December 2002

MAKE OR BUY OR BAG IT: THE SREF FRAMEWORK FOR ANALYZING IT OUTSOURCING DECISIONS

Stanley Birkin
University of South Florida

T. Grandon Gill
University of South Florida

George Steigner
University of South Florida

MAKE OR BUY OR BAG IT: THE SREF FRAMEWORK FOR ANALYZING IT OUTSOURCING DECISIONS

Stanley J. Birkin
University of South Florida
sbirkin@coba.usf.edu

T. Grandon Gill
University of South Florida
ggill@coba.usf.edu

George Steigner
University of South Florida
gsteigner@coba.usf.edu

Abstract

The "make or buy" decision for information technology applications, systems and infrastructure has been the subject of extensive analysis in the IT literature over the past decade. While outsourcing has garnered enthusiastic support from many quarters, a number of cautionary notes have also been sounded, and despite auspicious starts, many outsourcing arrangements have been less successful than had been expected. In this paper, we propose a framework intended to help managers weigh the benefits of outsourcing the development of an IT project against the negative consequences. The framework, referred to as the SREF (Strategy-Risk-Economics-Future benefits) framework, is derived by marrying portfolio theory of selecting IT projects with recent research findings relating to outsourcing versus in-house development decisions.

The "make or buy" decision for information technology (IT) applications, systems and infrastructure has been the subject of extensive analysis in the IT literature over the past decade. Interest in the subject has grown particularly high in recent years with the advent of application service providers. While outsourcing has garnered enthusiastic support from many quarters, a number of cautionary notes have also been sounded. Despite auspicious starts, many outsourcing arrangements prove to be less of a panacea than their champions had hoped. Indeed, outsourcing arrangements frequently become more contentious as customer and vendor gain experience working with each other, rather than becoming more comfortable as would be expected.

In this paper, we propose a framework intended to help managers weigh the benefits of IT outsourcing, particularly in the area of application development, against the negative consequences. The framework is derived by marrying portfolio theory of selecting IT projects (McFarlan, 1981), with research findings relating to outsourcing versus in-house development decisions. We refer to the proposed analytical technique as the SREF (Strategy-Risk-Economics-Future benefits) framework, named after its key dependent variables.

IT Outsourcing: Conventional Wisdom

It is beyond the scope of this paper to review and synthesize the many IT outsourcing books and articles that have been published over the past decade. (Useful reviews of the subject can be found in Klepper and Jones, 1998; Clark, Zmud and McCray, 1995). In examining this literature, we found a number of themes that have consistently emerged:

1. *Many current trends in IT seem to favor increases in outsourcing relationships* (Klepper and Jones, 1998, p. 35-36; Clark, Zmud and McCray, 1995). Among these are: emergence of standards for information exchange and electronic markets (Malone, Yeats and Benjamin, 1989) making outsourcing easier to initiate; difficulties in hiring highly competent IT personnel in non-IT companies (Pearlson, 2001); increasing availability of outsourcing vendors and new vendor options (Sprague and McNurlin, 1993); and, an increasing experience with outsourcing relationships.

2. *Outsourcing of systems conferring competitive advantage should be avoided* (Sprague and McNurlin, 1993; Venkatesan, 1992). It is also noted that the strategic/non-strategic distinction is not always an easy one to make, and is subject to political pressures within the organization (Lacity, Willcocks and Feeny, 1996).
3. *Outsourcing areas of the firm's core competence should be avoided* (Klepper and Jones, 1998).
4. *All of IT management should never be outsourced* (Klepper and Jones, 1998; Sprague and McNulin, 1993).
5. *Outsourcing tends to be considered only where significant financial gains can be realized from the process* (Sobol and Apte, 1995; Sprague and McNulin, 1993).

Numerous researchers and practitioners have also warned of the practical perils of outsourcing (Clark, Zmud and McCray, 1995; Fried, 1995), sometimes summarized as abdication of control, high switching costs, lack of technical innovation and loss of ownership (Pearlson, 2001). This basic conventional wisdom on outsourcing does raise a number of questions that have been identified in the literature:

- If we deem an application or system to be truly strategic to our company, shouldn't we choose the most competent possible developer—even if it happens to be a third party?
- How do we account for the fact that an area that is not currently a core competency of the organization today may need to become one if we are to survive in the future?
- If we go beyond simple economic analysis in choosing internal applications, why should mildly unfavorable short-term economics be a screen that precludes consideration of outsourcing?
- If outsourcing decisions are frequently political in nature, how do we prevent such politics from subverting the process to the detriment of the organization?

