

December 2005

Make-or-Buy in the Age of Open Source: A Transaction Cost Analysis

Martina Greiner
University of Georgia

Dale Goodhue
University of Georgia

Follow this and additional works at: <http://aisel.aisnet.org/icis2005>

Recommended Citation

Greiner, Martina and Goodhue, Dale, "Make-or-Buy in the Age of Open Source: A Transaction Cost Analysis" (2005). *ICIS 2005 Proceedings*. 55.
<http://aisel.aisnet.org/icis2005/55>

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2005 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

MAKE-OR-BUY IN THE AGE OF OPEN SOURCE: A TRANSACTION COST ANALYSIS

Martina E. Greiner and Dale L. Goodhue

Terry College of Business

University of Georgia

Athens, GA U.S.A.

mgreiner@uga.edu

dgoodhue@terry.uga.edu

Abstract

Since their appearance, open source communities have become increasingly successful and seemingly pose a real threat to traditional proprietary software vendors. Because open source software has now achieved both recognition and legitimacy, obtaining products and services from communities offers firms an additional alternative in traditional make-or-buy decisions. Transaction cost economics has been widely used as a theory to explain and predict the appropriate governance structure for make-or-buy decisions. By comparing transaction and production costs along a continuum of variable asset specificity, transaction cost economics helps to explain and predict the circumstances in which the open source community is the appropriate governance structure for specific make-or-buy decisions. Our work contributes to existing open source software research by shedding light on the factors that influence the appropriateness of this form of software production for firms. We are also contributing to the body of research surrounding transaction cost economics by incorporating into the original analysis the addition of “communities” as a unique governance alternative.

Keywords: Make-or-buy decisions, transaction cost analysis, open source software, open source communities

Introduction

Open source communities are “virtual” communities in which people come together, generally without regard to time and space limitations, and collaboratively develop publicly available software. Since their appearance, open source communities have become increasingly successful, with some now threatening the market dominance of major proprietary software vendors. Apache and Linux, for example, are two of the most popular and well known open source communities. Linux is now considered the most significant competitor to Microsoft’s flagship operating system, Windows. Apache now effectively owns the Web server market with a 68.83 percent share—more than three times that of its closest competitor, Microsoft’s Internet Information Server, with a 20.85 percent share.¹ Open source projects and their related communities come in many shapes and sizes, and the growth in the number of projects and participants has been astounding. Sourceforge is the largest repository of open source projects and lists almost 99,700 projects and has more than 1,067,000 registered users (May 2005, www.sourceforge.net).

Because open source software (OSS) has now achieved both recognition and—to varying degrees—legitimacy in the marketplace, obtaining products and services from communities offers firms an additional alternative in traditional make-or-buy decisions (Glaeser 2003). OSS is usually available with little or no upfront cost. However, firms do not always choose OSS alternatives, despite the seemingly significant cost savings. The controversial discussion about total cost of ownership shows that OSS may have more costs than initially considered (DiDio 2004; Giera 2004). In the following, we will focus on open source communities and investigate how the make-or-buy decisions of firms change when OSS is involved.

¹As of March 2005 (see http://news.netcraft.com/archives/web_server_survey.html; accessed May 2005).

Transaction cost economics (TCE) has been widely used as a theoretical basis to explain and predict the appropriate governance structure for make-or-buy decisions in organizations and, of most interest to this discussion, of information systems within organizations (Aubert et al. 1996, 2004; Ngwenyama and Bryson 1999; Riordan and Williamson 1985; Wang 2002). The basic tenet of TCE is that the firm will choose whichever governance structure minimizes the total cost associated with a transaction. It has been argued that virtual communities producing public goods are becoming a viable and competing form of organizational governance alongside hierarchies and markets (Benkler 2002; Butler 1982; Demil and Lecocq 2003; Glaeser 2003; Watson et al. 2005). Thus, if TCE is to be useful in this domain, it should be able to explain and predict the circumstances in which open source communities are appropriate governance structures for specific make-or-buy decisions.

Our study contributes both to the study of OSS as well as to the body of research on TCE. First, our work contributes to OSS research by shedding light on the factors that influence the appropriateness of OSS for firms. Second, we are contributing to TCE by incorporating “communities” as a unique governance choice into Williamson’s (1981a) original analysis. The paper is organized as follows. First, we explain the basic tenets of TCE. Second, we describe OSS and open source communities, explaining some of the attributes of OSS as a governance structure. Then, we explain how the governance structure changes as a function of asset specificity by comparing transaction and production costs for alternate governance structures. We will end the paper with a short evaluation of the limitations and advantages of using TCE for understanding software make-or-buy decisions.

Transaction Cost Economics

Transaction cost economics was initially introduced by Coase (1937) and Williamson (1975) to explain why some assets are produced internally within a firm with a hierarchical governance structure while others are produced and purchased on an external market. TCE focuses on the transaction as the basic unit of economic activity. A transaction is defined as “a good or service... transferred across a technologically separable interface” (Williamson 1981b, p. 1544). The basic tenet of the theory is that, when undertaking exchange transactions on the market, transaction costs in addition to physical production costs are involved (Coase 1937). Transaction costs include indirect costs, such as those related to searching, information gathering, negotiation, bargaining, and the eventual monitoring of a contractual relationship (Coase 1960). Production costs are either the costs of producing a good or the price a firm has to pay to procure it. For TCE, the basic criterion for organizing transactions is to economize on the sum of production expenses and transaction costs (Williamson 1981b). Although TCE mostly addresses transaction costs, production and transaction costs are not independent from each other and need to be examined simultaneously because they both contribute to the total cost of making or buying a product or service (Williamson 1981b). If the sum of the transaction and production cost of using a market is too high, other governance structures, such as hierarchical production in a firm, are more appropriate. Depending on the characteristics of a transaction, different governance types lead to lower or higher transaction and production costs when conducting the transaction. TCE originally included only two opposing governance structures, the market and the hierarchy, but has since been extended to include others. For example, hybrid structures include contractual arrangements such as exchange agreements, reciprocal trading, and franchising (Williamson 1991).

TCE is based on two basic behavioral assumptions which influence transaction costs. The first behavioral assumption is bounded rationality and the second is opportunism (Williamson 1981a). Bounded rationality means that people behave rationally but within the boundaries of their limited knowledge and imperfect cognitive capabilities. Bounded rationality produces uncertainty from various sources and makes it impossible to *ex ante* determine all contractually relevant aspects of a transaction (Williamson 1981a). The second behavioral assumption is that transaction partners might behave opportunistically. Opportunism is based on the notion of self-interest and further enhances the uncertainty associated with a transaction.

