

Product...Or Service

By Paul Magel, Senior Vice President, BlueCherry Application Solutions at Computer Generated Solutions, Inc.



Paul Magel

The lag time between writing and publication means that I am writing these words in December 2005. And, as the year comes to an end, I find myself considering why it is that BlueCherry has outsold all of the other “soft goods software” packages for the past few years in a row. It occurs to me the reason is that BlueCherry represents a funda-

mental change in software technology. Where previous software required much programming to make it “fit” the business, BlueCherry is much more a core product, like Excel, that can be manipulated to fit many scenarios without unique customizations. Companies using BlueCherry have enjoyed a very different experience in growing with their software than those using older legacy type systems.

The first computer systems used by apparel manufacturers some 30+ years ago were all pretty much “home grown”; that is, programs that were custom coded by and for each company with the financial wherewithal to even afford what was then rather costly hardware. In due time, as the industry matured and hardware became affordable to smaller organizations, so-called “packaged” software appeared in the market. The intent was noble; since all apparel firms were basically performing and tracking the same operations, the pioneering programmers wrote software in languages like RPG (for the AS400) or BASIC (for Unix) to handle the complexities of style-color-size data and track the details of production that were otherwise handled manually on index cards or the like.

Of course, the reality was that even though various manufacturers were basically performing the same functions, each had its own way of doing things. For example, some companies liked to allocate goods to sales orders based on the customer’s order date, while others decided to give more important customers first dibs even if they placed their orders later in the season. A very common differentiator was reporting; everyone has their own ideas of how a cut and sold, production or sales order report ought to look.

To accommodate these real-world variations, programmers had to be called upon to write customizations to the computer code. As a result, each installed version of a given “packaged” program became more unique over time—I would venture to say that among our own 300+ legacy ACS Optima software customers, there are 295 “versions” thanks to this constant need to tweak the software on a case-by-case basis. It was common for larger companies to have a department of programmers on staff whose job it was to update their code as needed.

Understand, this was not the result of some underlying conspiracy on the part of software vendors. The simple fact is that in those earlier days, programming languages and techniques were less mature than today. Reports and inquiries, for example, were the end result of specific programs written to create them. So, if a client asked that his commissions report be sorted by division rather than by sales rep, it took a programmer’s skills to rewrite the portion of the code that created that report to change those sort values.

Of course, as time passed, not only did computer hardware become more affordable to ever-smaller companies, but software technolo-

gies and programming techniques matured as well. Two key areas of improvement have been object oriented programming (OOP) and the advent of report writers.

OOP is a technique where code is written in small modules, or objects, that can be reused as needed. For example, an object called “Available to Sell” could be created that would calculate ATS based on some formula (say, on-hand minus demand plus due to arrive). Once created, that ATS object can be reused in any number of reports and inquiries, and in every one it will operate exactly the same way. This eliminates the issue of having two reports, say, displaying different values for ATS for the same style-color—a problem that has long plagued legacy systems, where reports (and inquiries) were the results of specifically written unique sub-programs. In the old days, it was quite possible that the programmer who wrote report “B” never met the one who wrote report “A” and thus used a different formula for various calculations.

Report writing tools came into existence about 10 years ago; these are user-friendly tools that allow for the creation of reports using “wizards” and understandable English names rather than programming in a fundamental computer language. While there is typically a programming requirement to them (to maintain the translation from the underlying file names to the English ones the users see), they allow for the creation of reports tailored to each user as desired without needing to keep a staff of programmers. Since reports are the means by which users glean useful information from their computers, enabling users to create the reports they need quickly and easily is no small feat.

Even while technology was advancing, our industry was changing. Where before virtually all apparel firms “bought fabric and made clothing,” in time more companies outsourced some or all of the production, and many companies bought one finished good (like a blank T-shirt) to “produce” a different item to sell (the garment dyed or embellished T-shirt). As well, the retail community has been finding more ways to complexify the sales cycle, from reducing turn times to demanding more electronic correspondence.

These real-world changes pushed each company to modify their software, essentially making their copy of a “packaged” system a one-off. This, in turn, interfered with the ability to receive software updates from the vendor, the concern being that the new version would “break” some special modifications at a given company. Over time, clients would stop trying to accept software version updates altogether, and would instead pay for more customizations to add whatever features could be had in the update were it usable.

Essentially, where earlier ERP software was really an excuse to deliver services to the client, BlueCherry is a product that can be maintained en masse. Next month, I shall discuss how this shift has affected the consulting side of the practice as well...

For more information, visit www.cgsinc.com.