Vendors’ incentives to invest in software quality in enterprise systems

Wooje Cho a,1, Ramanath Subramanyam b,1, Mu Xia c,*,1

a Ulsan National Institute of Science and Technology, Republic of Korea
b University of Illinois at Urbana-Champaign, United States
c Key Lab of Financial Intelligence and Financial Engineering, Southwestern University of Finance and Economics, Chengdu, Sichuan, China

ARTICLE INFO

Article history:
Received 24 April 2012
Received in revised form 8 November 2012
Accepted 24 April 2013
Available online 4 May 2013

Keywords:
Enterprise systems
Software quality
Game theory
Synergy

ABSTRACT

In the enterprise system market, software quality is often unobservable at the contracting stage between the vendor and the customer. Two factors complicate the vendor’s decision to invest in software quality. First, as a required part of the transaction, services such as installation and maintenance are bundled with the product. Second, the vendor’s cost of delivering these services is directly affected by quality of the software. Incorporating these factors, we develop an analytical framework to examine when vendors of enterprise systems have an incentive to invest in software quality under different market structures and market participant behaviors. We also investigate economic consequences of such quality decisions by enterprise software vendors, highlighting certain unique characteristics of these markets. We consider a duopoly setting, with simultaneous and sequential moves of the vendors. Our results show that in the duopoly market, even when customers are uninformed about quality, an investment-equilibrium exists. We find that there exist conditions under which customers might have reasons to trust that vendors would invest in high software quality.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The market for enterprise systems has unique characteristics that distinguish it from both traditional product and software markets. First, the adoption decision requires significant long-term financial and organizational commitments. Second, an enterprise system purchase consists of both a product – the system – and the associated services such as installation, customization, maintenance, and upgrade. The service component is unique compared to a non-enterprise prepackaged software product. In addition, the quality of the software not only is of concern to the adopter, but also affects the vendor’s cost of delivering the services. The vendor’s product-quality decision, therefore, has to be made based on the understanding of all the above factors and their interactions and can have significant economic impacts on both the customer and the vendor.

Adding to the complexity of the problem is the considerably high switching costs for customers and their inability to perfectly observe the real quality of the system at purchase time. This gives vendors additional incentives to exaggerate the quality of their software products, hoping if and when the customer does discover the low quality of the software it is too expensive and too late to switch to a competitor. Such information asymmetry between vendors and customers at the onset of the agreement raises the question: How does the information asymmetry of product quality affect the vendors’ incentives to invest in software quality? Also, should customers believe the vendor-speak about their high product quality?

In this paper, we develop an analytical model to answer these research questions. During the development process, a software vendor decides on the level of investment in the quality of packaged software. On the one hand, creating high quality software requires non-trivial investments and process quality decisions. On the other hand, these decisions might not directly influence the customer’s decision as many software quality attributes are unobservable at the time of signing the contract. Thus, the vendor may have incentives not to invest in those unobservable attributes. In addition, due to the fast-changing technological environment and the presence of market uncertainties, spending extensively on the quality of solutions over alternatives such as additional marketing effort might not seem desirable to the vendors.

Nevertheless, quality investments, even on unobservable product attributes, are very likely to benefit the vendor during the customization and maintenance processes (e.g., [5]). For instance, investing in flexible and modular design choices would enable the vendor to customize the software more effectively in response to a customer’s request, thus saving time and effort. Similarly, investing in a high level of reliability and security would reduce maintenance costs and prevent potential security breaches. In addition, in the long run, a high-level investment in quality is also likely to benefit the vendor through good brand reputation and high customer satisfaction. Thus, the quality of enterprise software is likely to affect the customer’s utility and the software vendor’s profitability in providing customization and maintenance services.

The aim of the current paper is to examine strategic behaviors of the two enterprise software vendors in the duopoly market with
asymmetric information about software quality ex ante. The present study could provide software vendors with guidelines on making decisions about product quality during the development process\(^2\). This research could also help organizations in the process of making strategic decisions about enterprise solutions and contracting with vendors. We intend to provide researchers with a better understanding of economic consequences of quality-related decision in the domain of enterprise systems, a rapidly growing area.

From the perspective of the customer, purchasing and adopting an enterprise system are critical decisions that may highly impact its organizational performance. However, selecting an enterprise system product is a complicated decision because information about the quality of enterprise software is privy only to the software vendor. Customer would have to make the purchase decision before they are able to systematically determine whether a vendor’s enterprise systems’ quality would be sufficient to meet the firm’s expectations. In addition, once an organization chooses a software vendor with a specific enterprise system product, they might have to live with the choice for several years.

We find that there exists an investment-equilibrium in the duopoly market, even if customers are uninformed about the quality at the time of purchasing decision. This implies that the vendors have incentives to invest in the unobservable quality dimensions of enterprise software in the presence of competition.

2. Background and model development

Much industry and anecdotal evidence suggests that asymmetric information on the quality of enterprise software can cause tension between vendors and customers. As a case in point, a leading firm in the waste management service industry, Waste Management, sued SAP over ERP implementation seeking recovery of more than US$100 million\(^{[10]}\). Waste Management’s argument was that:

SAP promised that the software could be fully implemented throughout all of Waste Management within 18 months. ... From the beginning, SAP assured Waste Management that its software was an ‘out-of-the-box’ solution that would meet Waste Management’s needs without any customization or enhancements, ... Unfortunately, Waste Management ultimately learned that these representations were not true. ... Product demonstrations by SAP prior to their deal were fake software environments, even though these demonstrations were represented to be the actual software.... mock-up version of that software intended to deceive Waste Management...

Before the agreement was formalized between the two companies, SAP had promised to Waste Management that the solution was mature. According to Waste Management’s argument, the SAP’s talk turned out to be a significant exaggeration.