TCE argues that transactions have distinct characteristics that, in combination with the attributes of alternate governance structures, produce different production and transaction costs. The characteristics of a transaction are enumerated along three key dimensions: (1) asset specificity, (2) uncertainty, and (3) frequency of transactions (Williamson 1981a).

Asset specificity refers to the degree of to which the investments necessary for a transaction are specific to that particular transaction (Williamson 1981a). That is, if the transaction should fail, durable investments in transaction-specific human or physical assets have much less value in some second best use (Williamson 1986a). Such a situation can lead to dependencies between buyers and suppliers, for example, buyers cannot turn easily to an alternative supplier, and are thus locked into the transaction for a considerable time after (Williamson 1981a). Asset specificity can arise from many different sources such as site specificity (e.g., a special location necessary for a transaction), physical assets (e.g., specialized tools for the production of components), and human asset specificity (e.g., worker skills which are specific to the transaction). Regardless of the form of asset specificity, the issue is always the same: asset specificity creates a dependency and the party (i.e., seller, buyer, or both)

who has to invest in specialized assets is vulnerable to opportunism (Vining and Globberman 1999). Consequently this party will make special efforts to safeguard against opportunism (Rindfleisch and Heide 1997).

Asset specificity is very relevant to software procurement transactions. Asset specificity can depend on the degree of specialization of the software itself. Hospitals, for example, need specialized software that is not generic to all firms, but only to those in the healthcare industry and often to a specific hospital. The buyer cannot easily switch to another supplier. Conversely, supplier-invested assets to serve the specific needs of a specific hospital or to the healthcare industry cannot easily be sold to other hospitals or other industries. Thus, both buyer and seller are committed to design relationships that have a good chance of continuity (Williamson 1981a). Another example is software that consists of several smaller modules that are linked together. If one software module can be easily exchanged for another, it has low asset specificity. If it cannot be easily exchanged, it has high asset specificity. A third example is the purchase of a Web server or operating system that has consequences on the deployment of other applications. A variety of software runs on Windows but not on Linux and vice versa. Once a commitment to one system has been made, the buyer is locked-in and cannot easily switch to an alternative.

Uncertainty is the second key characteristic of a transaction and can come from different sources including environmental variability and behavioral uncertainty (Rindfleisch and Heide 1997). Uncertainty is especially interesting when asset specificity is present (Williamson 1986b). For nonspecific transactions, continuity of the relationship between buyer and seller is less important because the buyer can easily switch to another seller if the business situation changes, making the original transaction undesirable. Increasing the uncertainty does not change this situation. However, if the buyer is interested in a long-term relationship due to high asset specificity, then higher uncertainty increases the necessity of insurance against opportunism or the need to adapt the contract (Williamson 1986b). With high asset specificity, increasing uncertainty thus increases transaction costs.

Software procurement transactions are especially vulnerable to environmental uncertainty simply due to the complexity and volatility of the high-tech market (Rindfleisch and Heide 1997). In cases of high environmental variability, the writing of complete contracts might be difficult and contractual gaps may enlarge and increase the number of adaptations of the contract (Williamson 1986b). Behavioral uncertainty arises from the possibility of opportunism and the difficulty of monitoring transaction partners. With increasing behavioral uncertainty, costs for safeguarding and monitoring rise.

Finally, TCE asserts that the frequency of transactions influences both transaction and production costs. In empirical studies, frequency is often disregarded (Rindfleisch and Heide 1997). In our case, the procurement of new software is typically not a frequent activity in itself. Software customizations required to align an application with the specific and changing needs of an organization may indeed be a more frequent transaction; however, in this case the frequency of change would likely be highly correlated with the degree of asset specificity. Thus, for this study we also have chosen to disregard the frequency of transactions as having a distinct impact on transaction costs.

In summary, TCE argues that the combination of the dimensions of transactions with the attributes of governance structures causes different production and transaction costs. Each governance structure can be described “by an internally consistent syndrome of attributes [and] possesses distinctive strengths and weaknesses” (Williamson 2002, p. 175). The challenge is to align transactions to these governance structures in order to economize on transaction and production costs (Williamson 2002).

Governance Structures and Open Source Communities

Before we continue, we need to formally define the terms open source software, proprietary software, and open source community. We will use the term open source software (OSS) to refer to software whose licensing model guarantees the right to copy, modify, and distribute the source code of the program without discrimination (Feller and Fitzgerald 2000; Open Source Initiative 2005). OSS is usually available at no charge or for a minimal distribution fee. However, the price is not the distinctive characteristic of OSS. The key distinction from proprietary software lies in the license models that allow copying, modifying, and redistributing the source code. In contrast, proprietary software is either only available in binary form and cannot be read by a programmer or the license model of proprietary software prohibits changing and redistributing the software (Stallman 2002).

Perhaps the most interesting phenomenon surrounding OSS are the communities that evolve like self-organizing entities around the software development process (Stewart and Gosain 2001). Geographically dispersed developers work together to develop OSS (Cook 2001; Feller and Fitzgerald 2000). In addition to the core developers, users worldwide can contribute to the source code, test the software, report bugs, and suggest new features. The literature emphasizes that these communities are characterized by voluntary membership, high autonomy of the developers, and distribution of work via the Internet (Feller and Fitzgerald 2000;

Markus et al. 2000). The participants usually receive little or no financial rewards, although individuals might receive other nonmonetary rewards such as gaining reputation, career improvement, and increased skills (Hars and Ou 2002; Hertel et al. 2003; Lakhani and von Hippel 2003). Some firms employ developers to participate in open source projects (funded open source projects such as Apache and Mozilla). This may change the voluntary nature of open source projects although it does not change the principle that the source code is available for zero costs or a marginal fee and that licenses allow modification and distribution (Hertel et al. 2003).

The framers of TCE examine hierarchies and markets as two opposing governance structures, but as discussed previously, other distinct forms of governance have been proposed as complements to markets and hierarchies. Recent literature claims that open source communities are representative of a distinct form of governance called simply communities (Benkler 2002; Demil and Lecocq 2003; Glaeser 2003; Watson et al. 2005).

