MIS Quarterly

RESEARCH ARTICLE

BUSINESS FAMILIARITY AS RISK MITIGATION IN SOFTWARE DEVELOPMENT OUTSOURCING CONTRACTS[1]

By: **David Gefen**
Management Department
LeBow College of Business
Drexel University
101 N. 33rd Street/Academic Building
Philadelphia, PA 19104-2875
U.S.A.
gefend@drexel.edu

Simon Wyss
Department of Management Information Systems
University College Dublin
Dublin 4
IRELAND
si_wyss@yahoo.com

Yossi Lichtenstein
College of Management
7 Rabin Boulevard
Rishon Le-Zion
ISRAEL
Department of Management Information Systems
University College Dublin
Dublin 4
IRELAND
yossil@colman.ac.il

Abstract

This study examines the role of business familiarity in determining how software development outsourcing projects are managed and priced to address risks. Increased business familiarity suggests both more prior knowledge, and hence reduced adverse selection risk, and increased implied trust about future behavior, and hence implied reduced moral hazard risk. Preferring high business familiarity partners may also alleviate concerns about incomplete contracts. By reducing these risks, higher business familiarity is hypothesized to be associated with higher priced projects, reduced penalties, and an increased tendency to contract on a time and materials rather than a fixed price basis. These hypotheses were examined with objective contractual legal data from contracts made by a leading international bank. Integrating trust theory into agency theory and into incomplete contract theory and examining unique contract data, the contribution of the study is to show that the premium on business familiarity and the trust it implies is not in directly affecting price, but, rather, in changing how the relationship is managed toward a tendency to sign time and materials contracts. Implications about integrating trust into agency theory and incomplete contract theory, as well as implications regarding trust premiums and software development outsourcing, are discussed.

Keywords: Business familiarity, software development outsourcing, fixed price, time and materials, agency theory, incomplete contract theory, trust, contractual governance

Introduction

Information technology outsourcing has become a modus operandi in many organizations (Lacity and Willcocks 1998).

[1]Bernard C. Y. Tan was the accepting senior editor for this paper. Ravichandran Thiagarajan was the associate editor. Ron Thompson and Eric Walden served as a reviewers.

Reflecting this, already three quarters of all large firms are engaged in large long-term outsourcing contracts (Economist 2005). Despite this growing popularity, industry surveys claim that only half of all software development outsourcing (SDO) projects are successful and that about 28 percent of the projects that fail do so because of contractual management issues (McDougall 2006). Broadly speaking, the primary risks in SDO are monetary, project, and technology related (Benaroch et al. 2006). These risks, common to software development in general, are exacerbated in SDO by the fact that it is an *external* vendor who is developing the software. As we shall argue, knowing the vendor through previous business contracts should reduce these exacerbated risks because of increased mutual knowledge and trust.

Accordingly, this study examines how business familiarity with the vendor (i.e., the number and dollar amount of previous business relationships) reduces reliance on contractual controls that, at least partly, address these SDO enhanced risks. These contractual controls include: (1) splitting large pricy projects into smaller ones; (2) adding penalties to control quality; and, (3) choosing an appropriate method of contractual governance, primarily whether the contract will be managed on a fixed price (FP) or on a time and materials (TM) basis.[2] These controls are common risk-reducing contractual controls, and are also found in other types of outsourcing contracts (Bolton and Dewatripomt 2005; Kalnins and Mayer 2004).

The study examined the legally binding details of SDO contracts signed by a leading international bank during a period of over three years, amounting to over half a billion U.S. dollars. The analyses combined these objective contract data with the specific characteristics of the vendors. The study addresses a central topic in trust theory, namely what the premium on trust is. In this study, this premium was related to how external projects were controlled, but not to their contract price or penalties. This premium adds a new perspective to the current predominantly agency theoretical perspective of SDO in contracting research. Indeed, in TM contracts, because their requirements are more prone to changes, a client must rely more on the vendor and is more susceptible to its opportunistic behavior. The study is also suggestive about the relative difficulty in managing different sources of development risk. Risks resulting from the evolving nature of software and its specifications are relatively difficult to manage and require trusted vendors. In contrast, other risks, including those related to large and complex systems, are managed well by defining detailed specifications and deliverable schedules.

The contributions of the study are several.

1. This is one of the early studies to date to empirically examine the relationship between business familiarity and outsourcing contract price and penalty in the software development context. SDOs are unique because their deliverables are not as easily defined and their quality is not as verifiable as in other types of contracts, and so there are even fewer watertight warranties in SDO. SDO are also unique because they deal with an intellectual rather than a physical property (Walden 2005) and are subject to constantly evolving requirements (Beizer 1990) and price overruns (Benaroch et al. 2006).

2. The study integrates trust and incomplete contract theory perspectives into SDO research which previously dealt mainly with agency theory, showing that the premium on business familiarity and the trust it implies in SDO is not in higher prices, as suggested by previous research on trust in online business-to-consumer product contracts (Ba and Pavlou 2002), but is in a tendency to sign TM contracts. Our *post hoc* interviews suggest the reason behind this.

Literature Review and Research Model

Business Familiarity and Risks in SDO

Managing risk correctly is the key to successful outsourcing, and indeed outsourcing contracts include controls to explicitly address expected risks (Bolton and Dewatripomt 2005; Gopal et al. 2003; Kalnins and Mayer 2004). SDO risks can be classified into two broad categories: (1) client–vendor relationship risks that exacerbate software development risks, and (2) unforeseen contingencies related to changes and additions to the software specifications during the development period. Past SDO research focused mainly on client–vendor relationship risks (e.g., Gopal, et al. 2003; Snir and Hitt 2004). This study also examines unforeseen contingencies, and the coordination and pricing problems these entail (Kern and Willcocks 2001). The risks discussed in the next sections, while by no means meant to serve as a comprehensive list, are representative of major client and vendor SDO risks as discussed in the literature (Benaroch et al. 2006).

[2]Most SDO contracts are of these two types (Banerjee and Duflo 2000; Gopal et al. 2003). A third type of contracting, dealing with a hybrid TM with a fixed price cap, is often seen as a transition stage between the FP contracting of initial relationships and the consideration of TM contracting in repeated relationships (Kalnins and Mayer 2004). This was also the case with the bank where the data were collected. This hybrid type of contracting was therefore of less theoretical interest.

SDO risks arguably determine the risk-reducing controls found in SDO contracts, as they do in other outsourcing contracts (Bolton and Dewatripomt 2005). These risks and their related contractual controls are split in our research site into the pre- and the post-contract signing periods, as illustrated in the bottom half of Figure 1a. This is consistent with economic theory on contracting where risks are divided into the pre- and the post-contract signing periods, dealing primarily with adverse selection and moral hazard risks, respectively (Bolton and Dewatripomt 2005). The bank had three primary contractual controls: price, contractual governance, and penalty. Price, by splitting large projects into smaller ones, controls risks associated with size. This is applied before the requests for proposals (RFP) goes out. Contractual governance is applied throughout the selection and development phases. Penalties are applied at the software acceptance testing phase.

The study examined how business familiarity reduces some of these risks and, accordingly, how the contractual controls are affected. This is done in view of the two aspects of business familiarity, namely knowledge based on *prior* relationships and the implied *future* trust it brings about (Gulati 1995), as illustrated in the broken-line boxes in Figures 1a and 1b. Trust is the willingness to rely on a business partner based on its past trustworthy behavior (Hosmer 1995). Within this context, before the contract is signed, both client and vendor can reduce their respective risks through increased knowledge about each other (i.e., business familiarity). Also at contract signing time, both client and vendor can reduce their respective risks about the future behavior of their partner by preferring a trusted partner (i.e., a trusted partner based on performance in previous contracts).

Precontract Risks in SDO

Precontract risks deal primarily with software development risks in the context of the client–vendor relationship before the project begins. These risks, as they relate specifically to SDO, are primarily: (1) monetary risk such as cost overruns; (2) project execution risks, such as the lack of skills and experience, over-complex projects, and inadequate design; and (3) technological risks relating to immature technology (Benaroch et al. 2006). These risks relate to both client and vendor. These risks can be reduced through appropriate planning, monitoring, and control (Keil et al. 2003).

SDO risks relate to both the client and the vendors. On the client side, viewed from an agency perspective (Eisenhardt 1989; Salanie 1997),[3] these risks are increased by adverse selection risk (Snir and Hitt 2004), that is, the risk of choosing an inappropriate vendor. These risks and how business familiarity reduces them are shown in the top left hand side of Figure 1a. Choosing an appropriate vendor is crucial because the client cannot fully observe and control the development. Furthermore, as software development is not as standardized a process as other engineering activities, and as quality warranties for software are rare, it is difficult to identify vendors with specific competences at the bidding stage (Snir and Hitt 2004), although adherence to CMMI[4] can to some extent alleviate such concerns. Moreover, this lack of quality control may encourage some vendors to oversell their abilities (Snir and Hitt 2004), a problem aggravated by the lack of warranties in SDO, which makes legal recourse more problematic than in other instances of outsourcing. On the other hand, choosing an appropriate vendor increases the chances of employing appropriate expertise and competencies (Levina and Ross 2003). In terms of software development risks, an appropriate vendor should be able: (1) to reduce monetary risk by making more accurate cost estimations and by reducing costs; (2) to reduce project risks by employing the right skills and experience, by recommending when to avoid projects that are too complex, and by designing the system appropriately; and (3) to reduce technology risks by knowing the available technologies and what they can produce. As recent industry reports indicate (Artunian 2006), outsourcing contracts often fail because the vendor lacked appropriate capabilities. In other words, client risks may be increased when the client lacks knowledge about the vendor prior to signing the contract.

Turning now to the vendor's perspective, vendor risks are related to partial knowledge of the client and the project. These risks and how business familiarity reduces them are shown in the top left hand side of Figure 1b. This lack of knowledge may increase monetary risk because the vendor may bid too low or allocate insufficient personnel. Similarly, limited knowledge of the project may increase project and technology risks because the vendor may be unaware that it lacks the necessary skills, experience, and technology.

[3]Agency theory deals with relationships between independent parties with competing goals in which one party, the principal, delegates work to another, the agent, who does the work and as a result has more information about the project than does the principal. In SDO, the client is the principal and the vendor who is developing the software is the agent. Both client and vendor make their contracting decisions under conditions of partial information about the objectives and behavior of the other party (Gopal et al. 2003).

[4]Information on the Capability Maturity Model® Integration (CMMI) is available at http://www.sei.cmu.edu/cmmi/.