Details of this case suggest that the failure could be explained by lower levels of quality of the SAP’s software to Waste Management along some unobservable dimensions such as flexibility, functionality and scalability. For instance, one could maintain that limited design flexibility might have affected the ability of SAP to customize the enterprise solution within the stipulated time. However, from a selling standpoint, vendors might have an inclination to exaggerate the software quality. Because it sometimes takes several months to a few years for a customer to observe the quality of enterprise software, it is not pragmatic to expect the customer to terminate the agreement before it is too late. Instead, in many cases, the customer’s investment is locked in with the vendor during implementation, and the bargaining power of the customer is very little or non-existent once the contract is signed (e.g.,\(^{[17]}\)).

In the academic literature, product quality has been studied as a strategic choice of companies in industrial organization (IO). For monopolists, the cost of improving quality is incurred as a sunk design cost that does not change with total output\(^{[16]}\). A monopolist’s choice, then, will depend on whether quality affects demand. In the duopoly and oligopoly markets, a company’s quality choice will be determined by competition as well as the cost and demand structure in the market\(^{[2]}\). In the traditional setting, with information asymmetry between companies and customers, it is found that a monopolist does not have any incentive to invest in product quality\(^{[6]}\).

In this paper, we adopt the lens of product quality models in the IO approach, while taking into account of the differences between the enterprise system market and the traditional product markets studied in existing IO models. We discuss properties of enterprise systems markets in the following sections.

2.1. Characteristics of the enterprise systems market

As noted earlier, there are several unique characteristics that distinguish the enterprise system market from personal software markets as well as traditional product markets. First, sales of enterprise systems involve a combination of products and services associated with it. Customers will need to purchase both in order to generate positive value. The product sale is often in the form of licenses to install and use the packaged software and services include customization, enhancements, maintenance (repair and updates) and training. Vendors earn their revenue by selling license fees and service fees, but the combination of the two types of sales is often inseparable. Though many kinds of manufacturers of typical product markets sell warranty services with their products, such as PCs and televisions, the purchase of the service component is a consumer’s option in most cases. Consumers may not buy the warranty service, and even when they purchase the warranty, the price of the service sales is often lower than that of the product. However, in the case of enterprise systems, the price of services including customization and maintenance could be higher than or at least on the same scale of the price of user licenses. Further, customers generate value from the product only in the presence of the service component.

Second, it often takes considerable time before customers can evaluate the software quality. The quality of personal and individual productivity software such as Microsoft Office\(^{\text{TM}}\) programs can be revealed right after customers purchase the software because those programs do not need to be customized before user experiences the product fully. However, enterprise systems require significant time to be implemented and installed before they are used by organizations. Depending on the size of the adoption and the degree of customization, it could take up to a few years. What is worse from the perspective of the customer is that their bargaining power over the vendor decreases over time due to switching costs for the firm and possible holdup by the vendor.

Third, given that there are two kinds of operational costs in enterprise systems business: (1) the cost of developing packaged software and (2) the cost of delivering services including customization and maintenance, enterprise systems are characterized by the relationship between these two kinds of costs. Specifically, the first operational cost affects the cost of the second component. If the vendor invests more in the development of packaged software, then the quality of the enterprise software would be higher and the higher quality can reduce the cost of service delivery. For example, as flexibility, scalability and capability of the software increases, it will cost less to customize the enterprise system and to meet customer’s needs.

2.2. The quality of enterprise systems and unobservable attributes of quality

Customers need to confirm whether the quality of an enterprise system is at least as great as the level of need on the observable

\(^2\) In this paper, the term software quality refers to the quality of the packaged software, which is generally developed before the sale of the enterprise system.
dimensions before the decision is made, but some quality attributes are hard to measure and evaluate by the customer prior to using the system. We characterize the quality of enterprise software as unobservable with a consideration of observable and unobservable quality attributes. Prior findings indicate that quality attributes of software could be captured along dimensions such as functionality or capability, reliability, flexibility, usability, modifiability, performance, maintainability and security (e.g., [3,4,12,13,15]). Though customers might purchase a software application after preliminary demonstrations and tests, the quality attributes are rarely fully observable until the customers use the software. Therefore, there exists uncertainty in the selection of enterprise systems, because the purchaser is able to at most evaluate only a portion of all the quality attributes.

2.3. Vendors’ investment for software quality

According to the Akerlof’s [1] market for lemons, buyers of a used car whose quality is unobservable will not be willing to pay a high price because their best guess is that the car is of below average quality. Thus, owners of a good car do not place their car in the used car market because they are not able to get a high enough price. Likewise, in a traditional product market where product quality is unobservable, the firm would not have incentives to invest in the quality. Our study extends this body of research on product quality to the enterprise software market.

We posit that such unobservable qualities are determined by the amount which a vendor invests for the quality in the software development process, including budget, personnel, cognitive effort and related resources. We make this assumption since more resources are required to enhance the quality attributes of the software. For the software vendor, the software development process is similar to the production process of traditional product companies. Just as greater investment in product lines and raw materials can lead to a better-quality product in the traditional market, so is greater investment in software development likely to lead to higher-quality of the enterprise software.

One approach for a vendor to produce high quality packaged software is to adopt a quality-centric rigorous development methodology. Software development using such a methodology might result in higher costs for the vendor, and it might take longer to successfully release the developed software. For example, if a vendor adopts mature processes such as those recommended by the Capability Maturity Model (CMM) [15] for the software development process, it might involve relatively longer schedules and require higher investment in training and development processes, but the vendor might be able to produce high quality software. The vendor could recover such costs by creating economies of scale and generating complementary value, both of which have implications for the customer.

2.4. Basic models

We consider a vendor of the enterprise system (henceforth ES) that develops packaged software and provides support services for the packaged solution, such as customization and maintenance. Major factors of a vendor’s profit include sales revenues, the cost of delivering support services, the cost of developing the packaged software, and the cost related to the project duration [5]. Among them, we consider the first three determinants only because our study focuses more on the relationship between software quality and the support cost in ES market settings rather than on the time-constraint characteristic from outsourcing project settings.