There are two key aspects of communities that should be highlighted. First, production in a market is coordinated by pricing mechanisms and production inside a hierarchy by formal rules (Coase 1937). Open source communities, on the other hand, do not rely on price mechanisms for coordinating transaction exchanges because most often no money is exchanged as part of the development or procurement process. Further, developers within an open source community usually work voluntarily, with high autonomy, and can choose what and for how long they want to develop. Hence, communities most often do not rely on the formal rules-based governance mechanism characteristic of a hierarchy (Demil and Lecocq 2003). Second, open source communities are characterized by low intensity of incentives since they are not coordinated by price and low control² because they are not contractually governed (Demil and Lecocq 2003; Watson et al. 2005). Table 1 presents a summary of the key characteristics of the two original governance forms alongside communities. We propose that communities are not a mix between hierarchies and markets but rather have their own distinct structure.

We will assume three distinct governance structures and one hybrid for our analysis of the make-or-buy decision for software. First, we assume that software can be purchased from a proprietary vendor (i.e., on the market). Second, software can be developed in-house (i.e., in a hierarchy). Third, software can be developed by an open source community and deployed in a firm without changes to the source code (i.e., by a community). Fourth, software can be developed by an open source community and the source code modified by the firm in order to adapt the software to its proprietary needs (i.e., through a hybrid of hierarchy and community).

**Table 1. Comparison of Governance Structures
(Demil and Lecocq 2003; Glaeser 2003; Watson et al. 2005)**

	Hierarchies	Markets	Communities
Contract law regime:	Employment contract	Classical contract	Open license
Normative base:	Forbearance	Market exchange	No copyright
Primary adjustment of actions by:	Formal rules	Price	Common subject matter of work (i.e., product)
Membership determined by:	Formal rules	Exchange offer	Perception of being a member
Nature of incentives:	Career advancement, status concerns	Competition	Reputational concerns, signaling
Intensity of Incentives:	Low	High	Low
Control:	High	Low	Low

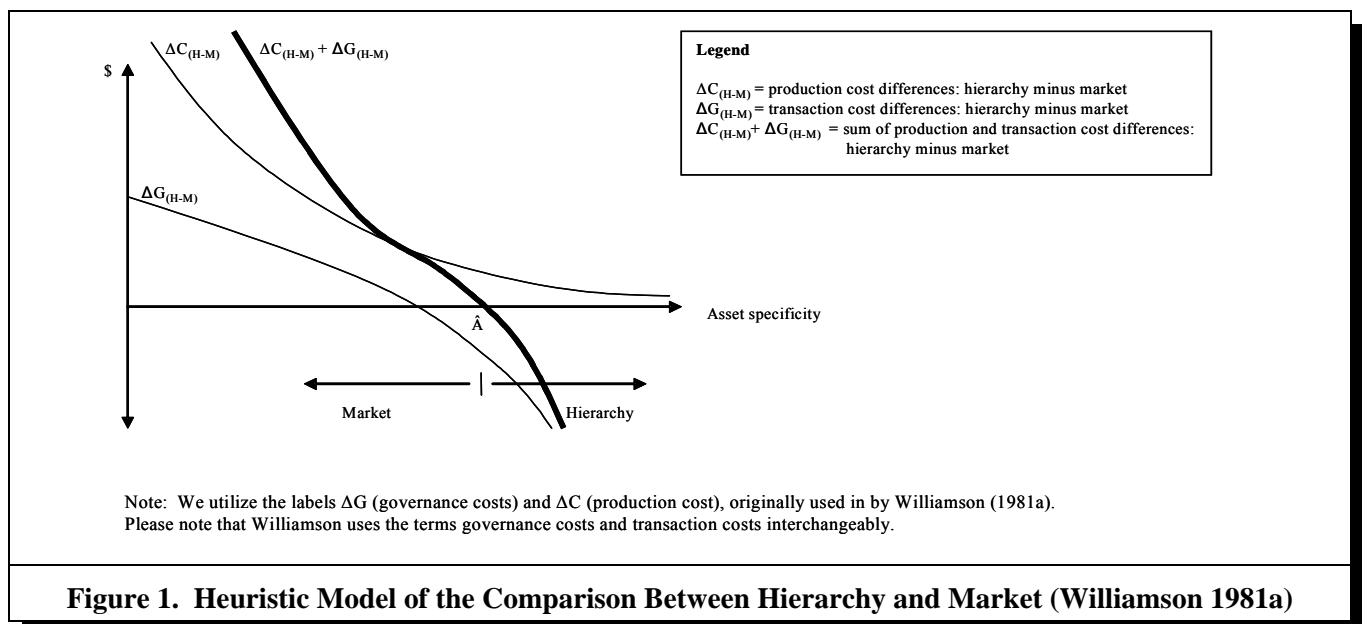
²Incentives direct people toward a desired output. Control refers to the capacity of a governance structure to reduce opportunism.

Open Source and Transaction Cost Economics

Comparison of Transaction and Production Costs Between Hierarchies and Markets

Riordan and Williamson (1985) point out that asset specificity is the principal factor responsible for differences in production and transaction costs. For production costs, they argued that a hierarchy (i.e., a firm) and market (i.e., outside suppliers) are identical in production cost technologies and therefore do not differentiate in production costs. However, this assumes that a firm is able to produce and sell products outside its core competency as efficiently as an outside supplier. According to Williamson (1981b), markets generally have a production cost advantage because they are more likely to realize economies of scale, economies of scope, and risk pooling benefits by aggregating uncorrelated demands. As asset specificity increases, the market (i.e., outside suppliers) specializes its investments and economies of scale and scope as well as risk-pooling benefits decrease (Williamson 1981a). Conversely, as assets become more unique, a firm is able to replicate the investments of outside suppliers without penalty (Riordan and Williamson 1985). Therefore, the production cost advantage of markets is a function of asset specificity and decreases with higher asset specificity, as depicted in Figure 1 by the curve $\Delta C_{(H-M)}$ (production costs of hierarchies minus production costs of markets). The production cost difference curve approaches zero with higher asset specificity but remains positive until the case of very high asset specificity (e.g., production unit is one) where production costs of markets and hierarchies may become indistinguishable (Riordan and Williamson 1985).

