

The Enterprise Shops for Packaged Applications

BY PHILIP J. GILL

When considering whether to buy enterprise applications rather than build them, a strategic approach is called for.

Off-the-shelf software is taking on new significance for information systems (IS) managers. In past years, such packaged software typically has been desktop productivity applications. Recently, however, a number of factors have combined to make packaged software more appealing to managers higher up the enterprise computing ladder: the success of popular desktop products; the economies of scale inherent in mass-market software; the growth of independent software vendors (ISVs) who specialize in applications for the enterprise; and the general shift from proprietary, centralized legacy systems to open systems-based, distributed client/server environments. More and more, major IS organizations are turning to packaged software to meet their enterprise computing needs, particularly for so-called back-office operations, such as accounting and financials, purchasing and inventory, distribution and warehousing, human resources, and manufacturing.



"There's been a strong trend in that direction for the last couple of years," says Clare Gillan, director of applications and information access research at International Data Corp. (IDC) in Framingham, MA. "Companies are trying to provide users greater access to information and to reengineer their businesses. Generally, when they reengineer, they want to become more process-oriented than department-oriented. Legacy systems have a hard time adapting to that."

IDC's research shows momentum in that direction. In 1994, combined software license and service revenues for packaged client/server applications approached \$6 billion worldwide. And Gillan forecasts the market continuing to grow at a compound annual growth rate (CAGR) of 37 percent for the next three to five years.

But buying packaged software for the enterprise isn't like buying it for PCs. An IS manager shouldn't expect to stroll down the aisles of a local computer store to pull an enterprise accounting suite off the shelf, for example. Buying and using packaged software at the enterprise level are considerably more complicated and require a well-thought-out strategic approach.

Better to Buy

The first step is to make the buy-versus-build decision. Fewer and fewer IS orga-

nizations would prefer to build their own applications, given the clear benefits of buying packages.

For one thing, packaged software almost always costs less than internally developed applications. ISVs can spread development costs over many customers during the life cycle of a given product; when an IS organization decides to roll its own software, it is the only one bearing the costs. "Buying is always better than building, because of the large investment needed to build a client/server application from scratch," says Bobby Cameron, director of the software solutions service for Forrester Research of Cambridge, MA.

A second consideration is that buying packaged software allows IS organizations to better deploy their own existing internal resources. For instance, instead of building accounts payable or general ledger software, programmers at an investment brokerage could invest their scant resources in developing trading applications or stock evaluation algorithms that could give the company a competitive edge and contribute directly to the bottom line.

Third, packaged applications almost always can be deployed more quickly. After all, most of the work is already done by the time it gets into user hands.

Fourth, packaged software is likely to be more robust and stable, with fewer bugs. Naturally, if an IS organization decides to beta-test a new package or if the application chosen is relatively new (less than a year or so), it should expect more problems. However, most of the popular packaged software products are several years old and mature enough to have most of the bugs worked out.

Finally, buying packaged software allows the user organization to leverage an ISV's collective expertise in a particular application area, such as financials or human resources. It's often simply not possible to have that kind of depth in-house. And even if it was, recruiting such personnel would certainly drive up costs.

Case in Point

James Madison University, a 10,000-student, state-run institution of higher learning in Harrisonburg, VA, recently chose PeopleSoft of Walnut Creek, CA, to supply three key applications—human resources (HR), accounting, and a student information system (SIS)—for its move from a legacy mainframe environment to a campus-wide distributed client/server network.

Allen Cerveney, associate vice president

of enrollment services, says the university was impressed with the quality and features of PeopleSoft's Human Resources Management System (HRMS) and its Financial accounting suite, but what sealed the deal was that PeopleSoft has staffed its new educational software division with professionals seasoned in SIS software.

"We felt we would benefit from being involved with people who have developed student information systems before," says Cerveney. A prominent software vendor can recruit top talent that a single university couldn't afford on its own, he adds. In this case, JMU and PeopleSoft will jointly develop the new SIS, and JMU will act as a principal beta-test site.

Users and analysts also agree that it's easier for ISVs to keep pace with new technologies. An IS organization's time and resources are better spent addressing the needs of the business than studying every facet of IT advances.

When to Build

Despite all these reasons, there are times when it makes sense or in fact may be necessary to build. If a specific business need is not met by any packaged application, the company that wants the app may have to develop it.