IT Portfolio Analysis

Many of the same questions that managers must address in considering selective outsourcing decisions must also be addressed in considering an organization's overall choice of IT projects. One approach that has been widely used is the portfolio approach (McFarlan, 1981, Cash, McFarlan and McKenney, 1988) which is based on the following:

- Because individual managers are likely to have their own preferences for risk and visibility in choosing projects that may not mirror organizational needs, some overall strategy for choosing among alternative projects is critical if an optimal mix of projects is to be obtained.
- One approach to developing such a strategy is to view an organization's choice of projects in portfolio terms, with each project having its own risk characteristics
- The organization then establishes a target portfolio for its projects (determining both total funding and targeted level of risks). The key criteria used in establishing the portfolio goals are typically 1) the degree to which IT currently contributes to the strategic goals of the organization, and 2) the degree to which that contribution is expected to change in the future. A simple grid showing the impact of these two factors on appropriate portfolios is presented in Figure 1.
- Managers, in proposing IT projects, are then held accountable not only for the expected benefits of the projects they propose (e.g., ROI, payback) but also for how well the mix of projects they propose meet the organization's portfolio goals.

In a full portfolio analysis, an organization would typically take the "future importance" dimension and become far more detailed—identifying the specific technologies, architectures, applications, etc. expected to become critical over time. Managers would then be graded on their ability to propose quality projects utilizing such technologies.

Portfolio analysis has traditionally been used in choosing which IT projects are developed and which are discarded (the "make-or-bag-it" decision) and determining an appropriate balance of maintenance versus development activities. We find, however, that the types of analyses conducted in developing a portfolio strategy are also highly relevant to the make-or-buy decision. Concerns raised about the risks of outsourcing nearly always focus on how the outsourcing decision will impact the organization's IT positioning and its vulnerability in the future. Thus, we believe the process of performing a portfolio analysis will reveal highly useful information that can be applied to the outsourcing decision.

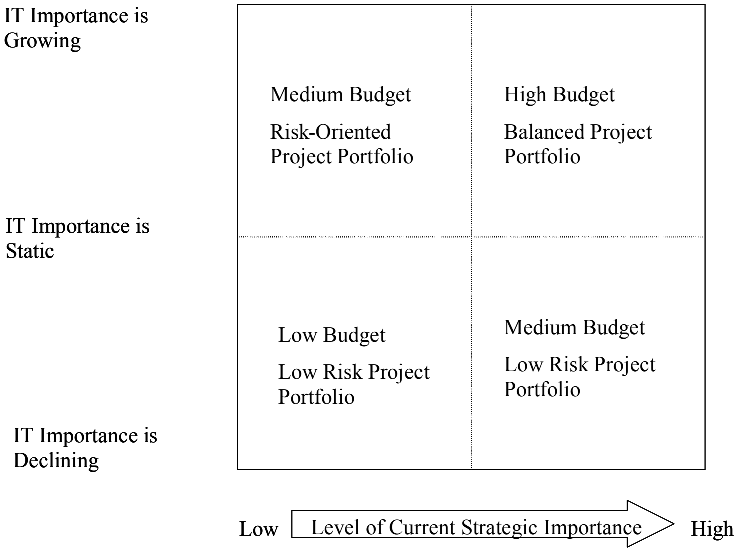
IT Importance is Growing
Medium Budget
Risk-Oriented Project Portfolio
High Budget
Balanced Project Portfolio
IT Importance is Static
Low Budget
Low Risk Project Portfolio
Medium Budget
Low Risk Project Portfolio
IT Importance is Declining
Low
Level of Current Strategic Importance
High

Figure 1. Simplified portfolio model, adapted from McFarlan (1981) Exhibit III

The SREF Framework

The SREF framework that we are proposing is an attempt to synthesize the portfolio approach to IT project selection with existing findings in research relating to outsourcing. For the purposes of the remainder of the present paper, we limit our scope in two important ways:

- First we focus our attention on the decision of whether or not to outsource application development, as opposed to the problem of whether or not to transfer ownership of existing systems.
- Second, we assume that, in applying the framework, a portfolio analysis has already been conducted, and that the project being analyzed has already been deemed consistent with the portfolio criteria.

