Maximizing Value, Minimizing Cost in Software Selection

ccpace

Table of Contents

I. Introduction: Why Purchase New Application Software

There is only one reason to consider implementing new application software. Your business wants/needs to do something that your current system (if any) is not able to handle and doing the process manually is just not efficient. The need can be strategic, tactical, revenue generating, market differentiating, or (most likely) expense saving.

Software functionality is a result of three variables: the requirements it is meant to fulfill, the time period during which it is developed, and the skill of the individuals who develop it. The biggest problem with systems written as little as five years ago is obsolescence; **system capabilities and business models have changed so dramatically that the best systems of yesteryear do not leverage the capabilities of technology and the requirements of business processes today.**

So, what's a manager to do? The most popular choices are to upgrade existing systems, purchase a software package, or some combination of the two. There's also the option of leaving an outdated software system in place, but in an economy where technological advantage is often synonymous with business advantage, the competitive implications of inertia can be significant.

Key Issues

Implementing an application software package demands significant investment of corporate resources; a structured process can mitigate many of the risks.

Software Selection Quotes In the News

- *"The average ERP project will cost over $10 million and will take more than a year to complete." Tech Republic*
- *"The purchase price of software makes up only 20% of the total cost of owning a software package." Tech Republic*
- *"During a five year life cycle, GartnerGroup expects that many enterprises implementing application software packages will find that software licensing costs constitute less than 5 percent of the total project cost." GartnerGroup*
- *"50% of packaged application projects fail because evaluation criteria are misunderstood or incomplete." GartnerGroup*
- *"Realizing that world-class leaders spend far less than average to process various transactions, enterprises are trying to streamline operations, and many look to technology for solutions." GartnerGroup*
- *"Realizing that world-class "Envision the selection process as a car trip: Before setting out, make the significant decisions about where to go, how to get there, and how much to spend." Computerworld*

II. Why This Isn't Your Average Decision

Business managers are in the decision business, choosing a software system (or choosing to build your own) should be just another day in the office, right? Maybe not. Consider your risks:

- Process Change: Any software will introduce radical change in the way that your company does business. Change without proven results is always risky.
- Learning/Customization Expense: Once you choose a package, you must invest staff time and money to learn about the package and to make necessary customizations. If the package is a poor choice, these charges become worthless and you must start again.
- Vendor Viability Risk: All vendors promise ongoing support, but the landscape proves that not all deliver. Even if you choose the right package, having the vendor drop support is a risk most companies cannot afford.

Software Evaluation Difficulty: Software is by its very nature intangible; often the sales capability of the vendor is clearer during the purchasing process than is the functionality of the software. For such an important decision, a lot is left to subjective chance.

III. Top 10 Common Mistakes of Software Selection Projects

1) Not gaining buy-in from all organizational areas affected (note: this may include your supply chain and/or customer chain).

2) Maintaining the status quo: automating inefficient processes or processes that don't take advantage of available technologies.

3) Developing criteria as you look at software: bells and whistles often look better on the sales floor than they do in operation.

4) Believing everything the vendor tells you: well-written scenarios that the vendor should demonstrate will serve you better than sales person hype.

5) Devoting only part-time resources to the project: if your project staff is working only part-time on the project, how can you expect optimal results?

6) Not considering process change implications: the organization must be willing to adjust to the software; don't expect the software to do all the bending.

7) Confusing desirable features with essential ones: a close look at your business goals will help you prioritize what's essential and what's not.

8) Fuzzy purpose: set the project expectations and scope upfront. What is your desired business result? What objectives (short-term or long-term) are you expecting to achieve?

9) Not considering vendor financial viability: all of the best technical capabilities don't help your organization if the vendor is going out of business and can't help you support the software.

10) Unrealistic cost projections: make sure you factor all relevant costs including customization, training, ongoing support, and data migration (if any).

