

---

**Ernie Schell**

---

## Selecting Catalog Management Software

Ernie Schell is President and Founder of Marketing Systems Analysis, Inc., the author of the *Guide To Catalog Management Software* and the *Marketing Software Review Newsletter*, Executive Director of the Catalog Systems Management Network, and MIS Columnist for *Operations and Fulfillment Magazine*. In 1993, he launched Marketing Services Alliance as a division of MSA, Inc., to provide outsource services management for third-party telemarketing, product fulfillment, and database marketing maintenance and analysis. A former Editor of *Target Marketing Magazine*, he was software reviewer for *Catalog Age* from 1984 to 1992, and is the Founder and President of Industry Publications International, Inc., which has produced annual editions of the *Guide To Catalog Management Software* since 1986. In September 1992, he launched the *Catalog Software Review* (now the *Marketing Software Review*), a monthly newsletter covering new systems and major upgrades of integrated software for catalog order entry and fulfillment, as well as specialized hardware and software for managing and tracking catalog inventory and customer files.



Ernie Schell

### ABSTRACT

In deciding on a system to meet your needs how can you be sure you're getting the right one? Here are some guidelines to help you make the right choices.

---

©1996 John Miglautsch  
International Society for Strategic Marketing

## Introduction

This article is adapted from a chapter in the Guide to Catalog Management Software, published by Industry Publications International, Southampton, PA, and reproduced with permission of the publisher.

**The most often asked question about Catalog Management Software has no answer!** That question is: "What is the best system?"

Best for what? According to what criteria? The most richly featured system may be so difficult for you and your people to learn that it loses most of its value. The system with the most list management capabilities may not meet any of your inventory needs, and the system with the most flexible and dynamic order entry and fulfillment may be totally inadequate for your accounting or reporting needs.

You know the old saying, "If you don't know where you're going, any road will get you there."

So let's do some reconnoitering. Selecting an integrated, packaged system from a third-party developer can be a little "EASIER" than you fear. Here is a summary of the five steps you need to take in order to make the most reasonable decision on what software to buy:

### Establish your Objectives

Assess your Operational Parameters, (order volumes, file sizes, etc.), your Existing Resources (hardware and software), and your Needs and Requirements (features and functions)

### Set Priorities

Investigate your Options (basic platforms, software systems)

### Evaluate the Best Solutions

Review your Alternatives (final sanity check)

You should already have a good idea of what your own objectives are. But you'd be surprised what happens when you sit down to put them on paper. They start to seem more nebulous than you thought they were. And when you ask even your closest colleagues what their objectives are for an order entry and fulfillment system, you'll be even more surprised that they don't agree with you or that they see the same problems from a completely different angle.

Here are some typical objectives you might come up with:

- increase our peak order processing capacity
- lower the cost of taking an order
- provide same-day fulfillment for 90% of in-stock orders
- manage inventory more efficiently
- segment customer files more effectively
- use the inventory data to support desktop publishing
- give customer service reps better access to customer contact histories and order files
- do gift shipments more easily
- provide more customer payment options (per order or across the board)
- reduce picking/packing staff by 40%
- reduce MIS staff by 15%
- reduce total MIS costs by 40%
- improve average customer LTV by 20%
- put all accounting on the same platform
- give TSRs better scripts and prompts
- automate outbound call management
- get better financial reports and net profit analysis
- get better ad hoc query capabilities
- generate better projections of response and response curves
- reduce or eliminate paperwork
- manage an off-site warehouse more directly
- achieve the ability to handle international customers
- streamline the manifesting process
- improve handling and processing of returns
- create a serialized inventory and better lot control

acquire MRP and assembly capabilities  
 get a better idea of who our customers really are  
 integrate a retail system with catalog inventory  
 beef-up order entry capabilities (for  
 personalization, special orders, future ship  
 scheduling, etc.)  
 monitor, track, and analyze clerical productivity  
 provide TSRs with more/less discretion to  
 override system parameters for pricing and  
 discounts  
 get a system that is easier to  
 maintain/learn/upgrade  
 get a system that has documentation  
 get a system we won't outgrow in another two  
 years.

