

Are We Becoming Addicted to Computers?

by Arthur Fink
Arthur Fink Consulting
Wilton, New Hampshire 03086

(A talk prepared for the ACM 1983 Annual Conference
New York City, October 1983)

Are we becoming addicted to computers? There's a deliberate ambiguity in the "we":

Are we professionals in the computer field becoming addicted to computers?

Is our society becoming addicted to computers?

Are there significant numbers of people in our society who are becoming addicted to computers?

Actually I believe that all three statements are in some measure true, but it will be helpful to examine each statement separately.

Computer Addiction

What do we mean by "addicted"? The Oxford English Dictionary defines it as: "(1) Delivered over devoted, destined, bound. (2) Attached by one's own act; given up, devoted, ... naturally attached bondage, which displaces free will. It is a condition in which really

objective self-examination does not and probably cannot guide our behavior. It is generally regarded as a pitiful state, over which one has little control.

Merely spending a lot of time with something does not by itself signify addiction. Thus although I spend countless hours writing with my word processing IBM/PC, I don't feel addicted to this machine. Writing is creative work for me, and I feel quite able to print out a listing, turn off my machine, and continue working with words and ideas.

But when occasionally I start writing a small program I can find myself locked into an encounter with the machine. Expecting my program to be plagued with trivial bugs, and probably with a more substantial logical flaw as well, I find myself seized with a compelling feeble program and the compiler or interpreter that seems to interfere with my will. It must work, and does.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1983 ACM 0-89791-120-2/83/010/0180 \$00.75

Is there any harm in this? Probably not, in my trivial example. But without an explicit specification of what is needed in a complex system, we may never finish our task of taming the bug infested program that has not yet reached its saturation level of complexity.

We sometimes seem to run in place to keep up with the technology. Do we, at the same time, let go of people, of the real needs of the organization, and of our own sense of balance or of what constitutes appropriate technology? Are we then uncomfortable being called a "computer person" when this happens?

I would suggest that we can cultivate an informed disrespect for the computer, which will help bring about more appropriate applications and also a more satisfying work experience. We can overcome our compulsion and critically examine the social context of our work with computers. And as the computer itself becomes less important, we will find that we can develop more functional and useful systems.

What I'm describing here is the kind of computer addiction that might affect us--computer professionals. The technology is so interesting that it becomes an end in itself rather than a means to attain other goals that are clearly in sight.

As we work developing the technology, does anybody address the real need for our system in terms of information needs; in terms of accountability and control; in terms of power, politics, and budget? Who is able to recommend that in certain cases a computer not be used, or that a system be less comprehensive than originally planned? Is there a

felt need for the system by those who hope for direct benefit from it, or is the main enthusiasm coming from those who see the project as a way to apply the new systems technology?

You might protest that such questions are outside the realm of the professional computer scientist, or even of the applied database designer. It probably is outside our role to take these questions on ourselves. But if the eventual system users are also avoiding these issues, then a second kind of computer addiction is evident.

Computer Deification

Computer deification occurs when people have so much faith or belief in computers that they stop asking the kinds of questions that would let them take control over their own application. Again, the process of computerization becomes an end in itself, and it often appears to supplant critical thinking about the application.

I encountered an extreme example of this in installing a database system to help plan material requirements for a large manufacturing company. The materials manager asked me if we could suppress printout of all computer-generated management suggestions that required action in the first week or two. He was willing to see suggestions that were not so urgent, but didn't even want to know what our program identified as most timely if his staff couldn't respond. Of course the immediate problems have to be solved first, and merely erasing them from the printout would defeat the whole purpose of our system. Rather than think about his material planning problems, this bright gentleman was totally focused on his new computer system

and was seeking a technical solution (or cover) for his management problem.

Computer Mystification

A related attitude is Computer mystification--the fear of anything even vaguely connected with the computer. This attitude leads to the common request that a technical person specify and implement an information system with little user guidance about the real application requirements, and to a surprising acceptance of poorly designed and disfunctional systems. It leads people who really do understand their own needs to believe that "the computer has the answer", while it often reinforces the arrogance of those of us who know the technology and not the application--even though we frequently encounter whole layers that we barely understand.