Vendors invest in the development of the packaged solution before the time of sale. At the onset of the agreement, the vendor’s sale price includes license fees for the software and service fees for the support component. In sum, the vendor bears the costs of the development of packaged software and incurs operational costs while delivering those services. In our model, as we emphasized earlier, the cost of the service delivery is influenced by the quality of the packaged software, which in turn is determined by the development costs. Thus, the profit function, \( \pi \), can be given by:

\[
\pi = (\text{sales revenue}) - (\text{operational cost of service delivery}) - (\text{cost of packaged software development})
\]

The consumer surplus, \( W \), from purchasing an enterprise system is determined by customer’s valuation of the ES, the payment amount, and disutility arising from a low software quality.

\[
W = (\text{customer’s value of ES}) - (\text{payment}) - (\text{disutility from low quality})
\]

The disutility in Eq. (b) occurs when the software quality is lower than the required level of quality to meet customer’s needs. In case that the quality is unobservable, it would hardly affect demand functions. However, if the quality became observable, the quality would affect the demand functions. Because the quality is determined by the cost of packaged software development, the cost would ultimately influence demand functions. In the following section, we develop a two period model. The does not affect demand functions because of information asymmetry in quality in the first period, but it does affect demand functions in the second period when the quality is disclosed.

3. Duopoly model of the enterprise system market

Consider a duopoly market of ES, where customers are characterized by two independent attributes, \( \varepsilon \) and \( \lambda \), as illustrated in Fig. 1. \( \varepsilon \) represents the organizational needs of a customer. The relative fit between customer’s needs and vendors’ ES packaged products based on observable product features is the horizontal distance between the customer and the vendor locations. Without loss of generality, we
assume Vendor A’s enterprise system is located at point 0 and vendor B’s is at point 1. Customers located at point \( c = 0 \) observes a perfect fit between their organizational needs and vendor A’s enterprise systems based on its observable attributes, and customers located at point \( c = 1 \) observe a perfect fit with vendor B’s solution. We assume \( c \) is common knowledge because a customer can find its \( c \) from observable features of each vendor’s ES and vendors can assess a priori organizational attributes during the vendor selection process and client visits. In the vertical axis, \( \lambda \) refers to the level of software quality that the customer requires to be satisfied with the ES, particularly based on unobservable quality attributes. In practice, \( \lambda \) of a customer organization would be determined by its organizational size, complexity of business processes and the intensity of information systems (IS) usage. Fig. 1 summarizes the basic setup.

3.1. Simultaneous game

There are two periods in the model. The two vendors start with a minimum level of software quality. In period 1, the two vendors decide on the degree of software quality improvement, \( x \). Vendor A chooses \( x_A \) and vendor B chooses \( x_B \). The cost of improving the software quality is \( mx^2 \). However, \( x \) is vendors’ private information and does not influence customers’ valuations in period 1. These assumptions parallel a traditional incomplete contracts setup with the additional complexity introduced by the private information of the vendor (example of inefficient ex-post bargaining outcome is highlighted in Matoushek [14]).

When a customer adopts an ES from either vendor, the generic value of the ES can be considered \( V \). The price of ES consists of the software license fee, \( p \), and the unit price of delivering services for the gap between customers’ need and packaged software, \( s \). Then, to a customer at point \( c \) between 0 and 1, the consumer surplus of adopting vendor A’s ES is \( V - p_A - s_A \), while that of vendor B’s is \( V - p_B - s_B(1-c) \). \( p \) is the price of ES software license and \( s \) refers to the price of the service that the vendor \( i \) (\( i = A \) or \( B \)) delivers to customize its ES. Since a customer knows its own location \( c \), the customer chooses vendor A if and only if its \( c < x_A - x_B \). Thus, in Fig. 1, customers located in AA and AB choose vendor A, whereas customers in BB and BA choose vendor B in period 1.

In period 2, the two vendors sell the next version of the ES. We assume that the two vendors do not change the degree of improvement \( x_A \) and \( x_B \) when they develop the next version. We made the assumption because once a vendor establishes its development processes, the organizational practices often do not change. The software quality of a vendor’s ES is disclosed in the second period to customers that adopted the ES, paralleling the incomplete contracts setup [7,8]. Then, the disclosed software quality, which was determined by investment in software development, affects consumer demands. Customers who are not satisfied with the quality in the first period will not renew the contract with the vendor in the second period. Customers located in AB either switch to vendor B or develop in-house systems because they were not satisfied with the quality of vendor A’s ES in period 1\(^4\). Similarly, customers located in BA either move to vendor A or choose in-house development because their quality requirement is greater than \( x_B \). Customers that are not satisfied with the quality of a vendor’s ES in the first period will select the other vendor with the probability \( \theta (0 \leq \theta \leq 1) \) and develop in-house systems with the probability \( 1 - \theta \) in the second period.

Illustrative Scenario: Consider a retail company that adopts a new Enterprise Resource Planning (ERP) system, and the market is dominated by two giant vendors. Let us suppose that client visits and system demonstrations suggest that vendor A’s ERP is more functional and suitable to the needs of the retailer in comparison to vendor B’s of-fering, i.e., \( \epsilon < \epsilon_A \). Then, the retailer would choose vendor A in period 1. However, in the first period, the retailer will not be satisfied with A’s ES if the retailer’s \( \lambda \) is higher than \( x_A \). Firm-specific factors such as organizational complexity, excessive reliance on the ERP, extremely high transaction volumes, and insufficient vendor’s effort can all contribute to the dissatisfaction. In such a case, the customer would either switch to vendor B’s offering or choose to develop its own in-house systems in the next period.

3.1.1. Model settings and assumptions

The setting and assumptions of the model are summarized as follows.