For instance, organizations on the leading edge of IT may outstrip the market. Russell Lewis, CIO at Jeffries & Co., a Wall Street investment brokerage firm, says that when his previous employer, Salomon Brothers, another Wall Street brokerage, made its first moves from centralized legacy systems to distributed

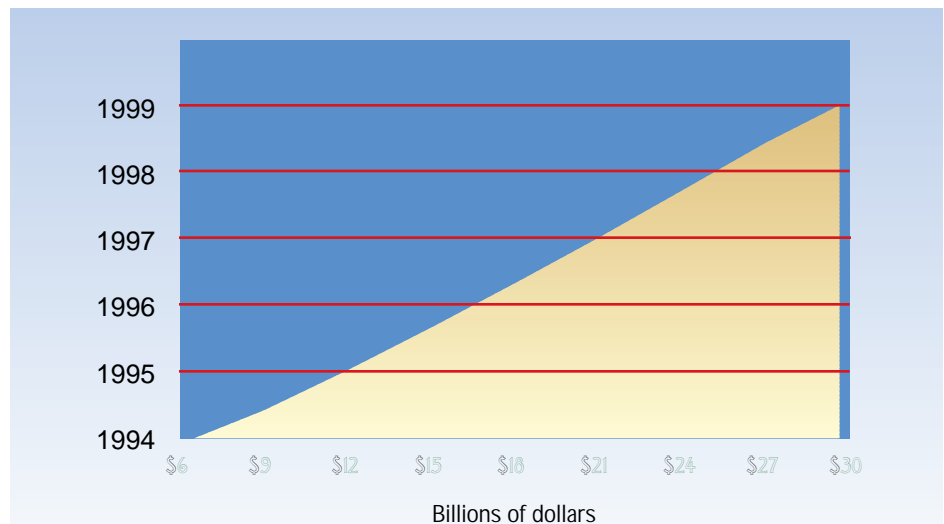
client/server computing, it had no choice but to build. "Back in 1989 there wasn't any packaged software out there to speak of," says Lewis, who works out of Jeffries' IS operations offices in Jersey City, NJ, across the Hudson River from New York.

Today, he says, things are different. In his new job at Jeffries, the IS department is consciously choosing to concentrate its expertise and internal resources on a few core applications it considers essential to achieving a competitive advantage. For the rest of the applications, Jeffries will buy packaged applications wherever possible, in large part simply because now it can.

But not everyone is so lucky. Mark Cates, director of logistics systems for Seattle-based Airborne Freight Corp., says some aspects of operations are so industry- or company-specific that his company sometimes has no choice but to build applications, or parts of them, on its own.

Bill Connor, vice president and director of IS for Motorola Corp.'s General Systems Sector in Tempe, AZ, says his organization buys 75 to 80 percent of all its software from outside suppliers. "I wish it could be 100 percent," he says. But he acknowledges that there are times when this isn't feasible. "It's probably better to build when it's directly related to your line of business or when the application provides a revenue-producing service to customers," he says.

Yet Lewis is adamant about avoiding major in-house development. "I see building software as an option of last resort," he says.



Source: International Data Corp.

Projected client/server software revenues, 1994-99

Group Decisions

Once the buy decision has been made, IS should work with users to determine the features and specifications they need. Users should be in on the process at the beginning, but it's part of the IS role to help them see beyond the constraints of their current system.

In some cases, going with packaged software may involve not just a new IT infrastructure, such as a move from mainframes to client/server networks, but major reengineering of business and workflow processes. That may be a change the organization needs to make anyway. It can learn, say IS managers, from the wealth of experience that established client/server ISVs bring to the party.

"You have to be prepared to change the way you do things to fit the software," says Connor of Motorola. His group provides information services and support to Motorola's computer systems and cellular telephone divisions.

Part of buying packaged applications means being willing to accept the business and workflow processes embedded in them, says Connor. Often, a user organization can learn a useful thing or two that would improve efficiencies of operations and cut costs. But Connor adds that users shouldn't take extreme measures to adapt to packaged applications. "It shouldn't be the tail wagging the dog. It's got to be a good balance."

The most important thing, IS managers

agree, is to instill a sense of ownership in the new client/server software from the outset among its target user population. If they don't feel that this is their choice but something being forced down their throats, the chances increase that the project will fail.

Even then, expect last-minute resistance. Connor recalls that recently one department tried to back out of using a new packaged application shortly after it was installed, saying they didn't think it would work out after all. The department's users had been in on the requirements, evaluation, and selection processes, almost from the outset—a fact that gave Connor's IS organization great leverage at this critical moment. "I told them that they chose it, so they'd have to live with it," he says.