Obviously, you can't have everything! You might not even come close. And if you say you want whatever you can get for less than \$35,000, you might not get very much at all.

Thus, it is important not only to put together your "wish list" of major goals and objectives, but to make sure that you cover the bases that are most important to your entire organization, and to *shorten the list of objectives to no more than four primary and six secondary objectives.*

#### **Four Key Steps**

To get to that point, you need to prioritize your "blue sky" list. There are four steps you need to take to do that.

1) Take stock of what resources you have and what *costs and benefits* are associated with them. A cost/benefit analysis will look at the age and condition of the physical resources and the training and experience of the human resources (in general terms and with respect to knowing your operation). Which of these need to be upgraded and replaced, which ones have already been amortized, which ones cannot be realistically replaced at this point, and which ones should be considered for replacement because the newer resources available can demonstrate a much higher and compelling return on investment?

Current resources include:

all computer hardware (and existing network hardware)

all computer peripherals

- modems
- printers
- tape/cartridge drives
- CD-ROM drives
- Universal Power Supplies

all operations-related hardware/software

- bar code equipment
- electronic scales
- radio frequency units
- fulfillment/picking light systems

all applications software

- order entry/fulfillment
- inventory management
- telemarketing
- point of sale
- accounting
- spreadsheet
- database management
- analysis
- desktop publishing
- word processing

Your analysis should also include an evaluation of MIS personnel in terms of skill levels, depth of staffing, and management capabilities. You'll want to include:

- MIS Director/DP Manager
- systems analysts
- staff programmers
- contract programmers
- computer operators

You might discover, for example, that you have been paying more for contract programmers than you realized, and that you can replace three part-timers with one full-time whiz who will be more likely to get to the heart of your business and be a leader, not a follower.

2) Determine your operational parameters. What are your past, current, and projected:

order volumes and seasonality peaks, valleys, and averages

volume of customer service calls (peaks/valleys/avgs.)

volume of technical support calls (if appropriate)

volume and seasonality of outbound prospecting calls

volume and seasonality of inquiry fulfillment

number of times you run your house list for segmentation and for label generation or image tape preparation

number of customers and inquirers/prospects

number of SKUs

- single SKUs sold from stock

- personalized SKUs sold from stock

- kitted SKUs sold from stock

- assembled/manufactured SKUs sold from stock

- drop-shipped SKUs

How well do your current resources manage these parameters? Are you likely to need additional terminals for order entry, customer service, and inventory processing? Will you need additional printers? *Does it appear that one of your objectives will be to streamline your data processing so that you can minimize the growth of these resources . . . or is it an acceptable fact of life that you expect to double your order volume in the next 18 months and therefore must double the number of your terminals (there is no "right" answer here).*

As part of your inventory of "resources" you will also need to take stock of your corporate finances. Indeed, you can't do a realistic cost/benefit analysis without concurrently doing a pro forma balance sheet, cash flow projection, and P&L projection based on anticipated orders and sales (net of returns and other costs of doing business) for as far ahead as you can see (at least for the next 12 months; preferably for the next three years). After all, a \$300,000 system might bring a 500 percent return on investment, but you're wasting your time planning for it *if you can't finance anything more*

*than a \$100,000 system.* The point of the exercise then becomes something like the following: if two systems are able to meet most of my needs, but one that costs \$135,000 is likely to have an impact on my operations that will *save the company \$40,000* per year for the next three years, is this a better investment than a system that costs \$90,000 but will save me only \$17,500 per year over the same period. You can't get to this decision without doing the preceding. And you can't really get to it without doing Steps 3 and 4, either.