When organizing transactions in a hierarchy, bargaining, market contracts, and transaction costs are substituted by administrative decisions, employment contracts, and administrative costs. Thus, transaction costs in a market must be compared to administrative costs such as coordination costs in a hierarchy (Coase 1960). According to TCE, administrative costs are relatively independent of asset specificity (Riordan and Williamson 1985). For products with low asset specificity, markets will experience lower transaction costs than hierarchies, as shown by the curve $\Delta G_{(H-M)}$ (transaction costs of hierarchies minus transaction costs of markets). Low asset specificity means that buyers and suppliers are less dependent on each other and contractual safeguards are less necessary. However, with increasing asset specificity, opportunism—such as the seller increasing the price after the buyer is locked in—makes cost for contractual safeguards necessary. Bilateral or obligational market contracting appear, contractual gaps enlarge, and safeguarding and monitoring costs rise, thus increasing the transaction costs (Williamson 1981a; Williamson 1981b). Since the transaction costs of hierarchies (i.e., administrative costs) are not a function of asset specificity, the transaction costs advantages of markets will decrease until hierarchies are more favorable in terms of transaction costs. As long as the sum of $\Delta C_{(H-M)}$ and $\Delta G_{(H-M)}$ remains positive, markets will have a cost advantage. From point \hat{A} and to the right, however, hierarchies have a cost advantage over markets. Generally, with higher asset specificity, firms prefer internal production.



In the following sections we will compare how the introduction of open source communities changes the normative implications of TCE. We will first look at the area left of \hat{A} and compare markets to communities and markets to hierarchy-community hybrids. For the first part, we assume that a decision to buy the product has been made and the firm evaluates whether to obtain the software from a market or community. For the second part, we examine whether the introduction of open source as a hybrid form changes the decision to make or to buy. Finally, we will examine the area right of \hat{A} by comparing hierarchies to hierarchy-community hybrids. Here we assume that the decision to make the product is made and we evaluate whether we develop from scratch or start with available OSS.

Please note that all of these models are heuristic. Transaction costs are difficult to measure since they are context dependent and parts such as search costs and negotiation costs are difficult to quantify. Yet, transaction costs do not need to reach the level of production costs in order to play an important role in the choice between hierarchy and market production. As Williamson points out, transactions “will be organized in markets *unless* transaction cost disabilities appear” (1981b, p. 1547). In this case transaction cost differences play the role of a tipping point that causes firms to move from outside to internal production. More important than the exact slope and intercept of the curves or the exact position of our proposed thresholds (\hat{A}) are the changes in governance options as we introduce open source communities to the Williamson TCE model and the relative attractiveness as asset specificity increases.

Given a Buy Decision, Should it Be Community or Market?

Figure 2 shows the differences in transaction and production costs between markets and communities *from the perspective of the buying firm*. With higher asset specificity, production costs and the price of a product on the market rise, and because the price of OSS is usually zero or low, communities will have an increasing production cost advantage over markets ($\Delta C_{(M-C)}$). Please note that Figure 2 shows a heuristic model and the production cost difference curve $\Delta C_{(M-C)}$ does not necessarily need to be linear. More important than the slope of the curve is that in comparison to the market, communities enjoy a production cost advantage over markets that is likely to increase with higher asset specificity.

It is suggested that communities have lower transaction costs for the buyer because the negotiation and writing of contracts is not necessary (Demil and Lecocq 2003). Some transaction costs are also assumed to be lower because opportunism or modifying the pricing structure after a lock-in is not an issue in most open source communities. However, transaction costs are not as low as this argument might suggest.

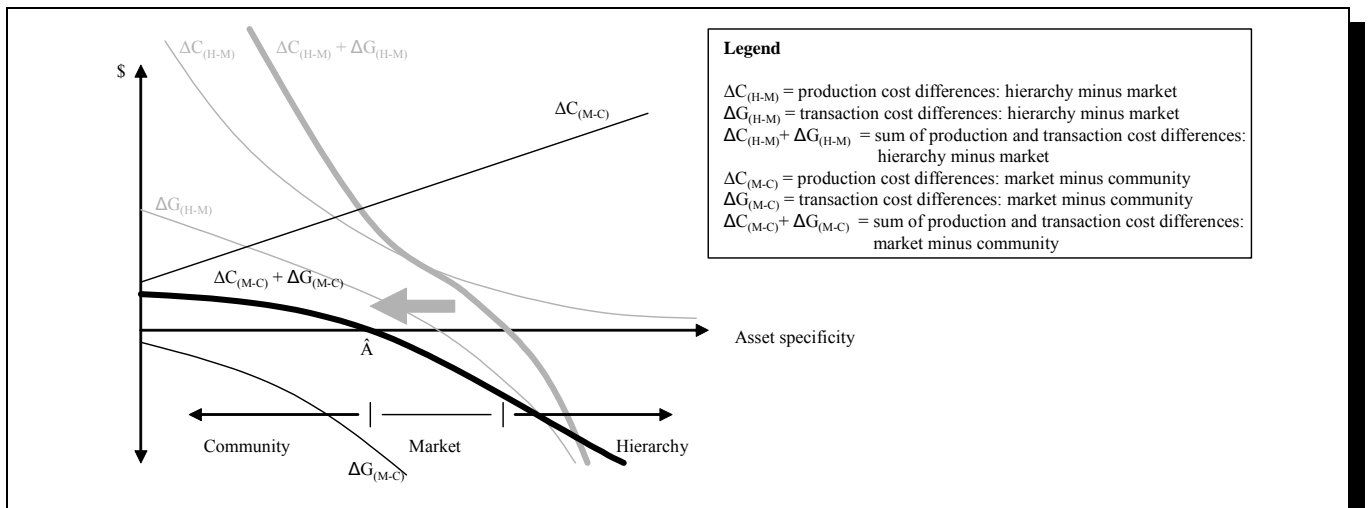


Figure 2. Comparison Between Market and Community

First, transaction costs include not only search, negotiation, and writing costs, but also costs from risk and higher uncertainty associated with open source. Potential risks—and the costs associated with their mitigation—include legal concerns, warranty issues, liability questions, patent violations, and copyright infringement (Hang et al. 2005). One major risk is project failure. Higher asset specificity creates a lock-in because software cannot easily be exchanged. Because open source communities are usually based on voluntary membership, there is no guarantee that the open source community will flourish and continue to maintain and further develop the software. If the open source community withers, it will lead to high costs because software needs to be either maintained by the firm or exchanged. Insuring against this risk is a type of transaction cost.

Another source of transaction costs is quality assurance. If a firm decides to buy the product on the market, it relies on the quality assurances of the market (i.e., the guarantees of the proprietary vendor). Proponents of open source assume superior quality due to peer production since bugs can be spotted and corrected quickly (Raymond 1998). However, this quality assurance mechanism of communities is likely to depend on the level of participation in the community. Although, some empirical evidence to support the superior code quality argument exists (Samoladas and Stamelos 2003; Samoladas et al. 2004), firms may want to deploy additional mechanisms to assure code quality and lower the uncertainty associated with code quality.

