

TANDY PRIZE LECTURE to the National Science Teachers Association

Las Vegas, NV -- 1998 April 17

"National Danger -- Disappearance of Human Science Teachers" By Bob Bemer

I have a simple message that I'd like you to believe and take away from this meeting:

The value and importance of the qualified teacher to our civilization becomes ever higher as the public is beguiled into the belief that computers can think for them. They cannot. Every student is different, but computer programs are the same, and we have not yet found how to adapt them to each student as a human teacher does.

In the hierarchy of teachers for our complex world, no teachers are as critical as teachers of mathematics, science, and logic. All human survival has depended upon logic, reasoning, and knowledge. And the study of the sciences enhances logical decisionmaking.

I admit that popular opinion does not reflect this. Many of you aren't paid like professional athletes. But then you never have been, as I can attest. My father being a mathematics teacher, my childhood was not noted for its extravagance.

Instead, you must accept the honor given you today, for your skills in imparting knowledge of aspects of the field of science, most commonly to younger people. As I understand the Tandy Prize, you are being honored not so much for your own knowledge, or for your own developments, as for your skill and dedication and success in passing that knowledge on to others. I do certainly commend the Tandy Corporation for starting and continuing the fine tradition that brings you here today. I applaud the motivation of these awards.

With that said, I'll get to the subject of my talk. You may have laughed at its title, about a possible disappearance of science teachers, even as knowledge increases. But I'd like to examine that eventuality in a scientific way, by considering you as a species, if only for reference purposes. Varied in attributes, perhaps, but a species.

We know that species arrive, and species die out. As you are already here, let us consider the dying out part. Some possibilities are:

NATURAL DISASTER

We can probably rule this out. Not all of you live in California! ---- And anything that would be worldwide would not affect only science teachers.

WANTON DESTRUCTION BY OTHER SPECIES

Such behavior is relatively rare, so we might dismiss this possibility as well.

DESTRUCTION BY OTHER SPECIES AS FOOD

Science teachers may have good taste, but I have never heard that they taste good. So you're probably safe.

LOSS OF FOOD SUPPLY

As it is likely that few of you are farmers, your food and other needs are purchased. That's done by money. Money that mostly comes directly from the public, or indirectly through business and government. Money they give you for the reason that most work is distributive. In other words, few want to be bothered teaching their own children, even if they know enough. It's just more efficient.

And this is likely to go on, UNTIL --

REPLACEMENT BY OTHER SPECIES

(contending for the same resources, doing it better)

What other species could possibly replace you? If you haven't noticed, computers. It started early on.

Many years back Dr. Joseph Weizenbaum of MIT wrote a little program called "Eliza". From another room it would ask questions, and then give canned dialogue fragments in reply. It was meant as a joke, actually to debunk the superstition that computers could think.

But to Dr. Weizenbaum's great surprise and disgust, the psychiatry profession took it up as a marvelous breakthrough in patient treatment. Papers were written about it in scholarly journals. PhD theses appeared on improvements. It did cause Weizenbaum to write another book, debunking computer thinking. And it made me wonder why, if it were feasible, psychiatrists would want to obsolete themselves.

For another example, have you seen the recent Microsoft TV spot (I first saw it on April 05)? It touts:

"Knowledge you can't get any other way".

I've had a hard enough time in my life arguing that data is not equivalent to information. If you don't have the program to interpret the data you don't have information. Guess whom I argued that with in the Halls of Congress? Dr. Fred Whipple, the astronomer. He did admit I was correct, but he just had all that unread data.

And I submit to you that information is not knowledge without the programming to interpret it! I also think that human teachers are the best vehicle to do that! I still like the old definition of a teacher as someone on the other end of a log from a student.

The problem is that the public is a market operation. A body that won't provide funding unless convinced that it is getting its money's worth. Can you prove that you can teach better than computers can? Or can you at least prove that you make a more effective team than computers alone?

This is the same public that does not understand fully that computers may be used to yield both good and bad results. The very same public that believes computers to be infallible, and so has agreed to give over much of its decisionmaking to those

computers, forgetting that they are programmed not by gods, but by just us plain folk, who sometimes are not very trustworthy!

