Association for Information Systems
AIS Electronic Library (AISeL)

MCIS 2009 Proceedings

Mediterranean Conference on Information Systems (MCIS)

2009

Why Do Commercial Firms Open The Source Code Of Their Products?

Jai Asundi
School of Management, University of Texas at Dallas, asundi@utdallas.edu

Octavian Carare
School of Management, University of Texas at Dallas, tavi@carare.net

Kutsal Dogan
Graduate School of Business, Ozyegin University, kutsal.dogan@ozyegin.edu.tr

ECONOMICS OF IS

WHY DO COMMERCIAL FIRMS OPEN THE SOURCE CODE OF THEIR PRODUCTS?

Asundi, Jai, School of Management, University of Texas at Dallas, 2601 North Floyd Road, Richardson, TX 75080, USA, asundi@utdallas.edu
Carare, Octavian, School of Management, University of Texas at Dallas, 2601 North Floyd Road, Richardson, TX 75080, USA, carare@utdallas.edu
Dogan, Kutsal, Graduate School of Business, Ozyegin University, Kuşbakışı Cad. No:2, Altunizade, Istanbul, 34662, Turkey, kutsal.dogan@ozyegin.edu.tr

Abstract

This paper is concerned with the economic trade-offs associated with open-sourcing, the business strategy of releasing the source code of a commercial software product. We model open-sourcing as a strategic option for firms that compete in the market for software products. At the core of our model is the effect of open-sourcing on customer values, as well as the relative ease of customizing the open-source products. We show that open-sourcing can arise as an equilibrium outcome in our two-stage game. If the enhancement of customer value from open-sourcing is moderate or high, in equilibrium firms may find it optimal to release the source code of their commercial software products even when this strategy may reduce their profits.

Keywords: *Open Source Software, Game Theory, Open-sourcing, Competition*

1 INTRODUCTION

We wish to investigate the observation, puzzling to many, that some firms in the current business environment choose to open-source some of their software products. In what follows, we refer to the release of a software product to the open-source community as open-sourcing. The observation that many firms choose to open-source their products is at first sight puzzling because open-sourcing can hardly be seen as consistent with profit maximization. Clearly, the commercial product and the open-source counterpart are substitutes. It seems quite intuitive that by making a substitute product available free of charge, any software producer would lower its profit from the sale of the commercial product. Thus, we are quite uneasy about the tension that arises between this apparent reduction in profit and the observation that an increasing number of software firms choose to open-source their commercial products.

An increasing number of firms release their products, free of charge. For instance, in October 2004, IBM released Cloudscape, a relational database product, to the Apache Software Foundation, an active member of the OSS community. It is interesting to note that consistent with what seems to be the norm for this type of situation, IBM offers full customer support for the product that was released to the OSS community. Similarly, in August 2004 Computer Associates released their database product Ingres to the OSS community. Notably, in November of 2005, Computer Associates created a new company, Ingres Corp., to provide support and services for their OSS database product.

Why would a firm that enjoys a sizable stream of profit from the sale and service of a product choose to create its own competition by releasing a free open-source product? How does open-sourcing affect the competitive environment faced by software firms? And, importantly, is there an economic mechanism through which open-sourcing can contribute to software firms enhancing their competitive position?

The following two quotes suggest some explanations. According to John Prial, IBM's vice president of marketing and information management software,

"By open sourcing Cloudscape, IBM hopes to accelerate development of Java-based applications and drive more innovation around Linux and Java. [...] We think it will especially create new business opportunities [...]."(Prial (2004))

Bertrand Serlet, senior vice president of software at Apple, argues that

"[With open-source code,] thousands of people look at the critical portions of source code and check those portions are right. It's a major advantage to have open-source code."(Serlet, 2004)

An increased pace of innovations and improved security through increased exposure are, indeed, two of the major candidate explanations for the recent pattern of open-sourcing. But are these sufficient reasons to open-source a product? We argue that the answer is a qualified yes. Open-sourcing may result in product innovation and quality. The literature mentions several other reasons for open-sourcing. An important such reason is the use of open-source products by firms who wish to gain an advantage over their competitors. Few of the explanations in the literature, however, discuss the impact of open-source products on the customer's perception of the commercial and open-source products. We argue in what follows that the release of an open-source product affects the customer's valuations. We show how this change in customer valuations, in turn, is an important determinant of a firm's open-sourcing strategy.

We intend to show by way of a simple model how open-sourcing can arise as an equilibrium strategy. Even though our model is somewhat stylized, we are able to capture some of the principal economic trade-offs involved in the software developer's decision to release open-source products. We find that open-sourcing can be profitable in some situations. Open-sourcing can arise as a result of competition, despite the reduction in profit that is caused by "customer loss" – i.e., the reduction in market share that may arise as a result of open-sourcing. We show that if the enhancement of customer value that results from open-sourcing is moderate, firms may find it optimal to release open-source products. When the value gains to the customers from open-sourcing are high, we show that firms cannot take full advantage of these gains. The firm's inability to funnel some of the customer value gains into higher profits is due to increased competition. Overall, our results indicate that it is the customers, not the firms, who are likely to benefit the most from open-sourcing.