Industry Lingo

- *Application software packages are prewrittenby a vendor to provide a set ofstandard functions usable by a variety ofcompanies.*
- *COTS (Commercial Off the Shelf) and Out ofthe Box are other terms that refer toapplication software packages.*
- *ERP (Enterprise Requirements Planning) isa kind of application software package thatencompasses many functions, such asproduct planning, parts purchasing,maintaining inventories, interacting withsuppliers, providing customer service, andtracking orders.*
- *CRM (Customer Relationship Management) is a type of application software package that improves revenue by providing acustomer-centric focus on sales andcustomer service processes.*
- *Software Tools, as differentiated from packaged software, are often products that enable technical staff to configure or develop systems but do not directly provide users with any functionality.*
- *Beta release or version is a first release program in which a software application or tool vendor works with clients to test a new product release in practice.*
- *RFI (Request For Information) is a preproposal document distributed by your company asking prospective vendors to describe their products.*
- *RFP (Request For Proposal) is a more formal document distributed by your company asking vendors for detailed proposals,emonstrations, and references includingproduct characteristics, cost data, and implementation estimates.*

IV. Software Selection Activities at a Glance

While every project is unique, there are major elements common to all software selection projects. CC Pace has developed SelectExpress™, an effective and flexible software selection methodology, that helps businesses make the best choices for their specific needs. Below is an overview of the methodology. The following pages provide the details and milestones for each activity.

CC Pace's Software Selection Methodology Eight Major Activities and Milestones

1. Initiate Project
2. Check Industry and Best Practices
3. Document Requirements
   - Strategic Requirements
   - Functional Requirements
   - Vendor Requirements
   - Technical/Operational Requirements
   - Interface/Conversion Requirements
4. Analyze Packages and Select Finalists
5. Analyze Build Vs. Buy Options (write new application vs. purchase packaged software)
6. Review Selected Finalists
   - RFP
   - Vendor Demonstrations
7. Document Recommendation/Gain Management Approval
8. Develop Implementation Cost and Staffing Plan
   - Upgrading of Infrastructure (HW, Network)
   - Installation
   - Enhancement/Customization
   - Existing System Interface Development
   - File Conversion/Migration
   - Training/Change Management
   - Documentation and Support

IV. Software Selection Activities at a Glance - Activity 1: Initiate Project

While this activity may seem self-evident, there are several steps that, while easy to forget, are crucial for project success. Questions you should consider include:

- Does the project team include all staff (business and technical) with a vested interest in the project outcome?
- Is the project sponsor powerful enough (in position and in reality) to implement the project?
- Is the project scope clear both technically and from a business perspective? Does it include approximate calendar and budget?
- Are project communication processes and calendars clear? Are there regular written and/or oral communications between the project team and the sponsor?
- Have the issues that are out of scope been clearly identified?
- Have key organizational issues (a.k.a. sacred cows) been identified and factored into the project plan?
- Have sources of project risk been identified and risk mitigation strategies developed?
- Is there a clear project calendar and budget with exception processes identified should they not be sufficient?

The milestones for this activity include project scope charter (including approximate budget and calendar), project team identification, and project sponsor identification.

IV. Software Selection Activities at a Glance - Activity 2: Check Industry and Best Practices

Application software packages support processes, which in turn support specific business models. Before specifying software requirements around your current business model and process, you should analyze how other firms have improved their processes and business models through innovative use of technology. While ame-too strategy rarely achieves competitive advantage, much can be learned from analyzing trends that have benefited (and hurt) others. Sources for this analysis may include:

- Industry trade associations
- Business periodicals and websites
- Web searches on (perceived) marketplace leaders
- Innovative business and technology publications
- Analyst organizations

Questions that should be answered include:

- Is there a competitor in my industry who is innovative in the process I am looking to automate? What are their business results (revenue increased, costs decreased)?
- Is there a firm in another industry that is doing something with the process that I can copy?
- Are there any horror stories of mistakes made (within or outside my industry) that should be considered in my project?
- What do the pundits say about the evolution of this process in the future? Are there new technologies or market trends that I should anticipate?

The milestones for this activity include a report on industry and technology trends relevant to the process being analyzed as well as a report on lessons learned (good and bad) from other known competitors and industry neighbor implementations.