3) Map your physical layout. Where are your terminals located in relation to your CPU? How many stand-alone PCs need to be connected to the database? What kind of access do they need to data (active, on-line or downloaded off-line)? Where are your printers? Where is your warehouse in relation to your CPU? Will this change, and will you be adding other warehouses? What other corporate entities in what other locations need to be connected to your system, with what kinds of access? What other computer systems does your company have that you need to connect with, and on what terms? What hardware does your CPU need to interface or communicate with (a telephone system, for example)? What service bureaus (order entry, fulfillment, credit card processing, or data/list processing) do you use and what type of interface do you have (on-line full-time, on-line intermittent/as-needed, magnetic media) to transfer data between you and them?

4) Undertake the grueling process of determining the features and functions you would like your new order entry and fulfillment software to perform. In brief, the specification process should include:

An assessment of the functions, routines, and tasks you need your system to perform. The technical requirements for performing them.

The business constraints on each of these functions.

To elaborate --Assessing functions refers to the broad, general areas you want the system to support. These would typically one more of the following:

order entry (mail and phone)  
 outbound telemarketing  
 customer service  
 technical service  
 customer correspondence  
 fulfillment
 

- picking
- packing
- manifesting

 inventory management
 

- receiving
- storage/transit
- pull-forward
- returns processing

 assembly  
 manufacture  
 purchasing  
 vendor correspondence  
 customer file management  
 mailing list management/generation  
 response analysis  
 advertising/media analysis  
 accounting functions
 

- accounts receivable
- accounts payable
- sales reports
- corporate G/L
- asset management
- payroll (complete or part-timers

only)

Once you have decided which of these types of functional activities you want your new system to handle, you can specify what routines and tasks you want the system to perform.

A "task" is a very specific activity, such as "display a cross-sell prompt for each SKU to which a cross-sell is attached in the inventory Master File" or "replicate the order header automatically when entering a second order for the same customer during an order entry phone call."

### **Nailing Down Routines**

A "routine" is a collection of tasks that comprises a more complex series of steps. For example, "when processing a return, we want to be able to log a

Return Materials Authorization number against the original order; automatically update the order record and the inventory sales record when the returned merchandise is received with Reason for Return and condition of returned item; update accounting files for the accounting period(s) in question; automatically update the inventory files if the item is resalable; and issue a credit or refund to the customer, as required." If you care about the exact nature of how those tasks are performed, you need to specify that routine at the task level ("we must have up to 99 user-definable Reasons for Return" or "the refund check must include a personalized offer to redeem for an additional 10% value on the next order"). If you don't care about the details -- only that the system must do them somehow -- then don't bother specifying the tasks, just list them as a way to define the routine.

### **Technical Requirements**

The Technical Requirements for performing the tasks and routines will include such things as: benchmarks for system response times at various activity levels (less than 2 seconds to find 1 customer record out of 650,000 when 20 order entry and customer service terminals are active) record layouts for customer files, inventory files  
 forms layouts (picking, packing, accounting)  
 must operate 24-hours per day, 7 days per week  
 must be compatible with Rolm phone switch  
 must interface with manufacturing system on the AS/400  
 must be compatible with TCP/IP and x.25 protocols  
 customer file must be a relational database  
 must interface with specific manifesting systems  
 must interface with specific POS system  
 must generate audit trail with date/time stamps  
 must have laser output for certain documents  
 autofaxing of correspondence to customers and vendors  
 must provide 4 levels of security access and user-specific menu layouts  
 we must have source code!

The Business Constraints might include:

- budget restrictions/desired leasing terms
- training must be at our site
- the system has to be up-and-running by July 1
- all modifications must be completed by Sept. 1
- we must have round-the-clock access to technical support
- we must have 4-hour response time for field service
- the system must be fault-tolerant (won't go "down")

Issues like the last one, and issues such as the record layouts you require and other software that will remain in place, are both technical and business issues. Are certain fields necessary in the inventory file because you want the information, or because they are needed to improve data analysis routines or serve a look-up function? Do you want the system "parameterized" with look-up tables for key data elements to improve performance, eliminate clerical error, or provide consistency? Are you keeping a POS system because it is state-of-the-art, because it would take too long to convert to something else, or because you like the way your current system operates and if it ain't broke, don't fix it? Thus, you need to include representatives from all divisions -- data processing, operations, marketing, accounting, upper management -- in making these assessments.