In the SREF framework, we propose that four questions are particularly critical in choosing whether or not to outsource project development:

- *How strategic is the project in question to the goals of the organization?*
- *What is the difference in risk level between in-house and third party development?*
- *In what direction are the economic benefits (e.g., towards outsourcing or towards in-house development)?*
- *What, if any, future benefits are we likely to accrue in the event we choose in-house development over outsourcing?*

Strategic Level

Since analyzing whether or not a system is strategic is the mainstay of many an introductory IS course (e.g., Sprague and McNurlin,1993, Chapter 3), we choose not to dwell on the mechanics of such analyses (see Exhibit 1 for examples of the types of questions that could be asked). In the broadest terms, a strategic system tends to confer competitive advantages to its owner and, if the are sustainable over time, the relative strategic importance of the system grows dramatically. We also contend that some systems that may appear to be *support* systems are actually a form of *life support system*, in that their failure would preclude the company from doing business.

The conventional wisdom seems to favor the notion that strategic systems are poor candidates for outsourcing. There are, however, important counter-examples of strategic systems whose development was outsourced, such as DEC's XCON (Leonard-Barton and Sviokla, 1988). One common reason for using a third party to develop such systems was the failure of earlier attempts to build systems using alternative technologies more familiar to the organization. Thus, it is reasonable to believe that the "rule" that strategic systems should not be outsourced may be relaxed in the case of applications whose development exceeds the capabilities of the in-house team.

Relative Risk Level

The second factor in the SREF framework is the relative risk level between in-house and outsourced development. Again, using the fundamental framework provided by portfolio theory, we find that such risk typically arises from three sources (Cash, McFarlan and McKenney, 1988, 163-164):

1. *Project Size:* The larger the project, the greater the associated risk.
2. *Experience with Technology:* The greater an organization's experience with the specific technologies being employed, the lower the risk.
3. *Project Structure:* The less well defined a project, the more risky it is likely to be.

Because the determination of project risk would already have been done as a component of portfolio analysis, the focus of the SREF framework would be to assess the relative risk of in-house versus outsourced development. Furthermore, such a risk determination would likely need to be performed for each potential vendor. In Exhibits 2 and 3, we present two "sample" questionnaires that illustrate the types of questions that might be asked to assess risk, similar in intent to those provided by Cash, McFarlan and McKenney (1988, p. 166-168).

Economic Benefits

Since the proposed project has already met the firm's portfolio criteria, the primary economic issue is comparing the costs of in-house development with that of outsourcing for each vendor or contract proposal. The fact that economic analysis focuses on cost does not mean such an analysis will be simple.

On the vendor side, the nature of most outsourcing contracts makes it difficult to do a straight "apples-to-apples" comparison with in-house estimates. Indeed, several recent books have made the complexity of outsourcing contract provisions a central focus (e.g., Klepper and Jones, 1998; Halvey and Melby, 1996; Mylott, 1995). Lack of experience in managing outsourcing contracts and a poor understanding of the costs associated with administering such contracts are also cited as important sources of unexpected economic costs (Clark, Zmud and McCray, 1995). Some examples of the economic issues in vendor contracts are noted in Exhibit 4, under "External Costs". Similarly, internal cost estimates, already likely to be inaccurate for high-risk projects, may not take into account indirect costs. Examples of two such costs that are often missed are the opportunity costs associated with choosing one project over another and the addition of fixed costs associated with hiring new development staff.

Future Benefits of In-House Development

The final factor in the SREF framework is the anticipated benefits of in-house project development. Consideration of these benefits is central to most outsourcing analyses and is at the heart of nearly all critiques of outsourcing. The types of benefits likely to arise from in-house application development tend to fall into two categories: learning benefits and benefits resulting from reduced future vulnerability.

Learning benefits are those long-run contributors to profitability (such as cost reductions and enhanced revenue) that are not captured by the economic analysis limited to the project itself. Some examples include:

- *Developing expertise in key technologies.* Even if development of an application fails, as long as the individuals who attempted to develop the application remain with the organization, the risk of developing applications using the same technologies in the future declines. For the technology-related learning benefits of a project to be accurately captured, it is critical that the project's fit with the long term IT direction of the firm, as articulated through portfolio analysis, be assessed.
- *Developing better knowledge of internal processes.* Many applications force developers and managers to rethink internal processes, thereby identifying opportunities for improvements that are likely to be better recognized when development is done in-house.
- *Developing better knowledge of customer or supplier processes.* Since many of today's systems cross organizational boundaries, developers may gain access to customer or supplier information that helps the organization identify further opportunities for service or profit.