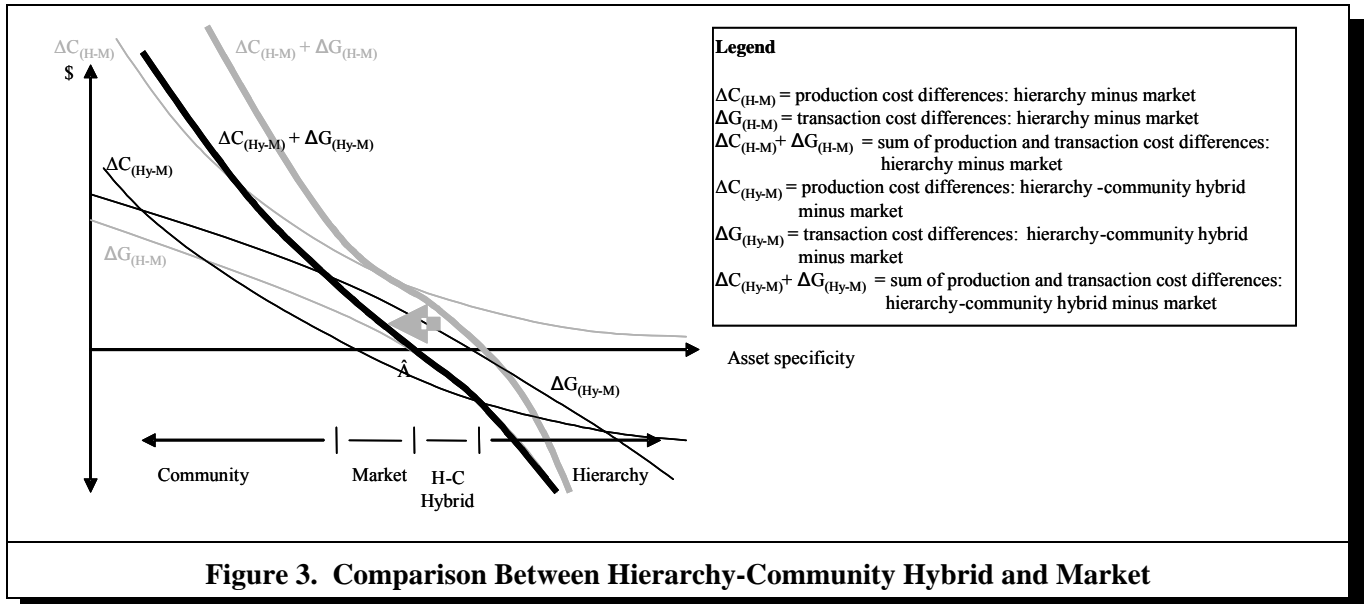
Behavioral uncertainty is also an important source of transaction costs. Open source communities are characterized by a strong developer culture. Developers usually work voluntarily and have low incentives to fulfill customer needs. Coordination within a community is not regulated by a contract or price but developers choose a development task based on their interest and competency (Demil and Lecocq 2003; Glaeser 2003). Even if the open source project is funded by an outside company (e.g., IBM and Apache), the *buying* firm has usually only little influence on the developers and the direction of the project. Transaction costs thus occur because communities are characterized by low intensity of incentives and low controls (Watson et al. 2005). Costs to ensure control and enable incentives—such as providing funds for projects or hiring developers to participate in the OSS communities—may be substantial. By not including these potential costs, total costs may be underestimated.

Due to the risk of project failure and legal, quality, and behavioral uncertainties, we propose that initial transaction cost advantages of a community will decrease and make the market the preferable form of governance with higher asset specificity ($\Delta G_{(M-C)}$). In Figure 2, we can see that as asset specificity increases, community, markets, and then hierarchies constitute a continuum that economizes on total costs. For low asset specificity, communities are preferable since they enjoy a production cost advantage compared to markets. With increasing asset specificity, however, we argued that the transaction costs of communities increase and firms will purchase software on markets until the transaction costs of markets in comparison to hierarchies cause the firm to switch to internal production.

This will hold true as long as we assume availability of similar software (e.g., quality, functionality) in markets and communities. If no adequate alternative is found in open source communities, market will be the preferred governance mode regardless of asset specificity.

Hierarchy-Community Hybrid and the Make-or-Buy Decision

In this section, we examine whether and when the introduction of open source changes our choice of buying on the market if an OSS alternative is available that can be modified and adapted to the firm's needs (hierarchy-community governance). We argued in the previous section that the transaction costs of communities will significantly increase due to the risk of project failure and legal, quality, and behavioral uncertainties. However, due to open source licensing, a firm is not forced to rely entirely on production within a community. The open source license guarantees the right to copy, modify, and distribute source code. Thus a firm may take OSS as a basis for its own customized, in-house development. Note that the distinction to the "pure" form of community lies in the degree of reliance on the community to develop the software to meet the specific needs of the organization. In our former example, a firm is not changing the underlying source code. All production is done entirely within the community. In the hierarchy-community hybrid mode, the firm takes on part of the software development effort. Thus, production costs of a hierarchy-community hybrid (C_{Hy}) can be assumed to be lower than would be seen in a pure hierarchy (C_H) because a significant portion of the development has already been done by the community ($C_{Hy} < C_H$). The firm is able to transfer development effort from the community to in-house developers at a relatively low price. If we take $C_{Hy} < C_H$ as given and subtract from both sides the production costs of markets, $C_{Hy} - C_M < C_H - C_M$ we conclude $\Delta C_{(Hy-M)} < \Delta C_{(H-M)}$, that is, the production cost difference curve between hierarchy-community hybrid and market ($\Delta C_{(Hy-M)}$) is lower than the production cost difference curve between hierarchy and market ($\Delta C_{(H-M)}$) as shown in Figure 3.



Please note that we assume that starting with an available software solution might be preferable to starting from scratch. This argument depends on the complexity of the software. The more complex the software, the more time developers would need to get accustomed to the source code and effectively change the source code. On the other side, OSS software requires utilizing a modularized design (e.g., object-orientation) so that parallel and decentralized development is possible. Hence, modules may be reused and not all modules need to be changed, thus reducing the development effort and development costs.