Pogo Possum said:

"What we need are more expert experts
than the experts we've got." and also

"The depths of human stupidity have not yet been plumbed."

Those phrases apply to many programmers. That's sure! So let's ask some questions about this upstart species called computers.

In answering, let us remember that computer programs do not by themselves have any more virtue, any more force, or any more correctness than the knowledge of their creators, and it foolish to succumb to the notion that they do. The story of the Emperor's New Clothes is very appropriate to the computer business. We often embed our stupidity in computer programs that cannot be overridden, even if and when we find them faulty.

ARE THEY RELIABLE? HARDWARE?

The hardware itself has become remarkably reliable. At the RAND Corporation in 1951 I made a program that did answer-checking by replicating the process mod 99 on the data. If a regular and a mod 99 calculation did not match, it was done over automatically. You could see the hardware deteriorate under your eyes, or as we said, you could see the bug grow. But not anymore.

That is for the materials side. The design side is still subject to faults, and periodically we hear of microchip design flaws. Not so dangerous as earlier cases.

I was one of three supervisors in the computational department of Lockheed Aircraft in 1952. One of the others was in charge of flutter calculations. The IBM 701, the large computer of that time, employed a kind of laissez faire arithmetic that some teachers espouse these days. It didn't exactly get correct answers! The floating point arithmetic unit gave answers of minus infinity when two identical values were subtracted. Not a zero value with a reduced power exponent to indicate loss of significance, mind you, but minus infinity!

The computer program saw no flutter in the wings, but the wings themselves did! The vibration just built up to a point where the metal fractured. And then they fell off the airplane. Hundreds of people flying in Lockheed Electras died. None of them knew, when the aircraft was plunging to the ground, that some engineer had been badly trained.

ARE THEY RELIABLE? SOFTWARE?

I will make the flat statement that software has become less reliable! Possibly because, when we went to COBOL and FORTRAN and such, more people with fewer skills could suddenly learn to use computers. I now wonder if I was wise in having influence in that trend.

When I was Director of Systems Programming for UNIVAC (if any of you still remember that name), we were trying to sell an 1107 to the U.S. Air Force, which had

up until then been using mostly IBM computers. We were quite proud of our FORTRAN compiler. At the Air Force site some existing programs were compiled. One run gave a diagnostic message about entry into the middle of a DO loop. A NONO if there ever was one. The General in charge said that was impossible; they'd been running that program for three years. A programmer was sent to recheck the source program. He came back saying "Sorry, General. Three years of wrong answers!"

ARE THEY ADAPTABLE? COST-EFFECTIVE?

When we talk cost-effectiveness we must necessarily talk downsizing, which is a large part of any push to replace humans with computers. We have a bad example, from among many, in the railroad business.

My wife and I drove from Phoenix to Dallas last month. You're probably aware that Interstates 10 and 20 run fairly parallel to the major railroad trackage. We were astonished to see that only three of the 20+ long trains that we saw were actually in motion.

Suspensions were confirmed by a Web search for stories. Two railroads had merged -- Union Pacific and Southern Pacific. What a paradise for business! First you can replace people by the computer, and then one computer system can do the same work as two.

That is, if their programs and databases correspond! Unfortunately they did not! Downsizing by computer can be fraught with pitfalls.

One would think that when the problem was found, they could get the trains to run temporarily by going back to the system they had before they were computerized, using people again.

What people? The workforce been downsized, but so had the equipment. Like signal towers, where, as Senator Bennett of Utah says, a lost car used to be found by someone reading the physical numbers on the sides of the cars with binoculars. But the towers had been torn down because the computer didn't need them anymore. The checker was gone. The numbers in painted letters gave way to sensor reading, and the people with the requisite knowledge had been laid off or retired.

A civilization now run extensively by computers cannot go back to BC, which now also means "Before Computers".

A constituent of the Senator's lost \$5 million in perishable goods. He was ticked off!

How could this happen? Well, some management is not as smart as they themselves think. All they saw was the lovely profit that could be made. Computers aren't paid highly. No need to plan for contingencies.