In some cases, the long term learning benefits of a project may prove far more important than the economics of the project itself. Thus, a high-risk project may fail and yet still prove to be a net positive contributor to the organization.

Future vulnerability is one of the most commonly cited concerns related to outsourcing. Most of these vulnerabilities appear to stem from either 1) high switching costs (e.g., Pearlson, 2001), reflecting the difficulty of changing vendors or bringing an application in-house after it has been outsourced, or 2) loss of control (e.g., Klepper and Jones, 1998), wherein certain critical aspects of an application (e.g., functionality, legal protections, future enhancements) are likely to be under the control of the vendor. Specific indicators of high vulnerability may include:

- *Limited availability of suitable vendors*. Where there is only one vendor, or where a single vendor is substantially more attractive than any of its competitors, vulnerability tends to be high. Even if the vendor works well with the organization, and contractual specifications reduces the risk of activities such as price gouging, the organization is still likely to be dependent on the financial health of vendor.

- *Tight coupling with the organization's existing systems*. Where a project is customized to fit the organization's existing systems (or where existing systems are customized to facilitate the development of the project), switching costs are likely to be high. This is sometimes referred to as asset specificity (Cheon, Grover and Teng, 1995).
- *Lack of experience negotiating and managing outsourcing contracts*. As well as unexpected costs, inexperience with outsourcing relationships can lead to long term vulnerability to problems such as unexpectedly poor service (Lacity, Hirschheim and Willcocks, 1994), failure to innovate (Pearlson, 2001), and loss of proprietary information/technology (Klepper and Jones, 1998).
- *Lack of emerging standards*. In situations where a specific project is undertaken in a domain where clear standards are emerging (e.g., XML), long term vulnerability is likely to be reduced

Examples of specific learning and reduced vulnerability future benefits issues are listed in Exhibit 5.

Applying the SREF Framework

The SREF framework presented in this paper represents our attempt to integrate some of the most important issues that have already been identified in the outsourcing literature.

This present paper is not intended to report empirical results and the purpose of including Exhibits 1-5 is to clarify the framework and to suggest a methodology that could be validated in future exploratory field studies. In this section, we present a proposal of how the SREF framework might ultimately be applied.

Data Gathering and Adjustments

We assume that questionnaires, such as those suggested in Exhibits 1-5, will be used to gather data on the project and on potential vendors. Each answer would then be scored (an untested scoring system, proposed based upon our study of the literature, is included with the exhibits). After the raw scores are compiled, we then anticipate a number of adjustments:

- *Sustainability Adjustment:* We propose that the strategic score of the system needs to be adjusted for evidence of sustainability. Our initial hypothesis is that the effect will be multiplicative, which is to say that strong evidence of sustainability will dramatically increase the strategic value of the system (Exhibit 1).
- *Relative Risk Adjustment:* We propose the raw risk scores for each vendor (Exhibit 3) need to be subtracted from the risk score for in-house development (Exhibit 2), leading to a risk difference. Thus, a risk difference score is developed for each vendor.
- *Cost Overrun Adjustment:* Because the ability to accurately cost a project is likely to be dependent on level of risk, we believe it would be prudent to adjust expected cost differences between in-house and outsourced development in the event a high risk differential exists (Exhibit 4). In the long run, we also believe it would be desirable to calibrate economic values in monetary terms as opposed to using a point score, but there is simply not enough data in the existing literature to support even exploratory propositions along these lines.
- *Vulnerability Adjustment:* Although indirect long-term benefits of in-house development may not depend on how strategic a project proves to be, we believe that vulnerabilities associated with outsourced development need to be amplified according to how strategic the project is likely to become. As a preliminary hypothesis, we treat the effect as multiplicative (Exhibit 5).

Upon completion of these adjustments, we would have scores for:

- the strategic level of the system,
- the relative risk of in-house versus outsourced development for each vendor,
- the relative economics of in-house versus outsourced development for each vendor
- the expected long-term benefits likely to accrue from in-house development

Analysis

Determining the strategic level of a project would be the first step in applying the SREF framework. We do not propose that strategic systems should not be outsourced. Instead, our own analysis suggests the major impact of a system being strategic should be to reduce the importance of the economic variable in making the outsourcing decision, leaving relative risk and future benefits as the key variables, as illustrated in Figure 2.