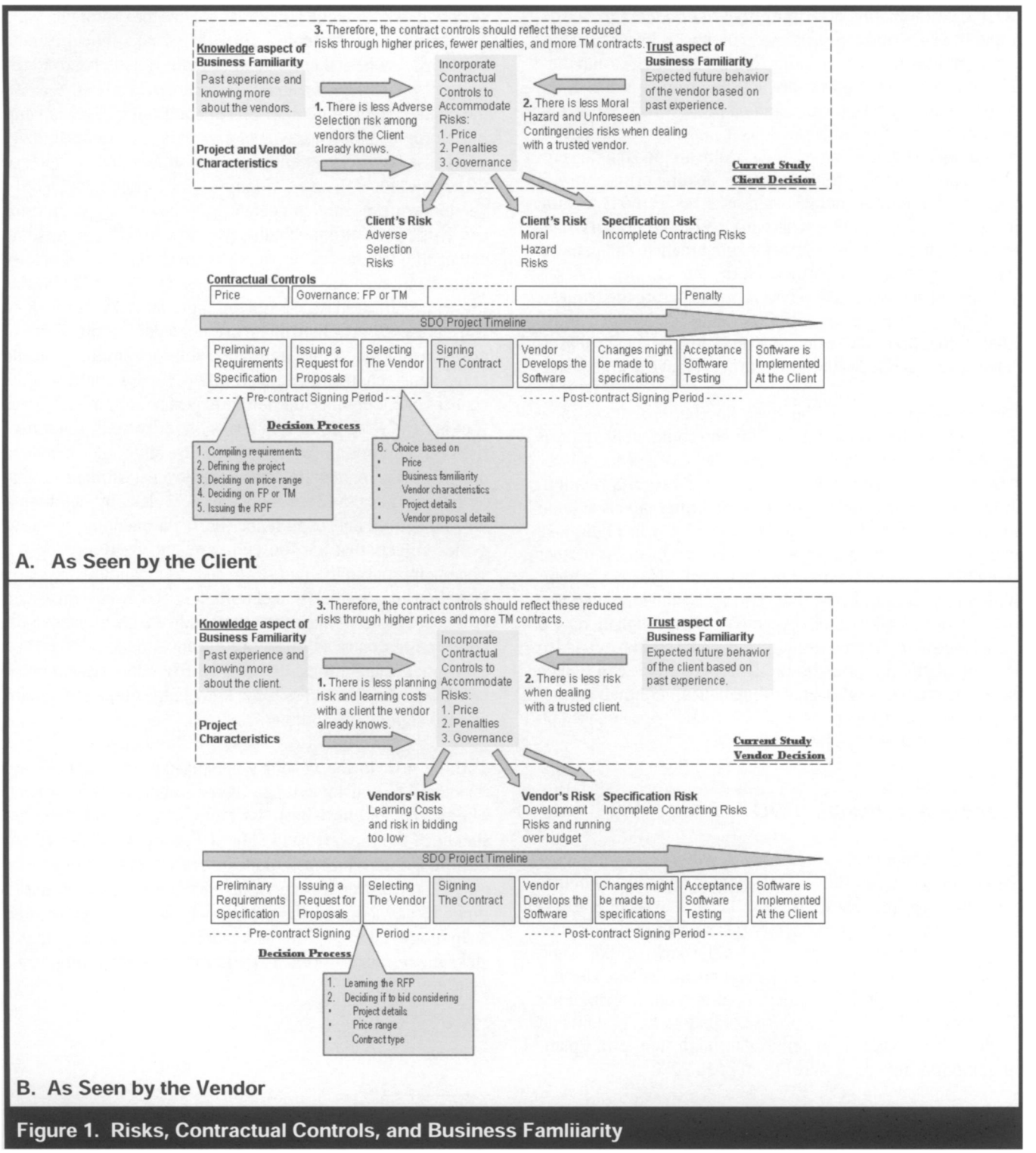

Figure 1. Risks, Contractual Controls, and Business Famliiarity

Reducing Precontract Risks in SDO Through the Knowledge Aspect of Business Familiarity

Addressing such precontract risks is where the first aspect of business familiarity (i.e., knowledge based on past contracts) comes in. Contract theory posits that repeated contracting (i.e., business familiarity) reduces private information about both client and vendor (Bolton and Dewatripomt 2005). In particular, information about the vendor's capabilities are revealed with repeated contracting, allowing the client to screen vendors for appropriate capabilities and performance (Snir and Hitt 2004). Similarly, business familiarity increases the vendor's knowledge about the client, allowing it, for example, to allocate the right resources for each client and to make more accurate price estimates. Business familiarity also increases the vendor's tacit knowledge about the client, its business environment, and its processes (Ethiraj et al. 2005), eases communication and understanding between client and vendor (Williamson 1979), and reduces the need for elaborate and costly contracts (Corts and Singh 2004). The result of these factors is a reduction in the various SDO risks, including monetary risks (through better cost estimation and less costly contracts), project execution risks (through better communication, client-specific resources, and tacit knowledge), and technological risks (thanks to vendor screening, and knowledge of the client's processes and systems).

Post-Contract Risks in SDO

Post-contract signing risks deal with risks during the development and implementation phases. These risks, as they can be anticipated at contract signing time, also deal with monetary risk, project execution risks, and technological risks (Benaroch et al. 2006). Again, these risks are not unique to SDO but are exacerbated in the case of SDO by the fact that the developer is an outside contractor.

On the client's side, viewed again from an agency perspective, these risks are exacerbated because of moral hazard (Snir and Hitt 2004), that is, the need to deal with hidden unsatisfactory vendor actions, effort, performance, etc. after the contract has been signed. These risks and how business familiarity reduces them are shown in the top right hand side of Figure 1a. These moral hazard risks are especially high with software because software development processes are not fully standardized, requirements evolve, and deliverables are complex, and their quality is not easily verifiable. Outsourcing increases these risks because the client often lacks relevant technological resources and the vendor may not wish to share its proprietary knowledge; after all, doing so may compromise its negotiating position for future contracts and make it easier for the client to include other vendors in the bidding (Pisano 1989). Moreover, these moral hazard risks extend beyond the development phase, the main topic addressed by previous SDO research (Ethiraj et al. 2005; Gopal et al. 2003), into the software acceptance testing phase, where the cost of defects may be higher, in particular when delays cause missing business opportunities.

The vendor also faces increased risks due to hidden behavior at the post-contract phase, in this case by the client. This is shown in the top right hand side of Figure 1b. Supportive client behavior is crucial to the vendor because the vendor needs support from the client to access knowledge and to gain the cooperation of client personnel. Similarly, client actions, access, and cooperation are critical when changes are required, developed, and implemented.

Reducing Post-Contract Risks in SDO Through the Trust Aspect of Business Familiarity

Addressing such post-contract risks is where the trust aspect of business familiarity comes in. While the knowledge aspect of business familiarity reduces the client's adverse selection risk and the vendor's miscalculations prior to signing the contract, the trust aspect of business familiarity reduces the client's moral hazard risk and the vendor's development risks. If trust-based relationships can be achieved, then in these long-term orientation relationships the parties involved can depend on each other to do the right thing and not take undue advantage of the situation (Gulati 1995; Kumar et al. 1995). This is why trust is crucial in enabling and shaping many business relationships, especially if the parties lack control over, but depend on, each other (Kumar et al. 1995), as they do in SDO. Trust also allows companies to behave as though doubts about the conduct, ability, and intentions of the other party can be put aside, and so be confident about the constructive behavior and intentions of the other party (Gefen 2004).

Assuming there is such a trusting relationship, both client and vendor can expect to partially alleviate their post-contract risks, knowing that the other will take constructive steps to cooperatively remedy problems without being opportunistic about it. Misunderstandings and the ramifications of unforeseen contingencies, such as changes in the requirements as the project progresses, are common in software development (Beizer 1990; Kern and Willcocks 2001). Conversely, if the other party cannot be trusted, the increased risks, because of its potentially uncooperative attitude and behavior, can make any arrangement prohibitively risky and expensive (Kumar et al. 1995). After all, even in meticulously written contracts some details often cannot be fully spelled out and must be

resolved later (Koh et al. 2004), making it essential for both client and vendor to trust each other beyond the explicit details in the contract (Gefen 2004; Wang et al. 1997). Having a trusting relationship contains the additional benefit of reducing contract costs and monetary risks, thanks to a reduced need to rely on extensively detailed contracts and on monitoring as protective measures to assure expected outcomes (Kumar et al. 1995). Indeed, past SDO research has speculated, although it did not measure, that trust may be a factor also in SDO, at least when the market is perceived as risky (Gopal et al. 2003; Lichtenstein 2004).

Business Familiarity as a Proxy of Trust

Business familiarity has been treated by previous research on company relationships as both a central antecedent of trust (Corts and Singh 2004; Gefen 2004; Gulati 1995; Kumar et al. 1995; Zucker 1986) and as a practical objective and directly measurable proxy of the otherwise subjective and elusive trust construct in business alliances (Gulati 1995; Gulati and Singh 1998). To be precise, business familiarity is not the same as subjective trust (Luhmann 1979). Trust deals with current beliefs and willingness to depend on another, while business familiarity deals with actual past behavior (Gefen 2002; Luhmann 1979). However, business familiarity, especially when the parties involved can choose between many other partners, is also a sign of actual past trusting behavior (Gulati and Singh 1998).[5] Accordingly, business familiarity is both a knowledge-based antecedent of current trust and a proxy of actual trust in these past transactions (Gulati 1995).

Typically, trust is the product of beliefs in the trustworthiness of the other party (Mayer et al. 1995; McKnight et al. 1998; Mishra et al. 1998) and is usually defined and measured as a subjective intention or assessment. (For detailed reviews, see Gefen et al. 2003; McKnight et al. 2002.) Obviously, such subjective cognitions are not included in business contracts. This is where, extrapolating from Gulati (1995), business familiarity is of special importance in SDO. Assuming there are many vendors and assuming untrustworthy vendors got weeded out by the client, the extent of previous contracts should be a good indicator of how much the client actually trusted the vendor.

[5]This was the case with the bank as well. The bank avoided vendors it did not trust. Having said this, it should be explained that trust is not a yes or no variable. It is a scale. In the case of the bank, as in other IT contracts (Gefen 2004), as vendors completed their contracts properly they were trusted to a greater degree. And so, while vendors who did not prove trustworthy were weeded out, those who remained could improve the degree of trust the bank had in them.

Hypotheses

Business Familiarity and Contract Price

A high degree of business familiarity should accordingly reduce the respective risks the client and the vendor have both before and after the contract has been signed, and therefore both client and vendor should be less compelled to add risk-reducing controls into the contract. Their respective risks should be smaller in such a case, as explained above, because both should know each other better based on previous projects and should trust each other more concerning future ones. Conversely, clients and vendors who do not know each other well can rationally be expected to take stronger steps to reduce their vulnerability and increase their control of the situation through contractual controls. Contracts are explicitly designed to address risks and increase control (Bolton and Dewatripomt 2005).