You might ask "But how could this happen to me?" Do you think you can't be downsized? Surplussed? Who would do your job if you didn't? Or is it a situation like Dagwood Bumstead returning from vacation to ask who did his job while he was gone, and being told "Nobody"?

RELEVANCE OF THE YEAR 2000 PROBLEM

You probably know that I'm in the software business, striving to ameliorate one of

the biggest boo-boos that mankind has ever made. I am a toiler in the Year 2000 "vinegaryard". Yes, I said vinegaryard, not vineyard. Because the smell is bad, the product distasteful. But it gives a lot of perspective.

So far we've had only a preview of the Year 2000 bug. We'll continue to find the largest chunk of evidence ever assembled against the assumption that computers can be omnipotent and replace people, including you teachers!

The attention to this problem has been much delayed, but recently the fear and awareness has mushroomed. I watch the evidence grow. Frightening growth, even though there are many dunderheads in high places and the press that dismiss the significance or the danger. Let me give you some samples:

On this last March 23, Pat Robertson interviewed Ed Yourdon, a software guru that you in the software area surely respect. Yourdon said there were about 100,000 mainframe computers, 300 million PCs, and 25 Billion embedded systems! Of the 300 million PCs, the IRS has 130,000.

He confirmed the cost to be in trillions of dollars. Citibank, for example, is spending about \$600 million. He detailed 9,000 electric utility plants and 11,000 banks in this country. "Zero of which are ready for the Year 2000 rollover". Three weeks ago, with less than 1 2/3 years remaining until deadline. The US Social Security system has been working on it, they say, since 1991. They still are not ready, and may not be. Actually it's closer to just one year, as the Government fiscal year 2000 starts on 1999 July 01.

Yourdon says he plans to have at least a month supply of cash on hand for that Jan 1st. We should all make plans. If living in the North, how's your fuel supply?

The New York Fed's daily payment settlements are some \$2.5 TRILLION dollars, about 1/3 of the yearly GNP of the US. Remember, "Zero banks are compliant today".

The US Government Accounting Office supports this. They have said that "there was a danger of as many as 700 banks failing" because of this problem.

Just ONE week ago! Charles Rosetti, head of the IRS, predicted collapse if his staff couldn't get the IRS computers fixed for Year 2000. He said "The whole financial system of the United States will come to a halt". To prove that he means it, he asked Congress to delay its bill on IRS restructuring.

"If we don't fix them, there will be 90 million people 21 months from now who won't get refunds."

Check Y2K on the Web. You can get more survivalist literature there than you thought existed.

WHY I AM HERE

Because you teach basic science, you may have interest in how it happened that you have me, not a teacher, but a computer programmer, standing here arguing why you teachers are so important in a world run by computers. Computers do run it, you know. And we can never go back to the earlier time when they didn't!

It started with the Wall Street Journal in 1996. I had read an article about impending difficulties with computer programs where the way the year was represented was shortcut. Sort of like having only two wheels on the odometer of a used car you were thinking about buying, and you didn't know whether it had gone just 97 miles or 100,097 miles.

It happened that "Time and the Computer" was a hot button of mine. I had published an article of that name in 1979 February. In it I warned about the consequences of the folly of implying, rather than representing specifically, the millennium and century values in computers. Unfortunately nobody paid much attention.

And I was interested even before that. In 1969 I proposed a National Computer Year of assessment and planning, somewhat like the previous International Geophysical Year. One of the things I hoped to fix was this very same problem, based upon the newly-issued international time standard that mandated a 4-digit year representation. Might have worked, but President Nixon didn't think computers were that important, and he wouldn't sign such a proclamation.

With these bitter memories of failure, I resurrected a scheme for effectively storing all 4 digits of the year in the same space that the previous 2 digits had used, together with a method of using them in computation, and applied for a patent.

The Wall Street Journal ran the story. Your director Kaye Thornton appreciated the implications this might have in the coming years, seeing correctly that by the time of this meeting the problem would be understood by many more people. And I can promise you that you're going to be more than understanding it. You're going to be feeling it! Soon!