Because of computer mystification, programmers and analysts often receive almost priestly status. I sit on the regional committee of a nonprofit organization that had experienced serious delays in the in-house implementation of a fundraising information system. Not surprisingly, I was asked to look into reasons for the delay. "You understand computers", my colleagues told me. In fact, I didn't know much about this kind of application or the particular machine and DBMS being used.

When I reported that the problem stemmed from staffing difficulties, probably aggravated by low salaries and a restrictive set of affirmative action hiring guidelines, the committee again said to me "Arthur: You understand computers, so follow up on this." Did anybody notice that the issues I identified were not technical at all?

In all these examples, I wonder why simple common sense seems not to prevail, or even to be very evident. Is there something about the technology, or about popular perceptions of it, that reinforces such notions? Whatever is operating here, I believe that this is how we experience society's addiction to computers.

Computer Literacy

Underlying this addiction is a widespread but rarely stated belief that the computer is the solution, and that the act of programming is a substitute for creative thinking. Most of you know that precisely the opposite is true: designing a good program often requires more knowledge of the subject area than solving the problem without computer assistance.

One of the strongest, and I will venture to say the most destructive, expressions of society's love affair with the computer is the current push for "computer literacy" in elementary and secondary schools. The popular press tells us that unless our children get a sufficient dose of LOGO or BASIC in third or fifth grade, they won't be ready to take their place in the technological society. Schools are spending money on computers now, and have less money available for everything else.

Just hearing that computers are the "wave of the future" or the "key to employment" doesn't convince me that writing programs in BASIC or LOGO or playing video games is more important than some other learning activity. If such computer work helps the children in their learning, then it may be justified as a teaching technique without the added pretense that it contributes to "computer literacy".

Although the popular name for school computer programs is "Computer literacy", I think they should more properly be called "computer proficiency". I've yet to see a definition of computer literacy that really builds on my understanding of regular English "literacy". Just as a literate person can say "That's a hollow argument" or "That's an unsupported assertion" or "That's a beautiful poem", so a computer-literate person should be able to say "That computer output is useless (or worse) in this situation" or "That's an elegant model".

Such literacy goes beyond computer penmanship. Learning to write simple computer programs leads only to a minimal computer proficiency. I've met too many students who can speak and write in a computer language with great fluency--and have nothing to say in that language.

I believe that children (and adults) need to know what is appropriate or inappropriate computer use, that we all need a healthy dose of computer assertiveness training so that there is no doubt as to who is in control, that we need enough understanding of the subject we are working with to recognize instantly when the computer output is absurd, that we need to understand how our problem is represented as a model in the computer, and that we need to know what important aspects of our problem are not represented (perhaps cannot be) in the computer program.

A Social Audit of a Computer Application

Just as environmental impact statements are prepared before undertaking construction projects, social audits are needed to understand possible consequences

before a computer application is begun. Much more inquiry is needed to understand the general implications of our whole reliance on information processing tools which can expand the power of our intellect but don't similarly augment or assist our social or spiritual nature.

Here is the beginning of a list of questions to ask the users, the system designers, and the programmers before a project is begun and as it proceeds:

Do users have enough "feel" for the application that they will clearly recognize bad output?

Are other concerns or problems in the organization more pressing, and less likely to get addressed if the proposed system is built?

Is it possible at critical points throughout the system development project to deliberately delay or terminate the project if that is appropriate?

I'll leave the list for you to complete. I have a longer version in draft form, but know that much more dialogue is needed to complete this project.

A Center for Appropriate Computing

The purpose of this paper is to invite discussion on the social implications of the computer revolution. I've deliberately refrained from discussing most of the well acknowledged issues, such as privacy, physical or psychological effects of working with a video display, etc. These are important, but we have to look deeper to understand what the computer is bringing into our world, and what that demands of us.

In running workshops for computer professionals on "Computers and Personal Values", and in speaking with teachers and other people, I have become convinced that an organization is needed to raise questions about the appropriate ways to use and to regard computers, and to channel creative energy into this area.

Therefore, I am starting a nonprofit Center for Appropriate Computing, with a mission to foster discussion on issues of computers and values. The center will conduct a program of seminars and conferences, distribute papers that we believe will stimulate helpful discussion, develop teaching materials, and publish a newsletter. I hope that we can work closely with existing organizations, including the ACM Special Interest Group on Computers and Society (SIGCAS). And I hope that all of you will have something to say about how most appropriately to work with today's computer technology.