SAPPHIRE '99 SINGAPORE
3rd Annual SAP Asia Pacific
Institute of Higher Learning Forum
Maximizing the Synergy Between Teaching, Research & Business

OPTIMAL TIMING FOR ERP UPGRADE: A PRELIMINARY MODEL

Celeste See Pui Ng, ngc@fit.qut.edu.au, 617 3864 1922
Taizan Chan, taizan@fit.qut.edu.au, 617 3864 2533

Fax No: 617 3864 1969
Information System Management Research Centre
Queensland University of Technology
GPO Box 2434 Brisbane, QLD Australia 4001.

ABSTRACT

Similar to in-house developed software, the post-implementation cost associated with an ERP tends to increase over time. The post-implementation cost of an ERP, however, consists of not only software maintenance cost, but also users' dissatisfaction cost due to unrealized requirements and expectations. This cost has been found to increase over time as the modified codes in the ERP deteriorated and backlogs of users' requirements increased. As a result, there might be an optimal point in time when it is more economical for an organization to invest in a new ERP upgrade that clears the backlogs of requests and offers fresh codes. This paper develops an economic model of ERP life-cycle cost to determine the optimal time to choose an upgrade and the optimal time to complete the implementation of that upgrade. It is found that, among other things, it is more economical for an organization that deals with a volatile business environment to delay its upgrade to later versions than to upgrade early.

INTRODUCTION

The significant benefits potentially offered by Enterprise Resource Planning (ERP) systems, such as integration of all core functions [11] [28], improvement of overall business practices [19] [8] [3], and lower cumulative maintenance cost [16] [2] [28], have lead many organizations to adopt ERP over the past 10 years. The adoption and implementation of ERP product, such as SAP's R/3, which is Y2K compliant, has increased even more dramatically over the past 3 years as Y2K is approaching [5] [11] [19]. As a result, more than 70% of Fortune 1000 have now implemented ERP [26] [19] [1]. It is also expected that ERP adoption will continue over the next 5 years at a compound annual rate of 37% [12].

Although there is relatively very little research on ERP (and large packaged application in general), some effort has been directed towards addressing the issues of ERP selection and implementation (see for example, [9] [15] [27]). While this stream of research will continue to be relevant as more and more organizations turn to ERP for their IS solutions, there is also an urgent need to address the issues faced by those organizations *after* their ERP have been successfully implemented and become operational. This is especially so given the large base of organizations which have already adopted ERP. However, research addressing the post-implementation issues appears to be non-existent.

While ERP offers complete functionality for supporting all the core functions of an organization, it has been found that many expected benefits from ERP are not realized [6] [22] [10] [20]. Furthermore, as environment changes, users' requirements for its ERP will also change. As a result, there is still a need to continuously adapt and enhance an ERP after it is implementation to meet users' requirements and expectation [22]. Similar to in-house developed software, it has been found that maintenance of ERP is also an understaffed function which often result in backlog of users' requests. These backlogs represent the gap between users current expectations and requirements and the actual functionality and benefits delivered by the ERP, which is often measured as users' dissatisfaction [6] [15]. Furthermore, as ERP also suffers deterioration to its code as a result of frequent changes, its maintenance cost also increases over time [5], [28].

A common but expensive approach to solving the problems of users' backlog and deteriorating code with an in-house software is the rewriting of the software (usually from scratch). For packaged application such as ERP, a more attractive option, that of *system upgrade*, is available to organizations that have adopted the ERP. As ERP vendors attempt to remain competitive, they will constantly introduce new versions of their products that not only perform more efficiently but also include new features and functionality needed by their existing customers and potentially new customers [30]. Furthermore, as part of the vendors' effort to remain attractive to their customers, the codes introduced by these vendors to provide the new functionality are usually of high quality to ensure that they are easy to understand and modify. Consequently, an organization can turn to upgrading its ERP as a means to reduce users' dissatisfaction and maintenance cost [28].

Deciding whether to upgrade an ERP to the latest version or when an ERP should be upgraded, however, is not a trivial decision. Unlike upgrading of small, stand-alone package, such as a word-processing software, purchasing and implementing ERP upgrade is a costly and lengthy endeavor [28]. It is therefore not optimal for an organization to upgrade its ERP each time a new version is introduced. In fact, because of the cost and time involved, many organizations may decide not to upgrade at all or to upgrade only after a long period of time. Even if an organization decides that it may be worthwhile to upgrade its ERP sometime in the future, it still has to deal with the decision of when is the optimal time to upgrade. On one hand, if it upgrades too early, it will miss some of the state-of-the-art designs or new functionality introduced in newer version [9]. On the other hand, if the organization upgrades its ERP too late, the organization has to incur higher users' dissatisfaction cost and allocate a larger budget provision for maintenance of the existing system [29]. Given the huge cost associated with ERP life-cycle, it is therefore imperative to provide these organizations with insights concerning the optimal time to implement and upgrade its ERP to minimize its operation cost. This paper develops an analytical model to determine the optimal timings to upgrade and to complete the implementation of the upgrade.