SOME HARSH FACTS

For every computer program you can show me that can adapt to change, I can show you thousands that cannot.

For every program you can show me that conforms to the rigorous methodology of Edsger Dijkstra and company, I can show you a million that do not, that are full of mistakes, and ignore conditions that will arise. These are documented in a way that defeats their reuse and adaptation by others (if documented at all). Even now researchers still seek the truth about what makes a program satisfactory in all ways. We just don't know! And if we did, how would we get everyone to conform to that methodology?

Dr. Edward Yardeni, Chief Economist of Deutsche Morgan Grenfell, runs a wonderful website at:

"<http://yardeni.com>

He has testified often to the U.S. Congress on the Year 2000 problem, faulting the shoddy nature of existing computer programs heavily as a major deterrent to fixing the Year 2000 problem. I quote:

"Software programming is far less disciplined and rigorous than most people realize. Two different programmers can and do write completely different programs that will perform exactly the same task. Programming is more of an art than a science. One of the biggest Y2K headaches is that few programmers take the time -- or are even asked -- to document

the logic of their programs..."

This is the same Yardeni that one month ago raised his percentage chance for a worldwide recession from 40% to 60%. In USA Today, no less. Did you see it? Did you comprehend it? Last week, with the inter-national bankers in Switzerland, he raised it to 100%.

Some of you may teach computer science. What makes you think THAT has reached the level of a science? I've been told that the scientific method makes use of facts and knowledge previously proven. After my 49 years in the business there is still very little reusable code. Even now the likes of Java applets make only a small dent in our corpus of programs.

Sounds very pessimistic, doesn't it? But dangers often provide opportunities and benefits. All effects of the Year 2000 crisis may not be bad. Instead of penalizing our GNP some 5-10% by existing tax rules, we can go to a flat or sales tax way to get revenue to run our government. Forget anyone's wishes, folks. The IRS, due to its Year 2000 troubles, may not survive in its present form.

WHEN AND WHEN NOT TO USE COMPUTERS

A program is a process. Once you think you have the process correct you can turn out copies like cookies. That's both a boon and a danger. Right or wrong, it still gets spread widely, like false Web rumors.

We humans remain corruptible -- by constraints of time and deadlines, by thinking we know the whole nature of a problem when we do not, by believing our program is temporary and fixable later (perhaps by others), by the sheer motive power that computers provide. We revel in how fast it runs, what Intel chip is inside. We don't revel in how correctly it runs, and the usefulness of the work it does.

I stress the way change affects programs because if a program can't twist and turn like a sheepherding dog as the flock veers, it is not trustworthy. Certainly not one program has corrected itself for the Year 2000.

What teacher has a homogenous set of students? Until cloning becomes the norm, that is? Will the student set of 10 years from now be identical to the set of today? Anyone who has followed culture patterns knows that this will not be.

NOT ALL COMPUTERS ARE USED WELL

I have been a computer programmer since early 1949 -- 49 years of programming -- and I still love it. But as I often caution, computers are just a tool. Not gods. And tools can be used for good or evil. Fire is such a tool. I like a fire in a fireplace, and to cook a steak. I don't like arson.

Remember that I'm talking of bad use of computers, not how bad computers are. I'm not blaming computers per se, just their misuse. Banks don't like electronic theft of funds, even if electronic transfer can be a great boon. People don't like it at all if a virus is slipped into their PC. To my mind the worship of the computer as a teaching tool is not justified.

LET'S EXTRAPOLATE BACKWARD

Suppose we had computers like those of today when I was in college. 1937-1940, if

you must know.

Surely their teaching programs would have had the atom as the smallest particle of matter. As far as most knew then, it was. Now the person who programmed that has died or moved to Borneo. Believe me, the program was poorly annotated or explained. Almost all are.

Who will update such an erroneous teaching program everywhere it is used? The press can flash a story worldwide very quickly. Upgrading all existing programs is difficult.

Surely their teaching programs would not have said that volcanoes had the habit of occurring at slice planes tangential to the earth's core. It wasn't known then. Who could understand the old program well enough to add this in context? Or would every thing have to be reprogrammed? Y2K gives us incontrovertible proof of this impossibility.