Since one of the major ways of reducing risk and increasing control is to split a large contract into several smaller projects, there should be a tendency to sign smaller contracts with vendors with whom the client is less familiar. Splitting large projects into smaller contracts should benefit both client and vendor in reducing risks. To be precise, although the decision on the actual project price is determined on the basis of the bids the vendors make, clients often decide before issuing an RFP on the price range of the project and whether to split larger, and therefore more risky, projects into smaller ones. Aware that larger projects are more risky, clients should rationally prefer to give these projects to higher business familiarity vendors, vendors whom the client knows better and conceivably trusts more. As a result, lower business familiarity vendors should initially receive smaller contracts. Contributing to this decision, in smaller projects the client can exert more control on the software development process by being able to intervene or stop the project when something goes wrong or should the vendor not live up to expectations without having to wait as long as it might have had to had the large project not been broken up into smaller segments. Indeed, breaking a large project into many smaller steps is an accepted method in software engineering to increase control over the process (Barki et al. 1993; Beizer 1990). Splitting large projects, the client can also include within them incentives to the vendor: if the vendor meets its obligations adequately at this stage of the project, it has a good chance of winning the next stage. Also, in other types of business relationships (Rao and Bergen 1992), preferring to contract with the same vendors repeatedly, assuming they fulfill their obligations, gives these vendors an incentive to invest more in the relationship with the client (Bakos and Brynjolfsson 1993; Walden 2005). Additionally, to some extent, clients

can also reduce their adverse selection risks this way by screening vendors based on their performance in the previous stages of the project (Snir and Hitt 2004).

Reducing risk by splitting large contacts should hold true also from a vendor perspective, especially if the vendor is not certain yet what to expect of the client because of low business familiarity with it. Since larger projects mean more risk also to the vendor, and since this risk is reduced to some extent through business familiarity, there should be a tendency for low-business-familiarity vendors not to bid in large, and hence more risky, contracts. After all, it is clearly in the interest of the vendor to make sure the project is a success, assuming the vendor wants future contracts with the client. Should the project fail, some of the fault, rightly or not, is likely to be placed on the vendor, thus decreasing its chances of winning future contracts with this lucrative client. Maintaining a good reputation with the client is crucial in this regard (Lichtenstein 2004; Taylor 2007). Splitting large contracts into smaller projects reduces the risks the vendor is taking in each project because there is less money involved and because size is highly correlated with software complexity and the number of glitches (Beizer 1990). Conversely, vendors with a high degree of business familiarity, arguably, should know the client better, how it operates, and its business environment (Walden 2005), and should therefore be at less risk when taking on large projects. Accordingly, large, pricy SDO projects should go to familiar vendors, while smaller SDO projects should go to less familiar vendors.

> H1: Greater business familiarity with the vendor is associated with higher contract price, even after controlling for project duration and complexity.

Business Familiarity and Contract Penalties

Another consequence of increased business familiarity should be its effect on reduced contractual penalties. Penalty clauses are a common way in which clients protect themselves from inadequate behavior by vendors. Should the vendor breach the contract, the penalty will come into effect and the client will be compensated. This is quite common, as shown in a study of 34 contracts using telephone interviews where Saunders et al. (1997) found that over 80 percent of the outsourcing arrangements that were perceived as successful had tight contractual controls. We can therefore assume that with less familiar vendors, the client, knowing less about the vendor, should adopt more risk controls in its contracts (Eisenhardt 1989). Penalties, however, are a double-edged sword. On the one hand, the penalty allows the client to control financial risk and to deter the vendor from inappropriate conduct. On the other hand, a large penalty clause has the disadvantage of increasing the overall price tag because the vendor will likely charge a higher price if it faces a bigger possible penalty. Accordingly, since there is less risk with more familiar vendors, and the client can therefore rely less on the penalty clause as a way of controlling the vendor, there should be lesser penalties with these vendors.

> H2: Greater business familiarity with the vendor is associated with smaller penalties, even after controlling for project duration and complexity.

Business Familiarity and Contract Type

A Risk Control Perspective

It is widely recognized by both industry and research (Gopal et al. 2003) that the FP or TM decision is a key legal and managerial one in SDO (Kalnins and Mayer 2004). SDO contracts are mostly either FP or TM (Banerjee and Duflo 2000; Gopal et al. 2003). In an FP contract, the software development project is managed by the client from afar. The client defines the project and then invites vendors to bid on it. The winning vendor then receives detailed requirement specifications and deliverable timetables. The vendor is contractually obliged to deliver the software as defined and in working order by the specified dates or pay a penalty. In contrast, in a TM contract the client takes an active part in the software development process, actively engaging with the vendor as the software is being developed and paying the vendor according to the vendor's investment in time and related materials. TM contracts are generally more efficient when the project is more complex (Bajari and Tadelis 2001).

From the client's risk perspective, FP and TM represent a tradeoff: unloading most risks but losing flexibility versus bearing most risks but retaining responsiveness to inevitable changes, respectively. FP projects are based on well defined and supposedly stable requirement specifications as well as detailed timetables for deliverables and acceptance testing. The cost of the project to the client can be assessed rather accurately when signing the contract, but only providing the specifications remain the same, and as a result much of the monetary, project, and technology risks are borne by the vendor. FP contracting, however, is not always the most advantageous way to contract software development. Creating such detailed and verified specifications is often too complicated to be practical, especially as the requirements may change as the software is being developed and as previously unforeseen additional features and connections with other software modules need to be included. Such changes in the requirements are to be expected in large software projects (Beizer 1990; Kern and Willcocks 2001). This need for flexibility by

the client is compounded by the typically long time it takes to develop software and the systemic and business changes that are therefore likely to occur after the specifications were originally signed. To account for these almost inevitable changes and additions, many projects are contracted on a TM basis where the client, because it is paying for the time and materials invested by the vendor, has more flexibility and can therefore modify the specifications while the software is being developed. However, TM shifts the monetary, project, and technology risks mostly to the client. As a result, the decision to contract TM or FP is made by the client based on project characteristics before the RFP is published. This process is shown in the "Decision Process" section at the bottom half of Figure 1a.

Accordingly, clients should prefer to grant the TM contracts to more familiar vendors because the increased monetary, technological, and project management risks in TM are going to be magnified if the vendor cannot be trusted. Moreover, business familiarity should directly reduce client risks by revealing private information about vendor capability, by easing communication and understanding, and by increasing vendor investment in client-specific resources. In practical terms, in the case of SDO, this private information could deal with vendor expertise or lack thereof, its development pace, how it operates with its clients, how it manages to retain its programmers, and to some extent its internal work processes. Knowing these could put the client in a better position to know what to expect in projects managed as TM contracts.

The vendor perspective is to prefer TM contracts. It bears little risk and, should there be difficulties, the client is expected to pay anyway. Sophisticated vendors, however, should consider the long-term as well, in particular, the influence of a successful current project on their chances to gain future contracts with the client and to retain or establish reputation in the market (Bolton and Dewatripomt 2005). Consequently, vendors should prefer TM projects with clients they are familiar with because the chances of project success in this case are arguably higher. Moreover, doing so also allows them to employ client-specific resources, for example, by making more accurate assessments based on their increased knowledge about the client, its existing information systems, and its business environment (Ethiraj et al. 2005). Research in other scenarios lends indirect support to this conclusion: in the oil drilling industry repeat business partners also tend to prefer TM contracting (Corts and Singh 2004).

An Incomplete Contract Perspective

Another reason why more TM projects should go to high business familiarity vendors is related to incomplete contract theory. A major consideration in making the FP or TM decision by the client is the extent of expected changes to the specifications. Such changes are quite common in large software projects (Beizer 1990). When the contract can be spelled out in detail, the risks are mainly adverse selection and moral hazard, extensively discussed in contract theory based on a principal-agent relationship (Tirole 1999). However, when the contract cannot be fully spelled out, there is also the risk of unforeseen contingencies (Tirole 1999). Such contingencies are dealt with in incomplete contract theory (Bolton and Dewatripomt 2005; Tirole 1999) based on the assumption that one of the core problems with unforeseen contingencies is ***ex post*** opportunism (Williamson 1979). This literature has suggested several contract types in this regard, mainly in the context of employment relationships (Simon 1951) and ownership and property rights (Grossman and Hart 1986). The theory interprets employment as a way to control the risk of ***ex post*** opportunism as a result of unforeseen contingencies (Bolton and Dewatripomt 2005; Simon 1951; Tirole 1999). When a service can be precisely specified at the time the contract is being signed, a sales-contract is preferred; however, when the services are not fully known ***ex ante***, and are to be determined by the client, an employment contract is preferred (Simon 1951). Conceptually, FP contracts are close to market transactions with predefined outcomes and price, while TM contracts resemble employment relationships where the client (employer) controls the work performed by the vendor (employee) and compensates it for its time. Extending this reasoning, TM controls the risk of vendor opportunism when unforeseen contingencies require changes in specifications, because the vendor, resembling an employee, is paid the same rate regardless of changes.[6] Incomplete contracting, except in the case of an employment contract, may also require ***ex post*** renegotiations to account for new requirements after the agent has ***ex ante*** implemented them (Bolton and Dewatripomt 2005).

Relying on business familiarity presents an alternative mechanism to control ***ex post*** opportunism. Given that contracts cannot be renegotiated as a matter of company policy by the bank and that vendors too prefer not to have such ***ex post*** renegotiations, having a trusting relationship (i.e., preferring higher business familiarity partners) could alleviate some of the problems with incomplete contracts. In trust-based rela-

[6]Alternatively, in theory, the venture could be sold to the agent to encourage it to invest optimally in the venture (Bolton and Dewatripomt 2005). However, such selling is problematic in SDO because by the time the software has been completed it is a sunk cost for the vendor and hence its ***ex ante*** investment may be compromised by the client's ***ex post*** opportunism. In the case of the bank, such an option is anyway inappropriate because most systems are part of an interconnected portfolio of applications the bank operates and owns.

tionships, the parties can be expected to avoid opportunist behavior toward each other and can reasonably expect the other to compensate them in future contracts for higher unexpected costs in the current project (Kumar et al. 1995). Indeed, in theory, incomplete contracts are treated more efficiently regarding both *ex ante* investment and *ex post* renegotiation if principal and agent are integrated into one company so they do not compete with each other (Grossman and Hart 1986). In other words, business familiarity presents an alternative *trust perspective*, where risk is put aside because the business partner can be trusted not to be opportunistic, to the *risk control perspective* of incomplete contracts theory, where risk is central because the business partner may be opportunistic. Practically, this trust perspective is crucial because contracts are often incomplete and it is common to have more going on between the companies than the contract stipulates (Kumar et al. 1995). Accordingly, based on both a risk control and an incomplete contract perspective,

> H3: Greater business familiarity with the vendor is associated with more time and materials contracts, even after controlling for project duration and complexity.