- The attributes of vendors’ ES consist of observable and unobservable attributes.
- The two decision attributes of customers, \( \epsilon \) and \( \lambda \), are uniformly distributed from 0 to 1.
- The vendors do not change their prices over two periods: \( p_A \) and \( s_A \).
- Customers know their \( \epsilon \) and \( \lambda \), but not vendors’ improvement until the end of the first period.
- Two products, A and B, are located at the ends of the unit interval.
- All customers purchase the next version of the ES in the following period.
- The marginal cost of producing packaged software is zero.
- Cost of quality varies in a quadratic manner with quality improvement.\(^5\)

---

\(^4\) The assumption that all customers at the end of the first period purchase the next version or switch to the second vendor is made for the following reasons. First, if the customer abandons the current solution and that of the vendor, the central organizational problem that was intended to be addressed originally by the ES still remains unaddressed (e.g., non-scalable legacy systems). Second, the idea of a next version of the product also applies to the context where a firm chooses a phased ES implementation where less risky modules such as Human Resources and Financial modules are first procured and deployed. Adjustments to the deployment processes based on the phased implementation are incorporated when the organization switches to a full-scale implementation in the following period. Further, we did consider an alternate model setup where the second period decision of the client is based on the anticipated quality level of the second vendor. However, the complexity of the solution and the need for significant additional assumptions that might reduce the usefulness of our model discouraged us from taking this path.

\(^5\) Since our focus of the paper is on the effect of software quality on the cost of delivering services, we assume that other factors that might affect the service delivery cost, such as compatibility issues from system variation, are given, and do not change with the quality of the software. Therefore, these factors do not affect our model results.
3.1.2. Vendor profits and incentives to invest in quality improvements

The profit functions of the two vendors are given below, where \( f \) and \( g \) are the probability density functions for the distribution of \( \epsilon \) and \( \lambda \), respectively and \( \delta \) is the cost of service delivery of vendor \( i \).

\[
\pi_A = \int_0^\delta pf(\epsilon)d\epsilon + \int_0^\delta \left[ (s_A-c_A)\delta(\epsilon) - m x_A^2 \right] d\epsilon
+ \delta \left[ \int_0^\delta pf(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left( s_A-c_A \right) f(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left[ (s_A-c_A)\delta(\epsilon)g(\lambda)d\lambda - m x_A^2 \right] d\epsilon \right] \\
\pi_B = \int_0^\delta pf(\epsilon)d\epsilon + \int_0^\delta \left[ (s_B-c_B)(1-\epsilon)f(\epsilon) - m x_B^2 \right] d\epsilon
+ \delta \left[ \int_0^\delta pf(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left( s_B-c_B \right) f(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left[ (s_B-c_B)(1-\epsilon)f(\epsilon)g(\lambda)d\lambda - m x_B^2 \right] d\epsilon \right] 
\]

\( i = A, B \). Notations for the duopoly model

\[
\pi_A = \int_0^\delta pf(\epsilon)d\epsilon + \int_0^\delta (s_A-c_A)\delta(\epsilon)d\epsilon - m x_A^2 \\
+ \delta \left[ \int_0^\delta pf(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left( s_A-c_A \right) f(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left[ (s_A-c_A)\delta(\epsilon)g(\lambda)d\lambda - m x_A^2 \right] d\epsilon \right] \\
\pi_B = \int_0^\delta pf(\epsilon)d\epsilon + \int_0^\delta \left[ (s_B-c_B)(1-\epsilon)f(\epsilon) - m x_B^2 \right] d\epsilon
+ \delta \left[ \int_0^\delta pf(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left( s_B-c_B \right) f(\epsilon)g(\lambda)d\lambda + \int_0^\delta \left[ (s_B-c_B)(1-\epsilon)f(\epsilon)g(\lambda)d\lambda - m x_B^2 \right] d\epsilon \right] 
\]

Descriptive terms are provided in Table 1. Since we assume uniform distribution for \( \epsilon \) if \( \epsilon \in [0,1] \), otherwise \( f(\epsilon) = 1 \) if \( \epsilon = 1 \) if \( \epsilon \in [0,1] \), otherwise, the profit function is given by:

\[
\pi_A = p c_0 - \frac{\epsilon}{2} (s_A-c_A) \left[ 1 - \frac{\epsilon}{2} \right] - m x_A^2 \\
+ \delta \left[ \int_0^\delta x_A p c_0 + \frac{\epsilon}{2} x_A (s_A-c_A) \left[ 1 - \frac{\epsilon}{2} \right] d\epsilon \\
+ \theta (1-x_B) p c_0 + \frac{\epsilon}{2} x_A (s_A-c_A) \left[ 1 - \frac{\epsilon}{2} \right] d\epsilon - m x_A^2 \right] \\
\pi_B = p (1-\epsilon_0) + \left( s_B-c_B \right) \left[ \frac{1}{2} - \epsilon_0 + \frac{\epsilon_0}{2} \right] - m x_B^2 \\
+ \delta \left[ x_B (1-\epsilon_0) + \frac{\epsilon}{2} x_B (s_B-c_B) \left[ 1 - \frac{\epsilon}{2} \right] d\epsilon \\
+ \theta (1-x_A) p c_0 + \frac{\epsilon}{2} x_B (s_B-c_B) \left[ 1 - \frac{\epsilon}{2} \right] d\epsilon - m x_B^2 \right] 
\]

Notations pertaining to these equations are summarized in Table 2.

For ease of computation, we assume that the two vendors have the same price for the service fee, \( s_A = s_B = s \), and zero license fees for the packaged software, \( p_A = p_B = 0 \), which lead to an \( \epsilon_0 = 1/2 \), and that the customers do not consider the option of in-house development in the second period, \( \theta = 1 \). Since the marginal reproduction cost of software products is negligible, neither vendor would want to be less attractive to customers by pricing higher than the competitor. We posit \( c_i \) is a linear function of \( x_i \), \( c_i = a - b_i x_i \). Here \( a \) refers to the cost of delivering service with lowest level of software quality, and \( b \) refers to the impact of software quality on reducing service delivery costs.

