

# 12 Deadly Mistakes of Software Selection

**The** selection of business software is far from being an easy task; with more products coming on the market everyday; more low-end products reaching up; more high-end products reaching down; and with mergers, acquisitions and name changes, it is actually getting more confusing.

In the midst of this confusion it is tempting to look for short cuts in your software selection process. Whether you decide to hire a consultant to help or go at it by yourself, it is crucial to be aware of the pitfalls you should avoid. In this article we provide some pointers on how to recognize and avoid what we consider to be the 12 “deadly” mistakes of software selection.

## 1. Buying the same software that your industry peers have bought.

We placed this as the number one mistake because it embodies so many errors in understanding. Here is what is so deadly about this mistake: First, it assumes that industry peers are qualified to make good software selection decisions. Second, it excludes many other more important items such as analyzing how they made their selection, what their criteria was, how much their total cost of ownership is, and how well the software has met their needs. Last, it fails to consider your own unique requirements and differences. A simple illustration will help clarify what we mean.

Assume two companies are in the same industry in the same town doing roughly the same magnitude of business. Now look at their different histories. (See Figure 1)

As you can see the list could go on and on as to major differences in the history and needs of

these two companies and how they see the world that shapes their perception and approach to software. What company “A” sees as too complex and cumbersome, company “B” sees as flexible and feature rich.

Consider what other companies are doing, but buy the software that your company needs.

## 2. Buying software based on features alone.

When you buy software on features alone you overlook other critical factors such as: scalability (software’s ability to grow with your company), flexibility (ability to meet your changing needs, rather than you having to conform to the software), excess complexity (the software is too complex and cumbersome for you to effectively implement and manage), technology fit (you don’t have the technical sophistication to use and manage the software), cost (the total cost of ownership is more than you can afford), support (insufficient support resources exist where and when you need them), and cultural fit (the

software, software vendor, and/or implementation vendor are so incompatible with your business culture that you are not able to build a partnering relationship).

Figure 1

Company A	Company B
New Company	Old Established Company
Flexible policies	Successful Routines
Coming up from a DOS platform	Coming down from a mainframe platform
Wants off-the-shelf package	Use to customizing
Intends to do e-commerce	No interest in e-commerce
40% of sales international	100% domestic sales
In western states	In all 50 states
Central Warehouse	Decentralized focus
Needs HR/PR	Outsources HR/PR
Cash strapped	Cash cow
Analytical emphasis	Transactional emphasis
Buys into Intel/Microsoft strategies	Likes AS/400 Platform
Has a Euro dollar problem	Has a Year 2000 Problem

### **3. Buying software without proper consideration of the Software Company.**

We recently ran into this problem when we were talking to a group of company CFO's about their software choices. We basically said, "The software product is a superb fit for your company. It can handle the small divisions and large divisions. It has the right modules. It has the right blend of complexity vs. usability. The technology is very current. It will be your best-priced option, but stock prices are dropping, executives are leaving, morale is weakening, financing has dried up and the future of the company looks bleak."

It is important to consider a vendor's momentum, litigation issues, profitability, ability to successfully deliver its new product or revisions, employee and senior management turnover rate, employee morale/attitude, business focus, leadership vision, and strategic direction.

This is the general problem with using books, databases, surveys, etc. that compare your requested features with features the product has. It is stagnant data that does not ebb and flow, not to mention that the data in the survey may be inaccurate or incomplete.

### **4. Buying software without seriously focusing on the implementation service provider selection.**

This one is far more deadly than you might imagine. You spend months or even years matching your requirements to software options and then just take whatever implementation service provider that comes along with the territory. Except with lower cost software (even then there are usually 10's and 100's of possible choices of who can sell, install, implement, train, and support your company on the software), the implementation service provider is a big part of the equation.

On a rating scale of 1 to 10, consider the following scenario. Assume choice "A" was software with a rating of "9", and an implementation vendor with a rating of "5". Assume choice "B" was software with a "7" rating and an implementation vendor with a rating of "8". We would probably choose option "B".

In too many instances, we have seen major software change requests when there was nothing wrong with the original software. The original software was never implemented properly, or people were insufficiently trained to use its features as a tool to do their jobs.

Remember, it is not the software that helps you; it is the implementation and use of the software that helps you.

### **5. Buying software that has lower initial costs but significantly higher ongoing costs (especially hidden internal costs).**

We have, unfortunately, seen a major company make this mistake. Their decision criteria were heavily weighted towards initial costs. They had an inkling that this was the wrong approach, but felt politically tied to it. Later on, they realized that the true cost of upgrades involved a several month mini-implementation process, the software required excessive consumption of scarce Information Technology ("IT") human capital, and there were unnecessary reconciliation processes for most concurrent users. The maintenance and support requirements were so overwhelming that they had to cancel other current projects, and scale back pending projects due to the unforeseen drain on IT resources.

### **6. Buying software with the input of a small elite group that did not get the buy-in from the organization at large.**

You need the input and buy-in from the constituent parts that make up your organization. You cannot leave out IT, departmental heads, power users, or foreign branches. The value proposition for the usefulness of software begins in the selection process. If inside politics shape the deal, if outside interests prevail, or if a biased attitude is used, the implementation will usually be problematic. It will be another one of those corporate HQ initiatives that does not go anywhere because the appropriate feedback and buy-in were never secured up front.