Contract Type, Penalties, and Price

The next hypothesis deals with the effects of contract type on contractual penalties. It is in the nature of TM projects that the client engages the vendor much more, is more involved with the software development process, and can keep closer tabs on what is going on than in FP projects. In addition, TM projects are often onsite, making it easier to evaluate the software during development, rather than as a black box during testing as in FP projects. As a result, in TM projects the client should be less exposed to the risk of inappropriate vendor behavior, and hence should need to rely less on the penalty clause in the contract to keep the vendor on track. Most importantly, penalties are often impractical in TM contracts because doing so requires a clear definition of deliverables and schedules at contract signing time, and the lack of such a definition is typically the reason why many projects are designed as TM in the first place. In contrast, in FP projects the client needs to add some bite to its contracts to make sure the vendor delivers on time and of the quality specified.

> H4: Time and materials contracts are associated with lower penalties than fixed price ones, even after controlling for project duration and complexity.

The relationship between contract type and price is an open-ended question. On the one hand, large, pricy, and thus risky projects are not suitable for FP contracts, because vendors' risk premiums might be too high, and because the client's risk controls are too weak. On the other hand, FP contracts are amenable to aggressive competitive bidding that reduces price, and so may be tempting to clients especially in large projects. Past SDO research did examine the related question of how contract type affects vendor profit but did not examine project price. Regarding profit, according to Gopal et al. (2003), TMs are associated with more vendor profit. On the other hand, Ethiraj et al. (2005) found that FP contracts were correlated with more profit to the vendor, although the empirical evidence, in their words, was modest because when other factors were included its effect became insignificant. Accordingly, no explicit relationship is hypothesized between contract type and price. Any correlation between the two is expected to be on account of both being theoretically correlated to business familiarity.

Project Characteristics Controls

Additional controls, albeit not related directly to business familiarity, were included in the research model to account for expected effects and to show that the hypothesized paths are significant even when these control paths are included.

Project Duration

Typically, the price of a software development project is determined mostly by time invested (i.e., the product of project duration by team size). Since, as a matter of bank policy, the contracts did not specify how many people would be assigned to the project, this variable is a lower-bound estimate of time invested. Longer projects mean more time invested, which means more salary expenses, and hence a higher price tag. Moreover, longer projects are typically larger, harder to describe in detail in the requirement, and are generally more prone to changes during development (Barki et al. 1993), and so should affect the three dependent variables. Project duration was included as a control also by previous SDO research (Ethiraj et al. 2005; Gopal et al. 2003).

Software Complexity

The model also controls for the expected effects of software complexity. More complex software takes longer to develop (Beizer 1990), and hence should carry a higher price tag. Additionally, more complex software typically requires more quality control (Beizer 1990), and hence may result in a preference by the client to add more contractual penalties and to have the software developed with a TM contract because of the closer monitoring this might provide. Indeed, larger and

more complex SDO projects are often outsourced on a TM basis (Gopal et al. 2003). Likewise, the construction industry prefers TM in complex projects (Bajari and Tadelis 2001) and the offshore oil drilling industry prefers FP in low complexity projects (Corts and Singh 2004). Similar controls, such as effort and requirements uncertainty, have been applied in other SDO research (Gopal et al. 2003). We additionally controlled for contract size in terms of pages and the extent of external documents as indirect measures of project size and complexity.

Vendor Characteristics Controls

Vendor characteristics, its size as measured by the number of employees and its degree of internationality (i.e., being international), are also added as controls. As a matter of reducing the risk involved with ever-present employee turnover in the high technology industries, clients should rationally prefer to give TM contracts to larger vendors because employee turnover will affect smaller companies more than larger ones in their ability to replace employees with specific skills. Support for this can be found in the oil drilling industry where clients prefer to contract on an FP basis with their smaller vendors (Corts and Singh 2004). Vendor internationality is included because it is harder to control overseas vendors and because cultural misunderstandings may presumably make it harder to explain project details to them. Moreover, there are different orientations toward business partnerships and long-term orientation across cultures (Fukuyama 1995; Hofstede 2004), hence another good reason to avoid outsourcing long and complex projects abroad. The research model is shown in Figure 2.

Research Methodology

The Bank and How it Outsources

The bank is among the largest in Europe. Its size and methods of managing SDO are typical of other banks and large institutions. It manages an IS department with nearly 3,000 permanent employees and around 2,000 contractors. The department's responsibility is the acquisition, development, maintenance, and operation of the bank's information systems. The bank distinguishes between four different outsourcing contractual arrangements: IT Consulting, Body Leasing, Maintenance, and SDO. In the past, 60 percent of the contracts were Body Leasing, 15 percent were IT Consulting, 15 percent were Maintenance, and only about 10 percent were SDO. IT Consulting contracts are mostly for independent developers, specialists, and consultants who are hired to perform specific small-scale work oftentimes on an ongoing basis. Body Leasing contracts are typically for programmers or other experts who are hired from larger suppliers for a limited period of time. These contract types typically deal with work done within the bank's premises and are managed by the bank. SDO contracts, the topic of this study, in contrast, are projects that are developed and managed by external vendors typically outside the bank's premises. These are priced as FP, TM, or a hybrid TM with a fixed price cap. The bank receives full ownership of outsourced code and documentation so it can maintain the code internally. When system maintenance is outsourced, a separate maintenance contract is signed. Other maintenance contracts are for standard software, packages, and hardware. The SDO contracts we studied totaled over half a billion U.S. dollars and were signed between January 2000 and April 2003.

The bank maintained guidelines and checklists to assist its project managers in picking the right contract type and had standard versions for both TM and FP contracts. The contract management process included six steps: evaluation, product/supplier selection, contract negotiation, closure by fulfillment, extension, and cancellation. Various teams were involved throughout the process, especially the bank's legal team, IT management, and project managers. The bank preferred to use its standard contracts as much as possible. The standard contract for SDO is about 10 pages long, not including appendices for scope and schedule. It is similar to forms suggested in the literature, for example by Pearson (1984) and Kutten (1998). These standard contracts define the purpose and content of the contract, the supplier's obligations, scope changes, acceptance testing, warranty and guarantee, expenses and payment terms, training, ownership of intellectual property, breach penalty, liability, confidentiality and data protection, and prohibition against enticement or employment of staff. The appendices describe the scope of the project, payment details, obligations, and project organization. The standard contract further defined actions to be taken when the schedule is missed—either setting a new schedule or the option to cancel the contract. The bank maintained guidelines that defined the boundaries of possible changes and explains their impacts. All the software, whether developed in FP or TM contracts, went through separate elaborate testing because of the high impact glitches could have.

Data Description

The unit of analysis, as in previous research (Corts and Singh 2004; Ethiraj et al. 2005; Gopal et al. 2003; Gulati 1995; Kalnins and Mayer 2004), was the actual SDO contract. The

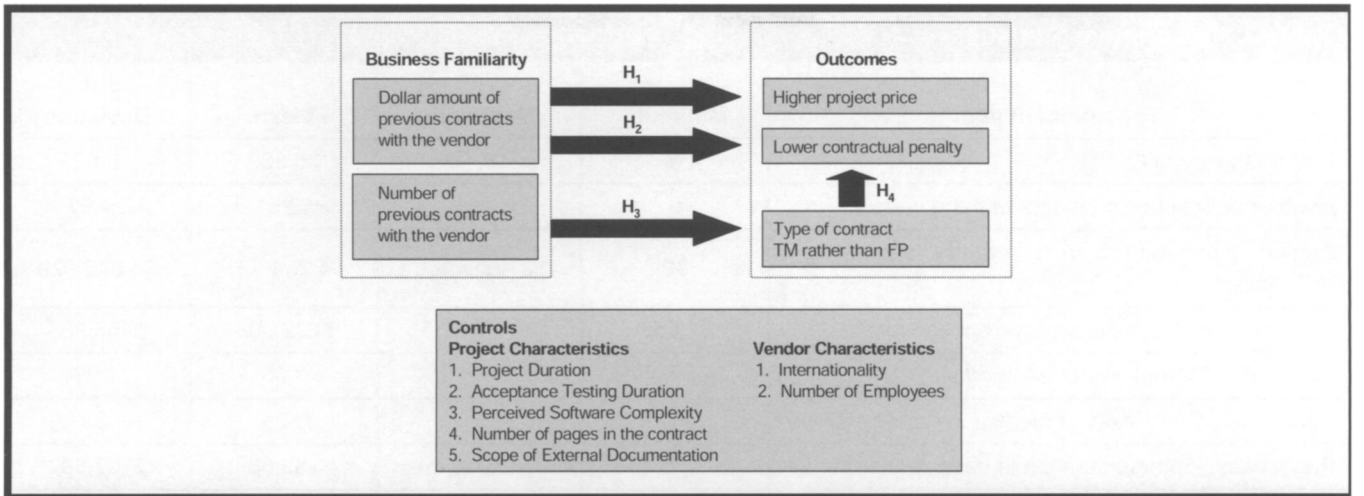

Figure 2. Research Model

price of each contract and its penalty in U.S. dollars were copied as explicitly stated in the contract. Project duration was determined as the time interval in days between when the project was set to start and its expected delivery time. All these measures were explicit, objective, legally binding, and straightforward details in the standardized software outsourcing contracts of the bank. We registered whether the contract was FP or TM as a value of 1 or 2, respectively. This distinction was explicitly stated in the contracts. Also coded were testing duration and the number of external documents referenced. The coding was done by copying the contract details exactly as they were, except for the one subjective scale of perceived software complexity. In all, the bank signed a total of 424 outsourcing contracts during this period: 292 FP, 96 TM, and 36 hybrid projects. We had access to contract type, price, and vendor on all these contracts, but, due to limitations the bank placed, we had access to complete data on only 274 of these contracts. Among these 274 there were 181 FP, 57 TM, and 36 hybrid projects. The hybrid projects were excluded from the analyses because they were applied only in special circumstances which precluded our ability to generalize based on them.

The bank's business familiarity with each vendor was calculated as (1) the number of previous contracts the vendor had with the bank, and as (2) the total U.S. dollar value of these previous contracts. We had access to both these related, but distinct, measures of business familiarity. Measuring trust in contracts this way is common (Corts and Singh 2004; Gulati 1995) because trust cannot be measured directly in evaluating past contracts. To measure trust directly in our study would have required asking the bank managers to evaluate each vendor, an unlikely event, and to estimate this as it was when the contract was signed. The business familiarity measures deal exclusively with contracts previous to the current one and occurring between January 2000 and April 2003. We were not granted access to projects before January 1, 2000, and so these measures should be regarded as lower bound estimates. To partly account for this unavoidable bias when calculating business familiarity, these two measures were calculated based on all the 424 SDO contracts during this 40 month period to provide a more accurate measure. Mode project duration was 115 days. With such a small mode, the chance of misclassifying a vendor as new or grossly underestimating the bank's business familiarity with it are small, especially considering that the most recent history is what really counts.