Our paper has two important managerial implications. First, we show that open sourcing is more likely to be an outcome of competition when firms anticipate that the presence of an open-source product enhances customer values for the commercial product. This value enhancement may be primarily attributable to new product features, to bug fixes and to improved security that result from open-sourcing. Second, as intuition suggests, increased competition from the free open-source products of their competitors erodes the profits of the firms that do not release open-source products.

The next section provides a review of the relevant literature. Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.** gives a brief outline of the market for open-source products. We develop our model in Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.** and collect results in Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. Concluding remarks are in Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. The proofs and calculations are relegated to an Appendix that is available by request.

2 RELATED LITERATURE

Our work is related to the literature on OSS and to the literature on pricing of information goods. Schiff (2002) provides a comprehensive survey of the early literature on OSS. The current research on OSS can be classified into three broad categories (see von Krogh and von Hippel, 2006). Analysis of the motivations of open source contributors is by far the most popular research topic, perhaps because it has at its core the puzzling observation that cohorts of talented programmers choose to contribute to OSS projects with no apparent compensation. Unlike the programmers of most commercial software projects, OSS project contributors are volunteers located in various parts of the world. Topics concerning the governance, organization and innovation processes associated with OSS constitute the second main stream of OSS research (MacCormack et al., 2006; Koch and Schneider, 2002). The third stream of research is focused on the competition between open source and traditional, closed-source software (Casadesus-Masanell and Ghemawat, 2006; Economides and Katsamakas, 2006; Bonaccorsi et al., 2006; Mustonen, 2005).

Our paper belongs to the third stream of OSS research. We seek to provide some economic explanations for the increased incidence of firms that compete by releasing open-source counterparts of their proprietary software products. Like our analysis, a few studies examine hybrid business models that include proprietary and open source software (Bonaccorsi et al., 2006; Krishnamurthy, 2005). An important motivation in Rossi and Bonaccorsi (2005) is that firms that open their code expect to obtain contributions and feedback in order to fix bugs and improve the software. Other explanations for open-sourcing include, as perhaps best articulated by IBM’s Jon Prial (2004), an increase in the rate of innovations and the resulting increase in demand for a complementary commercial product of the same firm.

A few studies examine the competition between commercial software and OSS (Casadesus-Masanell and Ghemawat (2006), Economides and Katsamakas (2006)). These studies assume the existence of an OSS product without specifically addressing the determinants of a firm’s decision to open source.

Hawkins (2004) makes an important point that the release of code may be profitable because it entails a reduction in the cost of maintaining the code. In Mustonen (2005) the firm’s decision not to support the rival software results in incompatibility between its commercial program and the freely available substitute. The model is similar to ours in that it considers customers who are heterogeneous with respect to their valuations of the competing products, but in his model only one firm acts strategically. A similar analysis by Sen (2007) explores the competition between proprietary software, an OSS product and a commercially-supported offering of the OSS product. August et al. (2007) consider a model in which a firm chooses between open- and closed-source architectures. Profits are obtained from services such as integration, support and consulting associated with the open source product.

Our work is also related to the literature on the pricing of information goods, in particular, to the work on versioning of information goods (Bhargava and Choudhary, 2008; Weber, 2008; Jing, 2007; Ghose and Sundararajan, 2005; Sundararajan, 2004; Dewan et al., 2003; Jing, 2003; Bhargava and Choudhary, 2001; Raghunathan, 2000). While, like most models of versioning, our model views the open source product as a differentiated version of the closed-source product, our model does not view open-sourcing as a tool for price discrimination or as a cause of network effects.

The explanations given in the literature for the existence of open source software, while providing valuable insight, go only some way toward identifying the reason why software firms choose to open-source their products. In the academic literature and in the media, two stories seem to coalesce as the most likely candidate explanations for open-sourcing. First, the release of open-source products increases market size, so that firms benefit from the sale of complementary products or services. Second, the release of open-source products reduces the cost of maintaining and debugging the code. The logic of both arguments relies on the fact that open-sourcing may be, from a dynamic perspective, profitable for a software firm. Both explanations rely on the intuition that a favorable trade-off exists for the software firm between short run losses in revenue that stem from "customer loss" (i.e., the reduction of revenue that arises as a result of making a substitute product available free of charge) and, in the long run, increased profitability that is due to higher revenues or lower costs. We think that in this particular case intuition is misguided. If the main consequence of open-sourcing is an increase in the number of customers who use the product, market size could also be increased through free distribution of closed-source software. Free distribution could also result in better testing and reporting of bugs. Furthermore, the reduction in the cost of maintaining or debugging the source code achieved as a result of "more eyeballs" scanning the released source code for bugs could be achieved through the release of the source code to a set of qualified firms or individuals, and not to the community at large.