### **Managing Change**

Ideally, these Four Steps for Assessment should be done by a committee or Task Force composed of representatives from each area of the business, and given a mandate to manage the process of system specification and implementation. These people should be conscious of the obligations they have to bring about orderly *change* in the organization, and to include all potential users of the system in a process of inquiry and participation in system definition so that they will have a sense of "ownership" when the new system is in place. Change is disruptive and disorienting *even when it is "good."* The Task Force should strive to reduce

the anxiety and fear that change inevitably brings in its wake.

Once the Assessment procedure is complete, you can Set your Priorities. You know what resources you have and what objectives you would like to achieve. It is time now to separate the dreams from the reality, to determine which objectives you can meet and must meet with the resources you can afford or wish to acquire.

One way to do this is to separate your objectives into five categories:

- A. Broad goals that must be achieved
- B. Specific goals that must be achieved
- C. Broad goals that would be nice to have, but not necessary
- D. Specific goals -- nice but not necessary
- E. All "wish list" items that are marginal of marginal benefit  
(no matter how nice they would be)

With the objectives prioritized, you can now go back and review all of your assessment results to see what goes and what stays. If you have Dom Perignon aspirations but only a Gatorade budget, you may find that your objectives have shrunk to two and your functional requirements have dwindled to a handful. You can then take this brief listing and proceed to the next step: Investigating your Software Options.

### **The Request For Proposal (RFP)**

You are now reading to prepare a Request for Proposal, or "RFP" (sometimes called a "Request for Quote" or RFQ) for software vendors.

The RFP will include the following:

- a company background summary
- a list of all primary, secondary, and tertiary objectives
- the inventory of your resources
- the list of your operational parameters
- discussion of hardware platform issues

a detailed list of the routines and tasks you require  
discussion of database and file structures required  
your budget requirements  
your expectations for training and support  
your timeframe for system selection and implementation  
directions for responding to the RFP (what types of answers, to what degree of specificity, do you want the vendor to provide; how much detail -- time and cost involved -- should the vendor provide about modifying the software to meet your needs; when is the vendor's response due and to whom should it be sent? How much background on the vendor's business and what references to you require?)

Optionally, you may include a discussion of how you will be undertaking the decision-making process: what people will be reviewing the proposal, what elements of the proposal will receive the most weight, what types of system demos do you want to see?

A final word on this exercise. In general, it is better to request as few changes, modifications, and customizations as possible to a vendor's software package. Keep your requirements to a minimum without sacrificing essential functionality. The ability to do that is, after all, one of the primary reasons for undergoing this entire process.

Obviously, you won't know how much customization you require until you look at the software options available. That's your next step.

### **Software Review**

The systems reviewed in the Guide to Catalog Management Software (published by Industry Publications International, Southampton, PA, 215-396-0650) represent virtually the entire spectrum of *fully integrated order entry, fulfillment, and customer database systems* for catalog management. Of course, you may have very specialized requirements, for telemarketing let's say,

and very simple inventory requirements, and you may therefore want to look at some telemarketing systems as well as the programs in the **Guide**. Or you may have a lot of wholesale distribution or retail requirements, and may also want to look at inventory management or POS systems, too.

The **Guide** review will tell you if the vendor is willing to make any changes to the software. In cases where the vendor can't or won't make changes, ask the vendor to respond by indicating, from his/her perspective, which of your requirements the software package can meet (and in what way), which ones it comes close to meeting, and which ones it can't meet at all. By doing so, you will reduce the chance that something you see in a demo or read in the review or hear on the telephone will be misunderstood.

Bear in mind that the inability of a vendor to make changes does not *ipso facto* eliminate that package from consideration. In fact, it is all the more important that you clarify what you'll be getting, because what you don't get you will never have.

