Versatile packaging: Software for all retail environments

by ELWIN E. LAGES
Ingram Book Company
Nashville, Tennessee

ABSTRACT

With 40,000 programs already, and the potential for expansion of possible retail outlets, the software publisher must package his product in such a way that the retailer can exercise any number of options for shelving and display. The package must be versatile, attractive, and must display enough information to be able to sell itself. In addition, the package has to be able to assist the retailer in terms of security.
INTRODUCTION

Retail shelves cannot accommodate all the available programs, and salesmen cannot learn how to use and demonstrate even a significant number of them. According to P.C. Telemart, of Fairfax, Va., there are more than 15,500 titles and 40,000 software products. In addition, this title base, according to P.C. Telemart, is increasing by the hundreds each month. What this means is that the competition for space is all retail environments is intense.

As a buyer for a distributor, and a former retailer of software, I am familiar with many of the problems that beset all concerned with how to improve software sales. Being in the middle I hear of all the problems from both sides of the retailing fence. There is one problem, though, that seems to be talked about more than any other. That problem is packaging. It seems so obvious. Packaging is a natural facet of product development in all other aspects of retailing, from automobiles to books. It seems odd that software developers, all looking to make big bucks in the burgeoning software market, forget packaging in the rush to get their products out.

In the rest of this paper, one word stands out from all others: versatility. This is the ability of the package in which a piece of software is presented to serve various functions in different retail environments. Today a store owner is inundated with hundreds of products monthly, each claiming to be the premier program of its type available, each claiming to be able to sell itself. Sadly enough, no program is currently that unique that it can afford any negatives if it hopes to be placed on enough retail shelves to be noticed by the buying public.

The first person that must be impressed by a product is not the user, but the distributor or the retailer. Tough choices have to be made since no store can stock 40,000-plus titles and the package, not the program, is the first thing seen. The other sections of this paper will touch upon the new retail environments in which software will be sold in the near future, and the packaging needs of these new and old retailers.

According to P.C. Telemart, of Fairfax, Va., there are more than 15,500 titles and 40,000 software products. In addition, this title base, according to P.C. Telemart, is increasing by the hundreds each month. What this means is that the competition for space is all retail environments is intense.

As a buyer for a distributor, and a former retailer of software, I am familiar with many of the problems that beset all concerned with how to improve software sales. Being in the middle I hear of all the problems from both sides of the retailing fence. There is one problem, though, that seems to be talked about more than any other. That problem is packaging. It seems so obvious. Packaging is a natural facet of product development in all other aspects of retailing, from automobiles to books. It seems odd that software developers, all looking to make big bucks in the burgeoning software market, forget packaging in the rush to get their products out.

In the rest of this paper, one word stands out from all others: versatility. This is the ability of the package in which a piece of software is presented to serve various functions in different retail environments. Today a store owner is inundated with hundreds of products monthly, each claiming to be the premier program of its type available, each claiming to be able to sell itself. Sadly enough, no program is currently that unique that it can afford any negatives if it hopes to be placed on enough retail shelves to be noticed by the buying public.

The first person that must be impressed by a product is not the user, but the distributor or the retailer. Tough choices have to be made since no store can stock 40,000-plus titles and the package, not the program, is the first thing seen. The other sections of this paper will touch upon the new retail environments in which software will be sold in the near future, and the packaging needs of these new and old retailers.
RETAIL ENVIRONMENTS

Just a few short years ago there were comparatively few software packages, and these were largely sold in hardware-oriented computer stores. These stores carried software as supplements to the hardware they carried, and most of the sales efforts were with the hardware.

Today, the retail environments for software are expanding continuously. There are large software-only franchises; bookstores, both chains and independents; such mass merchandisers as Sears, Target, etc.; record stores; video stores; even a recent experiment where software was carried on some newsstands in New York City.

Future Computing, a marketing research organization based in Dallas, Tex., estimated (Figure 1) that in 1983 almost half of all software sales would be in computer specialty stores, but that by 1988 that share would shrink to approximately one quarter. Software specialty stores and mass merchandisers will account for almost half of all software sales.

In the differing markets for home computers and office computers, the potential distribution channels are staggering. Figure 2 shows that potential growth in home computer software. In such outlets as bookstores, the growth potential is phenomenal. Software in correct packaging is a natural product to be distributed in an outlet that is perceived by its customer base as the purveyor of information and entertainment.

Figure 3 shows the potential market for business software. Although not as varied as the home market, the potential growth is many times that of the potential growth of computer stores. Fully one-third of all software sales are going to be in outlets not specializing in computers or their programs. These are outlets where the personnel will not be as fully trained and where self-service is the key. That requires the publisher to prepare a product that has the potential of selling itself.