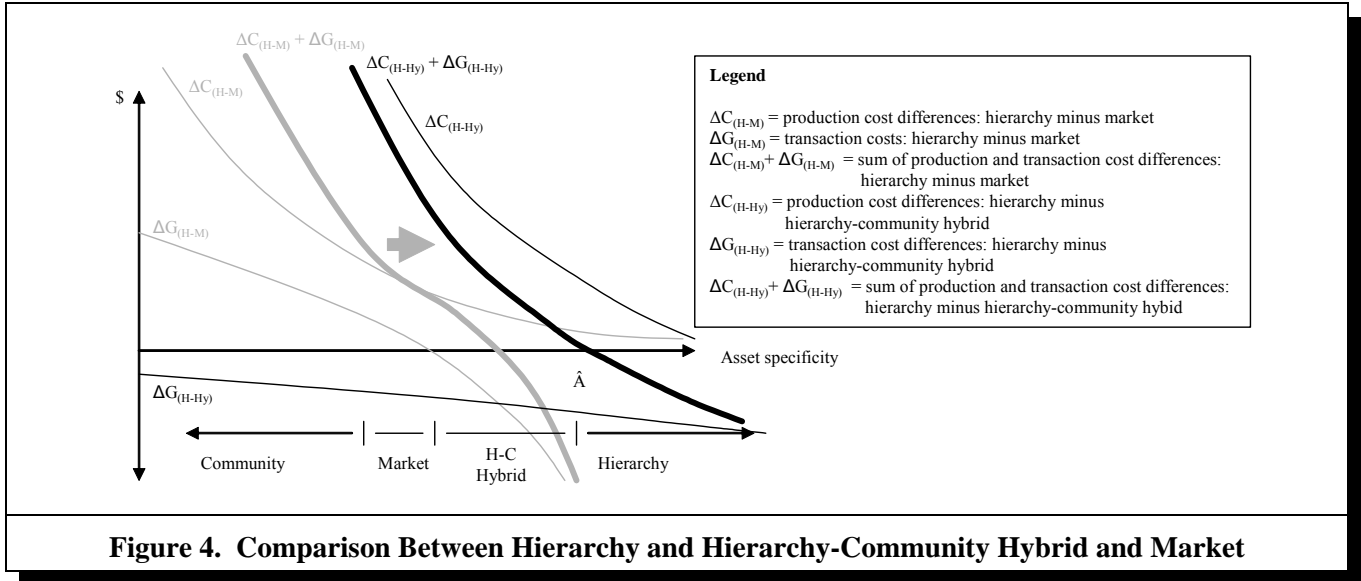
Transaction costs may be higher when compared to hierarchies because the transaction costs of hierarchy-community hybrids include both those of hierarchies and of communities. However, the important uncertainty of project failure can be mitigated because the firm is now a member of the community and can ensure the longevity of the open source project. Conservatively, we assume that the transaction costs of the hierarchy-community are higher than the transaction costs of hierarchies ($G_{Hy} > G_H$) due to the additional uncertainties and risk associated with using OSS. Therefore, the $\Delta G_{(Hy-M)}$ line is higher than the $\Delta G_{(H-M)}$ line, that is, the transaction cost penalties of a hierarchy-community hybrid versus a market ($\Delta G_{(Hy-M)}$) are larger than the penalties of a hierarchy versus a market ($\Delta G_{(H-M)}$) as shown in Figure 3.

In summary, if the production cost advantages of hierarchy-community hybrids exceed the transaction cost advantages of markets, the threshold (\hat{A}) will shift to the left. Please note that it is not certain that there is a level of asset specificity where this is the case. However, since OSS can be acquired at such a low cost, it certainly could move the threshold to the left because production costs of internal production will decrease.

Given a Make Decision, Should it Be Hierarchy-Community Hybrid or Hierarchy?

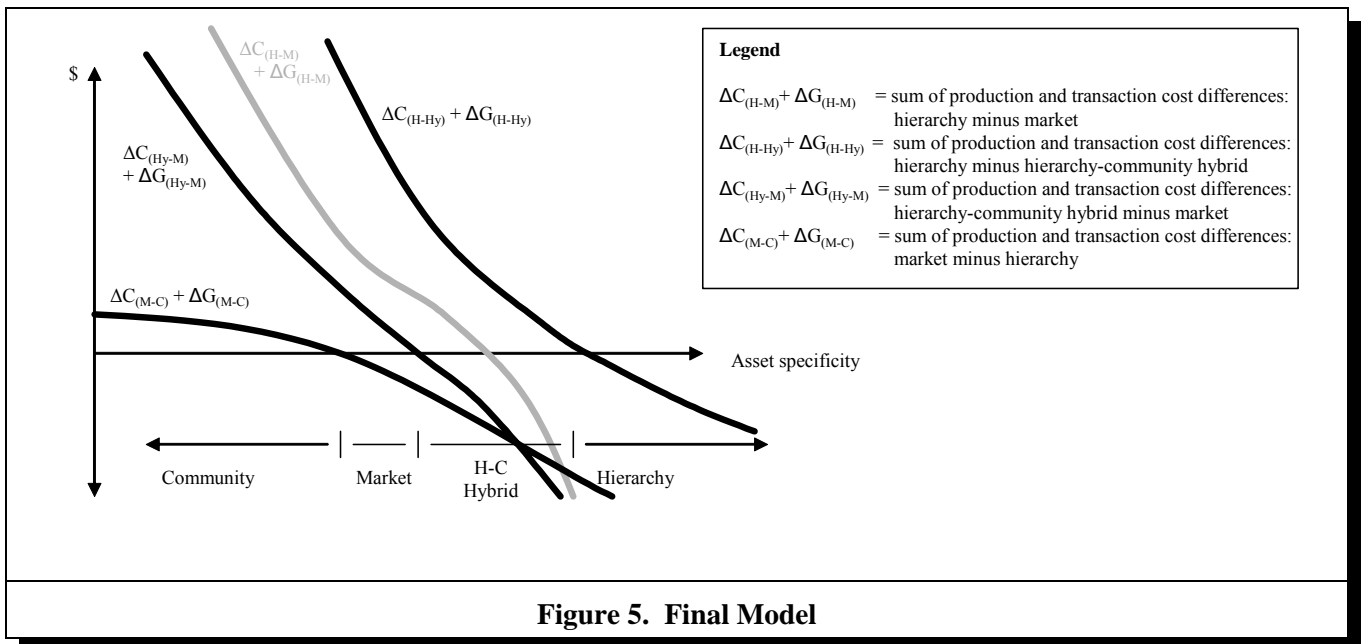
Our last evaluation concerns the comparison between hierarchy and hierarchy-community hybrid governance structures. Note that the decision to develop the software is already made. At relatively lower asset specificity, the hybrid mode will have a production cost advantage over the hierarchy. With increasing asset specificity, this cost advantage should decrease as it becomes more attractive to design the software from scratch rather than make such extensive changes to existing software.

