

The Limitations of Language

BY MICHAEL NORTON

This month, *Technical Support* magazine introduces a new and slightly different column, "Evolutions," that will focus on processes, or, more accurately, the process of the processes that dictate our computing milieu. This column explores how we got here from there, and how we're going to get there from here.

"Evolutions" is a slightly different way of thinking about what we do for a living, giving a different meaning to the detail and tedium we face every day. The intent is to force the technician to unfocus his or her eyes for a moment from the endless minutiae of technical work to glimpse inter-relationships, perhaps even a "bigger picture." However, it must be stressed from the onset that "Evolutions" is not a proclamation; rather each individual column is like one of the proverbial blind men attempting to discern the essence of an elephant: One is convinced it is the trunk, another the ears, and so on. While this approach is certainly not the deductive logic with which we are familiar, the fact is that is how things really happen in the computer world. We are all specialists with limited vision attempting to connect our specialties with other specialties. Consequentially, induction is one of the fundamental mechanisms of IS evolution.

We'll discuss these and other concepts at length in the months ahead. Like Darwin aboard the HMS Beagle, we'll examine an amazing variety of computer phenomena that will challenge us to look beyond our parochial assumptions. So, with that hope, let's set sail.

It is accepted as axiomatic that language is the distinguishing characteristic of man and that his sophisticated use of symbols is the most cogent explanation for the rise of technology and, consequently, his pre-eminence in the evolutionary chain. Using

words, man has carved out civilization, and with mathematical symbols he has made his way to the moon. However, what often goes unrecognized is that language is every bit as restricting as it is liberating; the symbols we share are mainly limitations on how we perceive the world as well as what kind of world we can create. Racial and sexual slurs are glaring examples, but the phenomena is inherent in all language and thus present in all our symbols, including those we use in the computer world. An obvious example is the term "Personal Computer."

**... what often goes
unrecognized is that language
is every bit as restricting as
it is liberating; the symbols
we share are mainly limitations
on how we perceive the world
as well as what kind of world
we can create.**

Before the IBM PC, the machines we now call "Personal Computers" were labeled a variety of names: hobby computer, home computer, microcomputer, etc. The adoption of the IBM architecture as a standard for clones proliferated PCs to the point the term has become generic. People use the acronym PC much in the same way they ask for a Coke when they mean a soft drink or Kleenex rather than tissue. The point is that while the term "Personal Computer" was originally a marketing phrase, it has now ultimately affected the architecture and

design of a particular type of machine to the extent that a new animal, the Network Computer, has had to evolve to take advantage of the realities of a suddenly networked world. There is no debate, however, about whether this evolution is simply a matter of semantics.

So what, exactly is a PC? Although it is well defined among computer professionals, lay people make no distinction between IBM PCs and Macintosh machines. Why should they? The external appearance is similar; they both have monitors, disk drives and keyboards. The distinctions are internal, of course, and thus, are significant from a purely technical point of view. However, it is important to remember that as computer professionals we sometimes make distinctions others do not. I've read that Eskimos have a hundred words for snow. Computer professionals have PCs, Macs and now NCs.

Why is this important? Because the industry infighting is beginning to spill out into the streets. Until very recently, most people in the "real world" did not even know who Bill Gates was. Due to the Internet phenomena and an economic upswing that has allowed more people to place computers in their homes, such ignorance is rapidly changing. Outside the office I actually heard more conversations about the Justice Department action against Microsoft than I did in the office. Unfortunately, the conversations weren't well informed, and the lone newspaper account contained some blatant technical errors that only confused the issues. These issues tend to be difficult for lay people to understand.

And that is the point: For the most part the decision-makers are these lay people, not the technicians. They are non-technical, a fact with enormous implications for information services. The question may be

legitimately asked, then, whether IS managers are technicians or businessmen? So what is a “technician?” Even among technicians there is disagreement as to what constitutes one. Systems people condescend to application programmers who peer down on support personnel who lampoon users who (gasp!) occasionally have more technical expertise than anyone in the IS department — certainly with PCs. The formerly hard-coded technical hierarchy is beginning to break down as technicians become users and users become more technical. A mainframe application programmer may very well seek out the generation X misfit who hauls out the trash to run CHKDSK against the disk drives on his PC — except it’s not CHKDSK anymore, it’s SCANDISK. In this new world order, what are the requisite qualifications for IS management (in addition to a passion for Dilbert)?

This brings me back to the issue of language. Philosopher Ludwig Wittgenstein noted that language has meaning only in context of a “language game.” “Touchdown” means nothing in baseball, for example, but everything in football. It is important to remember that while our IS jargon only has meaning within the IS “game,” these same words are used in other disciplines with variant meanings. Consider the current dispute over the words “integrated” and “separate” in the DOJ action against Microsoft. Millions of words will be bandied about to reach a consensus on the meaning of

those two words in the computer world. Interestingly, those same two words were focal points in an entirely different struggle in the ’60s — the struggle for civil rights.

Did the word “evolution” exist before Darwin’s infamous Origin of the Species? Certainly, and a cursory examination of any dictionary will reveal that what we think of when we hear the word “evolution” is not what an eighteenth century man would have understood. The word “evolution” itself has evolved from its origin as an “unfolding” or “rolling out” to the hierarchical “changing over time” connotations that the word conjures for us today. The impetus behind the evolution of the word “evolution” was Darwin’s theory and the way it changed the game. And it is precisely this borrowing of words between games that constrains the concepts underlying our technology.

Take, for instance, the humble Desktop. User interface designers strove to emulate the physical work environment; thus we actually have a trash can icon sitting on our Desktop. Why? Because it was easier for new users to learn how to use these new contraptions called PCs if the user interface modeled the reality they had always known. Deleting a file required learning a new paradigm; dragging an iconic representation of the document to a trash can seemed so much more natural. But what about the new generation whose first experience with a computer was as a toddler playing games and who will never truly experience the

paper shuffle their parents took for granted? What artificial limitations will “Desktops,” “trash cans” and “folders” impose on the next generation of user interfaces?

Who can say? Every generation struggles against the chains of language until one link breaks and a new paradigm emerges. “Computerese” will evolve as well, with one of the most fundamental forces shaping the change being the increasing need to communicate technology with a semi-technical world. As our terms become more “popular,” of course, they become more rigidly defined to allow such communication, and the entire process is repeated; but such are the limitations of language. 



Michael Norton is the network administrator at Softouch Systems, Oklahoma City, Okla., which provides both mainframe and PC software solutions. He has written mainframe manuals in addition to articles for a number of publications. Michael can be contacted at mnorton@softouch.com.

©1998 Technical Enterprises, Inc. For reprints of this document contact sales@naspa.net.