Although the problem of determining the optimal timing to upgrade an ERP is similar to those of hardware and software maintenance/replacement, the existing models in these streams of literature are not directly applicable. In hardware replacement literature [4] [23] [24] [25], it is assumed that a piece of equipment can be replaced immediately. However, ERP upgrade involves an implementation period between the time a decision to upgrade is made and the time the implementation is completed. As the cost of upgrade implementation is directly related to the implementation period, this implementation period has to be considered explicitly in an ERP upgrade model. On the other hand, the existing models on software replacement [7] [13] [14] do not consider the cost of users' dissatisfaction. However, users' dissatisfaction, which measures the gap between users' requirements and expectations of the benefits derivable from ERP and the actual functionality of the ERP, is an essential component of ERP cost model as many organizations that adopt ERP do also hope to gain the benefits they expected from the ERP [21] [5] [8].

This paper develops a preliminary model of ERP life-cycle cost for determining the optimal timings to upgrade and to complete the implementation of the upgrade. It considers not only the maintenance cost involved in enhancing and maintaining the ERP, but also considers the users' dissatisfaction cost and the implementation period explicitly. Furthermore, the model accounts for the cost of the entire ERP life cycle by also considering the initial investment cost involved in the purchase and implementation of the ERP.

The rest of the paper is organized as follows. The next section describes the framework of the model by describing the parameters and variables involved in the model and the objective function to be minimized. The third section describes the results derived from the model and the sensitivity analysis of the decision variables and draws implications of our results for ERP upgrade. The last section concludes the paper by discussing the limitations of the model and the future extensions to the model.

MODEL FRAMEWORK

We consider the total cost of purchasing, implementing, operating, and maintaining an ERP over a planning horizon T[1]. We assume that at the start of the planning horizon, 0, the organization has already made an investment of K_0 to purchase and implement an ERP to make it operational. Due to changing requirements, the IT department responsible for the maintenance of the ERP would receive a stream of requests from the users over time. These requests represent users' expectations of the functionality and hence the benefits provided by the ERP. As in most organizations, not all of these requests will be satisfied by the IT department. Requests, or parts of requests, that are satisfied required effort to implement and hence will incur maintenance cost. Those requests that are not satisfied represented a gap between users' expectations of the functionality and the actual functionality provided by the ERP, and has been found to result in users' dissatisfaction [10] [22]. These users' dissatisfactions result in poor users' morale, lower productivity, and high turnover, which could all be translated directly into cost for the organization.

Both the maintenance cost and users' dissatisfaction cost per unit time are expected to rise over time. The maintenance cost will increase over time because the codes will deteriorate as more and more changes are made to the ERP, thereby making it harder to understand and change. The users' dissatisfaction cost will increase over time because users' requests that are not satisfied usually remain unsatisfied, adding to a backlog of users' requests, thereby increasing the gap between users' expectations and actual functionality over time. As a result, the total life-cycle cost, which is the sum of the initial investment, the users' dissatisfaction cost, and the maintenance cost tends to increase dramatically over time. However, by upgrading to a new version that incorporates all the backlogs of users' expectations with fresh code, the users' dissatisfaction and maintenance cost can be contented. It may therefore be more economical for the organization to upgrade to a new version of the ERP at some point in time over the planning horizon T.

At a time denoted by Ti, the organization may decide to upgrade to a new version available at that time, and start its implementation. However, unlike hardware replacement, which could occur instantaneously, the new upgrade requires an extended period of time of implementation. The length of the implementation schedule will depend on the version being upgraded as well as the number of people involved in the implementation period. It is assumed that the amount of effort required to implement a later version of ERP will be higher as the gap between the existing version of ERP and the new version will be higher. It is also assumed that by increasing the number of people assigned to the implementation of the upgrade, the implementation period will be shorter. However, increasing the size of the implementation team implies higher implementation cost. Thus, in addition to deciding when to start implementing the new version, i.e. Ti, the organization also needs to decide when the implementation should be completed, i.e. Tu (and hence how much resources to be allocated to the implementation). Note that as the new version will not be available until Tu, the existing version still needs to be maintained during the period Tu – Ti. Thus, the total life cycle cost consists of four basic components: (i) the initial investment, (ii) the total cumulative life-cycle cost (i.e. sum of users' dissatisfaction and maintenance cost) of the existing ERP from time 0 to time Tu, (iii) the investment cost associated with the purchase and implementation of the new version of ERP, and (iv) the total cumulative life-cycle cost of the new ERP from time Tu to T.

The problem to be addressed is that of determining the optimal Tu and Ti such that the total ERP life-cycle cost over T is minimized. The following subsections describe the components of the ERP life-cycle cost model and formulate the objective function to be minimized.