IV. Software Selection Activities at a Glance - Activity 3: Document Requirements

Identifying requirements is probably the self-evident task in software selection. However, it's easy to overlookfunctional requirement categories that do not address immediate need and to underestimate the need for prioritization to enable trade-offs. Documented requirements should contain:

- STRATEGIC REQUIREMENTS: These are business drivers that will change over the term of the installation. Theymay include the firm's anticipated volume, anticipated businesscycles, changing competitive differentiation, and anticipated industry changes.
- FUNCTIONAL REQUIREMENTS: These are process and business related functions thesoftware is expected to accommodate. This may include the functions of the current systems, the functions the current system is no longer able to perform, the anticipated process flow for the new application, the volume requirements of the new application, the quantity and specifics of system reporting requirements, and/or the system's ability to handle legal and/or tax changes. It is especially important to quantify functional requirements so that trade-offs can be accommodated.
- VENDOR REQUIREMENTS: These are the minimum set of vendor qualities needed to provide the team comfort in dealing with the product vendor. These may include a vendor's years in business, market capitalization, number of successful installations, vertical industry of vendor references, number of consultants trained in product, lessons learned in the company from past vendor projects, and the firm's willingness to choose a vendor with a trial (beta) project.
- TECHNICAL/OPERATIONAL REQUIREMENTS: This is the area of requirements definition most focused on the technical specifics of the application software. Issues include compatibility between the system's operating environment and that of the firm's other applications, and evaluation of the system's stability, modifiability, security, and documentation. System scalability (for both software and hardware) should be considered as well, since the new software system should be able to handle volume increases without a significant change in the firm's hardware and systems infrastructure.
- INTERFACE/CONVERSION REQUIREMENTS: The new application package will most likely need to integrate with existing applications. There may be a great deal of existing data that must be converted by the firm into a new system format. By documenting the description and volume of data that must be exchanged between the systems, as well as the prospective system's ease of interface and conversion, the project team can further qualify potential software packages.

The milestone for this activity is the Applications Requirements Document, which includes descriptions and prioritizations of all software requirements.

IV. Software Selection Activities at a Glance - Activity 4: Analyze Packages and Select Finalists

The first challenge here is to find the 'universe' of potential candidate packages. Don't focus only on packages that are well advertised. Expand your search using industry associations, IT analysts, technical and industry publications, and peer contacts. There are specialty organizations, conferences, and websites that specialize in many application types (including ERP, CRM, Sales Force Automation, Supply Chain Management). Make sure that you differentiate software tools (see glossary) from software packages, since vendors often market the capabilities of the two interchangeably.

Before you begin evaluation, develop your criteria. These should be a subset of the criteria identified in Activity 3, since the purpose here is to narrow the field to the best-fit alternatives.

This review is preliminary, unlike the detailed review that takes place in Activity 6. Your purpose is to develop a high-level analysis of the overall costs, capabilities, and closeness of fit between packages on the market and your capabilities. Depending on the time and effort available as well as the importance (risk) of the implementation, you may analyze packages based on generally available information or you might want to issue a general RFI (request for information), which requires prospective vendors to provide information to you in a standardized format. The benefits of an RFI include standardization of response and the ability to have vendors answer specific questions relevant to you. The costs include the time to develop and review the RFI.

A spreadsheet is a good tool to evaluate packages by comparing and weighting scores on the various criteria. The outcome of this activity should include a list of finalist products (3 – 5 usually) as well as ballpark estimates of cost of software, degree of fit, and time (and cost) of customization.

The milestones for this activity include the evaluation matrix and the choice (and functionality/cost statistics) of the software finalists.

IV. Software Selection Activities at a Glance - Activity 5: Analyze Build vs. Busy Options (Write New Application vs. Buy Application Software)

Now that you have analyzed the best packages available, it's a good time to evaluate the costs and benefits of customizing a package versus writing your own. Given equivalent cost and capability, most organizations would choose to implement a packaged solution to ensure technology currency and minimize the cost and effort of ongoing maintenance.

The first step is to work with your technical staff to evaluate the approximate cost of building a new system (or a set of system enhancements) to meet your needs. The evaluation matrix from Activity 3 may be a good tool to describe your needs. Remember to factor into your analysis not just the cost of writing the system, but also of maintaining it in the future. This includes scalability (handling increasing volumes), incorporating new functionality, and leveraging new technologies.

The milestones for this activity are the build vs. buy decision and documenting the criteria on which the decision is based.

IV. Software Selection Activities at a Glance - Activity 6: Review Selected Finalists

This is the in-depth part of your software analysis. The three most important elements of detailed review are system demonstrations, review of vendor proposals, and reference checking.

