The Dark Secret of ERP ROI
Want Better "R"? Lower Your "I"

N o one seems to disagree that typically **ROI on ERP has been atrocious.** A quick scan of recent articles comes up with the following limited list:

1. ERP was rushed in for Y2K remediation; (Note 1.)
2. No periodic reconsideration of goals after installation; (1.)
3. Dissolution of ERP team; system turned over to operations; (1.)
4. Unnecessary customization, including replication of legacy systems; (1.)
5. Proliferation of multiple instances of the software; (1.)
6. ERP simply mirrors existing, inadequate business processes; (2.)
7. Organizations unable to conform to best practices templates; (2.)
8. Failure of management to ask "How do we want to run the business?" (2.)
9. Inexperienced consultants/system integrators; (2.)
10. Lack of executive sponsorship; (2.)
11. IT focuses on "technical wizardry," not business value; (2.)
12. No metrics for intangibles; (2.)
13. Systems are complex and hard to learn; (3.)
14. Companies are naive and optimistic in evaluating software capabilities; (3.)
15. Training that focuses on key strokes, not flow of information; (4.)
16. Failure to educate middle management. (4.)

Your personal favorite may not be on the list, but one thing these reasons tend to have in common is that they emphasize the "R"--return--side of the equation. It's astonishing that no one seems to state the obvious: **A major reason ROI on ERP has been so bad is that the "I"--investment--is simply way too big.** Even now, everyone--industry experts, CEOs, board members, and especially most ERP vendors and consultants--seems to accept that costs will be measured in the millions of dollars. Similarly, there has been only the slightest attention paid to the fact that a typical implementation takes years and that the cost of implementation services is often five or more times the cost of the software.

In the late 90s some companies threw up their hands at ROI analysis and accepted that ERP was simply something they needed to do in order to "get in the game;" the real benefit would come from applications such as business intelligence, supply chain management, e-business, and advanced planning and scheduling taking advantage of the data stored in the ERP system. This is actually true, but it does not go far enough in bringing a vital truth to light: **ERP systems tend to resemble each other in that they all process transactions, maintain a database of information, and allow standard and ad hoc reporting against that information.** The metaphor for ERP is often that of "backbone," not "brain." So why would any company spend millions, and years--with no certainty of a successful culmination--to create this infrastructure if it could obtain it at a fraction of the cost, in a fraction of the time, with predictable results? The answer to that question is beyond the scope of this paper. Let's try to answer a better one: How can the monstrous costs of ERP be tamed?

Software cost. The most expensive ERP systems come from a small number of public companies. Like private companies, they have virtually no marginal cost to produce an additional copy of their software. What is at stake for them is the quarterly need to report license revenues against the amortization of development expense. The greater the revenues, the greater the gross margin, and the better the market likes it. However, that is their problem, not yours. Each extra dollar of software revenue boosts their

gross margin, not the value you receive. Their counter-argument would be that their prices reflect the development expense necessary to provide the functionality included in their product. (You probably won't hear that the bottom line requires that they also recover the cost of the hype that helped to capture mind-share). You need to decide whether what you're receiving is functionality that will add value to your business, or complexity that you don't need, reflecting, among other things, the vendor's trying to be all things to all kinds of companies, rather than concentrating on your industry.

A small number of companies (I'm thinking private, but let me know of any public examples) have adopted a different model that has been proven to serve the needs of both customers and company ownership. It helps customers to control costs by providing:

- **Comprehensive, integrated function at one low price**
- **Unlimited license**: no user- or tier-based pricing
- **New modules at no cost** to existing users

Cost to implement and maintain. These costs--as mentioned previously, often a staggering five times or more the (inflated) cost of software--include costs of:

1. Integrating retained legacy systems
2. Integrating "best of breed" point solutions
3. Creating customizations
4. Training users
5. Converting data
6. Creating and administering a project plan
7. Ongoing support (New releases, interim releases and bug fixes, telephone support, etc.)
8. Hiring new personnel

Any of these functions might be provided by a "systems integrator"--it's amazing how often the phrase turns up in articles about ERP. What is less often mentioned is the economies that can be realized by minimizing integration and customization: costs 1., 2. and 3. **A mature, robust application that is integrated out of the box is required**, as is an experienced, credible consultant who advocates avoiding: 1) slavish retention of legacy systems, 2) often catastrophic attempts to integrate best of breed, and 3) unnecessary customization.

As mentioned earlier, ERP ROI has suffered from training users in key strokes, rather than information flow. No matter what system you choose, it is likely that the vendor can have it installed on a server at your facility within days of coming to an agreement. And then what have you got? Empty data files, untrained users, no policies or procedures for using it for decision making in your company, and, usually, no plan for getting from there to the point where you're using it without compromises to run your company. It's clear that the first order of business after installation should be a business simulation workshop that will allow someone familiar with the system to become familiar with your operations and the way in which you do and would like to do business. The outcome of this workshop should be a detailed project plan outlining each party's responsibilities on a day by day basis. There should be no doubt in anyone's mind as to what needs to be done on a particular day. As the project progresses, users of the system can then be trained in the particular way a function is to be performed in your company. Enormous savings in time and money can be realized if neither you nor your vendor has to travel for training. **Interactive video conference-based training**--available and priced for as little as an hour or two of exactly the training that you need--is highly desirable.

Somewhere along the way, **yearly maintenance fees** of 20% or more of software list became routine. Whether corresponding value is being delivered is problematic, especially since, as previously discussed, the price of the software is frequently arbitrarily high. In negotiations, vendors will often plead they have no control over the costs of providing maintenance and therefore have no ability to adjust the fees charged. This is probably not entirely candid in many cases. Their model for handling updates may be inherently flawed and flabby. Efficient models include electronic transmission of problems from the customer, electronic transmission of updates on a daily basis to all customers, and one-key retrieval and application of updates by the customer. Adherence to best practices like this should allow vendors to maintain yearly maintenance fee percentages in the low teens (even if software prices are not artificially inflated).

A Final Note on Hiring and ROI. It's pretty clear that if you're hiring in order to implement or maintain an ERP system, your ROI is headed in the wrong direction. Perhaps the technical infrastructure is too fragile and demanding; anymore, your server and database management system should not require much more attention than a refrigerator (only occasional cleaning required). Your people should be able to learn any new skills required in a short time at a reasonable cost. If you're hiring programmers to rewrite or add functionality, it sounds like you either have not located a vendor who specializes in your industry, or you're helping to subsidize functionality you don't need. With regard to hiring, a story may be in order. A few years back, this writer noticed in the classified ads that a company was looking for both an application programmer and a database administrator for a system whose name even then was synonymous with blown budgets, endless implementations, and bloated infrastructures. A quick and dirty analysis of the costs, with benefits, of those two people (if they could be found) suggested that **one year's savings** in avoiding their hiring would **completely fund** a three year lease for the acquisition (including server), implementation and three years of maintenance on a system meeting the guidelines suggested by this paper.

Notes
1. "ERP ROI Depends on Ongoing Planning and Adjustments to Meet Changing Business Needs," META Group.
2. "Why the Controversy over ROI from ERP?" R. Michael Donovan.
3. "Making ERP Add Up," Tom Stein, Information Week Online, May 24, 1999.
4. "ERP Training Stinks," Malcolm Wheatley, CIO Magazine, June 1, 2000.

_____

For sixteen years, CMS Manufacturing Systems has delivered its promise of "ERP Made Easy" to customer-driven discrete manufacturers operating in tight supply-chain relationships. Please contact Bob Paley, CPIM at 303.433.0317 or bpaley@cms400.com. You may also visit CMS on the web at www.cms400.com.