Initial Investment Cost

This cost component is the initial investment cost of the existing ERP. Unlike the case of in-house software, which usually costs only up to millions of dollars to develop, the initial investment cost of ERP could be as high as several hundred million dollars [5] [8] [11]. This significant amount of money has signaled its importance to be considered explicitly in this model. A large portion of this investment cost is due to the ERP implementation cost. Other costs

---

[1] This planning horizon is the time frame over which the IT manager wants to plan for the upgrade of the ERP. The limitation of this assumption is addressed in the Discussion section of the paper.

involved under this cost component include ERP purchase and employees training cost. In this study, we assume that this initial investment cost of the existing ERP is a constant figure denoted by:

$$K_0 \tag{1}$$

For ease of reference, the subscript 0 is used to refer to the cost components of existing ERP while subscript 1 for the new version.

Existing ERP Life-Cycle Cost

The existing ERP is assumed to be in operation from time 0 until the new version of ERP is completely implemented and ready to be put into operation at time, Tu. During this period of time, Tu – 0, the existing ERP will undergo maintenance whenever the maintenance request arrived. Similar to [7], we assume that users' maintenance request arrives at an average rate of λ requests per unit time. The value of λ is determined by the volatility of the business environment of an organization. A highly competitive organization and external environment will lead to a higher value of λ, as there is a higher need to synchronize the ERP with the changes in the environment [17]. However, the speed in incorporating the changes into the existing ERP is usually at a lower rate than the rate at which users' requests arrive [18]. Consequently, there is always a discrepancy between what the user requests and what is fulfilled. In this model, we use the function points metrics as a measure of the functionality that is requested by a user (as well as a measure of what is fulfilled).

Let η_0 be the average number of function points incorporated into the existing ERP in each maintenance request and assuming the maintenance cost per person-hours is a constant cost of σ. Therefore, the total maintenance cost of the existing ERP system from time 0 until the new version is fully implemented, Tu is:

$$\int_0^{Tu} \lambda\eta_0\,\sigma.\,t\,dt \tag{2}$$

On the other hand, assume θ_0 is the average number of function points failed to be incorporated into the existing ERP system in each request and the average users' dissatisfaction cost associated with each function point is ρ. Due to the backlog of users' dissatisfaction, as the unmet users' requirements remained dissatisfied, the users' dissatisfaction cost is always increasing from time to time. Thus the cumulative users' dissatisfaction cost of the existing ERP from time 0 until the new ERP implementation is completed is:

$$\int_0^{Tu} [\int_0^{t} (\theta_0\lambda\rho)\,dt]\,dt \tag{3}$$

The total cumulative life-cycle cost of the existing ERP from time 0 to time Tu includes both the cumulative users' dissatisfaction and maintenance cost. Therefore, it could be represented by:

$$\int_0^{Tu} [\int_0^{t} (\theta_0\lambda\rho)\,dt]\,dt \;+\; \int_0^{Tu} \lambda\eta_0\,\sigma.\,t\,dt \tag{4}$$

Investment Cost Of The New Version Of ERP

Similar to the initial investment cost for the existing ERP, the investment cost of the new version consists of both the (i) implementation cost and (ii) other costs such as the cost of purchasing the new version and training. The second cost component is assumed to increase with time as it is expected that later versions which incorporate more functionality will be more expensive and since it is more complex, will also involve a higher training cost. Assuming that this cost component starts at a base value of E_1 and increases over time at the rate of ϕ, it can be denoted as:

$$E_1 + \phi E_1 Ti \tag{5}$$

As the new ERP will also require implementation, this implementation cost must also be modeled explicitly. It is also assumed that by increasing the number of people assigned to the implementation of the upgrade, the implementation period will be shorter. However, increasing the size of the implementation team implies higher implementation cost. It is also assumed that by increasing the number of people assigned to the implementation of the upgrade, the implementation period will be shorter.

In this case, we adopt the formulation of linear rewriting speed used in [7], to model the relationship the between the size of the team, N and the implementation speed of the team, S. That is, we assume that the ERP implementation speed is a linear function of N given by:

$$S(N) = c + mN$$

Both c and m are constants, representing the minimum number of professional labors required in order to start the ERP project, and the productivity rate respectively. As mentioned earlier, the degree of customization is likely to increase with the version that an organization is upgrading to. We capture the rate of increase in the amount of customization necessary by β. Thus at time Ti, the amount of customization required is expressed as βTi. The implementation team of size N must be able to complete this implementation within the implementation period, i.e.

$$\beta\ Ti = S(N)(\ Tu - Ti\ ) \tag{6}$$

Thus, the total ERP implementation cost for the duration of the ERP implementation is:

$$N(\ Tu - Ti\ )\ \pi = \beta\ Ti\ /m\ \ - c(\ Tu - Ti\ )\ \pi\ /m \tag{7}$$

Total Cumulative Life-Cycle Cost Of The New Version Of ERP

Similar to the cumulative life-cycle cost of the existing ERP, the cumulative cost of the new version of ERP includes both the users' dissatisfaction cost and maintenance cost but it is incurred only from the time the new ERP is implemented, Tu until the end of the planning horizon, T.
Therefore, the total cumulative users' dissatisfaction and maintenance cost of the new version of ERP is:

$$\int_{Tu}^{T} [\int_{Tu}^{t} (\theta_1\lambda\rho)\ dt]\ dt + \int_{Tu}^{T} \lambda\eta_1\,\sigma\, t\ dt \tag{8}$$

where θ_1 is the average number of function points that is failed to be incorporated, and η_1 is the average number of function point incorporated into the new version of ERP in each maintenance request.