This does not mean you have to extend the process and interview everyone. However, the representative groups, including front-line people, should be heard.

### **7. Buying popular software without considering all the options.**

It is a popularity contest out there much of the time. Too many companies buy software because the vendor has the most customers, the greatest revenue, and/or has the highest magazine or analyst ratings. They fail to examine why the vendor has the greatest number of customers or revenue, what the criteria for the ratings were, and whether the software or vendor can meet their own unique requirements and needs. We deal with over 7,000 business software products and are welcomed in a selection because we are open

to so many options that would never have been considered. The popular software might not have even been on the original long list, yet they get onto the short list, and eventually are selected simply because they appear to be the most popular software. What rich and fertile software packages are generally left out of these popularity contests?

- *Foreign entries that are very strong overseas. (We are so provincial in our thinking in the U.S. that it even spills over into software selection.)*
- *New start-ups with a vision, and the momentum that is taking them well into the scope of execution.*
- *Old has-beens that have regenerated themselves to the surprise of onlookers.*
- *Software labeled with a description that inadequately describes its potential to your company.*
- *Software that has no “dot” on a Gartner chart.*
- *Software that is not implemented by any of the “Big Five” accounting firms because it goes in too easily.*
- *Software that was specifically developed by and for your vertical industry.*

There are a wealth of viable options out there that could be your best buy and competitive edge, but they would be overlooked if you were only looking for popular software.

### **8. Buying software that is more complex than you need.**

It is our observation that people tend to overbuy not under-buy. For example, consider all of you reading this article. If you paused for a moment and thought about it, you could more than adequately be served by Microsoft Works instead of the more powerful Microsoft Word. We like the glitz, the bells, and the whistles, even if we never use them. This puts an unnecessary burden on your IT resources and results in a lot of idle capacity of software potential. Worse yet, it results in buying software that your internal resources can not adequately implement and support.

### **9. Buying software without properly defining your requirements.**

In this day and age? Yes, but different than you think. Many companies over define their requirements and end up with so many of them that they lose the clarity, focus, and manageability so essential to an appropriate selection. We believe that companies should be able to put their “differentiating criteria”

on just a few pages. These represent the items that are unique and point out why some software is just a better fit for your company. It helps you avoid buying software that is missing a “deal killer” requirement, thus rendering it impossible to use (such as serial number tracking, if you are in the expensive parts warranty business). It also allows you to determine why certain options should be clearly eliminated from consideration.

### **10. Buying software that is at the end of its product life or the very beginning of its product life, at either extreme.**

Software is so complex these days that it is sometimes easier to do a complete rewrite rather than try to patch the old software. It is important to track the genealogical history of the software versions to get a feel for where it has come from and where it is going. When you are evaluating software on the short-list you should find out where it is headed in the future, when the next version is due, what it is expected to contain, and when someone would rationally consider being able to have enough support, training, bug fixes, etc. to implement the software successfully.

Because software is so complex, even if a large company comes out with a new version 1.0, rarely, if ever, are the features complete. It might still be in beta testing, be unstable, or lack modules you need. They have release deadlines to meet and like any 1.0 product or upgrade, for that matter, developers have difficult choices to make about how much they can code before the release deadline. And, if the software is relatively bug free, new versions usually do not have adequately trained implementation sources, training resources, or full technical support. If you like to be on the bleeding edge because you have a medical 911 IT support group, then consider it, but know your company is the exception not the rule.

### **11. Buying software based on dying technology.**

Another mistake that companies make is buying software based on a hardware platform, database, architecture and/or operating system that is declining, no longer supported, no longer taught in the schools, is too inefficient, inflexible or slow compared to the alternatives, or has too small of a customer base to survive in the future. This puts you into a corner with few alternative paths.

## 12. Buying software to fix business processes.

Software is an enabler that might help you focus attention on your broken processes, but purchasing software does not fix them. It might give the illusion that something is being done, but it is not the fix. Good software gets implemented successfully at good companies because:

- They have a capable team and executive support.
- The team is empowered to make quick decisions.
- There is a clear focus on correcting problems.
- They take advantage of new technologies.
- They reward the team.
- They are dedicated to train their users.

- They know how to utilize the software's capabilities as a tool to help them meet their business objectives.

When these 12 “deadly” mistakes are avoided, we know that the software selection and implementation are going to succeed.

Hopefully, you can use these tips to help you with your selection process. If you would like a copy of our booklet, “The Essential Guide for Selecting Today’s Business Software” or a copy of a speech we often give entitled, “What is the Optimal Software Selection Process?” email us at [mail@softresources.com](mailto:mail@softresources.com). If you would like some help on your project, we would invite you to look into our services. ☎ [www.softresources.com](http://www.softresources.com)

---

*This article is based upon the observations and real-world experiences of SoftResources LLC. SoftResources is an unbiased software selection company that offers four main areas of services: 1. Business System Assessments, 2. Software Selection, 3. Implementation Vendor Selection, and 4. Contract Review and Negotiations. To maintain their objectivity, unbiased approach and focus on their client's needs, SoftResources refuses to sell or implement software, or to undertake any other activities, which would compromise their unbiased focus.*



SoftResources LLC 2517 Eastlake Avenue East, Suite 100 Seattle, WA 98102-3278  
Phone (206) 860-2400 Fax (206) 860-2828 Web [www.softresources.com](http://www.softresources.com)