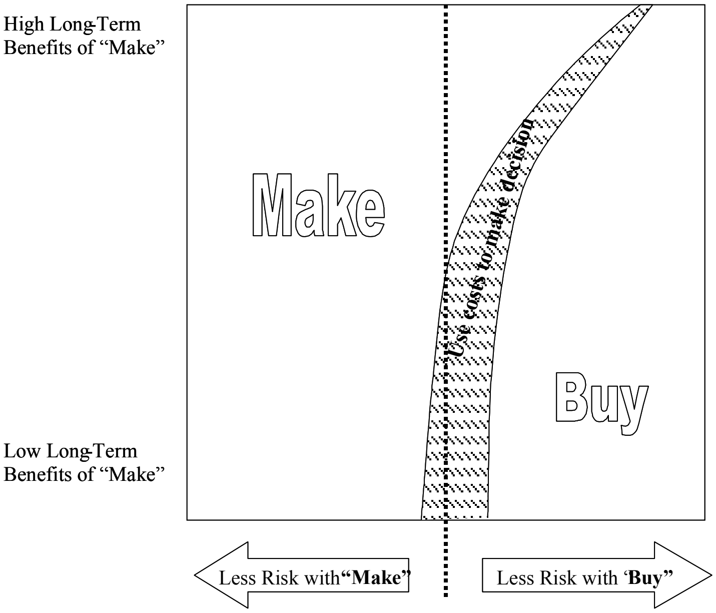
High Long-Term Benefits of "Make"
Low Long-Term Benefits of "Make"
Make
Use costs to make decision
Buy
Less Risk with "Make"
Less Risk with 'Buy"

Figure 2. Make-or-buy decision where project is strategic

The interpretation of Figure 2 is as follows:

- Where long-term benefits of in-house development appear to be low, then reducing development risk becomes the primary factor in the make or buy decision. Stated another way, *if* a system is strategic and *if* there do not appear to be long term problems with outsourcing, and *if* a vendor can build it with less risk, then outsourcing is recommended.
- Where the long term benefits on in-house development are high, then the decision will be a trade off between long-term benefits and relative risk, hence the rightwards curve of the boundary.
- Only in a situation where risk and long term benefits leaves us relatively uncertain regarding the make-or-buy decision does relative economics play an important role in the decision (hence the thin shaded area).

In situations where the proposed project does not appear to be strategic to the organization, then SREF recommendations change dramatically (Figure 3).

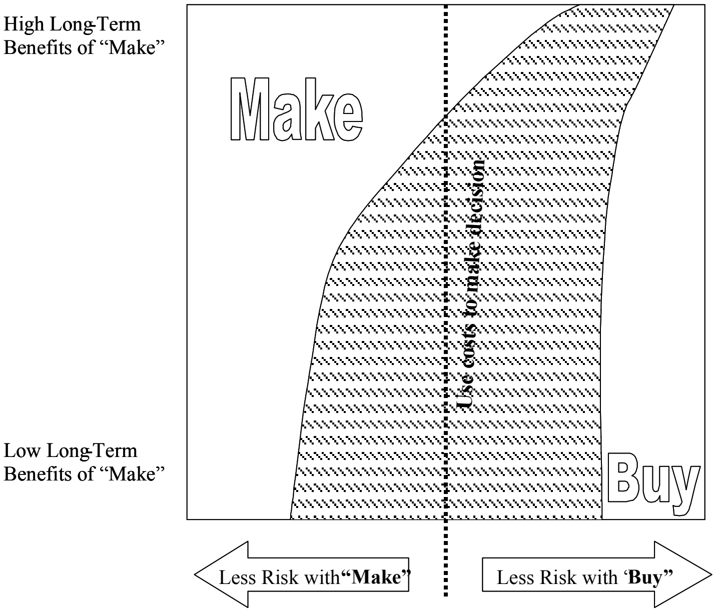
High Long-Term Benefits of "Make"
Low Long-Term Benefits of "Make"
Make
Use costs to make decision
Buy
Less Risk with "Make"
Less Risk with 'Buy"

Figure 3. Make-or-buy decision for non-strategic (support) projects

The interpretation of Figure 3 is as follows:

- Where long-term benefits of in-house development appear to be low, then relative economics becomes the deciding factor in choosing whether to make or buy as shown in the wide shaded area.
- Where the long-term benefits of in-house development are high, then the decision will be a trade off between long-term benefits and economics resulting in the rightwards curve of the boundary.
- Only in a situation where the mismatch between in-house and vendor risk is very high will risk be the basis for the decision. Since the system is not strategic, we assume its benefits to be mainly economic (e.g., reduced administrative costs), and we can therefore afford some likelihood of a "worst case" scenario where project risks conspire to prevent completion.