The next step, Evaluating the Systems, begins with reviewing the proposals from the vendors. The key here, after taking into account the non-negotiables such as the size of the vendor company, the type of programming language it uses, or the number of other installations the vendor has, is to review how well each system meets your functional requirements. To make sense of what can be an otherwise hopelessly chaotic jumble, you should *assign each requirement with a priority weight* (from a high of 1 to a low of 3 or 5), score the responses for each requirement with regard to how much change is required (5 being a lot and 1 being a little, with 0 for none at all), and multiply the score by the weight. The system with the lowest score set of weighted/ranked scores "wins."

### **Vendor Compatibility**

Make sure the vendor's attitude about systems is incompatible with your style of doing business or your "corporate culture." This is a very significant

consideration, for selecting a vendor is a little bit like choosing a spouse. There will be an intimate and reciprocal relationship, and you are "marrying into a family" as much as you are taking a partner. If the family or the partner doesn't have the right "chemistry," it can be a disaster.

I can honestly say that I have seen more systems fail because of a bad relationship between vendor and client than I have because of an inherently bad system. That's also why there is no such thing as a "good" system, in and of itself. There are only systems that are better or worse for *you*.

You will obviously want to see a demo of the system. If the vendor has a demo disk, get a copy and use it. If you can dial in to a demo on-line, do so. And if you can arrange to visit a live client site, by all means do so for those two or three vendors who have the systems you are most impressed with.

Be aware that a demo is not likely to show you how well the system is going to meet your needs. That's another reason why you have bothered to go through the exercise of formalizing your requirements. But a demo will help you to get a better sense of the "look-and-feel" of a system, how its pieces all function together, and what its greatest strengths and biggest shortcomings are *in terms that are meaningful to you*.

### **Users Group Meetings**

If possible, you should try to attend a vendor's Users Group Meeting. These are wonderful opportunities to get better acquainted with the vendor, the system, and the people who use it and support it. Ask other users in person or by telephone about the vendor's track record in diagnosing and troubleshooting problems, correcting bugs, and providing technical and user software (and/or hardware) support and training.

If it hasn't been covered in the RFP, review with the vendor candidates the conversion of your current files: who will handle this, what are the downsides,

when and how will it be scheduled and implemented?

Additionally, you should discuss the vendor's plans for future upgrades. How much will the system be changing, in what directions, and how much input will users have regarding this evolutionary process?

### **Contract Negotiation**

When you have finally settled on the vendor and the system you believe best meets your needs, your last chore is negotiating the best contract. In general, if the vendor is supplying hardware as part of the package (a "turnkey" contract), the price of the hardware may not be very flexible. After all, the vendor has to make some money on this to make it feasible, and the vendor is probably not making as much from it as you think. The real negotiables are the amount of training you receive free, what you pay for additional training, the cost of making modifications, and the timeframe.

While you're at it, you should also put in writing that all modifications will be fully documented, and that such documentation must be forthcoming before you pay scheduled technical support fees.

### **A Final Review**

A last word. When the system is installed, be sure to set aside a day or two to review the customized features and functions during the training and conversion period, just to make sure everything was done as promised. Don't make final payments until everything -- modifications, conversion, training, and documentation -- is completed to your expectations.

In general, it will take three to six months for you to get the new system completely in place and your people trained to the point where they can function effectively on it. If you try to rush this process, you'll find that "there's never enough time to do it right in the first place, but somehow always enough time to do it over."

Every six to twelve months your key systems managers (technical and non-technical) should refresh their acquaintance with the system. Go back for more training, sponsor a vendor visit to your site (perhaps an extension of a demo for a new system prospect) and have a system rep. go over unfamiliar or little-used features and functions, and reread the documentation. It is a general rule of software life that over time users tend to live within a very narrow "comfort zone" on any system. If you don't

watch out, you may find yourself increasingly *uncomfortable* with what you perceive to be system limitations in that so-called "comfort zone" and come to believe that your system is lacking in key features. You'll be starting prematurely, or perhaps with no justification at all, down the road once again that we have just traveled. Are you really that much of a masochist? To use the metaphor referred to above, maybe all you need is a little "marriage counseling!"