We conservatively assume that the transaction costs of the hybrid mode are larger than the transaction costs of hierarchies. With increasing asset specificity, this gap is likely to widen slightly because, on the one hand, the firm does not rely entirely on the OSS community as in the community governance mode but, on the other hand, there are inherent uncertainties involved in using OSS ($\Delta G_{(H-Hy)}$). Generally, we can assume that the new threshold (\hat{A}) is located to the right of the original threshold ($\Delta C_{(H-M)} + \Delta G_{(H-M)}$) because the firm is likely to save by using already available software to adapt to its own needs. However, as asset specificity increases, the hybrid production cost advantage will decrease, its transaction cost disadvantage will increase, and at some point developing a custom solution “from scratch” becomes more appropriate (\hat{A}).



Composite Model for Make-or-Buy Decisions in the Age of Open Source

Figure 5 shows the final model. There are three points worth mentioning. First, the introduction of open source options results in a decision continuum of community, market, hierarchy-community hybrid, and hierarchy as a function of asset specificity. Second, the community is an adequate governance structure for software with low asset specificity and should be expected to take “product market share” away from the market governance structure. Third, the case where the firm uses and develops open source leads to a change in the make-or-buy decision. Firms will consider some degree of in-house hybrid production at lower asset specificity levels than seen in Williamson’s original model.



Discussion

Limitations

TCE has been criticized for several reasons, most notably its tendency to over-simplify complex decision-making processes (Ghoshal and Moran 1996). This study may also share some of these limitations. For example, the benefits of OSS may not be adequately accounted for in the transaction costs. OSS development has been claimed to be superior in the identification and allocation of creativity (Benkler 2002). Access to a larger pool of ideas, creativity, and innovations may provide benefits larger than the transaction costs associated with community governance. Also, for example, the resource-based view of the firm argues that a firm is able to achieve a competitive advantage by focusing on the firm's unique internal resources that are rare, valuable, and difficult to imitate. These resources can be accumulated and built over time. If information technology is of strategic value as a resource, costs may play a secondary role to the value the resources create (Barney and Hesterly 1996). In this case, in-house development of own or OSS software may be desirable even if, from a production and transaction cost perspective, external software development or off-the-shelf software purchase is more cost efficient. Conversely, development of OSS may not be desirable if the OSS license requires firms to publish changes of the source code and firms worry that their competitive advantage diminishes because competitors have access to the same modified software. Such strategic considerations, along with in-house IT skills availability and political considerations, are often not adequately accounted for when solely examining transaction costs.

It is important to note also that software systems are often composed of a large variety of components tied together to create a composite business solution. The individual components often are procured individually, may each have differing degrees of asset specificity and uncertainty, and thus may each have to be created *optimally* under different governance structures. This study focuses on procurement at the most atomic level; attempting to apply our findings at the system level may not be applicable in many cases.

Please note also that our analysis focuses on obtaining and installing software. Maintenance costs are separate from production and transaction costs and not included in our analysis. Whether OSS is less expensive to administer depends on various factors such as in-house availability of IT experts and number of developers in the OSS community. Thus it is difficult to resolve the question of what governance structure is superior to the others in terms of maintenance costs. Proponents of OSS would argue that maintenance of OSS is less costly since maintenance tasks such as bug fixing and releasing new versions can be done within the OSS community and expensive maintenance contracts with vendors are not necessary. This would reduce the maintenance costs for the community and hierarchy-community hybrid governance form and consequently support rather than contradict our previous arguments.

Conclusions

TCE has been widely used to explain and predict the appropriate governance structure for make-or-buy decisions. We contribute to this body of research by incorporating community as a third governance choice into Williamson's original analysis. By comparing transaction and production costs along a continuum of varying asset specificity, TCE should help to predict when open source communities are appropriate governance structures for specific make-or-buy decisions. The basic tenet that with higher asset specificity the hierarchy becomes the appropriate governance structure was confirmed in our analysis. However, the introduction of open source communities changed the thresholds and added two additional governance forms. The pure *community* governance structure takes a share from the market governance structure because an open source community has production and transaction cost advantages for low asset specificity. At higher levels of asset specificity, the hierarchy-community *hybrid* takes some share from both markets and hierarchies.

Studies contrasting the total cost of ownership of OSS to proprietary software are both abundant and controversial today. TCE argues that the total cost of ownership depends on the asset specificity of the software for the particular buyer. TCE further asserts that transaction characteristics together with governance structure attributes lead to different transaction costs. Most current studies analyze total cost of ownership without consideration for concepts espoused by TCE, and thus may represent incomplete or faulty inputs into the decision-making process for organizations today. This study highlights these shortcomings and may serve to improve the decision-making process for organizations with regard to the adoption of OSS.

With our TCE model, we also might help explain an interesting development in the open source world. Professional open source firms—such as JBoss, MySQL, and Sleepycat—are second-generation open source organizations that have built viable businesses around OSS while still providing the software itself at low or no cost. They often rely on employees to perform the bulk of the

development for their OSS. Thus, the strength of control over developers is higher than in traditional open source projects and may reduce uncertainty as to the longevity of the organization and code quality. These firms also reduce legal uncertainties by offering their customers legal indemnity. These concepts, together with necessarily improved customer orientation versus traditional open source communities, serve to decrease the uncertainty related to deploying OSS while maintaining many of the benefits, not the least of which being low initial acquisition costs. The professional OSS firms are thus able to extend the community selection threshold further to the right by embracing some of the benefits of markets.