It is unlikely that the economic drivers of a firm's decision to release open source products could be clearly and easily enumerated. A firm's open-sourcing decision is affected by a multitude of factors. Some of these factors are identified in the literature. Our contribution is to bring to the fore an important, yet little explored aspect of open-sourcing: the impact of open source releases on the customer's valuation of the product. In the next section we highlight some of the important characteristics of open-source products and explore the ways in which these characteristics affect the customer's perception of open- and closed-source products .

3 OSS MARKET AND PRODUCTS

Hardly any online user forum devoted to a particular software product lacks complaints from users concerning the product features or, more often, the absence of desired features. In their out-of-the-box state most software products fail to meet each minute requirement of their users. In general, closed-source products cannot be viewed as highly customizable. By customization we mean changing the product to fit the existing infrastructure and needs of a firm. Conversely, OSS products may be freely customized to meet any user's precise needs. OSS users may manipulate the source code, either to make minor modifications or to go as far as to significantly change the product's functionality to integrate it to the existing information systems. While users of closed-source software products are generally restricted to making only minor changes to the product, they may modify their processes or practices in order to use the software more effectively. The time and effort spent incorporating desired functionality into OSS products or adapting to the requirements of closed-source products is reflected in costs incurred by the user. We believe that, for most products, the cost of customizing an open source product is lower than the cost of adapting to the requirements and customizing, to the extent possible, its commercial counterpart. Intimately related to open-sourcing is the issue of perception of OSS by the customer. Some customers may have a hard time assessing whether an open-source product has the same performance as the original, proprietary product (or a competing developer's product). SugarCRM is a provider of commercial open source customer relationship management software for companies with several deployment options to suit customer's security, integration and configuration needs. They offer two distinct products: Sugar Enterprise and Sugar Community Edition. The community edition allowed users to view and change the source as long as they follow the Sugar Public License (currently GPLv3). Unlike the community edition (which is free), the enterprise edition can only be acquired for a fee. Users of the Sugar software could perceive other significant differences between the enterprise and the community editions. The Community Edition lacks the functionality required to create teams or to assign access levels to the teams (Farber, 2005). The inability to keep users from deleting each other's contacts, schedules, leads, etc. makes the community edition relatively unfit for commercial use. The missing functionality is added in the Enterprise edition. SugarCRM employs full-time developers and the new features incorporated into the commercial product are generally missing from the free OSS product. Similar to the case of Sun's office products, the SugarCRM example indicates that the OSS counterpart of the proprietary software product is "crimped" in that it has reduced functionality. The concept of crimping is not new. Deneckere and McAfee (1996) describe it in the context of technology products. Our software case is similar in that the commercial developer incurs a cost to provide the (OSS ready) lower functionality product. However, the analogy breaks down when we consider that savvy OSS users have the freedom to re-establish the “crimped” functionality by re-writing the relevant code. The free availability of the source code allows the user to make changes to the product at a cost that we believe is lower than the cost of changing the functionality of the proprietary product.

4 THE MODEL

We consider the incentives for open-sourcing in a duopoly. The strategies of the firms in our model include a decision to open-source their output. Prior work by Sen (2007) and August et al. (2007) analyzed the competition between an OSS alternative and a closed-source commercial alternative. These authors showed that there may be benefits to open-sourcing when services are considered in conjunction with the software product. However, these papers do not consider the firm’s incentives to open-source their products in the absence of competition in the services market. In contrast, our model analyzes the competition between closed- and open-source products as an outcome of the firm’s strategic decisions to open source their products. Also, our model assumes that differentiation is embodied in the product itself and not due to differences in documentation and support services. While often software and service are not easy to disentangle, we gain some modeling flexibility by focusing only on the product market.

We model the variability of the fit of a software product to a firm’s existing systems and needs using Hotelling’s (1938) spatial model of product differentiation. Similar analyses of competition in open source environments abound; a recent example is the work of Gutsche (2005). As it is commonplace in the literature on product differentiation, we assume that the two firms are located at the ends of a line segment of unit length and share a measure of customers that we normalize to one without loss of generality. We also assume that the customers are continuously (and uniformly) distributed over the unit length segment and that a customer demands at most one product. We interpret the location of a customer relative to a firm as that customer’s ideal product requirement. A customer who is closer to a given firm incurs a smaller disutility to use the firm’s software than a customer who is farther away. As such, we model customers as heterogeneous in their fit for the products of the two firms. We assume that the two firms are symmetric in all relevant attributes, except location. To operationalize the notion of fit, we assume that customers incur a specific unit fit cost (τ) to use a given software. Given the symmetry of the firms, at equal prices, a customer prefers the firm that is closer. In this sense, customers located relatively close to a firm are “captive” and thus each firm does enjoy some degree of market power. Similar analyses in the literature have considered open source products as differentiated versions of their closed source counterparts, see e.g. Blitzer (2004) and the references therein.