Final Iteration

Upon completion of SREF analysis, a final check of the project's fit with the organization's portfolio may be required, since "project risk" (a dependent variable in portfolio analysis) may be impacted by the make-or-buy decision. For example, based upon large long-term benefits SREF could recommend in-house development of an application that would be much less risky if outsourced. In such a situation, the application might become misaligned with low-risk portfolio goals, and the inclusion of the application within the portfolio might have to be reconsidered.

Directions for Future Research

There are currently two important directions that we are pursuing with respect to the SREF framework:

- *Empirical Validation:* We are investigating possible case sites where outsourced development is commonly utilized, to be used for exploratory testing of the framework and refinement/validation of the instruments.
- *Extension of the framework to operational outsourcing:* Currently, SREF is focused on the issue of outsourced development. Through our study of the literature and discussions with managers, we intend to examine how the framework might be applied to the broader area of selective or total outsourcing of the IT function.

Conclusions

In this paper, we have reviewed the major issues facing managers as they make decisions concerning the outsourcing of information systems application development. We have proposed and developed a framework for analyzing these outsourcing decisions. The goal was to provide a mechanism designed to help managers in weighing the benefits of IT outsourcing against the potential negative consequences. This was done by building on IT portfolio theory and combining it with findings relating to IT outsourcing decisions. Our analysis led to the development of the SREF Framework, named after its key dependent variables—Strategy, Risk, Economics, and Future benefits. We have shown conceptually how the SREF model might ultimately be applied and have provided illustrative data gathering instruments showing how data could be gathered to generate scores for various measures dealing with (1) the strategic level of the system; (2) the relative risk of in house versus outsourced development; (3) the relative economics of in-house versus outsourced development; and, (4) the expected long-term benefits likely to accrue from in-house development. Consistent with our stated intent of limiting the focus of the present paper to theory development, no attempt was made to report on research applying the SREF framework to any empirical studies. We did, however, describe how this SREF methodology that we have proposed could be applied in future research and exploratory field studies.

References

Cash, Jr. J.I., McFarlan, F.W., and McKenney, J.L. 1988. *Corporate Information Systems Management: The Issues Facing Senior Executives. 2nd Edition.* Homewood: Irwin.

Cheon, M.J., Grover, V. and Teng, J.T.C. 1995. "Theoretical perspectives on the outsourcing of information systems". *Journal of Information Technology*. 209-219.

Clark, T.D., Zmud, R.W., and McCray, G.E. 1995. "The outsourcing of information services: transforming the nature of business in the information industry". *Journal of Information Technology.* 221-237.

Fried, L. 1995. *Managing Information Technology in Turbulent Times.* New York: Wiley-QED.

Halvey, J.K. and Melby, B.M. 1996. Information Technology Outsourcing Transactions: Process, Strategies and Contracts. New York: Wiley

Klepper, R. and Jones, W.O. 1998. *Outsourcing Information Technology, Systems and Services.* Upper Saddle River: Prentice Hall PTR.

Lacity, M.C., Hirschheim, R. and Willcocks, L.P. 1994. "Realizing Outsourcing Expectations: Incredible Expectations, Credible Outcomes". *Information Systems Management*. Fall. 7-18.

Lacity, M.C., Willcocks, L.P. and Feeny, D.F. 1996. "The Value of Selective Outsourcing". Sloan Management Review. Spring. 13-25.

Leonard-Barton, D. and Sviokla, J.J. 1988. "Putting Expert Systems to Work". *Harvard Business Review.* March-April 1988. Included in McGowan, W.G. (ed.) 1990. *Revolution in Real Time: Managing Information Technology in the 1990s.* Boston: HBS Press. 251-264.

Malone, T.W., Yeats, J. and Benjamin, R.I. 1989. "The Logic of Electronic Markets". *Harvard Business Review.* May-June 1989. Included in McGowan, W.G. (ed.) 1990. *Revolution in Real Time: Managing Information Technology in the 1990s.* Boston: HBS Press. 49-56.