References

- Aubert, B. A., Rivard, S., and Patry, M. "A Transaction Cost Approach to Outsourcing Behavior: Some Empirical Evidence," *Information and Management* (30) 1996, pp. 51-64.
- Aubert, B. A., Rivard, S., and Patry, M. "A Transaction Cost Model of IT Outsourcing," *Information and Management* (41) 2004, pp. 921-932.
- Barney, J. B., and Hesterly, W. "Organizational Economics: Understanding the Relationship between Organizations and Economic Analysis," in *Handbook of Organization Studies*, S. R. Clegg, C. Hardy and W. R. Nord (Eds.), Sage Publications, London, 1996, pp. 115-147.
- Benkler, Y. "Coase's Penguin, or, Linux and the Nature of the Firm," *The Yale Law Journal* (112:3) 2002, pp. 369-446.
- Butler, R. "A Transactional Approach to Organizing Efficiency," *Administration and Society* (10) 1982, pp. 323-362.
- Coase, R. H. "Nature of the Firm," *Economica* (4:16) 1937, pp. 386-405.
- Coase, R. H. "The Problem of Social Cost," *Journal of Law and Economics* (3:1-44) 1960.
- Cook, J. E. "Open Source Development: An Arthurian Legend," in *Making Sense of the Bazaar: Proceedings of the 1st Workshop on Open Source Software Engineering*, J. Feller, B. Fitzgerald, and A. van der Hoek (Eds.), Toronto, Canada, May 2001, pp. 16-19.
- Demil, B., and Lecocq, X. "Neither Market nor Hierarchy or Network: The Emerging Bazaar Governance," 2003 (available at http://opensource.mit.edu/online_papers.php, accessed August 21, 2005).
- DiDio, L. "Linux, UNIX and Windows TCO Comparison," The Yankee Group Report, Boston, April 2004.
- Feller, J., and Fitzgerald, B. "A Framework Analysis of the Open Source Software Development Paradigm," in *Proceedings of the 21st International Conference of Information Systems*, W. J. Orlikowski, S. Ang, P. Weill, H. C. Krcmar, and J. I. DeGross (Eds.), Brisbane, Australia, December 10-13, 2000, pp. 58-69.
- Ghoshal, S., and Moran, P. "Bad for Practice: A Critique of the Transaction Cost Theory," *Academy of Management Review* (21:1) 1996, pp. 13-47.
- Giera, J. "The Costs and Risks of Open Source: Debunking the Myths," Forrester Research, Cambridge, MA, 2004.
- Glaeser, J. "A Highly Efficient Waste of Effort: Open Source Software Development as a Specific System of Collective Production," paper presented at the Australian Sociological Association Conference, University of New England, Australia, December 4-6, 2003.
- Hang, J., Hohensohn, H., Mayr, K., and Wieland, T. "Benefits and Pitfalls of Open Source in Commercial Contexts," in *Free-Open Source Software Development*, S. Koch (Ed.), Idea Group Publishing, Hershey, PA, 2005.
- Hars, A., and Ou, S. "Working for Free? Motivations for Participating in Open-Source Projects," *International Journal of Electronic Commerce* (6:3) 2002, pp. 25-39.
- Hertel, G., Niedner, S., and Herrmann, S. "Motivation of Software Developers in Open Source Projects: An Internet-Based Survey of Contributors to the Linux Kernel," *Research Policy* (32:7) 2003, pp. 1159-1177.
- Lakhani, K. R., and von Hippel, E. "How Open Source Software Works: 'Free' User-to-User Assistance," *Research Policy* (32) 2003, pp. 923-943.
- Markus, M. L., Manville, B., and Agres, C. E. "What Makes a Virtual Organization Work?," *Sloan Management Review* (42) 2000, pp. 13-26.
- Ngwenyama, O. K., and Bryson, N. "Making the Information Systems Outsourcing Decision: A Transaction Cost Approach to Analyzing Outsourcing Decision Problems," *European Journal of Operational Research* (115) 1999, pp. 351-367.
- Open Source Initiative. "Open Source Definition," 2005 (available online at <http://www.opensource.org/docs/definition.php>; accessed August 21, 2005).
- Raymond, E. S. "The Cathedral and the Bazaar," *First Monday*, 1998 (available online at http://www.firstmonday.dk/issues/issue3_3/raymond/index.html; accessed August 21, 2005).
- Rindfleisch, A., and Heide, J. B. "Transaction Cost Analysis: Past, Present, and Future Applications," *Journal of Marketing* (61:4) 1997, pp. 30-54.
- Riordan, M. H., and Williamson, O. E. "Asset Specificity and Economic Organization," *International Journal of Industrial Organization* (3) 1985, pp. 365-378.

- Samoladas, I., and Stamelos, I. "Assessing Free/Open Source Software Quality," 2003 (available at http://opensource.mit.edu/online_papers.php; accessed August 21, 2005).
- Samoladas, I., Stamelos, I., Lefteris, A., and Oikonomou, A. "Open Source Software Development Should Strive for Even Greater Code Maintainability," *Communications of the ACM* (47:10) 2004, pp. 83-87.
- Stallman, R. M. *Free Software, Free Society: Selected Essays of Richard M. Stallman*, Free Software Foundation, Boston, 2002.
- Stewart, K., and Gosain, S. "An Exploratory Study of Ideology and Trust in Open Source Development Groups," in *Proceedings of the 21st International Conference on Information Systems*, V. Storey, S. Sarkar, and J. I. DeGross (Eds.), New Orleans, LA, December 16-19, 2001, pp. 507-512.
- Vining, A., and Globberman, S. "A Conceptual Framework for Understanding the Outsourcing Decision," *European Management Journal* (17:6) 1999, pp. 645-654.
- Wang, E. T. G. "Transaction Attributes and Software Outsourcing Success: An Empirical Investigation of Transaction Cost Theory," *Information Systems Journal*, (12) 2002, pp. 153-181.
- Watson, R. T., Boudreau, M. -C., Greiner, M., Wynn, D., York, P., and Gul, R. "Governance and Global Communities," *Journal of International Management* (11:2) 2005, pp. 125-142.
- Williamson, O. E. "Comparative Economic Organization: The Analysis of Discrete Structural Alternatives," *Administrative Science Quarterly* (36) 1991, pp. 269-296.
- Williamson, O. E. *Economic Organization: Firms, Markets and Policy Control*, New York University Press, New York, 1986a.
- Williamson, O. E. "The Economics of Organization: The Transaction Cost Approach," *The American Journal of Sociology* (87:3) 1981a, pp. 548-577.
- Williamson, O. E. *Market and Hierarchies: Analysis and Antitrust Implications*, The Free Press, New York, 1975.
- Williamson, O. E. "The Modern Corporation: Origins, Evolution, Attributes," *Journal of Economic Literature* (19), December 1981b, pp. 1537-1568.
- Williamson, O. E. "The Theory of the Firm as Governance Structure: From Choice to Contract," *Journal of Economic Perspectives* (16:3) 2002, pp. 171-195.
- Williamson, O. E. "Transaction-Cost Economics: The Governance of Contractual Relations," in *Organizational Economics*, J. B. Barney and W. G. Ouchi (Eds.), Jossey-Bass Publishers, San Francisco, 1986b, pp. 98-129.