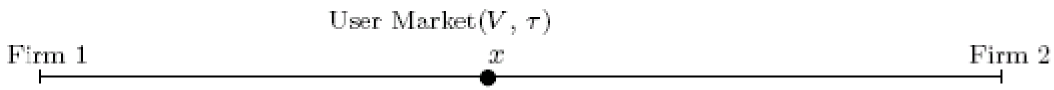

Figure 1. Market for a software product in duopoly.

Figure 1 provides a depiction of our main setup. Let *x* denote the distance of a customer from Firm 1 on the unit line. The customer could purchase the output of either firm. We assume that customers purchase at most one unit of output from either of the two firms. If the customer buys the closed-source product of Firm 1, the customer enjoys utility $V - \tau x - P_1$. If the customer buys the closed-source product of Firm 2, the customer enjoys utility level $V - \tau(1 - x) - P_2$.

Either firm has the option of open-sourcing its software product. We note that the closed-source product provides additional value through the use of proprietary features such as specialized tools, clip art, etc. Since these enhancements are available only to purchasers of the commercial product, the open-source product lacks these proprietary features. As such, the open-source product provides customers with less value than the commercial product. We denote this reduction in value by $_3$. In addition, we recognize that users of both products (commercial and open-source) gain additional value from the availability of the source-code of the open-source product. It must be noted that the commercial and the open-source products share the similar code-base. Hence, the availability of the source code allows all users to inspect the source code and identify bug fixes or develop enhancements that are available to all users. We denote this increase in value by$_1$. Accordingly, we assume that the value of the commercial product to a customer is $V+_1$. Letting $_2=_3_1$, the value of the OSS product becomes V_2. We assume that **Σφάλμα!** Note that the term $_1+_2$ represents the difference in value associated with purchasing the commercial product over its open-source counterpart. We view the open-sourced product as more customizable than the closed-source product. Since the OSS product is more easily customizable than the closed-source product, we assume that a customer's fit cost for the open-source product is where 01. It follows that a customer located at distance x in product space from the first firm enjoys utility levels $UC_1=(V+_1)xP_1$, and $UC_2=(V+_1)(1x)P_2$ if the customer buys the closed-source product from Firm 1 and Firm 2, respectively. If the customer chooses the open-source alternative of either firm, the customer nets utility level $UO_1=(V_2)x$, or $UO_2=(V_2)(1x)$. Implicit in our definition of open-sourcing is that the OSS products are offered free of charge by the two firms. We take as given in our model, without loss of generality, that customers have the ability to costlessly install and use the open-source products.

Before analyzing the possible outcomes in market configurations involving open-sourced products, we note that our analysis focuses only on those situations where all customers in the market are served prior to the firms' decision to open-source. The parameters of our model can be chosen so that the two firms are each local monopolies. In such situations, open sourcing by a firm may result in an increase in the market share of the commercial product, and indeed open sourcing may result in higher profits. To see this, note that the two firms in our model are local monopolies prior to choosing their open-sourcing strategy when $V<$. Intuitively, the higher the fit cost, the more customers become captive to the firm that is closest to them. If the fit cost is high relative to values, some customers would forgo purchases altogether, and thus a firm's pricing decision has no effect on the other firm's profit. It is easily shown that a firm's profit in a local monopoly configuration is equal to $V^2/(4)$. Open sourcing in such situations may increase the market share of a firm's commercial product. In particular, when (i.e, when the difference between the value of the open-source and the commercial products is small relative to the reduction of fit cost as a result of open-sourcing), the market share of each firm's commercial product is less than 1/2, so the two firms do not compete head-to-head with their commercial products. Furthermore, whenever $_1+_2$ also satisfies $_1+_2>V\sqrt{,1}$, that is, when the difference in value between the closed- and open-source products of a firm is large relative to value prior to the release of the open-source version, it can be shown that a firm's profit increases as a result of open-sourcing. However, we find these situations strategically less interesting because in equilibrium the open-sourcing decision of a firm that maintains its local monopoly status does not affect the profits of its opponent. We thus focus only on those situations in which open-sourcing has strategic implications. We discuss next the outcomes of the various modes of competition.

4.1 Duopoly with closed-source products

The simplest case in our environment is that of two firms competing with closed-source products. The profit-maximizing price be equal to $P_{10}=P_{20}=$. Intuitively, as customers incur a higher fit cost they become more captive, and thus the firms enjoy more market power and could afford to increase their prices. The equilibrium profits of the two firms are **Σφάλμα!**, consistent with the notion that more market power, indicated by higher customer fit costs, translates into higher profits for the two firms. Having established our benchmark, we turn next to an analysis of competition in which one of the firms also offers an open-source product.

4.2 Duopoly with only one firm offering an open-source product

Suppose Firm 1 decides to offer, free of charge, an open-source version of its commercial software product. The introduction of the open-source product has two main effects. First, some of Firm 1's customers would find it more profitable to choose the free open-source product. This effect works so as to reduce the profit of Firm 1. The second effect entails making Firm 2 compete with the free open-source product of Firm 1. This essentially works so as to dampen the effect on Firm 1's profits of changes in the price charged by Firm 2, and also as a way for Firm 1 to "steal" some of Firm 2's customers. It is important to note that the customer who is indifferent between acquiring the product of either firm is contemplating a choice between the free open-source product of Firm 1 and the commercial closed-source product of Firm 2.