Review of Vendor Proposals: Many companies write Requests For Proposal (RFPs) to which they askvendors to respond. This enables a more detailed description of vendor capabilities in a standardized format.RFP questions can be as specific or as general as you would like and should conform to the requirements matrix you developed in Activity 3. As with RFIs, the benefits of RFPs in terms of standardization and breadth of information collection trade off with the costs and time associated with RFP preparation and vendor review.

Demonstrations: Setting a level playing field for vendors is important for purposes of comparison, so we recommend developing specific scenarios that you want each vendor to demonstrate. In order to test standard vs. custom package functionality, you might want to watch each vendor install the systems and give them a limited time (under the review of your technical staff) to customize the product. If the software package is expensive enough (usually $100K or more), the vendor will be willing to work with you.

Reference Checking: Two factors help a great deal with reference checking: standardization and networking. Preparing a standard list of questions for every reference (reflecting your evaluation criteria) helps to gather equivalent information from every reference check. Networking (including using personal contacts, contacts featured in the press, and informally mentioned vendor sources) can help you to identify clients with whom the vendor may not want you to speak.

In the final analysis your decision will be based on a mixture of science, art, and intuition. Remember to document the rationale of your decision and your expectations so that you can go back and re-evaluate your decision later and improve your future selection process by comparing your expectations to reality.

The milestones for this activity are Consistent Vendor Evaluation Documents.

IV. Software Selection Activities at a Glance - Activity 7: Document Your Decision/Gain Management Approval

Gaining concurrence from your project and political hierarchies should be ongoing throughout the selection effort. However, when you are ready to make a recommendation, make sure that the multiple constituencies who will have to implement it agree upon the decision. This may include the technical staff, the business departments (even those only peripherally affected by the system), your supply chain, and even your customers. Think through techniques to achieve consensus. Often a one-on-one meeting with each group (or person) preceding the official project results is helpful. Once you have achieved consensus, make sure that you document it.

The milestones for this activity are the Decision Documentation and Sign-Off.

IV. Software Selection Activities at a Glance - Activity 8: Develop Implementation

So You've Chosen, Now What?

Now that you've chosen your software package, it's up to your team to plan for successful implementation because, after all, software alone does not achieve business results. Implementation plans include three key elements: staff, cost, and calendar.

STAFF: For many implementations, external staff or consultants are hired to provide skills in that specific product. These staff may be expensive and not trained in your business or systems, but they do possess knowledge about the software you plan to use. One caveat: software often follows fads and specialists in a popular software product may be both scarce and expensive.

COST and CALENDAR: A wise person once said that between the variables of cheap, good, and quick, any company implementing software could pick only two, since they negate the third (i.e., good and quick negate cheap). Calendar and cost estimates should be realistic and based upon the scope of work identified in your decision documentation.

It's important when you are developing your implementation plan to include all of the elements necessary for project success. This includes upgrading infrastructure (hardware, network, and system software), installing the software, customizing the software, developing interfaces, converting or migrating data, conducting training and change management, and providing documentation. Don't forget to plan for the cost and resources of ongoing help desk, application maintenance, and vendor support.

The milestones for this activity are Infrastructure, Installation, Customization, Interface, Conversion, Training, and Support Plans.

V. So, What's Most Important?

If your fairy godparent offered you five wishes for software selection and installation, we would suggest:

- Be very clear (gain organizational consensus) about what you are trying to accomplish from a business standpoint.
- Provide a level playing field (standard questions, demonstration scripts) in evaluating all software options (including custom development).
- Be comprehensive in formulating your project costs and implementation plans (include all process and technical activities both for the present and into the future).
- Learn from your mistakes; with proper focus, the more often you do this, the better you will get.
- Most importantly, measure your outcomes. The best vendors can go out of business and the best business models can become shattered by circumstance. Identify your risks, develop mitigation plans to address them, and constantly evaluate the costs and benefits of your system in light of your expectations. Remember, the project doesn't end when the software is chosen or installed but rather when the business results are achieved.

CC Pace

4100 Monument Corner Drive

Suite 400
Fairfax, VA 22030

(703) 631-6600

For more information on CC Pace,

please visit our website at www.ccpace.com