Then, the profits of the vendors will be:

\[
\pi_A = (s-a+bx_A) \left\{ \frac{1}{8} + \delta \left[ \frac{1}{8} \left( 1 - x_B \right) \right] \right\} - m x_A^2 (1+\delta) \\
\pi_B = (s-a+bx_B) \left\{ \frac{1}{8} + \delta \left[ \frac{1}{8} \left( 1 - x_A \right) \right] \right\} - m x_B^2 (1+\delta) 
\]

With the above assumptions, the first order conditions of Eqs. (5) and (6) yield:

\[
x_A^* = \frac{\delta(s-a+3b-3bx_B^*) + b}{16m(1+\delta)-2ab} \\
x_B^* = \frac{\delta(s-a+3b-3bx_A^*) + b}{16m(1+\delta)+2ab} 
\]

From Eqs. (7) and (8), we can get:

\[
x_A^* = x_B^* = \frac{\delta(s-a+3b) + b}{16m(1+\delta)+2ab} 
\]

Fig. 2a and b show the reaction functions for \( x_A \) and \( x_B \) with numerical examples. There is an investment equilibrium where both vendors improve their software quality equally as shown in both figures. Fig. 2a depicts the reaction functions when the effect of software quality in reducing costs is significant and Fig. 2b does when the effect is insignificant.

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Notations for the duopoly model — simultaneous moves.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V )</td>
<td>General value of the packaged ES (before additional investment in software quality)</td>
</tr>
<tr>
<td>( x_i )</td>
<td>Cost of (customization) service delivery; ( x_i ) is a function of ( x )</td>
</tr>
<tr>
<td>( c_i )</td>
<td>Software license fee (price of the product) of vendor ( i )</td>
</tr>
<tr>
<td>( s_i )</td>
<td>Service delivery fee (price of the service); ( s_i ) is a function of ( x )</td>
</tr>
<tr>
<td>( \epsilon )</td>
<td>the measure of relative fit between organizational needs of a customer and vendor A’s and B’s ES products</td>
</tr>
<tr>
<td>( \lambda )</td>
<td>the level of software quality that the customer requires to be satisfied with the ES</td>
</tr>
<tr>
<td>( \delta )</td>
<td>The discount factor for the next period</td>
</tr>
<tr>
<td>( m )</td>
<td>Coefficient of the cost for quality improvement</td>
</tr>
<tr>
<td>( f(\epsilon) )</td>
<td>the density function of ( \epsilon )</td>
</tr>
<tr>
<td>( g(\lambda) )</td>
<td>the density function of ( \lambda )</td>
</tr>
<tr>
<td>( k )</td>
<td>The parameter that refers to the customer’s intrinsic price of low quality when a vendor’s software quality is lower than the required threshold level of the quality</td>
</tr>
<tr>
<td>( \theta )</td>
<td>The probability that customers who are not satisfied with the quality of a vendor’s ES in the first period will select the other vendor’s ES instead of developing in-house systems</td>
</tr>
</tbody>
</table>

Fig. 2. a. Vendor A’s and B’s reaction function when quality decreases service costs \((a = 3, b = 2, s = 4, m = 1, \delta = 0.9)\). b. Vendor A’s and B’s reaction function when quality minimally influences service costs \((a = 3, b = 0.5, s = 4, m = 1, \delta = 0.9)\).
We observe that the level of the quality improvement at equilibrium is greater when the impact of software quality on reducing service delivery costs is higher.

3.1.3. Consumer surplus

Consumer surplus is determined by the value of ES, software package price, service price, and disutility from low software quality that occurs when \( x \) is lower than \( \lambda \). In Fig. 3, the surplus of the customers in AA over the two periods will be \( V(1 + \delta) - p(1 + \delta) - s(1 - e) \). Because they stick with vendor A over the two periods and are likely to be satisfied with the quality. Similarly, surplus for the customers in BB will be \( V(1 + \delta) - p(1 + \delta) - s(1 - e)(1 + \delta) \). However, customers in AB or BA will have disutility from low quality at least in period 1. The surplus of the customers in AB over the two periods will be \( V(1 + \delta) - p(1 + \delta) - s(1 - e) - k(\lambda - x_m) - k\delta(\lambda - x_e) \).

Among the customers in BA, if its \( \lambda \) is greater than \( x_m \), its consumer surplus will be \( V(1 + \delta) - p(1 + \delta) - s(1 - e) - s\delta - s\delta(1 - e) - k(\lambda - x_m) - k\delta(\lambda - x_e) \) if we assume \( x_m \geq x_e \).

Customers in BA in Fig. 3 may not have the disutility from the low quality if they are informed about the quality of the vendors’ ES. This implies that asymmetric information about software quality decreases consumer surplus of the customers.

3.2. Sequential move game

It is possible that both vendors do not “move” at the same time, and the leading vendor’s move can be observed by the late mover. Let us assume that vendor A is the leading firm in the ES market and vendor B is the follower. We assume that both vendors are aware of the existence of the other competitor, and vendor B will be informed about the quality of the leader’s ES. The timing of the game is as follows: (1) vendor A chooses a quality \( x_m \); (2) vendor B observes \( x_m \) and then chooses a quality \( x_e \). The payoff to vendor \( i \) is given by Eqs. (5) or (6). The notations follow from Table 2 with revisions in the timing of the game.

To solve for the outcome, we first recompute vendor B’s reaction to \( x_m \). If the software quality of vendor A’s ES is \( x_m \), vendor B’s response will be

\[ x_e = \frac{\delta(s - a - 3b + 3bx_m) + b}{16m(1 + \delta) - 256b} \]  

(10)

This value can be calculated from the first order condition of Eq. (6). Because the leader can solve the follower’s problem, vendor A should anticipate that vendor B chooses quality level \( x_e \) in Eq. (10). By inserting \( x_e = x_m^* \) into Eq. (5), we can get the leader’s profit function. The first order condition of the vendor A’s profit function yields

\[ x_m^* = \frac{4b^2\delta^2(a - s) + 5b^2\delta(1 + \delta) - 16mb\left(\delta^2(3b + s - a) + \delta(4b + s - a) + 1\right)}{14b^2\delta^2 + 64mb\delta(1 + \delta) - 256mb^2(1 + 2b + \delta^2)} \]  

(11)

3.3. Comparison of simultaneous and sequential move games

In this section, we compare results of simultaneous and sequential move games that we developed with settings in Fig. 1. In particular, we focus on the level of investment in quality, vendor profits, and consumer surplus by comparing numeric examples of the two model solutions including Eqs. (9), (10), and (11).