Varying Standards

As users are painfully aware, software vendors differ drastically in which standards their products support. A user organization should have at least a clear policy on what standards it demands. Some firms prefer strict allegiance to industry standards, both de facto, such as Unix, TCP/IP, or Microsoft Windows, and de jure, such as Posix. Other companies find it more important to adhere to internal standards, even if those don't conform to industry standards.

For example, Airborne Freight requires that all its packaged applications run on one of two internal system architectural

standards, either a legacy IBM mainframe environment with MVS or its newer, distributed processing environment, which is based on Hewlett-Packard HP 9000 servers, HP-UX Unix, and the Oracle relational database management system (RDBMS).

"We don't want to support more than two architectures," says Cates. "The costs are too high." In one recent case, the company chose not to purchase a packaged warehousing and distribution application, even though it met all of the company's functional requirements. The reason: It was available only on the IBM AS/400.

Evaluating Vendors

As well as checking standards conformance, the IS organization should evaluate a vendor of packaged applications in other ways. Is its technology sound and up-to-date? How long has it been since the latest update or revision? How old is the application? Is its architecture current, and will it suit both short- and long-term needs of the buyer?

Cates of Airborne says a key evaluation criterion was whether the vendor had built its application using a layered architecture. Right now, many users at the company's ALS subsidiary use dumb terminals, but Airborne intends to migrate them to X terminals or Windows PCs. "We want to be able to slide out one interface and slide in the other easily," he says.

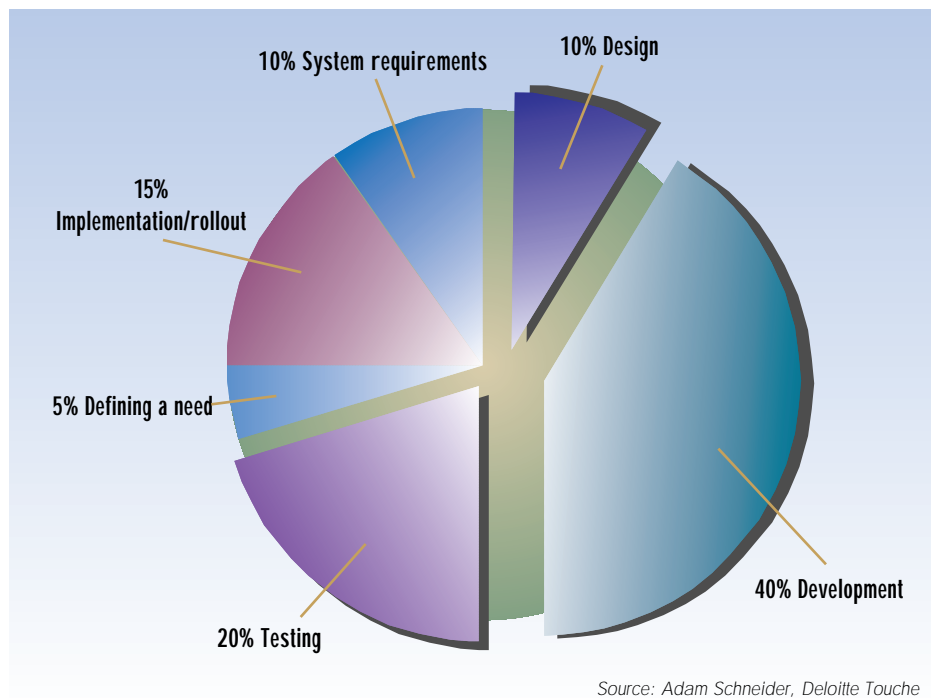
Another factor to consider is the ISV's financial health. Nobody wants to be left with orphaned software. That puts an IS organization back where it started, developing and maintaining key enterprise applications on its own.

What about service and support capabilities? If users need service 24 hours a day, seven days a week, IS should make sure the ISV can provide it.

Furthermore, there are personality issues. Is there a fit between the IS organization and the ISV? For instance, is upper management accessible and interested in having and keeping your business? Can you develop both personal and partnership relationships with the ISV?

Most users don't require that ISVs have comprehensive product lines. In certain fields, such as warehousing and logistics, it's simply not possible to find large ISVs with broad product lines, says Cates. Particularly in warehousing, he says, the smaller firms tend to have more innovative products.

On the other hand, James Madison Uni-



Source: Adam Schneider, Deloitte Touche

Reasons to buy packaged applications

(CONTINUED ON PAGE 22)

This page intentionally
left blank

(CONTINUED FROM PAGE 20)

versity made finding a single vendor to supply all three applications a priority. "We were interested in finding a single vendor because originally we had three separate committees, and we were all competing for the same scarce resources," says Cerveney. "We quickly realized we all had the same objectives and same goals." An added benefit of having a single vendor, he says, is that it eases the process of integrating applications across a network.