After collecting the contract data we contacted each of these vendors and established for each one the approximate number of employees at the time the data were collected and whether the vendor at the time operated only nationally, nationally with international ties, or internationally. National vendors are located in the same country as the bank headquarters and have no business abroad. There were 75 national vendors. National with international ties are those vendors whose headquarters are located in the same country as the bank headquarters but also have business abroad. There were 109 such vendors. International vendors are those whose headquarters are abroad. There were 54 such vendors.

Perceived software complexity was the only subjective measure. It was assessed by one of the authors who read the technical details of each contract. Perceived software complexity was included in the model because it highlights the fact that the hypotheses are supported even when including this known

Table 1: Descriptive Statistics

Measurement Item	Minimum	Maximum	Mean	Std. Deviation
Contractual penalty	0	279,166	25,502	36,823
Number of previous contracts with the vendor	0	96	18.61	24.97
Amount of previous contracts with the vendor in U.S. dollars	$0	$21,037,460	$4,254,321	$5,845,929
Total price in U.S. dollars of current contract	$2,083	$2,791,667	$272,696	$406,203
Subjective: Complexity of the module	1	5	3.00	1.08
Subjective: Complexity of module interconnectivity	1	5	3.24	.92
Project development duration in days	15	540	153.68	107.59
Vendor size in number of employees	1	300,000	5,915	37,965
Duration of the testing period in days	0	180	29.50	40.83
Number of external documents referenced	0	5	2.68	.80

antecedent and so including it provides a more complete picture. Removing this construct does not change the pattern of significant paths in the model. Perceived software complexity was measured as the average of perceived complexity of the outsourced software module itself (scale in Appendix A) and the complexity of its interconnectivity with other modules as revealed in the software requirements in the contract (scale in Appendix B). The items were measured on a five-point scale. Descriptive statistics are shown in Table 1.

Data Analysis

Hypothesis H1 was tested with a multiple linear regression with price as the dependent variable and the two measures of business familiarity, the contract type, and all the controls as the independent variables.[7] This regression is shown in the first row of Table 2. All the multicolinearity, autocorrelation, and heteroscedasticity statistics were within the accepted thresholds (Neter et al. 1990). There was no significant endogeneity. In Table 2 the T value of each coefficient and its p value are in parentheses. Hypotheses H2 and H4 were analyzed in a multiple linear regression with contractual penalty as the dependent variable and with contract type and business familiarity as the independent variables together with the controls. This regression is shown in the second row of Table 2. Business familiarity did not directly affect price or contractual penalty, not supporting H1 and H2, respectively. Contractual penalty decreased when the project was TM, supporting H4. Price was determined by project duration, perceived software complexity, and pages in the contract.

Hypothesis H3 was tested with a logistic regression with contract type being FP or TM as the dependent variable and with the two measures of business familiarity and all the controls as the independent variables. The results are shown in Table 3. The Wald value of each coefficient and its p value are in parentheses. Business familiarity significantly increased TM contracting, supporting H3. The analysis was repeated with a linear regression to enable a standardized calculation of multicolinearity, autocorrelation, and heteroscedasticity. TM contracts were determined by business familiarity, duration, perceived complexity, and acceptance-testing duration. In other words, while business familiarity did not directly result in lower penalties in the contract, it did result in a revealed preference to contract on a TM basis which itself was associated with lower penalties.

In all the analyses we included both the number of previous contracts and their dollar amount as measures of business familiarity. Consistently, the number of previous contracts

[7]To avoid the autocolinearity between the number of previous contracts and their dollar amount, all the analyses took the log of dollar amount. This is a standard way to reduced the skewness of the data (Neter et al. 1990) and was applied also by previous research with regard to the somewhat analogous variable vendor profit by Ethiraj et al. (2005). In the analyses we also took the log of project duration, as done also by Ethiraj et al., to account for the skewness of the data based on the recommended procedure in linear regression (Neter et al. 1990) and to allow for comparison with previous research.

Table 2. Multivariate Linear Regression Path Coefficients, Hypotheses H1, H2, and H4

		Hypotheses			Controls						
From / To	R^2	Business Familiarity: Log (Dollar Amount of Previous Contracts)	Business Familiarity: Number of Previous Contracts	Contract is TM rather than FP	Log (Project Duration)	Acceptance Testing Duration	Perceived Software Complexity	Pages in the Contract	External Documents	Vendor Size	Internationality of the Vendor
Log (price)	.62	H1: .06 (t - 1.23, p = .22)	H1: –.04 (t = –.76, p = .43)	–.02 (t = –.33, p = .75)	.63*** (t = 11.58, p < .001)	–.02 (t = –.40, p = .72)	.23*** (t = 4.28, p < .001)	.14** (t = 2.85, p = .005)	.09 (t= 1.85, p = .07)	.01 (t = .12, p = .90)	.05 (t = 1.04, p = .30)
Contractual Penalty	.29	H2: –.07 (t = –.83, p = .41)	H2: –.09 (t = –1.12, p = .26	H4: –.21*** (t = –2.65, p = .009)	.06 (t = .76, p = .45)	.26*** (t = 3.59, p < .001)	.22** (t = 2.63, p = .01)	.04 (t = .51, p = .61)	.06 (t = .76, p = .45	–.01 (t = –.10, p = .93)	–.14 (t = –1.79, p = .08

Table 3. Binary Logistic Regression Path Coefficients, Hypothesis H3

		Hypotheses		Controls						
From / To	Nagelkerke R^2 / % classified correctly	Business Familiarity: Log (Dollar Amount of Previous Contracts)	Business Familiarity: Number of Previous Contracts	Log (Project Duration)	Acceptance Testing Duration	Perceived Software Complexity	Pages in the Contract	External Documents	Vendor Size	Internationality of the Vendor
Contract is TM rather than FP	.42 81%	H3: 1.59* (W = 5.50, p = .02)	H3: –.02 (W = 1.43, p = .23)	1.33* (W = 3.48, p = .06)	–.03** (W = 7.70, p = .006)	.61 (W = 3.50, p = .06)	–.01 (W = .28, p = .60)	–.29 (W = 3.15, p = .08)	.00 (W = 2.24, p = .14)	.48 (W = .78, p = .38)

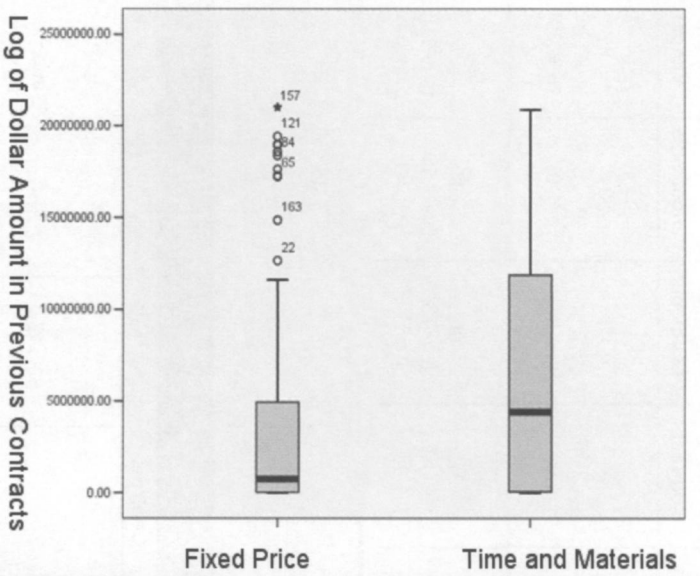

Figure 3. Contracting Governance by Business Familiarity

was insignificant, although it did become significant in predicting the type of contract when the dollar amount was excluded from the regressions. In other words, although the two measures are correlated ($r = .74$, p-value $< .001$), business familiarity is better measured as previous contract dollar amount.

The effect of business familiarity on TM preference, H3, is evident when shown graphically in Figure 3. Not only is business familiarity, measured as the dollar amount of previous contracts, significantly larger in TM projects (t-value = 2.981), but FP projects are mostly limited to the lower region of the business familiarity measure.

Discussion

Summary of Results

There are many risks in SDO, which is why correctly managing its contractual governance and control mechanisms are key to its success (Koh et al. 2004; Saunders et al. 1997; Wilcocks and Kern 1998). The underlying proposition advanced in this study is that because business familiarity reduces SDO risks it should affect contractual governance, price, and penalty. Based on the SDO contracts of a large European bank and independently collected vendor data, this study shows that business familiarity does affect contractual governance but not price and penalty. This means that the premium vendors can expect on business familiarity in SDO is not in higher prices, but in client preference to sign its TM contracts with them. These results are also informative in suggesting why despite the evolving nature of software requirements, and hence the advantage of TM, the tendency among clients is to prefer FP, as it is in other industries (Bajari and Tadelis 2001; Crocker and Reynolds 1993; Rogerson 1994): 55 percent in Lichtenstein and McDonnell (2003), 61 percent in Ethiraj et al. (2005), and 58 percent in Banerjee and Duflo (2000).

Limitations

The results of the study raise questions that require further investigation. The study examined the contractual records of a leading international bank. Hence, generalizing to other industries and clients requires additional research, especially as the trust literature (Luhmann 1979) has suggested, albeit never actually checked, that power may influence trust relationships. Relative power also determines principal–agent relationships according to contractual literature (Chiappori and Salanie 2002). The research model should also be examined in other national cultures to increase its generality because the way SDO contracts are outsourced may be culture dependent (Barthelemy and Geyer 2001). Examining vendors

who are not retained could also add interesting aspects, as could hybrid contracts.

Another topic which requires additional research is the effect of company policy. Companies typically have guidelines when signing large contracts. The bank was no exception. Its guidelines regarding SDO were based on its extensive experience over the years and its codified best practices. As far as we could determine, these guidelines were observed in the contracts we studied. The results of this study should be interpreted, therefore, as a combination of the managerial skills of the bank's managers, who were ultimately responsible for the success of the project, and these policies. Related to this, in many cases vendors also influence the contract type. Examining this aspect was not possible because the bank did not provide access to this information. How vendor and client preferences interact to determine contract type, therefore, requires additional research. Finally, although trust in contract research is typically measured through its business familiarity proxy (e.g., Corts and Singh 2004; Gulati 1995), the reader should remember that trust was only measured indirectly.