3.3.1. Investments in quality

Fig. 4a and b indicate the changes in the levels of quality investments with an increase in \( b \) (rate of reduction in cost of service quality with an increase in product quality) at two levels of \( m \) (the cost of quality improvements). We observe that the level of quality investments of the leader in the Sequential move game is higher than the levels of investments of both vendors in the simultaneous move game, which in turn are higher than the investment level of the follower in the sequential move game. Further, in line with our intuition, we observe that the gaps in the levels of these investments decrease with an increase in the cost of quality improvements (\( m \)).

3.3.2. Vendor profits

Fig. 5a and b show the changes in the vendor profits with an increase in \( b \) (rate of reduction in cost of service quality with an increase in product quality) at two levels of \( m \) (the cost of quality improvements). We observe that the profits of the leader in the sequential move game are higher than the profits of both vendors in the simultaneous move game, which in turn are higher than the profits of the follower in the sequential move game. Furthermore, we observe that the gaps in the levels of these investments decrease with an increase in the cost of quality improvements (\( m \)).

3.3.3. Sensitivity of investment levels and profits to changes in the cost of delivering poor quality services (a)

To examine vendor behavior with changes in labor prices and similar factors that increase the basic costs of delivering services with poor quality, Fig. 6a and b indicate the changes in the quality investment levels and vendor profits with an increase in a (cost of delivering service cost with lowest level of software quality). We observe that the investments in quality as well as the profits of the leader in the sequential move game are higher than the (quality investment levels, profits) of both vendors in the simultaneous move game, which in turn are higher than the (quality investment level, profit) of the follower in the sequential move game. Furthermore, we observe that the gaps in the levels decrease with an increase in drivers of low-quality service costs (\( a \)).

3.3.4. Consumer surplus

In comparing consumer surpluses across the simultaneous move and sequential move games, we set the value of ES, the price of packaged software, the price of service to be the same because they are not influenced by the uninformed quality (\( x_e's \)). Thus, we could leave out terms for those in the surplus calculations and compare only the disutilities due to low quality. The consumer surplus in AA and BB in Fig. 3 above will not have disutility from low quality. The disutility of the customer in AB over the two periods will be \( k(\lambda - x_m) + k\delta(\lambda - x_m) \). Among the customers who adopted vendor B’s ES, but were not satisfied with its quality, the surplus of the consumers in BA will be \( k(\lambda - x_m) + k\delta(\lambda - x_m) \) but
that of the consumers in BA′ will be \( k(\lambda - x_B) \). Let \( H(\lambda, \varepsilon, x) \) denotes the disutility from low quality; then, \( H(\lambda, \varepsilon, x) \) can be formulated as:

\[
H(\lambda, \varepsilon, x) = \begin{cases} 
0 & \text{if } \varepsilon < 1/2 \text{ and } \lambda < x_B \\
(k(\lambda - x_B) + \delta k(\lambda - x_B)) & \text{if } \varepsilon > 1/2 \text{ and } \lambda < x_A \\
(k(\lambda - x_B) + \delta k(\lambda - x_B)) & \text{if } \varepsilon > 1/2 \text{ and } \lambda > x_B \\
k(\lambda - x_B) & \text{if } \varepsilon < 1/2 \text{ and } x_B < \lambda < x_A 
\end{cases}
\]  

(12)

Since we assume the uniform distribution of \( \varepsilon \) and \( \lambda \) \( f(\varepsilon) = 1 \) if \( \varepsilon \in [0,1] \), 0 otherwise; \( g(\lambda) = 1 \) if \( \lambda \in [0,1] \), 0 otherwise, the expected consumer surplus can be given by:

\[
\int_0^1 \int_0^1 H(\lambda, \varepsilon) g(\lambda) f(\varepsilon) d\varepsilon d\lambda
= \int_0^{1/2} \left( \int_0^1 k(\lambda - x_A) + \delta k(\lambda - x_B) d\lambda \right) d\varepsilon
+ \int_{1/2}^1 \left( \int_0^1 k(\lambda - x_B) + \delta k(\lambda - x_A) d\lambda \right) d\varepsilon
\]  

(13)

Fig. 4. a. Comparison of the level of investment in quality \((x_{A}^* \text{ and } x_{B}^*)\) between the sequential and simultaneous move games with variations in \( b \) \((a = 3, s = 5, m = 1 \text{ (lower)}, \delta = 0.9; b = 0.3 \ldots 3.0)\). b. Comparison of the level of investment in quality \((x_{A}^* \text{ and } x_{B}^*)\) between the sequential and simultaneous move games with variations in \( b \) \((a = 3, s = 5, m = 2 \text{ (higher)}, \delta = 0.9; b = 0.3 \ldots 3.0)\).

Fig. 5. a. Comparison of vendor profits \((\pi_{A}^* \text{ and } \pi_{B}^*)\) between the sequential and simultaneous move games with variations in \( b \) \((a = 3, s = 5, m = 1 \text{ (lower)}, \delta = 0.9; b = 0.3 \ldots 3.0)\). b. Comparison of vendor profits \((\pi_{A}^* \text{ and } \pi_{B}^*)\) between the sequential and simultaneous move games with variations in \( b \) \((a = 3, s = 5, m = 2 \text{ (higher)}, \delta = 0.9; b = 0.3 \ldots 3.0)\).
Fig. 6. a. Comparison of the level of investment in quality ($x_A^*$ and $x_B^*$) between the sequential and simultaneous move games with variations in $a$ ($b = 2, s = 5, m = 1, \delta = 0.9; a = 1\ldots 5.0$). b. Comparison of vendor profits ($\pi_A^*$ and $\pi_B^*$) between the sequential and simultaneous move games with variations in $a$ ($b = 2, s = 5, m = 1, \delta = 0.9; a = 1\ldots 5.0$).