Negotiating Contracts

Once a vendor is chosen, negotiations begin. It is critical during the negotiating process to establish clear guidelines for the vendor in terms of who is going to customize the packaged application—the customer, the vendor, or a combination of the two.

IS managers say it's usually better to have the vendor do it, at least when making the initial purchase and installation. Later releases, updates, or upgrades could be handled by internal staff, once they're up on the product.

It's also necessary to establish a schedule of deliverables tied to payment. Payments should be made in portions, as the vendor hits targets. Don't be afraid to incent the vendor with rewards for early deliveries, and don't be afraid to hold back should it fall behind.

Flexibility is key. Motorola's Connor says that working with a packaged software vendor is a two-way street. Naturally, as a customer, users expect the ISV to be sensitive to their needs. But, he says, IS organizations have to be aware of their ISV's needs as well. For instance, Connor sometimes has paid a smaller vendor ahead of schedule, since it may not have the financial resources to finish a project on time. This will be especially true for more specialized application needs, in which case you'll have to deal with smaller ISVs.

Lewis of Jeffries & Co. says IS organizations should not be afraid to get involved with those smaller ISVs, who often employ the most innovative technologies or newest approaches. In some cases, where it fits with corporate business objectives, he recommends that IS consider funding development projects or even investing in a smaller ISV if its products or technologies are potentially beneficial to the business.

"This way, you can put your two cents in and also be assured of the quality of the products," says Lewis. At Salomon Brothers, he says, such strategic invest-

ment and partnership agreements with small ISVs were common.

And never negotiate so tight a deal that the vendor can't make money. "You're never smart pressing a small vendor to the point where they can't make money," says Airborne's Cates. In this regard, it's better to think of the ISV as a sort of partner than an adversary over cost.

Custom Packages?

Nothing truly comes off the shelf in packaged applications for the enterprise. There will always be some customization. How much depends on individual companies. Most users agree that the best guideline is to look for the standard application to provide at least 80 percent of the func-

Few IS organizations would prefer to build their own applications, given the clear benefits of buying packages.

tions you are looking for. Then be prepared to customize for the rest of the needed functions.

Luckily, according to Bob Bacon, vice president of finance at Groth, Inc., a Houston-based valve manufacturer, most packaged enterprise applications today are so feature-rich that a large part of any customization process is merely picking among options the package provides. In fact, Groth recently purchased ManMan/X, a client/server manufacturing resource planning (MRP) II system from Computer Associates International (CA) of Islandia, NY, because it contained such extensive configuration options that users could tailor the software to their business needs.

ManMan/X contained 80 to 90 percent of the features the business needed, says Bacon. "There were so many options and tables in the package that it allowed users to do the customization rather than go to the programmers to do it," he says. "We modified screens to meet the needs of our business."

Nevertheless, careful examination of

a package's development tools, usually included in the standard package, is a must. JMU's Cerveney says one of the reasons his institution chose PeopleSoft applications was the high quality of their development tools.

However, be careful not to customize packaged software so much that it becomes, in essence, a proprietary application. First off, that's what the IS organization is trying to get away from. Second, a high degree of customization will make updates and upgrades to new releases more difficult to implement. Finally, your organization would lose one of the basic advantages of going with packaged applications—that is, getting more function at less cost, because the vendor spreads development and ongoing maintenance costs over many customers.

Living with the Package

It's probably best to let the ISV do most of the initial installation and train the user departments, say IS managers. But vendor involvement can range from minimal to total.

Connor likes an ISV to get actively involved. "They know their product best," he says. He even prefers the ISV to do all the customization for the Motorola site. He says that such an arrangement usually helps build better long-term relationships with the supplier.

Airborne takes a different approach. When it purchased CA's MRP software for its ALS subsidiary, Cates says, the company chose to keep CA's involvement to a minimum. Cost was one reason; the less the vendor is involved, the lower the cost. The other was the desire to get its own IS staff up to speed on supporting the product as quickly as possible. "A CA consultant comes in once every two weeks or so to see how we're doing and answer questions," he says.

Once the software is up and running, that doesn't mean the relationship with the ISV is over. In fact, it's just beginning. By now both parties should have shared something—the customer shares its business needs and direction, and the ISV its current and future technology. The most successful business relationships between IS and ISV will be those in which both started at the outset to build close business and personal relationships. "Remember," says Connor, "they're people, too."

● **Philip J. Gill** is a free-lance writer and editor based in San Diego, CA. He can be reached at philipgill@aol.com.