have double-spaced (T is the terminal and U the user):

T: Want to know about the "netlpr" command,
where you type in "netlpr"?

U: Sure.

T: You can just use it between computer center
machines,
OK?

Only if you're up here.

U: Yeah,
I know.

T: OK.

It'll show you who owns net queue files,
if you want to know that.

You can use "netrm" to get rid of them,
and you can get them listed with "netq".

That clear?

U: Yeah.

One problem with this is that the user has to type at his or her normal typing rate, which will inevitably be much slower than speaking. But even so, the fragmentation and involvement which make this machine's output more like spoken language might significantly

increase the user's comfort and comprehension. To know whether that is really true calls for further detailed research on the features which distinguish spoken from written language, and tests of whether the introduction of such features into computer language indeed makes a difference. Such research ought in any case to be rewarding beyond the bounds of this particular application.