The model parameters, functions, and variables are summarized in Table 1, 2, and 3 respectively.

Category	Parameter	Definition	Unit
Initial investment	K_0	Initial investment cost of the existing ERP system	Dollar
	E_1	Initial investment cost of the new ERP system	Dollar
	Φ	Cost factor, the rate at which the price of the newer versions increases	1/month
Users' Environment	λ	Users' request arrival rate	Request/month
Users' Dissatisfaction	θ_0	Average number of function points failed to be incorporated into the existing ERP system in each maintenance request	Function point/request
	θ_1	Average number of function points failed to be incorporated into the new ERP system in each maintenance request	Function point/request
	ρ	Average users' dissatisfaction cost	Dollar/function point
Maintenance Environment	η_0	Average number of function points incorporated into the existing ERP system in each maintenance request	Function point/request
	η_1	Average number of function points incorporated into the new ERP system in each maintenance request	Function point/request
	σ	Average ERP maintenance cost	Dollar/person-hours
Implementation efficiency	β	Degree of customizations	Function point/month
	π	Average ERP implementation cost	Dollar/person-hours
	c	Minimum number of ERP implementation team size	Function point/month
	m	ERP implementation team's productivity rate	Function point/person-hours

Table 1: Model Parameters

Function	Definition	Unit
S(N)	Speed of ERP implementation team	Function point/ Month
P(Ti, Tu)	Total cumulative cost of the upgrade software system	Dollar

Table 2: Model Functions

Variables	Definition	Unit
Ti	The time when the implementation of the new version of EPR starts	Month
Tu	The time when the new version of ERP completed	Month
N	The number of people in the ERP implementation team	Person-hours/month

Table 3: Model Variables

Total Cumulative Life-Cycle Cost Over The Planning Horizon

The total life-cycle cost involved in owning and maintaining an ERP is the sum of the initial investment cost of the existing ERP given in equation (1), the total cumulative life-cycle cost of the existing ERP from time 0 to time Tu given in equation (4), the investment cost associated with the purchase and implementation of the new version of ERP given in equations (5) and (7), and the total cumulative life-cycle cost of the new ERP given in equation (8). Therefore, assuming that P(Ti, Tu) represents the total life-cycle cost,

$$\begin{aligned} P(Ti, Tu) = K_0 &+ \\ &\int_0^{Tu} [\int_0^{t} (\theta_0 \lambda \rho)\, dt]\, dt + \int_0^{Tu} \lambda \eta_0\, \sigma.t\, dt + \\ &E_1 + \phi E_1 Ti + N(Tu - Ti)\, \pi + \\ &\int_{Tu}^{T} [\int_{Tu}^{t} (\theta_1 \lambda \rho)\, dt] dt + \int_{Tu}^{T} \lambda \eta_1 \sigma.t\, dt \end{aligned} \quad (9)$$

which can be simplified as

$$\begin{aligned} P(Ti, Tu) = K_0 + E_1 &+ 1/2\, \theta_1 \lambda \rho T^2 + 1/2\, \lambda \eta_1 \sigma T^2 + \\ &(-c\, \pi\, Tu\, /m) + 1/2[\theta_1 \lambda \rho + \lambda \eta_1 \sigma]\, Tu^{2} + \\ &(\phi\, E_1 - 2\theta_1 \lambda \rho T - \lambda \eta_1 \sigma T + c\pi\, /m + \beta/m)\, Ti + (3/2\, \theta_1 \lambda \rho + ½\, \lambda \eta_1 \sigma)\, Ti^{2} \end{aligned}$$

Given this cost model, the goal is to determine the optimal Ti and Tu, i.e., the optimal policy that will minimize the total ERP life-cycle cost over the planning horizon, T.