Contributions

The importance of previous business relationships in determining current ones, in general, has been recognized in the past (Williamson 1979), but the jury is still out on the exact effect business familiarity has (Kalnins and Mayer 2004). Adding to the need to better understand the effect of business familiarity in SDO, no previous research has examined business familiarity in the context of price and penalties. Hence, a key contribution of this study is in being the first to examine how business familiarity is related to the monetary aspects of SDO contracts. These data are especially interesting considering the significant relationship with TM preference and the surprisingly insignificant relationships with price and penalty. Additionally, expanding previous theory in this context, this study incorporates elements of trust theory into the principal–agent relationship in SDO and allows a more refined look at business familiarity than previous research has done.

These contributions are highlighted by the unique data which support them. Much previous research about the FP or TM decision (e.g., Currie 1996; Gopal et al. 2003; Lacity et al. 1996) was based on case studies, interviews, and *post hoc* surveys which may be subject to recall bias (Gopal et al. 2003). While those studies presented valuable insights, they were based on subjective, mostly *post hoc*, one-time perceptions and beliefs. Several empirical studies did examine actual contracts but these either did so without a perspective of previous business relationships and with only very small projects (e.g., Snir and Hitt 2003), studied contracts from a vendor perspective (e.g., Ethiraj et al. 2005; Gopal et al. 2003; Kalnins and Mayer 2004), or did not deal with SDO (e.g., Corts and Singh 2004). The objective contractual measures applied in this study also have the distinct advantage of containing no common method bias and no after-the-effect rationalization, and so provide a better estimate of path coefficients and explained variance.[8]

Contribution in View of IT Research

Past research has identified several practices that may increase the successful contracting and management of outsourcing projects. These include relationship management (Koh et al. 2004; Lee and Kim 1999; Nam et al. 1996; Wilcocks and Kern 1998), restricting the extent of IT outsourcing (Lacity and Willcocks 1998), creating a trusting relationship with the client (Banerjee and Duflo 2000; Gefen 2004; Lee and Kim 1999), partnership and providing good service (Grover et al. 1996), a general fit between the ongoing outsourcing strategies and the outsourcing success criteria (Lee et al. 2004), and applying appropriate project controls (Gopal et al. 2002; Lee et al. 2004). This study adds business familiarity as another central aspect with its effect on contractual governance.

The study also highlights the need to differentiate between FP and TM governance, often pooled together in previous research (e.g., Ang and Straub 1998; Banerjee and Duflo 2000), and to consider vendor and project characteristics to better understand outsourcing projects, again a topic mostly overlooked by previous research.

To some extent the importance of business familiarity supports the observation by Kalnins and Mayer (2004) that key determinants of contractual governance are *ex ante* uncertainty about costs and specifications, costs of measuring quality *ex post*, and prior relationships.[9] This study extends

[8]Common method bias is known to inflate coefficients and explained variance (Podsakoff et al. 2003); after-the-effect rationalization and self-justifications are a known source of bias (Cook and Campbell 1979); Cote and Buckley (1988) estimated that between 15 percent and 40 percent, and on average 26 percent, of the variance in subjective measures is due to systematic measurement error.

[9]Kalnins and Mayer (2004) examined a supplier of IT services and computer hardware, where software development was included only in some of the contracts and measured only as a dichotomous, included or not, variable. This is not surprising. The data Kalnins and Mayer analyzed were collected between 1986 and 1998, mostly before SDO became popular. This places the current study in a very different ballpark. In hardware service outsourcing,

these conclusions to a purely software environment and qualifies this importance to contractual governance alone. The role of business familiarity also provides somewhat of an indirect support to the theoretical two-stage method proposed by Snir and Hitt (2004),[10] although our empirical results support incentives through future TM contracts rather than through increased pay in the second phase. The data also confirm the central role of project duration in determining the two central dependent variables, price and the decision to go TM, as well as the somewhat lesser weight of various measures of complexity. This is not surprising as duration and complexity are key contributors to software development cost (Beizer 1990).

Business Familiarity and Price

To understand why the hypothesized effects of business familiarity on price and on penalty were not supported, we presented the results *post hoc* to two software managers at the bank. Apparently, in purely IT related decisions, such as contracting FP or TM, the bank relies on its IT managers who are given full authority to decide, and therefore H3, dealing with IT management, was supported. In contrast, in regard to the monetary aspects of the contract, its price and penalties, Accounts Payable managers are also actively involved. Accounts Payable managers are less concerned with IT management risks and more concerned with money management. When it concerns money management, any type of implied lack of impartiality, including preferring one vendor to another or paying one vendor on a higher scale than another, contradicts regulations and may hint to illegal favoritism. Therefore, the hypothesized effects of business familiarity on price and on penalty, H1 and H2, were not supported, although from an IT manager's perspective, discussed in H1 and H2, these IT managers may prefer certain vendors on account of there being less risk with high business familiarity. This insignificant relationship between business familiarity and price is noteworthy because this is the only study to date on SDO with contract price as its dependent variable.

New Perspectives on Agency Theory in Contracts

The study supports agency theory's prediction that business familiarity should affect the nature of agency relationships (Eisenhardt 1989), but adds a new trust perspective into this reasoning. This trust perspective is important because an underlying assumption in agency theory is that the agent and the principal are in competition with each other, and hence the principal needs to address adverse selection and moral hazard risks (Bolton and Dewatripomt 2005). Because of this assumed competition and the uncertainty it represents, past research identified coordination costs (Gulati and Singh 1998) and appropriation concerns as key determinants of contractual governance in alliances (Pisano 1989), and suggested that companies address these *risks* through their choice of contractual governance (Pisano 1989). As far as the principal is concerned in contract theory, the agent should not be trusted outright because there is always an overwhelming threat of agent opportunism, and so risk controls should be added into the contract (Bolton and Dewatripomt 2005). These controls allow both parties to learn about each other and to protect themselves (Coase 1937). Trust adds an interesting twist here. Trust means the principal and the agent can rely on each other, and so do not need to rely as much on extensive controls in the contract to guarantee their respective outcomes (Fukuyama 1995; Kumar et al. 1995). The significant effect of business familiarity, and the trust it implies, on the preference to contract on a TM basis supports this perspective.

In this regard, the study addresses a central topic of trust theory, namely what the premium on trust is. Past research on one-time product delivery contracts has shown that the premium on trust, at least in online marketplaces, is measurable in terms of an increased price received by more trusted sellers (Ba and Pavlou 2002) and in increased behavioral intentions to transact with these sellers which leads to increased actual transactions with them (Pavlou and Gefen 2004). Trust also results directly in an improved assessment of the software implementation provider (Gefen 2004), and of outsourcing relationships (Lee and Kim 1999). In contributing to this line of research, the current study shows that the premium on business familiarity may actually be mostly a subtle one. Trust, as implied through business familiarity (Gulati 1995), does not directly affect the primary outcome, price. Rather, it changes how the relationship is managed. Increased trust results in reduced reliance on contractual safeguards, such as FP contracting. This conclusion presents

---

the deliverables, such as disk space and CPU speed, can be quantified and their quality verified, and so the risk can be calculated explicitly in the contract and its warranty. In SDO, in contrast, because of the unique nature of software, there is always an aspect of ingrained uncertainty in both the requirements and in the development process (Beizer 1990), and it is much harder to verify that the required service was delivered according to the requested quality.

[10]In their method, Snir and Hitt (2004) propose that in the first stage, the pilot, the principal sets a sufficiently low compensation so the agent actually loses money. To make up for this loss the second stage compensates adequately, but only vendors who do well in the first stage are retained for this second phase. In effect this creates an incentive for only high-quality vendors to apply. Low-quality vendors, assuming rationality on their behalf, will be discouraged from applying at all because they know they will be weeded out after the pilot stage and so not receive compensation in the second stage to offset their loss in the pilot.

trust in a more calculative manner. Trust means believing one knows what to expect of the trusted party (Fukuyama 1995; Gefen et al. 2003; Gulati 1995; Luhmann 1979), and this makes the relationship with these trusted companies more worthwhile (Lacity et al. 1996). The premium then is in how the relationship is managed.

Perspectives on Incomplete Contracts

The FP or TM decision can be viewed also through the lens of incomplete contract theory (Bolton and Dewatripomt 2005; Tirole 1999). Although, according to incomplete contract theory, all contracts are essentially incomplete (Tirole 1999), there is a distinction in practice between relatively complete and incomplete contracts. Relatively complete contracts are those where there are no clauses dealing with unforeseeable contingencies, although there are still adverse selection and moral hazard risks (Bolton and Dewatripomt 2005; Tirole 1999). These relatively complete contracts, resembling FP contracts, can be described in sufficient detail *ex ante* and are observable albeit not necessarily verifiable *ex post*. "Observable albeit not necessarily verifiable" means both parties know when the project is done correctly although it is not necessarily verifiable by a court of law (Tirole 1999). Even in theory, there is no need to renegotiate such contracts if the product being developed belongs to the principal at the end of the process. Incomplete contracts, in contrast, are more realistic in some settings as they include clauses about unforeseeable contingencies (Tirole 1999). Incomplete contracts, resembling TM contracts, are not sufficiently describable *ex ante* although they too are observable *ex post*. Incomplete contracts are especially pertinent to some cases of SDO because of the constantly evolving nature of software requirements, and hence the difficulty in determining *ex ante* the complete picture of what the software will do and how it will need to be rewritten to adjust to new needs. The mechanisms suggested in theory for making incomplete contracts optimal involve *ex post* renegotiation to undo suboptimal outcomes. In reality, however, renegotiation is so complex that it is rarely applied (Bolton and Dewatripomt 2005).

An alternative method to deal with these inefficiencies is to rely on trustworthy partners. Trustworthy partners generally look at the long term, do not mislead, refrain from opportunistic behavior, and generally care about the success of the partnership. Putting things into perspective, agency and incomplete contract theories discuss how to control risks that result from the possible opportunistic behavior of the other party. In contrast, creating a trust-based relationship is about creating a relationship where the parties *a priori* do not wish to be opportunistic with each other (Kumar et al. 1995), and this is what the bank unwittingly but logically did. The bank avoided the problems of *ex post* renegotiations through *ex ante* contracting TM mostly with vendors it had extensive business familiarity with (i.e., vendors it could trust). This strategy had the additional benefit of reducing information asymmetry in the more risky TM projects. In other words, the answer to *ex post* opportunism (Williamson 1979) as applied in this case was to choose trustworthy vendors rather than renegotiate.