Fig. 7. a. Comparison of consumer surplus between the sequential and simultaneous move games with variation of $b$ ($a = 3, s = 5, m = 1$ (lower), $k = 10, \delta = 0.9; b = 0.3\ldots 3.0$). b. Comparison of consumer surplus between the sequential and simultaneous move games with variation of $b$ ($a = 3, s = 5, m = 2$ (higher), $\delta = 0.9; k = 10; b = 0.3\ldots 3.0$). c. Comparisons of consumer surplus between the sequential and simultaneous move games with variation of $b$, at high $k$ ($a = 3, s = 5, m = 1, \delta = 0.9, k = 20$ (higher); $b = 0.3\ldots 3.0$).


in product quality) at two levels of $m$ (the cost of quality improvements). We observe that the consumer surplus in the simultaneous move game is greater than in the sequential move game. Next, with an increase in the intrinsic disutility of customer when faced with poor quality (high $k$), the difference in surplus between the simultaneous move and sequential move game increases. Further, increases in $b$, the delivery cost reduction of unit quality investment, are associated with reductions in the surplus. In other words, the more cost reduction that can result from a unit investment in quality, the lower the surplus for the customers. Finally, we observe that the gaps in the levels of these surpluses decrease with an increase in the cost of quality improvements ($m$).

### 3.4. Tradeoff between quality and fit

In the earlier models, we assume that software quality is the attribute customers value most, and customers do not adopt a vendor’s ES in the second period if the customer is not satisfied with the quality of the ES in the first period. However, we consider tradeoff between fit and quality of ES from the perspective of customers here. The tradeoff introduces sloped lines in the figure like Fig. 8.

In the second period, a customer knows the quality of ES that the customer adopted in the first period. Let $k$ also refer to the parameter that converts disutility from low quality to the dollar value. We assume $x_A$ and $x_B$ are small enough so that $s/k + x_A \leq 1$ and $s/k + x_B \leq 1$. Then, among the customers who are not satisfied with the software quality of vendor $A$ in period 1 ($AB$ in Fig. 1), only customers whose disutility from low quality, $k(\lambda - x_A)$, is greater than price for the misfit, $s(1-2c)$, will switch to vendor $B$ in period 2. Similarly, among customers in $BA'\,$ in Fig. 1, only customers under $k(\lambda - x_B) > (2c - 1)$ will switch to vendor $A$. Then, profits of vendors can be given by

$$\pi_A = \int_{s_A}^{x_A} p_x(x_A) dx_A + \int_{s_B}^{x_B} (s_B-c_B) f_x(x_B) dx_B - mx_A^2$$

$$+ \delta \left( \int_{s_A}^{x_A} p_x(x_A) dx_A + \int_{s_B}^{x_B} (s_B-c_B) f_x(x_B) dx_B - mx_A^2 \right)$$

$$+ \int_{x_A}^{1} p_x(1 - \frac{5}{k}(2c-1)-x_B) dx_B + \int_{x_B}^{s_B} (s_A-c_A) f_x(x_A) \left( \frac{5}{k}(2c-1)-x_B \right) dx_B$$

$$+ \left( \frac{5}{k}(2c-1)-x_A \right)$$

$$\pi_B = \int_{s_A}^{x_A} p_x(x_A) dx_A + \int_{s_B}^{x_B} (s_B-c_B)(1-\epsilon) f_x(x_B) dx_B - mx_B^2$$

$$+ \delta \left( \int_{s_A}^{x_A} p_x(x_A) dx_A + \int_{s_B}^{x_B} (s_B-c_B)(1-\epsilon) f_x(x_B) dx_B - mx_B^2 \right)$$

$$+ \int_{x_A}^{1} p_x(1 - \frac{5}{k}(1-2c)-x_B) dx_B$$

$$+ \int_{0}^{s_B} (s_B-c_B)(1-\epsilon) \left( 1 - \frac{5}{k}(1-2c)-x_B \right) dx_B$$

With the same assumptions in the previous section ($\epsilon_0 = 1/2, c_1 = a - b x_1$), we can get

$$\pi_A = (s-a + bx_A) \left\{ \frac{1}{8}(1 + 3\delta) - \frac{\delta s}{6k} + \frac{\delta x_A}{8} - \frac{3\delta s}{8k} x_B \right\} - mx_A^2(1 + \delta)$$

$$\pi_B = (s-a + bx_B) \left\{ \frac{1}{8}(1 + 7\delta) - \frac{\delta s}{6k} + \frac{\delta x_B}{8} - \frac{3\delta s}{8k} x_A \right\} - mx_B^2(1 + \delta)$$

From the first order condition of Eqs. (16) and (17), $x_A^*$ and $x_B^*$ can be calculated. The practical solutions and analyses are feasible with specific numbers similar to what we showed in the earlier sections.

### 4. Implications

We contribute to the IS literature by investigating the impact of software quality on service costs that are often a major portion in operation costs in ES markets. Though software quality has been studied extensively in prior literature, our analytical framework is unique in that it allows us to tease out the impact of quality improvement and operational service costs under alternative market structures.

Regardless of the market structure, prior research on quality (e.g., [9,13]) suggests that investments in upfront quality improvements can result in downstream service cost reductions. We explicitly incorporate this relationship into our models to infer implications for vendors and customers under variations in market structures. In addition, our model can be applied to other hybrid markets where firms earn revenues from selling both products and services highly related to those products. We apply the two-dimensional Hotelling’s spatial model to structure the problem of observable and unobservable attribute of ES. Because both observable and unobservable attributes of ES matter to customers, the two-dimensional model can provide a useful approach to problems in which decision makers should consider both types of attributes. These points of departure also enable us to differentiate our work from prior analytical research on product quality and innovation (e.g., [2,11,13]).