The optimization problem in this study is thus:

$$\begin{gathered} [G]\ \ \min P(Ti, Tu) \\ \text{subject to } Tu - Ti \geq 0 \\ \text{and } \ Ti \geq 0 \end{gathered}$$

RESULTS

Solving [G] gives us the following proposition:

Proposition: P(Ti, Tu) is strictly convex in Ti, and Tu. (See appendix for proof.) If $\{[(c\pi\, /\, m\lambda\, (\theta_0 \rho + \eta_0 \sigma) + (\phi\, E_1\, m + c\pi + \beta)\, /\, m\lambda\, (3\theta_1 \rho + \eta_1 \sigma)] > (2\theta_1 \rho T + \eta_1 \sigma T)/(3\theta_1 \rho + \eta_1 \sigma)\}$, an optimal point (Ti*, Tu*) exists and is characterized by the following:

$$Ti^{*} = (2\theta_1 \lambda \rho Tm + \lambda \eta_1 \sigma Tm - \phi\, E_1\, m - c\pi - \beta)\, /\, m\lambda\, (3\theta_1 \rho + \eta_1 \sigma) \quad (10)$$

$$Tu^{*} = c\pi\, /\, m\lambda\, (\theta_0 \rho + \eta_0 \sigma) \quad (11)$$

Other wise,

$$\begin{aligned} Ti^{*} &= Tu^{*} \\ &= [(2\theta_1 \rho + \eta_1 \sigma)\, \lambda T - \phi\, E_1 - \beta/m)]\, /\, \lambda\, [(\theta_0 + 3\theta_1)\, \rho + (\eta_0 + \eta_1)\, \sigma] \end{aligned}$$

The results obtained by solving the above objective function have indicated that optimal points exist for both Ti* and Tu*. By further analyzing this proposition in the following subsections, we are able to draw several implications that could serve as guidelines for organizations to make informed decision concerning the timing of ERP upgrade.

Sensitivity Analysis

By using equation (10), and (11), we analyze the sensitivity, of each of the parameters used in the cost model, with respect to the two optimal decision variables, Ti^* and Tu^*. (See the Implication 1-14 in the Appendix for proof.) Table 4 summarizes the result of the sensitivity analysis. 0 indicates that the parameter q has no impact on the decision variable. – indicates that the decision variable decreases with the parameter q while + indicates that the decision variable increases with the parameter.

q	Initial investment			Users' env.	Users' Dissatisfaction			Maintenance environment			Implementation efficiency			
	K_0	E_1	ϕ	λ	θ_0	θ_1	ρ	η_0	η_1	σ	β	π	c	m
dTi^*/dq	0	-	-	+	0	R1	R2	0	+	+	-	-	-	+
dTu^*/dq	0	0	0	-	-	0	-	-	0	-	0	+	+	-

Table 4: Sensitivity Analysis

Conditions:
R1: Ti^* increases with θ_1 if $3(\phi \boldsymbol{E_1} m + c\pi + \beta) > \lambda\eta_1\sigma Tm$
R2: Ti^* increases with ρ if $3(\phi \boldsymbol{E_1} m + c\pi + \beta) > \lambda\eta_1\sigma Tm$

Management Implications For ERP Upgrade

We draw implications of our results for ERP upgrade decisions. These implications are drawn with respect to the following factors affecting the ERP life-cycle cost:

Initial investment. The optimal policy and the sensitivity analysis suggest that the decision to upgrade should not be dependent upon the initial cost already invested in the ERP (i.e. K_0). This is an important insight. An organization that has already invested a huge sum of money on the initial purchase and implementation of an ERP is likely to feel that their investment has gone to waste if it has to make additional investment to upgrade to a new version before the expected return on the initial investment has been realized. As a result, it may either prolong the upgrade decision or decide not to upgrade at all. Our result suggests, however, that such investment represent a sunk cost and that if an upgrade could realize the expected benefits better than the existing version, then an additional investment should be made to upgrade. Indeed, our model suggests that the upgrade should be completed even earlier if the users' dissatisfaction cost associated with unrealized requests of the existing version of ERP (i.e. θ_0 and ρ) is higher. Also, part of the investment cost of the new upgrade could be alleviated by adopting a new version earlier, as later version will involve more customizations and hence a higher implementation cost.

Users' environment. Our model suggests that an organization that operates in a volatile business environment, characterized by higher λ, should wait for later versions to be available before it decides to upgrade. Furthermore, despite the fact that it should start to upgrade later, it also needs to complete the implementation of the upgrade earlier. It is therefore important for an organization dealing with a volatile business environment to compressed their implementation schedule by allocating more resources towards the implementation of the upgrade.

Users' dissatisfaction. The results from the model indicates that while optimal timing to start implementing a new upgrade (i.e. Ti) is not affected by the existing ERP level of users' dissatisfaction cost, the implementation has to be completed earlier, thus suggesting that the implementation schedule has to be compressed. This suggests that in an organization where users' dissatisfaction is high, the main issue is not which version should be adopted. Rather the issue is that if it is economical to upgrade to a new version, then once a version is decided, it should seek to spend additional money to speed up the implementation so that users could make use of the new version earlier.

Maintenance environment. Our results also suggest that an existing version, which cost more to maintain, should be upgraded earlier. On the other hand, if the newer version to be upgraded costs higher to maintain, then an organization should delay its decision to upgrade. Thus, a higher maintenance cost suggests that need to reduce the implementation schedule to reduce the potentially wasteful but expensive effort required to maintain the older version of the ERP while the newer version is being implemented.