Business Familiarity

The study measured business familiarity as both the number and the dollar amount of previous contracts. Previous research only included either the number of previous contracts (Corts and Singh 2004; Gopal et al. 2003; Gulati 1995), or their dollar value (Kalnins and Mayer 2004),[11] but not both. In this regard the data show that it is the actual amount paid, roughly corresponding in the case of SDO to hours spent in previous contracts, rather than just how many contracts there were, which determines contract type. Conceptually, these measures reflect, albeit imperfectly, the difference between whether the vendor was trusted and the degree to which it was trusted. Whether the vendor was trusted was measured as the number of previous contracts; how much it was trusted was measured in terms of dollar amount. In understanding the relationship between these two measures it should be remembered that trust is not a yes-or-no variable. Trust is, both in theory (Mayer et al. 1995) and in practice in IT services relationships (Gefen 2004), a matter of degree, and accordingly can grow and shrink over time. As a trusting relationship evolves and the trusted party proves itself repeatedly, the degree of trust increases and with it the degree to which the trusting party is willing to rely on the trusted party (Gefen 2004). In the case of the bank, while vendors who did not prove trustworthy were weeded out, those who remained could improve the degree of trust the bank had in them. Accordingly, although the bank could trust two vendors, granting both contracts, it could still trust one more than the other, being more willing to rely on this vendor by granting it larger projects. The bank may have felt more comfortable in trusting a given vendor to the extent of $10K but not $10M. And so, although we did not measure trust directly, arguably, the dollar amount is a reasonable reflection of how much the vendor was actually trusted.

[11]Although Ethiraj et al. (2005) did examine project contribution (i.e., revenue minus costs), as assessed at project completion from the vendor perspective, they did not examine price from the client perspective.

Implications for Industry

There is a growing trend in the banking industry to outsource software development. Although we do not know from the data whether a specific project was successful, the study provides some guidelines. IT managers at clients can learn from the successful management of SDO at the bank to sign TM projects primarily with vendors with whom they are more familiar. IT managers at vendors might be able, through this glimpse into the proceedings of a large client, to understand better the considerations of the companies with which they are bidding to contract. Gaining insight into such guidelines is hard to come by, as many commercial organizations are loath to make their actual contracts available to research, let alone the financial side of these contracts. The data provide a unique glimpse into this typically secret aspect and imply the need for vendors to manage SDO relationships based on trust because these are often managed with less reliance on expensive FP controls. Moreover, as increased business familiarity did not gain vendors a better price but only reduced contractual controls, it makes sense for vendors to invest in building a trusting relationship with the client rather than to compete primarily on price. New vendors could benefit from taking such a long-term approach to gain a foot in the door and eventual access to the less risky TM projects. Although vendors can do little to increase business familiarity in the short term, they can build trust through such long-term orientation.

The study also provides clients and vendors with a practical method to address unforeseen contingencies in SDO, such as incomplete specifications. Contract theory suggests that one way of addressing these is through *ex post* renegotiations (Bolton and Dewatripomt 2005), an option often impracticable in SDO. As this study shows, the industry prefers a simpler solution: rely more on higher business familiarity vendors when the project requires addressing such contingencies through TM contracting. A side benefit of this alternative reliance on trust is to encourage vendors to refrain from opportunistic behavior by rewarding vendors who had proven themselves (i.e., have remained long enough to achieve a high degree of business familiarity) with the more advantageous TM contracts. This resembles the observations by Rao and Bergen (1992) about how consumers who purchase from vendors on a frequent basis can assure vendor product quality through granting these vendors price premiums. A somewhat parallel incentive applies in SDO. Providing good service over time, as reflected through high business familiarity, awards such vendors with the easier to manage TM projects. Such a premium may also induce FP vendors to invest in the creation of a long-term relationship with the client, an incentive that counters the otherwise rational tendency by risk-averse vendors (who only earn a fixed fee) not to invest in the relationship (Bolton and Dewatripomt 2005). In this regard, clients may find it advisable to retain in addition to records of contract price also records of their vendors' technological capabilities, performance, and knowledge with specific departments and business processes. The results also suggest that one way to address cases where low trust causes clients to prefer FP contracts (Lichtenstein 2004), even when TM might be more appropriate, is to build trust gradually over time through many contracts.

Conclusion

Business familiarity, reflecting increased knowledge and trust, adds new theoretical perspectives to agency theory and to incomplete contract theory, and, in doing so, presents the IT side of SDO management in a new light. In SDO contracts, clients and vendors address their respective risks through several contractual controls. Primary among these controls are price, penalty, and contractual governance. Business familiarity, by adding a previous knowledge and future trust aspect into this client–vendor relationship, does affect some aspects, but not others, of this principal–agent relationship. In the SDO contracts studied, business familiarity affected contractual governance, with higher business familiarity vendors being awarded more TM contracts, but not the price and penalty aspects of the contract. In other words, business familiarity affected how the contractual relationship was to be managed but not its monetary side. Future research should look into this aspect of contracting in more detail. The prospect of integrating a trust management aspect into agency theory and into incomplete contract theory as an alternative to their currently prevalent risk control views is especially promising.

References

Ang, S., and Straub, D. W. 1998. "Production and Transaction Economics in Information Systems Outsourcing: A Study of the U.S. Banking Industry," *MIS Quarterly* (22:4), pp. 535-552.

Artunian, J. 2006. "The Seven Deadly Sins of Outsourcing: These Are the Transgressions That Can Doom You to Outsourcing Hell. Here's How to Avoid Them," *Computerworld*, May 8 (http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=111167&pageN).

Ba, S., and Pavlou, P. A. 2002. "Evidence of the Effect of Trust Building Technology in Electronic Markets: Price Premiums and Buyer Behavior," *MIS Quarterly* (26:3), pp. 243-268.

Bajari, P., and Tadelis, S. 2001. "Incentives Versus Transaction Costs: A Theory of Procurement Contracts," *RAND Journal of Economics* (32:3), pp. 387-407.

Bakos, J. Y., and Brynjolfsson, E. 1993. "Information Technology, Incentives and the Optimal Number of Suppliers," *Journal of Management Information Systems* (10:2), pp. 37-53.

Banerjee, A., and Duflo, E. 2000. "Reputation Effects and the Limits of Contracting: A Study of the Indian Software Industry," *Quarterly Journal of Economics* (115:3), pp. 989-1017.

Barki, H., Rivard, S., and Talbot, J. 1993. "Toward an Assessment of Software Development Risk," *Journal of Management Information Systems* (10:2), pp. 203-225.

Barthelemy, J., and Geyer, D. 2001. "IT Outsourcing: Evidence from France and Germany," *European Management Journal* (19:2), pp. 195-202.

Beizer, B. 1990. *Software Systems Techniques*, New York: Van Nostrand Reinhold Company.

Benaroch, M., Lichtenstein, Y., and Robinson, K. 2006. "Real Options in Information Technology Risk Management: An Empirical Validation of Risk-Option Relationships," *MIS Quarterly* (30:4), pp. 827-864.

Bolton, P., and Dewatripomt, M. 2005. *Contract Theory*, Cambridge, MA: The MIT Press.

Chiappori, P. A., and Salanie, B. 2002. "Testing Contract Theory: A Survey of Some Recent Work," CESifo Working Paper Series, No. 738, CESifo GmbH, Munich.

Coase, R. 1937. "The Nature of the Firm," *Economica* (4:16), pp. 386-405.

Cook, T. D., and Campbell, D. T. 1979. *Quasi Experimentation: Design and Analytical Issues for Field Settings*, Chicago: Rand McNally.

Corts, K. S., and Singh, J. 2004. "The Effect of Repeated Interaction on Contract Choice: Evidence From Offshore Drilling," *Journal of Law Economics & Organization* (20:1), pp. 230-260.

Cote, J. A., and Buckley, R. 1988. "Measurement Error and Theory Testing in Consumer Research: An Illustration of the Importance of Construct Validation," *Journal of Consumer Research* (14), pp. 579-582.

Crocker, K. J., and Reynolds, R. J. 1993. "The Efficiency of Incomplete Contracts: An Empirical Analysis of Air Force Engine Procurement," *RAND Journal of Economics* (24), pp. 126-146.

Currie, W. L. 1996. "Outsourcing in the Private and Public sectors: An Unpredictable IT Strategy," *European Journal of Information Systems* (4:4), pp. 226-236.

Economist. 2005. "Time to Bring it Back Home? Outsourcing Has its Limits," *The Economist*, March 3 (http://www.economist.com/research/articlesBySubject/).

Eisenhardt, K. M. 1989. "Agency Theory: An Assessment and Review," *Academy of Management Review* (14:1), pp. 57-74.

Ethiraj, S. K., Kale, P., Krishnan, M. S., and Singh, J. V. 2005. "Where Do Capabilities Come From and How do They Matter? A Study in the Software Services Industry," *Strategic Management Journal* (26:1), pp. 25-45.

Fukuyama, F. 1995. *Trust: The Social Virtues and the Creation of Prosperity*, New York: The Free Press.

Gefen, D. 2002. "Nurturing Clients' Trust to Encourage Engagement Success during the Customization of ERP Systems," *Omega* (30:4), pp. 287-299.

Gefen, D. 2004. "What Makes ERP Implementation Relationships Worthwhile: Linking Trust Mechanisms and ERP Usefulness," *Journal of Management Information Systems* (21:1), pp. 275-301.

Gefen, D., Karahanna, E., and Straub, D. W. 2003. "Trust and TAM in Online Shopping: An Integrated Model," *MIS Quarterly* (27:1), pp. 51-90.

Gopal, A., Mukhopadhyay, T., and Krishnan, M. S. 2002. "The Role of Software Processes and Communication in Offshore Software Development," *Communication of the ACM* (45:4), pp. 193-200.

Gopal, A., Sivaramakrishnan, K., Krishnan, M. S., and Mukhopadhyay, T. 2003. "Contracts in Offshore Software Development: An Empirical Analysis," *Management Science* (49:12), pp. 1671-1683.

Grossman, S. J., and Hart, O. D. 1986. "The Costs and Benefits of Ownership: A Theory of Vertical and Lateral Integration," *Journal of Political Economy* (94), pp. 691-719.

Grover, V., Cheon, M. J., and Teng, J. T. C. 1996. "The Effect of Service Quality and Partnership on the Outsourcing of Information Systems Functions," *Journal of Management Information Systems* (12:4), pp. 89-116.

Gulati, R. 1995. "Does Familiarity Breed Trust? The Implications of Repeated Ties for Contractual Choice in Alliances," *Academy of Management Journal* (38:1), pp. 85-112.

Gulati, R., and Singh, H. 1998. "The Architecture of Cooperation: Managing Coordination Costs and Appropriation Concerns in Strategic Alliances," *Administrative Science Quarterly* (43:4), pp. 781-814.

Hofstede, G. 2004. "Business Goals and Corporate Governance," *Asia Pacific Business Review* (10:3/4), pp. 292-301.

Hosmer, L. T. 1995. "Trust: The Connecting Link between Organizational Theory and Philosophical Ethics," *Academy of Management Review* (20:2), pp. 379-403.