In computer and software stores the staggering number of products available from an almost inexhaustible number of publishers requires again that the program in large part sell itself. Here, the package is the key. Not demonstrations, brochures, etc., all of which are indeed useful, but the package itself. The package is a retailer's and a potential customer's introduction to the program.

If any publisher is willing to forego selling his product to almost 75% of the potential market, then he is willing to see his competitors enter the market and grab that share uncontested.

THE PACKAGE

... consumer-oriented products that make learning fun, appealing packages, good advertising—are the software vendor's cost of admission into ... the market.

C. David Suess, President
Spinemaker Software Corp.
With hundreds of software packages finding their way into the hands of distributors and retailers alike, how does anyone decide what to stock? For that matter, with such a proliferation of products, how does anyone even decide what to look at? No one has the time, the personnel, or the ambition to examine every product from every manufacturer. This is especially true of the products of new entries into the list of companies trying to get their products into the network. A retailer or distributor will go out of his way to look at the products of an established vendor, and worry about details like packaging later, but the new entry will be placed on the bottom of any pile, anywhere. If the package looks like nothing, the program might never see the light of day.

Too many vendors concentrate on what they see as the "musts" of establishing their products in the minds of the public. So millions are spent on advertising to make retailers and consumers aware of a product and the company name, then this super program is shoved into a baggie or plain brown box, and left to fend for itself. The package is the best advertising currently available. It is the package that anyone sees first, and it is the perceived value as shown by the package that prompts anyone, the distributor, the retailer, or the consumer to go farther and spend more time to find out if the program is all it is advertised to be.

Packaging must be considered as part of the whole. Each facet of the network chain, from manufacturer to consumer, has specific needs. As a distributor, I am required to see every one, but especially those of the retailer. We all have a common purpose: to sell products.

As described earlier, the retail distribution of software is no longer limited to the hardware-oriented computer store, or even the software-only store. At the lower end "consumer" market, the potential retail shelf space is in many environments. Does the manufacturer have to do a package for each environment? The obvious answer is no. Costs would escalate, and the time problem would raise its ugly head. A package, though, does have to be versatile enough to operate adequately in all environments. The package should never be the limiting factor in distribution. A package should never disqualify a product from being sold anywhere.

All retailers have one thing in common: the desire to sell merchandise quickly and with the least amount of problems. With this in mind a package must have four facets: flexibility, appearance, information, and security. The first three help to sell the product, the last helps keep enough of the product in the store to sell.

**FLEXIBILITY**

Two years ago there were so few programs available at retail, that the retailer, usually a hardware dealer, had trouble find-
From the collection of the Computer History Museum (www.computerhistory.org)

386 National Computer Conference, 1984

...ing enough programs to fill available space. Today that is far from the truth. Two years ago, almost all software could be faced out or placed on pegboards with room to spare. Today that luxury is long past. The computer store has to dedicate more room to hardware, which, like software, also has proliferated over the past few years, and cannot afford to have as much space dedicated to relatively low-priced software. The nontraditional outlets (if any outlet only a few years old itself can be considered traditional) cannot or will not give 100% of their shelf space to software. And obviously, there is not a store large enough in any case to stock 40,000 titles.

Forcing any retailer to display a product in only one way is asking too much. Entertainment software is a prime example. Not very long ago, almost all games came in zip-lock plastic bags. They had little holes on the top for display on a pegboard and a store either gave over a large area to games or carried fewer than optimal sales would allow. Why? You can only shelve these bags one way, face out. Placed edge-on, and not only do they all look alike, but you cannot see them in any case.

Today's environment requires a package with more than just a face. In addition, a spine is needed that displays such simple information as title, manufacturer, and compatible hardware. In related retail ventures, books, records, video, etc., you might note that virtually all packages have spines. In the new retail environments, especially bookstores, packages must have the flexibility to be shelved and displayed in any way the retailer wants. No outlets that traditionally have dealt in other products are going to renovate their stores to carry products that they are unsure of to begin with. Unless the manufacturers of the programs are willing to forgo selling in one or more environments they must offer the retailer the ability to sell the product in any way he sees fit. And the retailer is the one who knows his customers better than any distributor or manufacturer.

APPEARANCE

This is so obvious that one wonders why it should be brought up at all. But if it is so obvious, why are so many programs brought out in such mediocre, if not downright ugly, packages? Each level of software has a different packaging requirement, but not one of them need be placed in an ugly or plain container. No line of software need be so uniform as to sow confusion.