McFarlan, F.W. 1981. "Portfolio Approach to Information Systems". *Harvard Business Review.* September-October. 142-150.

McFarlan F. W. and Nolan, R.L., 1995. "How to Manage an IT Outsourcing Alliance", Sloan Management Review. Winter 1995

Mylott, T.R. 1995. *Computer Outsourcing: Managing the Transfer of Information Systems*. Englewood Cliffs: Prentice Hall.

Pearlson, K.E. 2001. *Managing and Using Information Systems: A Strategic Approach.* New York: Wiley.

Sobol, M.G. and Apte, U. 1995. "Domestic and Global Outsourcing Practices of America's Most Effective IS Users" *Journal of Information Technology.* 10. 299-321.

Sprague, Jr. R.H. and McNurlin, B.C. 1993. *Information Systems Management in Practice, 3rd Edition.* Englewood Cliffs: Prentice Hall.

Venkatesan, N. 1992. "Strategic Sourcing: To Make or Not To Make". Harvard Business Review. November-December. 98-107.

Exhibits

Exhibit 1. Strategic Level Checklist

	Question	Yes	No	Points for Yes
Strategic Level	Will the proposed application lead to a significant increase in company revenue?			+5
	Once installed, would extended failure of th e application lead to major loss of revenue or inability to meet product or service delivery commitments?			+5
	Will the proposed application lead to significantly tighter integration with the company's customers that make it easier for them to acquire yo ur products/services?			+3
	Will the proposed application significantly increase the company's ability to respond quickly to high -value customer requests (e.g., reduce turnaround time)			+3
	Will the proposed application lead to significant increases in the quality of significant products or services supplied by the company?			+3
	Will failure to deploy the application lead to a significant loss in market share to competitors?			+3
	Will the proposed application lead to significant reduction in product or service unit costs?			+2
	Will the proposed application increase the company's capacity to deliver goods and services?			+2
	Strategic Level (add point scores for all 'Yes" answers):			
Sustainability	Is this system unique in the industry and, if so, wi ll it be difficult for competitors to replicate once it has been deployed?			+1
	If competitors do not have similar systems, will being the first company to deploy the application offer long -term advantages in market share?			+0.5
	If the application lea ds to tighter linkages to customers, will it be difficult or impossible for competitors to establish similar linkages with the same customers?			+0.5
	Sustainability Level (add point scores for all 'Yes" answers):			
	Strategic Score = (Strategic Level * (1 + Sustainability Level))/5:			
Rough Calibration: <= +1.0 - System is of limited strategic value +1.2 to +2.8 - System is strategic > = +3.0 - System is highly strategic				

Exhibit 2. Sample Internal Development Risk Checklist

	Question	Yes	No	Points for Yes
Size Risk	Will the proposed application consume more than 20% of your IS budget over the next year?			+5
	Is the proposed application significantly larger than the largest project you have ever constructed?			+10
	Is the proposed application comparable in size to the largest application the organization has ever constructed?			+2
	Is the proposed application significantly smaller than applications previously constructed by the organization?			-2
Experience Risk	Will the proposed application require the deployment of a new systems architecture to become operational?			+10
	Will the proposed application use an application architecture not previously used by the company?			+5
	Does the proposed application require the company to use tools not previously employed for development?			+2
	Is the application structurally similar to other applications successfully built by the organization?			-5
Lack of Structure Risk	Is the application intended to replace an application that already exists, closely duplicating the old functionality?			-5
	Are there other applications (e.g., competitors products, packaged software) with similar functionality that are available for inspection?			-2
	Are the intended users vague about their exact requirements for the system?			+5
	Are details regarding the application's exact architecture still to be determined?			+5
	Risk subtotal (add point scores for all 'Yes" answers):			
Rough Calibration: < 0 - Project has negligible risk, with completion likelihood near 100%, and budgets, timeframes likely to be accurate 0-10 - Project has modest risk, but is still likely to be within the capabilities of the organization 11-20 - Project is high risk. If it is to be undert aken, the organization must accept that a significant risk of failure exists 21+ - Project is extremely high risk. Likelihood of failure probably exceeds 50%, with cost/time overruns virtually certain				