I've known some inflexible people in my time, but none as inflexible as an old computer program!

WHAT CAN THE YEAR 2000 CRISIS TEACH?

Late in the day we've been surprised to learn that there were only two wheels on the year odometer. And we have found that to be a dangerous mistake, albeit one that I started warning about in 1971.

So who could fix it? People that used only COBOL? Most of them didn't even know that a compiler program changes their COBOL programs to a form the computer understands, in order to run. And if they did realize this, they would have no idea how a computer translated those "object" programs into action at the most basic level. COBOL is seldom taught any more, much less machine language and its principles.

Fortunately for some, but not for all, I and a few others learned the basics. The way a computer really worked on the inside. Of course then we went to COBOL and FORTRAN and such, and more people with fewer skills could suddenly learn to use computers. I now wonder if I was wise in having such a large influence in that.

The basics allowed me to create a method that may allow computerized correction of the problem, as opposed to new people studying programs someone else wrote 30 years ago without documenting the thought and logic processes that went into them. My method may give some relief, but certainly not for the computer chip inside your coffee maker, your automobile, the satellite from which you get your television signals, power plants, or the air traffic control system.

In our trade jargon, there is no "silver bullet". My own method was described in the Wall Street Journal as only a "silver-PLATED bullet".

I mention this because the "silver bullet" form of wishfulness is endemic. It has caused quite dangerous delay in addressing the impending catastrophe. It takes its most virulent form in the current President and Vice President of the United States. They claim that the fast way to better education is to put a PC on every child's desk.

When this happens, I want you all to alert me when you hear the first child saying:

"Wow! Now that I have a computer I can learn chemistry better. Where ARE those chemistry lessons? Point me to them. I can hardly wait. What's the icon for chemistry?"

Being trained without computers, I actually know the multiplication tables by heart, without a hand calculator. So --

If on January 01 of the year 2000 your parents receive their pension checks and their Social Security payments -- if your bank allows you to write a check on, or withdraw, money you know to be yours -- if airplanes can take you to see your children -- if the transport system brings farmers seed, and brings you food and clothing and heating oil -- if the water supply doesn't quit -- if stock markets haven't collapsed completely -- if your house isn't repossessed for nonpayment of the mortgage -- THEN

You can be grateful that I and some few others were not taught fuzzy thinking by computer programs that coddled us, but rather by real teachers that passed along the basics that they had learned. People like you! And like my father, who was so concerned about my education in mathematics that he himself taught the courses I took, in addition to being the school superintendent.

Do I think science and mathematics teachers critically important? Do I think your profession must survive despite computers? Who can doubt that I do?

You're not anachronisms. You're custodians of the human future, and defenders against inexactnesses that plague our lives. Why do I rail against "inexactness"? Aren't "approximately" or "sort of" just as good? If they were, you would all have credit cards with a "00" expiration date on them. But you don't! When your decisionmaking and your lives are given over to computers, you'll just have to accept the fact that very few of them operate via inequalities.

I've been leading up to my main caution, which is -- if you program our computers for our present world, and, knowing that they are running that world, fail to properly educate our children, who will reprogram those computers when new knowledge is found?

The kid who cannot make change without a cash register? The people that don't know how the computer is doing its thing? Even if no new knowledge is found, who can reprogram the computer when the old knowledge is found to be faulty? I certainly would not depend upon computers to reprogram themselves. Maybe in a couple more centuries, but not now.

So again I say, watch these next two years carefully, as those faultily-programmed computers make a mess of things on our way to the Year 2000. And leave here with the determination to teach exact science so that such catastrophes may not again arise.

Forget that IBM's "Big Blue" computer finally bested a human in chess. That was a specialized aberration. Don't let it be used as argument in favor of turning the world over to computerized decisions that are not countermandable. We humans can, and always should, outthink computers.

Reread, perhaps, E. M. Forster's "The Machine Stops". A frightening prediction, even in the Year 1909, of what can happen when people abdicate control of their lives to machines.

Let computers be our students, not our teachers.

Let computers be our servants, not our masters.