Kalnins, A., and Mayer, K. J. 2004. "Relationships and Hybrid Contracts: An Analysis of Contract Choice in Information Technology," *Journal of Law Economics and Organization* (20:1), pp. 207-229.

Keil, M., Rai, A., Mann, J. E. C., and Zhang, G. P. 2003. "Why Software Projects Escalate: The Importance of Project Management Constructs," *IEEE Transactions on Engineering Management* (50:3), pp. 251-261.

Kern, T., and Willcocks, L. 2001. *The Relationship Advantage: Information Technologies, Sourcing, and Management*, Oxford, UK: Oxford University Press.

Koh, C., Ang, S., and Straub, D. W. 2004. "IT Outsourcing Success: A Psychological Contract Perspective," *Information Systems Research* (15:4), pp. 356-373.

Kumar, N., Scheer, L. K., and Steenkamp, J.-B. E. M. 1995. "The Effects of Supplier Fairness on Vulnerable Resellers," *Journal of Marketing Research* (17), February, pp. 54-65.

Kutten, L. J. 1998. *Computer Software, Protection/Liability/Law/Forms*, St. Paul, MN: West Group.

Lacity, M. C., and Willcocks, L. P. 1998. "An Empirical Investigation of Information Technology Sourcing Practices: Lessons from Experience," *MIS Quarterly* (22:3), pp. 363-408.

Lacity, M. C., Willcocks, L. P., and Feeny, D. F. 1996. "The Value of Selective Sourcing," *Sloan Management Review* (37:3), pp. 13-25.

Lee, J.-N., and Kim, Y.-G. 1999. "Effect of Partnership Quality on IS Outsourcing Success: Conceptual Framework and Empirical Validation," *Journal of Management Information Systems* (15:4), pp. 29-61.

Lee, J.-N., Miranda, S. M., and Kim, Y.-M. 2004. "IT Outsourcing Strategies: Universalistic, Contingency, and Configurational Explanations of Success," *Information Systems Research* (15:2), pp. 110-131.

Levina, N., and Ross, J. W. 2003. "From the Vendor's Perspective:Exploring the Value Proposition in Information Technology Outsourcing," *MIS Quarterly* (27:3), pp. 331-364.

Lichtenstein, Y. 2004. "Puzzles in Software Development Contracting," *Communication of the ACM* (47:2), pp. 61-65.

Lichtenstein, Y., and McDonnell, A. 2003. "Pricing Software Development Services," in *Proceedings of the 11th European Conference on Information Systems*, Naples, Italy, pp. 1-12.

Luhmann, N. 1979. *Trust and Power*, London: John Wiley and Sons.

Mayer, R. C., Davis, J. H., and Schoorman, F. D. 1995. "An Integrative Model of Organizational Trust," *Academy of Management Review* (20:3), pp. 709-734.

McDougall, P. 2006. "Dexterity Required," *InformationWeek*, June 19, pp. 34-39.

McKnight, D. H., Choudhury, V., and Kacmar, C. 2002. "Developing and Validating Trust Measures for E-Commerce: An Integrative Typology," *Information Systems Research* (13:3), pp. 334-359.

McKnight, D. H., Cummings, L. L., and Chervany, N. L. 1998. "Initial Trust Formation in New Organizational Relationships," *Academy of Management Review* (23:3), pp. 473-490.

Mishra, D. P., Heide, J. B., and Cort, S. G. 1998. "Information Asymmetry and Levels of Agency Relationships," *Journal of Marketing Research* (35), pp. 277-295.

Nam, K., Rajagopalan, S., Rao, H. R., and Chaudhury, A. 1996. "A Two-level Investigation of Information Systems Outsourcing," *Communication of the ACM* (39:7), pp. 36-44.

Neter, J., Wasserman, W., and Kutner, M. H. 1990. *Applied Linear Statistical Models: Regression, Analysis of Variance, and Experimental Design* (3rd ed.), Homewood, IL: Richard D. Irwin.

Pavlou, P., and Gefen, D. 2004. "Building Effective Online Marketplaces with Institution-Based Trust," *Information Systems Research* (15:1), pp. 37-59.

Pearson, H. E. 1984. *Computer Contracts: An International Guide to Agreements and Software Protection*, London: Computer Law Series, pp. 113-130.

Pisano, G. P. 1989. "Using Equity Participation to Support Exchange: Evidence from the Biotechnology Industry," *Journal of Law, Economics, and Organization* (5:1), pp. 109-126.

Podsakoff, P. M., MacKenzie, S. B., Lee, J.-Y., and Podsakoff, N. P. 2003. "Common Method Bias in Behavioral Research: A Critical Review of The Literature and Recommended Remedies," *Journal Of Applied Psychology* (88:5), pp. 879-903.

Rao, A. R., and Bergen, M. E. 1992. "Price Premium Variations as a Consequence of Buyers' Lack of Information," *Journal of Consumer Research* (19:3), pp. 412-423.

Rogerson, W. P. 1994. "Economic Incentives and the Defense Procurement Process," *Journal of Economic Perspectives* (8), pp. 65-90.

Salanie, B. 1997. *The Economics of Contracts: A Primer*, Cambridge, MA: MIT Press.

Saunders, C., Gebelt, M., and Hu, Q. 1997. "Achieving Success In Information Systems Outsourcing," *California Management Review* (39:2), pp. 63-79.

Simon, H. A. 1951. "A Formal Model of the Employment Relationship," *Econometrica* (19), pp. 293-305.

Snir, E. M., and Hitt, L. M. 2003. "Costly Bidding in Online Markets for IT Services," *Management Science* (49:11), p 1504.

Snir, E. M., and Hitt, L. M. 2004. "Vendor Screening in Information Technology Contracting With a Pilot Project," *Journal of Organizational Computing and Electronic Commerce* (14:1), pp. 61-88.

SPSS. 2004. *SPSS 13.0 for Windows*, Chicago: SPSS Inc.

Taylor, H. 2007. "Outsourced IT Projects from the Vendor Perspective: Different Goals, Different Risks," *Journal of Global Information Management* (15:2), pp. 1-27.

Tirole, J. 1999. "Incomplete Contracts: Where Do We Stand," *Econometrica* (67:4), pp. 741-781.

Walden, E. A. 2005. "Intellectual Property Rights and Cannibalization In Information Technology Outsourcing Contracts," *MIS Quarterly* (29:4), pp. 699-720.

Wang, E. T. G., Baron, T., and Seidmann, A. 1997. "Contracting Structures for Custom Software Development: The Impacts of Informational Rents and Uncertainty on Internal Development and Outsourcing," *Management Science* (43:12), pp. 1726-1744.

Wilcocks, L. P., and Kern, T. 1998. "IT Outsourcing as Strategic Partnering: The Case of The UK Inland Revenue," *European Journal of Information Systems* (18:1), pp. 29-45.

Williamson, O. E. 1979. "Transaction Cost Economics: The Governance of Contractual Relations," *Journal of Law, Economics, and Organization* (11), pp. 335-361.

Zucker, L. G. 1986. "Production of Trust: Institutional Sources of Economic Structure, 1840-1920," *Research in Organizational Behavior* (8), pp. 53-111.

About the Authors

David Gefen is an associate professor of MIS at Drexel University, Philadelphia, where he teaches strategic management of IT, database analysis and design, and VB.NET. He received his Ph.D. in CIS from Georgia State University and a Master of Sciences in MIS from Tel-Aviv University. His research focuses on trust and culture as they apply to the psychological and rational processes involved in ERP, CMC, and e-commerce implementation management, and to outsourcing. David's wide interests in IT adoption stem from his 12 years of experience in developing and managing large information systems. His research findings have been published in *MIS Quarterly, Information Systems Research, IEEE Transactions on Engineering Management, Journal of MIS, Journal of Strategic Information Systems, The DATA BASE for Advances in Information Systems, Omega: The International Journal of Management Science, Journal of the AIS, Communications of the AIS,* and elsewhere. David is an author of a textbook on VB.NET programming.

Simon Wyss is a graduate of the Michael Smurfit School of Business, University College Dublin, and of the Zürich University of Applied Sciences, School of Management. Simon has 12 years of IT industry experience, including 7 as a project manager in the eCommerce and banking industries. His research interests include IT project management in the context of outsourcing and offshoring. Simon currently manages the development of a large-scale business-critical system at an international organization.

Yossi Lichtenstein is a lecturer at the College of Management, Rishon Le-Zion, Israel, and at the Michael Smurfit School of Business, University College Dublin. Previously, he was a staff member at IBM Research and at Hewlett-Packard Laboratories. His current research interests include the economic aspects of information systems development, and knowledge management at research organizations. Yossi has published in *MIS Quarterly* and *Communications of the ACM* and holds several software patents.

Appendix A

Perceived Complexity of the Module Itself

Level	Description
1	Simple enhancements, adaptive or perfective changes to a known system. System is well established and uses common and known technology. Surprises are very unlikely and not anticipated.
2	Simple enhancements, adaptive or perfective changes to a new but productive system. Surprises are unlikely and not anticipated.
3	Changes, new functionalities, upgrades or mitigations to an established system. Surprises are possible and anticipated.
4	Development of new systems, interfaces or applications. Common, known technology is used. Surprises are likely and anticipated.
5	Development of new systems, interfaces, applications, prototypes or specification documents, feasibility studies. Technology is new. Surprises are very likely and anticipated.

Appendix B

Perceived Complexity of Module Interconnectivity with Other Modules

Level	Description
1	Contract covers work on a standalone system, single module, or interface only, and the approach and specifications look very straightforward. No impact on other systems is expected and no knowledge of other systems is required. **Example:** Restaging of laptops of all customer advisors, adapting branding and logo of two e-courses.
2	Standalone system, single module, or interface only, however approach and specification look complex. No impact on other systems is expected but thorough knowledge of that specific system is required. **Example:** Complex regulatory change to web-based training program, deployment of standard global payment system.
3	System, module, or interface is interconnected with other systems, data sources, or applications. However, connections are well defined, standard or simple. Basic knowledge of other system(s) is required. **Example:** New release for Secure Messaging Application, migration of existing application to new architecture and platform.
4	System, module, or interface is complex in itself and interconnected with other systems, data sources, or applications. Connections are well defined, standard or simple. Thorough knowledge of that specific system and interfaces is required. Basic knowledge of other system(s) is sufficient. **Example:** Enhancement of interface to interbank clearing, customization of core banking software.
5	System, module, or interface is complex in itself and interconnected with other systems, data sources, or applications. Thorough knowledge of the organization's environment and system architecture is required; impact on other systems is possible. **Example:** New release for proprietary back-office application that allows full customer-service to call-center agents, new proprietary portfolio management system, such as data warehouse with many different data sources.