Analyzing competition in the presence of an open-source product is somewhat complicated because, depending on the model parameters, three configurations are possible involving varying measures of customers who acquire some of the three products. We depict the most general situation in Figure 2 below.

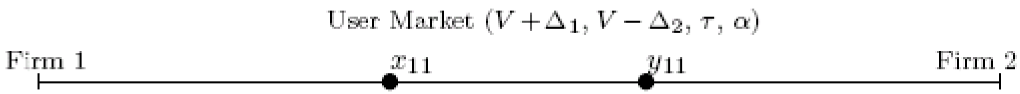

Figure 2. Market for software products in duopoly where Firm 1 has open-sourced.

The customer located at x_{11} is indifferent between the OSS product and the commercial product offered by Firm 1. The customer located at at y_{11} is indifferent between the OSS product (Firm 1's) and the commercial product offered by Firm 2.

Interior Solution

In this setting, all customers to the left of $x^{l,11}$ purchase Firm 1's closed-source commercial product, while all customers located to the right of $y^{l,11}$ purchase Firm 2's commercial product. The customers located between $x^{l,11}$ and $y^{l,11}$ find it optimal to use Firm 1's OSS product. It follows that a fraction $x^{l,11}$ of the customers purchase Firm 1's commercial product and that a fraction 1⍰$y^{l,11}$ of the customers purchase Firm 2's commercial product. The remaining fraction $y^{l,11}$⍰$x^{l,11}$of customers chooses Firm 1's OSS product. Solving for the two prices yields **Σφάλμα!** and **Σφάλμα!**. In equilibrium, these prices give rise to values of $x^{l,11}$ and $y^{l,11}$ that can be expressed as: **Σφάλμα!** and **Σφάλμα!**

In equilibrium the profits of the two firms are **Σφάλμα!** and **Σφάλμα!**

We need to ensure that, according to our assumption, the parameters of our model are chosen so that $0<x^{l,11}<y^{l,11}<1$. It is readily verified that, given our choice of parameters, $x^{l,11}>0$ and $y^{l,11}<1$. To ensure sure that $x^{l,11}<y^{l,11}$, we require that:

Σφάλμα! (1)

When condition 1 is satisfied, we term the segment of customers who choose the free OSS version "customer loss."

No Customer Loss (y_{11}□x_{11})

If condition 1 is not satisfied, all customers prefer Firm 1's commercial product to its free OSS version available. Intuitively, (1) is more likely to be violated if □$_1$ or □$_2$ – or both □$_1$and □$_2$– are relatively high, implying that the inherent value of the additional features offered in the commercial version is sufficiently higher than in the free OSS version. Firm 1's commercial product still benefits from the release of the open-source product (perhaps through a better management of code errors). Let superscript *II* denote this region. Straightforward calculations yield equilibrium prices chosen by the two firms that can be expressed as **Σφάλμα!** and **Σφάλμα!**. The equilibrium profits of the two firms can be written as:

Σφάλμα! (2)

and

Σφάλμα! (3)

Firm 2 is driven out of the market (x_{11}□1)

We use superscript *III* to indicate the parameter region where x_{11}□1. It can be easily checked that when the following condition holds, Firm 2 can no longer compete and Firm 1 becomes a monopoly:

□$_1$□3□ (4)

To maximize its profit Firm 1 chooses price $P^{III,11}$=□$_1$□□ and has profit □^{III,11}=□$_1$□□.

Having exhausted the set of possible outcomes when one of the firms opens up its source code, we turn to an analysis of competition when both firms offer an open-source product.

4.3 Duopoly with open-source products

When both firms decide to open their products, there are four products in the market. The most general market situation is depicted in Figure 3 below in which non-zero measures of customers choose to purchase one of the four products.

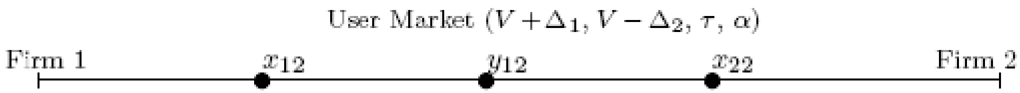

Figure 3. Market for software products in duopoly where both firms have open-sourced

In Figure 3, the customer at x_{12} is indifferent between the OSS product and the commercial product offered by Firm 1, while the customer located at distance y_{12} from Firm 1 is indifferent between the two OSS products. In addition, the customer located at x_{22} is indifferent between Firm 2's OSS product and the commercial product of Firm 2.