Implementation efficiency. Finally, our model reveals that with a more efficient team to implement the upgrade (i.e. higher m), an organization can not only wait to adopt later version of ERP that include more functionality but also complete the implementation within a shorter schedule. This result highlights the importance of having experienced people involved in the implementation of the upgrade, and is consistent with organizations' observations that ERP implementation period can be shortened with the help of experienced consultants. Our model, however, further reveals that higher labor cost involved in the implementation, (i.e. higher π), has a negative impact on the implementation period and implementation cost, which unfortunately is a lesson that many organizations learned only after investing millions of dollars in ERP implementation. This suggests that an organization can better prepare for implementing an upgrade later by better management of knowledge gained in the initial implementation of its ERP. This knowledge can be applied during the implementation of the upgrade so that it could reap the benefit of higher m without incurring the cost of higher π.

CONCLUSIONS AND FUTURE RESEARCH

The ERP upgrade/replacement model proposed in this paper represents only a preliminary model, which is based on several simplistic assumptions. Nevertheless, we are able to derive several results that provide some insights into the optimal decisions for ERP upgrade. To enhance the value of the model, and to be able to derive more insights into ERP upgrade, several extensions must be made to the model:

Firstly, since both the time of introduction and the extend of new upgrades by vendors are neither deterministic nor constant, the availability of new versions and the level of new functionality encompassed in each new version should be modeled explicitly as a stochastic Markov process.

Secondly, our assumption that a new version adopted will always clear the backlog of users' requests need to be relaxed. Often a vendor introduces a new version that encompasses the new functionality required by an *average* organization and not all the new functionality required by all organizations. This can be accommodated in the model by assuming that only a certain fraction of the users' dissatisfaction cost is resolved when an upgrade is made.

Thirdly, the assumption of linear relationship between the team size (L) and the implementation speed of the team (S) needs to be relaxed. In reality, it has been found that the total output increases only in a concave manner with respect to the team size as adding more people to a team also increases the amount of communication required among them, thereby lowering their productivity.

Fourthly, the model assumes that an organization will only upgrade once over a fixed planning horizon. This assumption can be relaxed to include multiple upgrade decisions over infinite horizon.

Fifthly, the formulation of the users' dissatisfaction and maintenance cost is simplistic. The formulation has to be better informed and tested with real data from organizations that have implemented ERP.

Finally, our model should be instantiated with real data to derive specific implications for organizations wishing to upgrade their ERP. The last two extensions (multiple upgrade decisions, and real data of users' dissatisfaction and maintenance cost) would require detailed case studies of organizations that have already implemented ERP and are considering to upgrade their ERP.

APPENDIX

Proof for Proposition 1:
The total cumulative life-cycle cost of the new ERP system from time Tu until the end of the planning time horizon T is given by:

$$P(Ti, Tu) = K_0 + E_1 + 1/2\, \theta_1\lambda\rho T^2 + 1/2\, \lambda\eta_1\sigma T^2 + (- c\, \pi\, Tu/m) + 1/2[\theta_1\lambda\rho + \lambda\eta_1\sigma]\, Tu^2 + (\phi\, E_1 - 2\theta_1\lambda\rho T - \lambda\eta_1\sigma T + c\pi/m + \beta/m)\, Ti + (3/2\, \theta_1\lambda\rho + ½\, \lambda\eta_1\sigma)\, Ti^2$$

$$f_{TiTi}\, P(Ti, Tu) = 3\, \theta_1\lambda\rho + \lambda\eta_1\sigma > 0,$$

and

$$f_{TuTu}\, P(Ti, Tu) = \theta_0\lambda\rho + \lambda\eta_0\sigma > 0.$$

Also,

$$f_{TiTu}\, P(Ti, Tu) = 0$$

Therefore, P(Ti, Tu) is convex in Ti, and Tu.

The K-T conditions for optimality are given by:

$$Ti^* [(3\, \theta_1\rho\, \lambda + \eta_1\sigma\, \lambda)Ti^* + \phi\, E_1 - 2\theta_1\lambda\rho T - \lambda\eta_1\sigma T + c\pi/m + \beta/m + \vartheta] = 0$$

$$Tu^* [(\theta_0\rho\lambda + \eta_0\sigma\lambda)Tu^* - c\pi/m - \vartheta] = 0$$

$$\vartheta(Tu^* - Ti^*) = 0$$

Whereby, ϑ is the Lagrangean constant. If $\{[(c\pi / m\lambda\, (\theta_0\rho + \eta_0\sigma) + (\phi\, E_1\, m + c\pi + \beta) / m\lambda\, (3\theta_1\rho + \eta_1\sigma)] > (2\theta_1\rho T + \eta_1\sigma T)/(3\theta_1\rho + \eta_1\sigma)$, $\vartheta = 0$, and $Tu^* > Ti^*)\}$ then

$$Ti^* = (2\theta_1\lambda\rho Tm + \lambda\eta_1\sigma Tm - \phi\, E_1\, m - c\pi - \beta) / m\lambda\, (3\, \theta_1\rho + \eta_1\sigma)$$

and

$$Tu^* = c\pi / m\lambda\, (\theta_0\rho + \eta_0\sigma)$$

If $\vartheta > 0$, then $Ti^* = Tu^*$. . Therefore,

$$Ti^* = Tu^* = [(2\theta_1\rho + \eta_1\sigma)\, \lambda T - \phi\, E_1 - \beta/m)] / \lambda\, [(\theta_0 + 3\theta_1)\, \rho + (\eta_0 + \eta_1)\, \sigma] .$$

