

On the foolishness of "natural language programming".

Since the early days of automatic computing we have had people that have felt it as a shortcoming that programming required the care and accuracy that is characteristic for the use of any formal symbolism. They blamed the mechanical slave for its strict obedience with which it carried out its given instructions, even if a moment's thought would have revealed that those instructions contained an obvious mistake. "But a moment is a long time, and thought is a painful process." (A.E.Housman). They eagerly hoped and waited for more sensible machinery that would refuse to embark on such nonsensical activities as a trivial clerical error evoked at the time.

Machine code, with its absence of almost any form of redundancy, was soon identified as a needlessly risky interface between man and machine. Partly in response to this recognition so-called "high-level programming languages" were developed, and, as time went by, we learned to a certain extent how to enhance the protection against silly mistakes. It was a significant improvement that now many a silly mistake did result in an error message instead of in an erroneous answer. (And even this improvement wasn't universally appreciated: some people found error messages they couldn't ignore more annoying than wrong results, and, when judging the relative merits of programming languages, some still seem to equate "the ease of programming" with the ease of making undetected mistakes.) The (abstract) machine corresponding to a programming language remained, however, a faithful slave, i.e. the nonsensible automaton perfectly capable of carrying out nonsensical instructions. Programming remained the use of a formal symbolism and, as such, continued to require the care and accuracy required before.

In order to make machines significantly easier to use, it has been proposed (to try) to design machines that we could instruct in our native tongues. This would, admittedly, make the machines much more complicated, but, it was argued, by letting the machine carry a larger share of the burden, life would become easier for us. It sounds sensible provided you blame the obligation to use a formal symbolism as the source of your dif-

faculties. But is the argument valid? I doubt.

We know in the meantime that the choice of an interface is not just a division of (a fixed amount of) labour, because the work involved in co-operating and communicating across the interface has to be added. We know in the meantime --from sobering experience, I may add-- that a change of interface can easily increase at both sides of the fence the amount of work to be done (even drastically so). Hence the increased preference for what are now called "narrow interfaces". Therefore, although changing to communication between machine and man conducted in the latter's native tongue would greatly increase the machine's burden, we have to challenge the assumption that this would simplify man's life.

A short look at the history of mathematics shows how justified this challenge is. Greek mathematics got stuck because it remained a verbal, pictorial activity, Moslem "algebra", after a timid attempt at symbolism, died when it returned to the rhetoric style, and the modern civilized world could only emerge --for better or for worse-- when Western Europe could free itself from the fetters of medieval scholasticism --a vain attempt at verbal precision!-- thanks to the carefully, or at least consciously designed formal symbolisms that we owe to people like Vieta, Descartes, Leibniz, and (later) Boole.

The virtue of formal texts is that their manipulations, in order to be legitimate, need to satisfy only a few simple rules; they are, when you come to think of it, an amazingly effective tool for ruling out all sorts of nonsense that, when we use our native tongues, are almost impossible to avoid.

Instead of regarding the obligation to use formal symbolisms as a burden, we should regard the convenience of using them as a privilege: thanks to them, schoolchildren can learn to do what in earlier days only genius could achieve. (This was evidently not understood by the author that wrote --in 1977-- in the preface of a technical report that "even the standard symbols used for logical connectives have been avoided for the sake of clarity". The occurrence of that sentence suggests that the

author's misunderstanding is not confined to him alone.) When all is said and told, the "naturalness" with which we use our native tongues boils down to the ease with which we can use them for making statements the non-sense of which is not obvious.

It may be illuminating to try to imagine what would have happened if, right from the start, our native tongues would have been the only vehicle for the input into and the output from our information processing equipment. My considered guess is that history would, in a sense, have repeated itself, and that computer science would consist mainly of the indeed black art how to bootstrap from there to a sufficiently well-defined formal system. We would need all the intellect in the world to get the interface narrow enough to be usable, and, in view of the history of mankind, it may not be overly pessimistic to guess that to do the job well enough would require again a few thousand years.

Remark. As a result of the educational trend away from intellectual discipline, the last decades have shown in the Western world a sharp decline of people's mastery of their own language: many people that by the standards of a previous generation should know better, are no longer able to use their native tongue effectively, even for purposes for which it is pretty adequate. (You have only to look at the indeed alarming amount of on close reading meaningless verbiage in scientific articles, technical reports, government publications etc.) This phenomenon --known as "The New Illiteracy"-- should discourage those believers in natural language programming that lack the technical insight needed to predict its failure. (End of remark.)

From one gut feeling I derive much consolation: I suspect that machines to be programmed in our native tongues --be it Dutch, English, American, French, German, or Swahili-- are as damned difficult to make as they would be to use.

Plataanstraat 5  
5671 AL NUENEN  
The Netherlands

prof.dr.Edsger W.Dijkstra  
Burroughs Research Fellow