Interior Solution

As above, we start with an analysis of the situation in which non-zero measures of customers choose each of the four products. Let superscript *I* denote the corresponding parameter region. In this setting, all customers who are located on the left of $x^{I,12}$ purchase Firm 1's closed-source commercial product and all customers located on the right of $x^{I,22}$ purchase Firm 2's commercial product. The customers who are located between $x^{I,22}$ and $x^{I,12}$ use either Firm 1's or Firm 2's free OSS product. Two firms' profits:

$p^{I,12}=p^{I,22}=(\text{?}_1+\text{?}_2)/2$. (5)

Hence, the equilibrium profits of the two firms can be written as:

Σφάλμα! (6)

Note that since the assumed solution entails non-zero measures of customers that use any of the four products, we need $x^{I,12}<y^{I,12}<x^{I,22}$. Since the firms are symmetric, this translates into a single condition involving the two ?'s. It can be checked that if the following condition is to be satisfied in order for the parameters to yield such a solution:

$\text{?}_1<(1\text{??})\text{???}_2$. (7)

No Customer Loss ($x_{22}=x_{12}$)

In this case, the two firms compete head-to-head with their closed-source commercial products. As above, we maintain the assumption that the release of the open-source product increases the value that customers derive from using the commercial version of a product, even though no customer could gain utility from using an open-source product. It turns out that, with or without this assumption, the equilibrium has the same properties as the equilibrium that we analyzed in the benchmark case above (so that $x^{II,22}$ is equal to 1/2 and the profits of the two firms are equal, $\text{?}^{II,12}=\text{?}^{II,22}=\text{?}/2$).

Having established the outcome of competition in all possible situations in our model, we turn next to an analysis of the incentives that firms may have to open-source their products.

5 OPEN-SOURCING EQUILIBRIUM

We first describe the sequential-move game between our two firms. The game proceeds as follows: In the first stage, the firms independently and simultaneously choose whether or not to release open-source versions. In the second stage, upon observing their opponent's open-sourcing decision, the firms, independently and simultaneously, choose their prices to maximize profit. Our equilibrium concept is subgame perfection (see Selten, 1975). A strategy profile for each of the two players is a subgame perfect equilibrium if it is an equilibrium in any of the subgames of the original game. We find the subgame perfect equilibria of our game using backward induction. We start with the second stage of the game. Depending on the firms' actions in the first stage, there are four possible open-sourcing configurations. Only three of which are distinct, due to symmetry. The optimal pricing decisions and payoffs in each of these second stage configurations are discussed in Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. The three main cases of Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.** provide the necessary payoff values for the first stage problem. Thus, we can evaluate the first-stage equilibrium outcomes using the payoffs we deduced in Section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. The profit of each of the two firms when no firm releases an open-source version we denote by $\text{?}_{neither}$. The profit of each of the two firms when both firms release open-source versions is denoted by ?_{both}. In the asymmetric case when one of the firms releases an open-source version, we denote by ?_{self} the profit of the firm that released the open-source version and by ?_{rival} the profit of its opponent. Table 1 summarizes the payoffs that correspond to the first-stage actions of the two firms.

The equilibrium outcome can be found by inspecting the firms' payoffs. The outcome of competition depends on the choice of the model's parameters since we have multiple solutions derived in sections

Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε. and **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. Thus, a different payoff structure may exist for different regions of the parameter space. There are four different symmetric payoff matrices to be considered.

	Firm Payoffs (Firm 1,Firm 2)	
	Firm 2	
	Closed	Open
Firm 1 Closed	$\square_{neither}$, $\square_{neither}$	$\square_{rival}$, $\square_{self}$
Firm 1 Open	$\square_{self}$, $\square_{rival}$	$\square_{both}$, $\square_{both}$

Table 1. Payoff Matrix Structure.

Relevant in the computation of equilibria is the ranking of the firms' payoffs in different competitive regimes. By choosing different values of the parameters of our model, the ranking of the profits that correspond to the first-stage actions of the two firms changes. Different equilibria obtain that correspond to the different ranking of the firms' payoffs. We explore the parameter space in terms of the value of $\square_1$, the incremental gain in the value of the commercial product brought about by the release of its open-source version, relative to the other parameters of the model. We find that the profits that result from the two firms' first-stage open-sourcing decisions can be ranked differently depending how the value of $\square_1$ compares to the other parameters of the model.

We start by assuming that $\square_2$ and $\square$ satisfy $\square_2<\square$. We turn to discuss next the ordering of the payoffs in the various parameter regions. The ordering of the payoffs is summarized in Table 2. We also relegate the definition of the cutoffs that define each region to an appendix that is available from the authors upon request.

Order of Payoffs for Different Regions		
Region	**A**	$\square_{neither}\square\square_{rival}\square\square_{self}=\square_{both}$
Region	**B**	$\square_{neither}\square\square_{self}=\square_{both}\square\square_{rival}$
Region	**C**	$\square_{neither}=\square_{both}\square\square_{self}\square\square_{rival}$
Region	**D**	$\square_{self}\square\square_{neither}=\square_{both}\square\square_{rival}$

Table 2. Order of Payoffs.