Sensitivity Analysis
Implications obtained from the table sensitivity analysis for variables Ti, and Tu with respect to parameter q as shown in the following table are as followed:

q	Initial investment			Users' env.	Users' Dissatisfaction			Maintenance environment			Implementation efficiency			
	K_0	E_1	ϕ	λ	θ_0	θ_1	ρ	η_0	η_1	σ	β	π	c	m
dTi^*/dq	0	-	-	+	0	R2	R1	0	+	+	-	-	-	+
dTu^*/dq	0	0	0	-	-	0	-	-	0	-	0	+	+	-

R1: ρ increases with Ti^* if $3(\phi \mathbf{E_1} m + c\pi + \beta) > \lambda\eta_1\sigma Tm$
R2: θ_1 increases with Ti^* if $3(\phi E_1 m + c\pi + \beta) > \lambda\eta_1\sigma Tm$

Implication 1:
Ti^* and Tu^* are independent from K_0.
Proof: It is obvious from expressions Ti^* and Tu^* .

Implication 2:
Ti^* decreases with E_1 and Tu^* is independent from E_1.
Proof: It is obvious from expressions Ti^* and Tu^*.

Implication 3:
Ti^* decreases with ϕ and Tu^* is independent from ϕ.
Proof: It is obvious from expressions Ti^* and Tu^*.

Implication 4:
Ti^* increases with ρ and Tu^* decreases with ρ.
Proof: It is obvious from expression Tu^*. For Ti^* ,
$f_\rho (Ti^*) = \theta_1 / m\lambda (3\theta_1\rho + \eta_1\sigma)^2 [\, 3(\phi E_1 m + c\pi + \beta) - \lambda\eta_1\sigma Tm \,]$
if $3(\phi E_1 m + c\pi + \beta) > \lambda\eta_1\sigma Tm$ then $f_\rho (Ti^*) > 0$. Therefore, Ti^* increases with ρ.

Implication 5:
Ti^* increases with σ and Tu^* decreases with σ.
Proof: It is obvious from expression Tu^*. For Ti^* ,
$f_\sigma (Ti^*) = \eta_1 / m\lambda (3\theta_1\rho + \eta_1\sigma)^2 [\theta_1\rho \lambda Tm + \phi E1\, m + c\pi + \beta]$.
Since $f_\sigma (Ti^*) > 0$ is always positive, therefore Ti^* increases with σ.

Implication 6:
Ti^* decreases with ϕ and Tu^* increases with ϕ.
Proof: It is obvious from expressions Ti^* and Tu^*.

Implication 7:
Ti^* is independent from θ_0 and Tu^* decreases with θ_0 .
Proof: It is obvious from expressions Ti^* and Tu^*.

Implication 8:
Ti^* increases with θ_1 and Tu^* is independent from θ_1.
Proof: It is obvious from expression Tu^*. For Ti^* ,
$f_{\theta 1} (Ti^*) = \rho / m\lambda (3\theta_1\rho + \eta_1\sigma)^2 [\, 3(\phi E_1 m + c\pi + \beta) - \lambda\eta_1\sigma Tm \,]$
if $3(\phi E_1 m + c\pi + \beta) > \lambda\eta_1\sigma Tm$ then $f_{\theta 1} (Ti^*) > 0$. Therefore, Ti^* increases with θ_1.

Implication 9:
Ti^* is independent from η_0 and Tu^* decreases with η_0.
Proof: It is obvious from expressions Ti^* and Tu^*.

Implication 10:
Ti^* increases with η_1 and Tu^* is independent from η_1.
Proof: It is obvious from expression Tu^*. For Ti^* ,
$f_{\eta 1} (Ti^*) = \sigma / m\lambda (3\theta_1\rho + \eta_1\sigma)^2 [\theta_1\rho \lambda Tm + \phi E_1 m + c\pi + \beta]$.
Since $f_{\eta 1} (Ti^*) > 0$ is always positive, therefore Ti^* increases with η_1.

Implication 11:
Ti^* increases with λ and Tu^* decreases with λ.
Proof: It is obvious from expression Tu^*. For Ti^*,
$f_\lambda (Ti^*) = 1 / m\lambda^2 \ (3\theta_1\rho + \eta_1\sigma) [\phi E_1 m + c\pi + \beta]$.
Since $f_\lambda (Ti^*) > 0$ is always positive, therefore Ti^* increases with λ.

