

ERP/SCM/eCOMMERCE: PACKAGE SELECTION PROCESS OPTIONS

A White Paper

by **Richard Luetgen**
ThoughtWing

This paper was developed to provide general background to assist clients in decisions related to selecting complex application software packages such as ERPs (Enterprise Resource Planning), SCMs (Supply Chain Management) and eCommerce packages. Please note that this paper presents professional opinions intended to apply generally and that clients must take appropriate care to evaluate them in light of their specific needs. ThoughtWing makes no representations, warranties or guarantees of any sort as to the applicability of the opinions presented in this paper to the specific needs of any client. Please also note that the techniques and materials presented in this paper are representative of a scalable selection methodology developed and used by ThoughtWing, and that other techniques and methodologies are commercially available.

The paper is organized as follows:

1. Overview
2. The *Classical* Approach
3. The *Intermediate* Approach
4. The *Proof of Concept* Approach
5. Conclusion.

1. Overview

Companies have employed many methods of selecting software packages, from the most rigorous at one extreme to selection at the other extreme based on prior experience of a key executive, with no discovery or extensive consideration of how a product satisfies real requirements. Care should be taken by a company in such matters to apply *appropriate* rigor to the process, for the following reasons:

- Acquisition and implementation of packages, particularly complex suites of applications such as ERPs, SCMs and eCommerce software, constitute a very expensive initiative, often requiring outlay of millions of dollars.
- It is a risky proposition. As much as 85% of such initiatives fail, when definition of failure includes inability to satisfy expectations – the project took too long, cost too much, did not deliver what was promised. And, in a significant number of instances, the pain becomes so great that the project is scrapped and the investment written off. Once in production, if the software fails to operate as intended the failure can have disastrous consequences, including dramatic involuntary reductions in shipments, inaccurate inventory accounting, pricing and invoicing.
- It is a highly disruptive process. It requires active and dedicated participation by key personnel over extended periods of time, just the people who are most needed elsewhere.
- Due to the high cost of acquisition and implementation, most companies will need to live with and use the software for years before they can afford to replace it with something else.

For these reasons, selecting products is a process that should be done once and done right. However, needs of companies in this regard differ. In determining real needs, a judgment balancing cost with risk needs to be developed.

Due to the high risk and substantial expense related to complex package software selection it is also in a company's interest to prepare extensively and early for both selection and eventual implementation of the products. One of the most important ways of doing this is to consider business process reengineering (BPR).

Business process analysis of some sort is inevitable in such implementations. The flexibility of current product offerings provides the opportunity to optimize business processes across the enterprise. On the other hand, remaining constraints in software's ability to support process variations can force companies to decide, even when they had not anticipated the need or desired it, to modify either a business process or the software itself. Each option has its significant costs, which need to be understood for true costs to be fully understood.

Accordingly, a company should prepare for a selection by identifying process areas where improvement is desired, and in quantifying the likely time and capital required to improve them. Such an exercise also paints a more realistic picture of the process requirements candidate products need to satisfy.

The remainder of this paper deals with the three major approaches to selecting package software, which are:

- The *Classical* Approach, which offers the most rigorous selection process,
- The *Intermediate* Approach, which offers less rigor but also less cost, and
- The *Proof of Concept* Approach, increasingly popular, which is likely to require least cost but provide greatest risk.

2. The Classical Approach

This approach is called *classical* because it is the approach used for many years, primarily by consultants, to apply a high degree of rigor to the selection process.

The *Classical* Approach consists of:

- A. Initial, high-level analysis of a client's supply chain and other business process, transaction processing and information requirements; determination of a client's major process and other objectives; identification of no more than seven, preferably no more than five candidate vendors who are most likely to satisfy needs.
- B. Comprehensive definition of business process, transaction processing and information requirements, as well as technology infrastructure requirements, including facilities, equipment and telecommunications needs.
- C. Development of *vendor-related materials*, including a *Request For Proposal* (RFP) document that formalizes business and other requirements. The *RFP* is used by the identified vendors as a turnaround document to identify the degree to which their products satisfy requirements; more due diligence might follow to determine the degree to which vendors' responses conform to reality.
- D. Vendor-related materials also include transaction mapping and scripting of important end-to-end transaction streams, such as the order-to-cash cycle. Such scripts are used as the basis for a series of high-level and perhaps more detailed structured demonstrations of the candidate product sets by the vendors, attended and scored by client personnel.
- E. Based on the *RFP* responses and the scores obtained during the product demonstrations, the field then is narrowed to no more than three candidates, preferably two.