Given these parameter regions and the ordering of the firms' payoffs in each region, we can finalize our equilibrium analysis. Figure 4 summarizes the equilibria in each of the regions of the parameter space that correspond to Table 2. We note that multiple equilibria co-exist in some of the regions of the parameter space. In region A, since $\square_{neither}\square\square_{self}$ and $\square_{rival}\square\square_{both}$, the firms' dominant first-stage action is not to release an open-source version. Regardless of its opponent's action, each firm is better off with a closed-source product. Therefore, in this case (the benchmark discussed in section **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**), in the unique equilibrium the two firms do not release open-source versions.

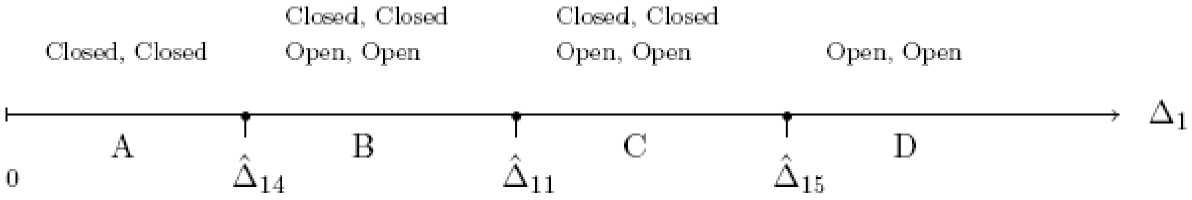

Figure 4. The equilibria with respect to $□_1$ *when* $□_2$□□.

The equilibria that correspond to parameters that fall in region B also contain outcomes in which the two firms do not release open-source versions. No firm would consider a release of an open-source version if its opponent were not to release an open-source version (since in this region $□_{neither}$□□$_{self}$). However, not releasing an open-source version is no longer the best action irrespective of the opponent's open-sourcing decision. Given that the other firm has an open-source product, the best response would be to release an open-source product as well, $□_{rival}$□□$_{both}$. Thus, opening the source code can also be part of the equilibrium. However, both firms are better off in the equilibrium that does not involve the opening of source code.

As above, in region C, there are two equilibria in which the firms either release or do not release open-source versions. Unlike the situation that arises when the model's parameters fall within region B, in region C the firms' payoffs in both equilibria are the same (so the equilibrium that involves releasing an open-source version is no longer payoff dominated).

Inspection of the payoffs in Table 2 indicates that, when the model's parameters are in region D, a firm could profitably and unilaterally open up its code. In this region, $□_{self}$□□$_{neither}$ and $□_{both}$□□$_{rival}$. Thus, irrespective of the action of its opponent, a firm's best first-stage action is to open yo its code. Thus, the unique equilibrium has both firms releasing open-source versions.

It can be easily seen by inspecting the values of the cutoffs that as the value of $□_2$ increases relative to □, the two regions A and B decrease in size. When $□_2$□□ the regions I and II vanish. In that case, both firms open their source code in the unique equilibrium for all values of $□_1$.

6 CONCLUSIONS

In this paper we analyzed the conditions under which firms find it optimal to release open-source versions of their products. Conventional wisdom suggests that open-sourcing increases the size of the market. In turn, greater exposure allows firms to reap higher profits through either increased sales of complementary products (e.g., hardware) or through reduced future costs of maintaining and managing the software code. This explanation is incomplete and somewhat fallacious, as clearly greater profit increases could be achieved through limited releases of the source code or through free distribution of the closed-source product. Recent research has considered the incentives for open-sourcing in connection with a complementary services market. While in today's business environment the software and services markets are hard to disentangle, we gain some insight by focusing only on the software product market. The main driving force of our model is the impact of open-sourcing on the customers' values. Open-source versions tend to provide less functionality than their commercial versions. However, customers could find the open source product more valuable than it closed source counterpart as a result of the better customization opportunities. In our model, the "crimped" product competes head-to-head with the products of the competing firm. As a result, the release of an open-source version better insulates a firm from the pricing strategy of its opponent. All things equal, this implies that the firm that releases the open-source version has a competitive edge over its opponent. Clearly, the firm that unilaterally releases the open-source version increases its profit, provided that it can maintain its customer base. If there is customer loss (i.e., when the release of the open-source version causes some of the releasing firm's customers to migrate to the free, open version) the outcome is influenced by the trade-off between higher prices and a smaller customer base. We have shown how these trade-offs affect the firms' decision to release open-source products. We identified parameter regions in which the equilibrium has the firms releasing open-source products. An important managerial implication is that open-sourcing is likely to occur when the difference in customer valuation between the proprietary and the open-source products is high relative to the fit cost. It is comforting to note that in most examples in which open-sourcing arises in a competitive environment, there is a sizable gap between the product valuations of the open- and closed-source products by customers. Not all firms in today's software business environment have included open-sourcing in their strategic repertoire. Another implication of our analysis shows that in order to stay competitive, software firms should consider a strategy to open-source their products in case a competitor chooses to do so.