Implication 12:
Ti^* decreases with c and Tu^* increases with c.
Proof: It is obvious from expressions Ti^* and Tu^*.

Implication 13:
Ti^* increases with m and Tu^* decreases with m.
Proof: It is obvious from expression Tu^*. For Ti^*,
$f_m (Ti^*) = 1 / m^2\lambda \ (3\theta_1\rho + \eta_1\sigma) [c\pi + \beta]$.
Since $f_m (Ti^*) > 0$ is always positive, therefore Ti^* increases with m.

Implication 14:
Ti^* decreases with β and Tu^* is independent from β.
Proof: It is obvious from expressions Ti^* and Tu^*.

REFERENCES

[1] Advanced Manufacturing Research, A. (1998), *The key players*, *Computerworld*, 68.
[2] Banker, R.D., G.B. Davis, and S.A. Slaughter (1998), *Software Development Practices, Software Complexity, and Software Maintenance Performance: A Field Study,* Management Science, **44**(4), 433-450.
[3] Barnes, M. (1999), *Customization of ERP apps requires development skills,* InformationWeek, (722), 9A-9D.
[4] Bethuyne, G. (1998), *Optimal replacement under variable intensity of utilization and technological progress,* Engineering Economist, **43**(2), 85-105.
[5] Bingi, P., M.K. Sharma, and J.K. Godla (1999), *Critical Issues Affecting an ERP Implementation,* Information Systems Management, **16**(5), 7-14.
[6] Cameron, B. (1998), *Soundoff! Is ERP in trouble?*, *Computerworld,* 62.
[7] Chan, T., S.L. Chung, and T.H. Ho (1996), *An economic model to estimate software rewriting and replacement times,* IEEE Transactions on Software Engineering, **22**, 580-598.
[8] Davenport, T.H. (1998), *Putting the Enterprise into the Enterprise System,* Harvard Business Review, **76**(4), 121-131.
[9] Davis, G.B. (1988), *Information Systems To Buy, Build, or Customize?,* Accounting Horizon, **March**, 101-103.
[10] Dryden, P. (1998), *ERP failures exact high price*, *Computerworld*, 1.
[11] Gable, G.G. (1998), *Large Package Software: a Neglected Technology?,* Journal of Global Information management, **6**(3), 3-4.
[12] Garner, R. (1998), *Still in the money*, *Computerworld*, 71.
[13] Gode, D.K., A. Barua, and T. Mukhopadhyay (1990), *On the economics of the software replacement problem*, in *The Eleventh International Conference on Information Systems* .
[14] Gupta, Y.P. and T.S. Raghunathan (1988), *A Preliminary Model for Information System Replacement,* Management Science, **16**(4), 289-296.
[15] Hecht, B. (1997), *Choose the right ERP software,* Datamation, (March), 56-58.
[16] Hicks, D.A. and K.E. Stecke (1995), *The ERP Maze: Enterprise Resource Planning and Other Production and Inventory Control Software., IIE Solutions*, **27**(8), 12-16.
[17] Kemerer, C.F. and S.A. Slaughter (1997), *Determinants of Software maintenance Profiles: an Empirical Investigation., Journal of Software Maintenance: Research and Practice*, **9**, 235-251.
[18] Martin, J. and C. McClure (1983), *Software Maintenance: The Problem and its Solutions* , Prentice Hall, Englewood Cliffs, New Jersey. 472.
[19] Minahan, T. (1998), *Enterprise resource planning: strategies not included,* Purchasing, **125**(1), 112-127.

[20] Morphy, E. (1999), *When ERP meets ROI,* Export Today, **15**(5), 42-47.
[21] Mullin, R. (1997), *ERP Software: The next Generation., Chemical Week*, **159**(20), 40-41.
[22] Radding, A. (1998),*Vendor directions, user wish lists*, in *Computerworld*. . p. 66.
[23] Rajagopalan, S. (1992), *Deterministic capacity expansion under deterioration,* Managament Science, **38**(4), 525-539.
[24] Rajagopalan, S., M.R. Singh, and T.E. Morton (1998), *Capacity Expansion and Replacement in Growing Markets with Uncertain Technological Breakthroughs., Management Science*, **44**(1), 12-30.
[25] Scarf, P.A. and O. Bouamra (1999), *A capital equipment replacement model for a fleet with variable size,* Journal of Quality in Maintenance, **5**(1), 40-49.
[26] Shepherd, J. (1998),*ERP: The Next Stage*, *Computerworld,* 55, 66.
[27] Simon, L. (1997), *Making the most of enterprise resource planning,* Chemical Market Reporter, **251**(19), 30-31.
[28] Stein, T. (1999), M*aking ERP add up*, *Information Week,* 59-63.
[29] Swanson, E.B. and C.M. Beath (1989), *Maintaining Information Systems in Organizations* , John Wiley & Sons Ltd., England. 255.
[30] Weston, R. and C. Stedman (1998), *Forget ROI - just install it*, *Computerworld*, 1, 14.