

Opinion:

The Software Vacuum

by
Chris Ryland
25 Follen St.
Cambridge MA 02138

There is a software vacuum. That fact has become increasingly clear in the past few months.

Take a look at the situation. At the time this is written there's only one company (MITS) offering a proprietary software product (BASIC) aimed directly at the microcomputer hobbyists. It's true that the People's Computer Company (PCC), the MITS Altair User's Group, and a handful of other user groups offer access to growing libraries of applications programs. Some microsystem manufacturers also supply "bare bones" system software with their machines, but the user is lucky if these bones consist of as much as a rather bare monitor, editor and assembler. Such software is just as important to a serious hobbyist's system as the hardware. So when I say there is a software vacuum, I mean there is an absence of commercially developed and marketed larger scale software products.

But why should this matter to me, as a hobbyist?

What do I mean by larger-scale, and why should the hobbyist market be invaded by commercial software vendors? In answer, let me give a concrete example. Suppose I want to write a "desk calculator" program for my system. I'd like it to include all the scientific functions of a powerful calculator, such as sine, cosine, tangent, arcsine, the hyperbolic functions, etc. I would feel competent to design and write the user interface (e.g., keyboard input) sections of the calculator, but when it comes to even the floating-point arithmetic (let alone the scientific) functions, I'm totally lost. What do I do? Give up? Scour the different user groups software offerings, piecing together little routines from here and there? Take a course in applied numerical analysis and learn enough theory to do all the programming myself? No. Ideally, I'd browse through several software catalogs from commercial vendors, picking out the best math package for my needs. It would be relatively inexpensive, because of competition among the various vendors. And since the time I save by

buying the package commercially is substantial, I'd consider it to be a larger-scale product (physically small though it may be). Finally, and most important, is the assurance of quality that I get — the programs that I buy should be backed up by a guarantee; if they don't work correctly, I can have the problem fixed by someone who's an expert at that sort of thing. In fact, any problems discovered by other customers of the supplier will be automatically corrected and reported to me.

I'm not pointing out anything surprising, though. What *is* surprising is that this lack of software is no one's fault. Why? The reasons will become clear as we examine the basis and effects of the vacuum more carefully.

The most obvious explanation for the software vacuum is the newness of our field. It's very easy to draw a parallel to the very early days of "real" computers, when every manufacturer was scrambling to produce hardware. The importance of software wasn't recognized, and it was simply left behind in the dust. Since then, however, software has become the major

component, in cost, size, and complexity, of any large computer system, since nothing can be done without it.

The parallel is completed by noting that, now, the big "scramble" in microcomputers is to get out the hardware. People *do* sometimes learn from history, though. Microsystem suppliers have realized that naked machines are close to useless, and many are offering at least the bare bones system software mentioned earlier. The problem here is that many of us want to *use* our home or business system for something (e.g., as a powerful desk calculator) and don't necessarily want to *do all the* programming ourselves.

This situation is eased, for example, by the availability of a BASIC system from one major supplier. Unfortunately, for the "from scratch" hobbyist, the difficulty still remains, since software like BASIC is much more expensive as a stand-alone product (and is usually out of the price range of most of us). Again, since the average hobbyist would be lost when faced with the task of constructing his own BASIC system, he's thereby automatically cut out of the

... I'd like to browse through several software catalogs ... picking out the best package for my needs.

legacy of widely-available BASIC programs.

Now why can't we throw the blame on the CPU chip manufacturers? Because, clearly, their proper market is the OEMs (Original Equipment Manufacturers), and thus their main concern regarding software is to provide their OEM customers with developmental systems of hard and software, leaving the hobbyist nearly completely, if not intentionally, out of the picture.

What about the other potential suppliers of software? Look at the situation of a typical software house, or vendor. They must ask, "What kind of hobbyist market exists? How big is it, and how can it be reached? What kind of return on investment can be expected from a venture in this area?" These are tough questions for a company whose existence depends on successful marketing of software products, even though the answers might seem obvious to us. But keep in mind that the existence of BYTE is among the first indicators of a widespread hobbyist interest in computing, and we're only a few months old.

Furthermore, there's the problem of proprietary programs, a problem hard to appreciate from the hobbyist's point of view. It stems from the softness of software: Hardware can only be physically in one place at one time, but software, because its copying cost is (relatively) so low, can virtually be in many places at once. For example, an institutional computer installation signs a proprietary contract when it buys a program product. Such contracts typically restrict use of the software to one computer system or customer site. This is one of the major reasons the institution is not likely to

lend or sell such a product to other installations. But a hobbyist, even though he might sign a similar contract, is much more likely to help out a friend by swapping a program for others, or by lending a copy to someone else. And this is deadly poison for any sort of commercial market. To take a concrete, if absurd, illustration, suppose you could make *working* Xerox copies of CPU chips — how long would the chip manufacturers be around?

Another simple fact of proprietary-program life is that such programs cost a lot to develop and market. Unless the market is large, this means high prices, which a big computer installation can usually justify, but which an individual just can't afford. It's true that there's a difference in scale between large and micro system software, but both, in their own spheres, are costly to commercially develop and maintain. So, the software vendors may simply not have been able, even if willing, to enter the microcomputer software market.

Continuing with our search for potential software suppliers, we arrive at the grass roots level: The home experimenter who's developed a good product with long hours and minimal tools. Why can't he go it alone? Well, he's usually operating on a small budget and thus can't mount any sort of larger marketing, packaging, or shipping effort. True, you might say, but couldn't he sell his product locally, at a neighborhood level? Yes, he could, but then he's not reaching the majority of people who are interested in his product, namely, *us*. Besides, how can he commercially support any product on even a medium scale, when such support might involve, aside from all the developmental and

promotional work and expense, handling dozens of trouble reports in a single week? He'd no longer be a hobbyist, but a full time one-man software house, and that puts him out of the grass roots class.

So, with our search ended, and no culprit in sight, what can be done about the software vacuum?

First of all, those of us in the personal computing field who have professional contacts can urge existing and potential software vendors to look hard at the hobbyist markets. *When the number of potential users of a package is multiplied by the profit margins to be expected, such software vendors should be economically viable.* Whenever there is a demand for a product, a free market will tend to fill that demand. It will be interesting to see what the market produces in system software for small computers.

Second, hobbyists can help generate commercial interest in this vital area in several ways. One is to make the software vacuum a topic for discussion at local computer club meetings. Another is to organize software trading posts in the newsletters which are very much a part of the hobby. Still another is to write manufacturers urging advanced software products to match the extremely high level of today's hardware technology. If you feel eloquent, write a letter to the editor of BYTE on the subject.

The first step in filling a need is identifying that the need exists. Hopefully these thoughts will start some BYTE readers off in a direction which will lead to commercially marketed mass produced software which can be plugged into one of the several microprocessor architectures which are on the market. ■

Suppose you could make *working* Xerox copies of CPU chips, how long would chip manufacturers be around?

Hobbyists can help generate commercial interest in producing software.