Our study enables us to understand plausible vendor behavior under competition in the duopoly setting, which is often observed in the marketplace. It attempts to understand competitor’s possible moves and conditions of equilibrium. Because the impact of quality on their moves, and the converse are not clear, our results can help understand vendor’s position incorporating the manner in which they handle organizational process initiatives such as quality control, service quality, and process improvements. For example, applying our research framework to their problem can help us understand whether there are incentives to invest in these organizational improvement processes depending on their position: whether they are the dominant players, whether it makes sense to be a follower in such initiatives, or maintain intense competitive behavior in the duopoly.

For customers, our framework can help them understand the vendor’s representation about their software quality from an economic incentive standpoint and whether their vendor-speak can be considered credible. This understanding can have a direct influence on the design of contract with such vendors. From the customer’s perspective, the quality of ES is the main concern. Because adoption of an ES requires significant financial and non-financial organizational commitment, it is the quality that most organizations care about most. However quality investments also have a self-interest component for the vendors in the sense that vendors could possibly invest only in those dimensions that help them reduce service costs. Our results have implications for the choice of a vendor.

### 5. Conclusions

Software vendors may hesitate to invest in quality, especially when it is difficult for customers to evaluate the quality attributes, such as
in enterprise software implementations. The packaged software is a ‘black-box’ to customers and it often takes a few years to judge its merit. For example, consider a software program that requires a long implementation period if the vendor follows a well-designed development process. Alternatively, the vendor might be tempted to develop a similar offering employing numerous ‘short-cuts’ (without following a well-designed development process). As a result of this choice, customer organizations may have to incur substantial organizational and financial costs after they adopt the vendor’s enterprise system. Our study finds that vendors can be better off and earn higher profits by investing in the unobservable quality dimensions in the duopoly market.

SAP® and Oracle® dominate the markets for ERP, CRM, SCM, Inventory Management, and Human Resource Management solutions particularly for large clients. Those markets can be viewed as a duopoly. Our analysis shows that there is equilibrium of the quality investment in the duopoly. One of our main contributions is that we examine the economic incentives that persist in the enterprise system market. Enterprise systems are inherently different from traditional software and traditional products. Economics of enterprise systems goes beyond the traditionally-defined boundaries of digital products, and thus we believe it is worth investigating participants’ economic activities and market equilibrium. Though sales of combinations of products and services are common in other industries, the weight of the services component of the transaction in enterprise system markets is much greater than in other product markets. It is observed that enterprise software companies, such as IBM and HP, have tended to sell their software more as a solution or a service rather than as a product. As this trend continues, the economic significance of customization, maintenance services and customers’ satisfaction will increasingly become more important, and the fact that product quality is a driver of these costs cannot be ignored. We hope that our study can initiate further research in this area.

Our models and the paper have certain limitations that we acknowledge here. First, we consider a two-period game as illustrative of the enterprise systems choice. Future research could consider a multi-period setup despite the added complexity of the solution. Second, the results of our comparisons can only be generalized to a certain extent since they are based on numerical solutions. We were not able to provide analytical results of the comparisons due to the significant complexity of the solutions. Though we obtained consistent results from multiple experiments with reasonable range of values for all our variables, it is possible that we might have missed some conditions that might yield different results. Next, we utilized a simplified service function for the vendor in that there is a negative linear relationship between software quality and service costs. A more nuanced service cost function can be developed and incorporated into the model. These models can also be complemented by means of continuous game analyses. Additionally, empirical data and analyses on the profitability and quality performance dimensions in these markets could significantly benefit our understanding of incentive issues in these complex environments. As several of our findings are based on financial benefits, under various market conditions, from investing upfront in quality practices, this notion provides certain empirically testable hypotheses. For example, future research in settings such as outsourced arrangements can examine relationships between quality and process improvement investments and the vendor’s ability to financially outperform their competition. The financial performance noted here goes beyond efficiency-related savings (inward looking for the firm) noted in software engineering literature, for example (e.g., [9]). Further, we employ a single parameter for quality and assume a single threshold for this measure for each customer firm. In reality, it could be maintained that quality is a multi-dimensional construct. Survey-based or archival-data based research that addresses this multi-faceted notion of quality could complement our findings. For instance, case studies that highlight the challenges and nuances that persist to the unobservable and observable quality dimensions can complement our findings, and can be conducted by investigating a single firm or a small sample of firms in detail.

Acknowledgment

The third author acknowledges financial support by the Major Program of the National Natural Science Foundation of China (Grant Number: 91218301), as well as Financial Service Innovation Team Development Project at Southwest University of Finance and Economics (2012).

References


Woosoo Cho is an Assistant Professor in the School of Technology Management at the UNIST (Ulsan National Institute of Science and Technology) in Ulsan, Korea. He received his PhD from the University of Illinois at Urbana-Champaign. He has industry experience at Samsung SDS as a Software Engineer. His research interests are in Strategy for IT firms, Strategic IT Investments, Information Security, and Information Systems Quality.

Ramanath Subramaniam is an Associate Professor in the Business Administration department of the College of Business, University of Illinois at Urbana-Champaign. He received his PhD from University of Michigan. His research interests include IS/IT Contracting, IT Governance, Software Measurement, Software Engineering Economics, Customer Satisfaction for IT, and IT Sustainability. He has published in journals such as Information Systems Research, Decision Support Systems, and Manufacturing & Service Operations Management.

Mu Xia is an affiliated researcher at the Key Lab of Financial Intelligence and Financial Engineering, Southwestern University of Finance and Economics in Chengdu, Sichuan, China. He received his Ph.D. in Management Science and Information Systems from the University of Texas at Austin’s McCombs School of Business. His research interests include e-business standardization, online communities especially peer-to-peer networks, electronic commerce, and combinatorial auctions. His work has been published in many scholarly journals including Information Systems Research, Journal of Management Information Systems, and Decision Support Systems.