- F. During this time, preliminary negotiations with the vendor candidates take place over software license and annual maintenance fees, and over cost of services, such as training, and preliminary results are included in candidate scoring. Integrator (a consulting firm which specializes in implementing specific software products into existing business environments) candidates also are identified at this time (or a vendor might strongly recommend one), and investigations of and preliminary negotiations with these candidates also take place.
- G. Based on the scoring, a final selection takes place; final negotiations with the selected vendor and integrator(s) then take place for license and maintenance fees and for the cost and availability of services. In some cases such final negotiation takes place with more than one vendor, and more than one integrator, in order to provide a competitive atmosphere that the client may exploit.
- H. The selected and acquired products then are implemented by prototyping them. During this process, internal company personnel, usually assisted by integrator consultants, model existing business processes, or create new policies, procedures and measurement criteria for processes, for the software.

Decisions also are made to modify current processes or to modify the software to accommodate existing processes. Changes are effected, training takes place, historical data is converted, and the final prototype is rolled out to the organization after thorough testing.

Some companies, usually very large ones desirous of minimizing the risk associated with selecting software, might defer final decision pending concurrent detailed prototyping of *multiple* products, selecting the set that more easily conforms to current business processes or anticipated future needs.

With respect to the initial recommendation of likely vendors and integrators, it should be clearly understood that while many reputable consultants would perform this service, none would offer guarantees or warranties of any sort regarding the fitness of the recommended software or integrator(s) to actual needs.

3. The Intermediate Approach

An *Intermediate* Approach is used that takes on greater risk but saves much time. It consists of expediting the narrowing process by adopting a recommendation from a consultant as to the three candidates most likely to satisfy probable requirements, *without actually defining detailed requirements*. The consultant also recommends potential integrators.

The initial study required by the consultant to gain an understanding of the environment also is significantly expedited. Mapping of key transactions still takes place, as in the *Classical* Approach, and demonstrations take place and are scored as above; but no *RFP* is developed.

Selection is based primarily on the demonstration scoring, and implementation proceeds immediately on selection and completion of vendor and integrator negotiations, as in the *Classical Approach*.

As with the *Classical Approach*, many reputable consultants would perform the initial vendor and integrator recommendation service, but none would offer guarantees or warranties of any sort regarding the fitness of the recommended software or integrator(s) to actual needs.

4. The Proof of Concept Approach

The last alternative approach might be used when the need is pressing to select and implement very rapidly. In this *Proof of Concept Approach* only one vendor/product set and one integrator is considered, usually offered up by a consultant as highly likely candidates. A presumption of fitness of the product is established, which is proved or disproved by the actual prototyping process. If proved, the product is ready to be rolled out; if disproved, the process starts again with another product, or another approach is taken.

Often, no commitment is made to a vendor until the concept is proven, and the vendor might provide an evaluation copy of the software at no cost until commitment is made. Negotiations regarding cost take place during the prototyping period. As above, reputable consultants would not make guarantees or warranties of any sort as to the actual fitness of the products or integrator(s) they would suggest.

5. Conclusion

The three approaches differ dramatically in three ways:

- A. The least business risk of failure is offered by the *Classical Approach*, while the greatest business risk of failure is offered by the *Proof of Concept Approach*. High business risk of failure is defined as software insufficiently supportive of current or anticipated business processes to allow it to be implemented without impractically extensive software or process modification.
- B. The *Classical Approach* is likely to require the most time and the greatest cost through implementation, while the least time likely to be required would be for the *Proof of Concept Approach*.
- C. The degree of dependence on consultants is usually lowest when using the *Classical Approach* and highest when using the *Proof of Concept Approach*.

In the *Classical Approach*, a structure is imposed on the process that relies extensively on internal company personnel to perform the actual work, with navigation and advice from consultants; and, more time is available to pass through the process.

In the *Intermediate Approach*, the process necessarily becomes more highly structured as time becomes a driving factor. Ready availability of specific skills also becomes mandatory, so consultants become more important to the process.

In the *Proof of Concept* Approach, requirements for structure, facilitation and other specific skills become critical as time becomes the overriding factor to be managed. Consultants, therefore, become very important.

The more complex a company – large in terms of revenues, geographic reach and employee head-count, multiple divisions, etc. – the more advisable the *Classical* Approach becomes, in order to effectively manage the correspondingly large risk and expense.

However, when speed or cost are overriding factors, the *Intermediate* and *Proof of Concept* approaches offer enhanced ability to meet objectives, so long as the greater risk of failure is well understood and can be effectively managed.