The perceptions of the retailer and his customer are the most important aspects to be considered in package design. In the mass market, the package that stands out from the pack has the best chance of selling. This works for cars, for books, and so on. Software, on any level, is no different.

The perception of the world is that a game should look like a game, and that business software should look like business software. Distributors, retailers, and customers all have one fault in common; the tendency to judge books, as well as software, by their covers. If the cover does not look like much, it is assumed that the program is not worth buying. That initial negative view will be almost impossible to overcome.

Game programs can be packaged in as bright or original a way as possible, within the framework mentioned in this paper. Other packages have to be done to fit the level for which they are intended. Professional software should look that way. But again it must be stressed that the package need not be plain or ugly. Visicorp has had two package types in its existence. At first, when VISICALC was the major program on the market, it came in a plain, brown loose-leaf binder. Looks did not matter; it was the only one. As other spreadsheet programs began to enter the market, Visicorp upgraded its packaging to reflect its position at the time as the preeminent professional software package. It created the software package equivalent to the grey pinstripe suit (in three pieces). Why? Because with previously unheard of competition it had to protect its image by presenting an outward appearance of professionalism and quality.

The appearance of the package is the first thing seen and is too important to ignore. It cannot be compromised.

INFORMATION

Somewhere out there in the computer world, a rumor has been spread that software cannot sell itself, that only by extensive demonstration could any customer hope to know what to buy. It was great for computer stores, they seemed like the gods of the new technology.

But then came 40,000 programs. No store could possibly hire a staff large enough to know them all. No store could afford the time to demonstrate a $20 game and risk losing the sale of a major software package. Then software left the computer store and entered the world of the mass market and the bookstore.

Software at all levels must either be able to sell itself or must provide enough information on itself to help guide the customer in narrowing down his choices before he goes and seeks help.

How does a book sell itself? It supplies the potential reader with enough information on what is contained within. This is called a dustcover. After the potential customer is enticed by the dustcover, he then has free access to the documentation; the book.

Should software be any different? Software, like books, should make the attempt to sell itself without any help from overworked, unknowledgeable, or uncaring salespeople. At the least, let the customer think about what he is buying, let him be able to ask intelligent questions if needed.

There are two things all people—especially those who are about to spend money—hate: feeling pressured and appearing stupid. Computer phobia exists even among those who own computers. They are using this new contraption for one reason or another, but often they do not feel competent to understand the computer or its programs. They know how books are written, in English, a language they understand, more or less. But a program! PASCAL? What is that?

There is no reason to have a package with large blank areas on it. Why? Why abandon the customer? Why not help the retailer? Why not try to make sales quickly by supplying basic information about the program at the outset?
SECURITY

This might be the last package requirement considered here, but it is not the least important, particularly to the retailer. Theft is as much of a problem with software, if not more so, as with any other product. "Shrinkage," as it is called in retail, is a special concern in nontraditional outlets. Because of its size and the fact that a program is often more costly than any of the other products the outlet might sell, retailers worry about theft.

Again, the retailer wants versatility to be inherent to the package without compromising his need for security. Some retailers are willing to display software behind glass counters or through plexiglass panels with holes in them, though for aesthetic reasons these are few. Most merchandisers want to leave the product out for self-service. They are willing to accept certain losses, but have no desire to make it overly easy for anyone to "shrink" a product out of the store.

Security in packaging is part of the manufacturer's responsibility. A 5⅛-inch disk is very thin and relatively small. No package should give free access to the disk itself. A loose disk will either be stolen, lost, or trampled on, costing the retailer the price of the program. Yet with the exception of most games, the customer should have access to the documentation. In that case the program disk should be sealed as part of the binding, the box, or the folder. While theft will not disappear and no package can be made theft-proof, there is no need to make it easy. After the purchase let the customer work a little to get the disk out.

In all other programs, the package simply need be of such a size that the program cannot be slipped into the pages of a newspaper and disappear.

CONCLUSION

It cannot be too strongly stressed that a program's package is often as important as that program's documentation. It is the package that brings the retailer or the customer to examine the program more closely. It is the package that presents the image of the program to all who view it. It is that same package that presents its perceived value to any who might consider using it.

With this in mind, the package should be as well designed as any other facet of the program. We all judge books by their covers. Software is no different. No program in today's competitive market needs a negative staring any potential user in the face. Conversely, perfect packaging will not make a perfect program out of a bad one. The product must be seen as a whole and it is the publisher's responsibility to his distributors and retailers to supply the best product available, on time, at a reasonable price, and in a package that will sell.

REFERENCES