The market for software products and services is under continuous evolution. Our model suggests that open- and closed-source software products are bound to co-exist. However, co-existence of the two types of products is more likely when the open-source product lacks significant features, or when the closed-source version becomes more valuable as a result of better code maintenance (like ridding the code of bugs). Also important is the ease with which customers could customize the open-source product. More facile customization of the open-source product implies that, all other things equal, an equilibrium is more likely to arise in which competitive firms release open-source versions of their software products.

References

August, T., H. Shin, T. Tunca. 2007. Open source software: Incentives and the market for services. Workshop on Infomation System Economics, (WISE). Montreal, Canada. http://www.wise2007.org/program.html.

Bhargava, H., V. Choudhary. 2001. Information goods and vertical differentiation. Journal of Management Information Systems 18(2) 89–106.

Bhargava, H., V. Choudhary. 2008. Research note - when is versioning optimal for information goods? Management Sci. 54(5) 1029–1035.

Blitzer, Juergen. 2004. Commercial versus open source software: The role of product heterogeneity in competition. Economic Systems 28 369–381.

Bonaccorsi, A., S. Giannangeli, C. Rossi. 2006. Entry strategies under competing standards: Hybrid business models in the open source software industry. Management Sci. 52(7) 1085 – 1098.

Casadesus-Masanell, R., P. Ghemawat. 2006. Dynamic mixed duopoly: A model motivated by linux vs. windows. Management Sci. 52(7) 1072 – 1084.

Deneckere, R. J., R. P. McAfee. 1996. Damaged goods. Journal of Economics and Management Strategy 5(2) 149–174.

Dewan, R., B. Jing, A. Seidmann. 2003. Product customization and price competition on the internet. Management Sci. 49(8) 1055–1070.

Economides, N., E. Katsamakas. 2006. Two-sided competition of proprietary vs. open source technology platforms and the implications for the software industry. Management Sci. 52(7) 1057 – 1071.

Farber, D. 2005. Commercial open source, a misnomer? downloaded on August 13 2007. http://blogs.zdnet.com/BTL/?p=1787.

Ghose, Anindya, Arun Sundararajan. 2005. Software versioning and quality degradation? an exploratory study of the evidence. Working Paper, CeDER-05-20, Center for Digital Economy Research, New York University .

Gutsche, Joerg. 2005. Competition between open source and proprietary software, and the scope for public policy. Proceedings of the First International Conference on Open Source Systems 196–199.

Hawkins, Richard E. 2004. The economics of open source software for a competitive firm - why give it away for free? NETNOMICS: Economic Research and Electronic Networking 6(2). http://slytherin.ds.psu.edu/hawk/research/opensource/opensource.pdf.

Jing, Bing. 2003. Market segmentation for information goods with network externalities. Information Systems Working Papers Series, Available at SSRN: http://ssrn.com/abstract=1281325 .

Jing, Bing. 2007. Network externalities and market segmentation in a monopoly. Economics Letters 95(1) 7–13.

Koch, S., G. Schneider. 2002. Effort, cooperation and coordination in an open source software project: Gnome. Information Systems Journal 12(1) 27–42.

Krishnamurthy, S. 2005. An analysis of open-source business models. Joseph Feller, Brian Fitzgerald, Scott A. Hissam, Karim R. Lakhani, eds., Perspectives on Free and Open Source Software. MIT Press, Cambridge, MA, 279–296.

MacCormack, A., J. Rusnak, C. Y. Baldwin. 2006. Exploring the structure of complex software designs: An empirical study of open source and proprietary code. Management Sci. 52(7) 1015 – 1030.

Mustonen, M. 2005. When does a firm support substitute open source programming? Journal of Economic and Management Strategy 14(1) 121–39.

Prial, J. 2004. Why ibm is open sourcing cloudscape as derby downloaded on August 13 2007. http://www-128.ibm.com/developerworks/db2/library/techarticle
/dm-0410prial.

Raghunathan, Srinivasan. 2000. Software editions: An application of segmentation theory to the packaged software market. Journal of Management Information Systems 17(1) 87–114.

Rossi, C., A. Bonaccorsi. 2005. Why profit-oriented companies enter the os field? intrinsic and extrinsic incentives. Fifth Workshop on Open Source Software Engineering. St. Louis, MO, USA.

Schiff, A. 2002. The economics of open source software:a survey of the early literature. Review of Network Economics 1(1) 66–74.

Selten, R. 1975. A reexamination of the perfectness concept for equilibrium points in extensive games. International Journal of Game Theory 4.

Sen, R. 2007. A strategic analysis of competition between open source and proprietary software. Journal of Management Information Systems 24(1) 233–257.

Serlet, Bertrand. 2004. Apple: Open-source pedigree will protect tiger downloaded on August 13 2007. http://news.com.com/2100-1016_3-5341689.html.

Sundararajan, A. 2004. Nonlinear pricing of information goods. Management Sci. 50(12) 1660–1673.

von Krogh, G., E. von Hippel. 2006. The promise of research on open source software. Management Sci. 52(7) 975–983.

Weber, T. A. 2008. Delayed multiattribute product differentiation. Decision Support Systems 44 447–468.