Exhibit 3. Sample Vendor Risk Checklist

	Question	Yes	No	Points for Yes
Lack of Structure Risk	Is the application an existing product of the vendor’s that is available “off the shelf”? *(If so, skip remaining questions)*			-10
	Is the application to be based on modifying an existing product of the vendor’s?			-5
	Are there other applications (e.g., competitors products, packaged software) with similar functionality that are available for inspection?			-2
	Does part of the vendor’s proposal include eliciting specifications and functionality from users?			+5
	Does part of the vendor’s proposal involve determining an appropriate architecture for the project?			+5
Size Risk	Will the proposed application represent more than 20% of the vendor’s business over the next year?			+5
	Is the proposed application significantly larger than the largest project the vendor has ever constructed?			+10
	Is the proposed application comparable in size to the largest application the vendor has ever constructed?			+2
	Is the proposed application significantly smaller than applications previously constructed by the vendor?			-2
Experience Risk	Will the proposed application involve a systems architecture that the vendor has not used before?			+10
	Will the proposed application use an application architecture not previously used by the vendor?			+5
	Does the proposed application require the vendor to use tools not previously employed for development?			+2
	Is the application structurally similar to other applications successfully built by the vendor?			-5
	Risk subtotal (add point scores for all ‘Yes” answers):			

Rough Calibration:

< 0 - Project has negligible risk, with completion likelihood near 100%, and budgets, timeframes likely to be accurate

0-10 - Project has modest risk , but is still likely to be within the capabilities of the vendor

11-20 - Project is high risk. If it is to be undertaken, the organization must accept that significant risk of non -delivery or cost overruns exists

21+ - Project is extremely high risk. Likelihood of failure probably exceeds 50%, with cost/time overruns virtually certain

Exhibit 4. Expected Costs Checklist

	Question	Yes	No	Points for Yes
Internal Costs	Are internal estimates of the cost of building the application substantially lower than bids from third parties?			+5
	Will developing the application require that other development projects with positive paybacks be shelved?			-2
	Will developing the application require the hiring of additional permanent person nel?			-1
External Costs	Are the potential third party developers of the project willing to commit to a fixed price contract?			-3
	If not fixed price, are third party developers willing to place a ceiling on eventual total cost?			-1
	Are third party d evelopers committing to deliver the application on a "turnkey" basis?			-2
	Does the proposed contract fail to require the return of all progress payments in the event of non -completion?			+3
	Is your organization uncertain about actual contract costs based on lack of experience with similar contracts?			-2
Relative Cost Subtotal (add point scores for all 'Yes" answers):				
Risk adjustment:				
Are internal project risks high, or extremely high, while vendor project risks are substantially lower?				-5
Are internal and vendor project risks high, or extremely high, while the total of external cost items is negative?				-2
Risk Adjusted Cost Subtotal (add point scores for all 'Yes" answers):				
Rough Calibration: < -2 - Acquiring application from vendor is likely to have a cost advantage -2 to +2 - Project is likely to cost about the same, whether done in -house or by a vendor > +2 - Cost advantages to constructing the application in -house are likely to exist				

Exhibit 5. Future Benefits of In-house Development Checklist

	Question	Yes	No	Points for Yes
Learning Value	Are the skills that would be acquired in building the proposed application internally likely to be important to the organization in the future?			+5
	Will the process of building the system internally lead the company to a better understanding of its customers' needs and economics?			+3
	Would a vendor building the system employ a significant number of proprietary technologies that will remain "black box" to the company?			+2
	Is it likely that in the process of building the application internally we will discover opportunities for increasing the productivity or quality of the company's operations?			+2
Learning Value Score (add point scores for all 'Yes" answe rs):				
Vulnerability	Will it be hard to "take the application back" in the event that things do not work out with the vendor?			+5
	Will installing the application require many significant modifications or customizations to other applications used by the company			+5
	Is a single vendor substantially better than other competitive vendors?			+3
	Does you organization have significant experience negotiating and managing similar outsourcing arrangements?			-2
	Is it likely that, over the long term, similar applications will proliferate around the industry and that a standard will ultimately emerge?			-2
Vulnerability Subtotal (add point scores for all 'Yes" answers):				
Vulnerability Score = Strategic Score * Vulnerability Subtotal:				
Long Term Benefits Sc ore = Vulnerability Score + Learning Value Score:				
Rough Calibration: < +5 - Building system internally is of limited long term value +6 to +15 - Long term benefits to building the application internally exist > +15 - The long-term benefits of keeping the system in -house are compelling				