MANAGEMENT SCIENCE
Vol. 52, No. 7, July 2006, pp. 1072–1084
ISSN 0025-1909 | EISSN 1526-5501 | 06 | 5207 | 1072

DOI 10.1287/mnsc.1060.0548

Dynamic Mixed Duopoly: A Model Motivated by Linux vs. Windows

Ramon Casadesus-Masanell, Pankaj Ghemawat
Harvard Business School, Boston, Massachusetts 02163 and Instituto de Estudios Superiores de Empresa, Avenue Pearson 21, Barcelona 08034, Spain, {casadesus@gmail.com, pghemawat@hbs.edu}

This paper analyzes a dynamic mixed duopoly in which a profit-maximizing competitor interacts with a competitor that prices at zero (or marginal cost), with the cumulation of output affecting their relative positions over time. The modeling effort is motivated by interactions between Linux, an open source operating system, and Microsoft's Windows and consequently emphasizes demand-side learning effects that generate dynamic scale economies (or network externalities). Analytical characterizations of the equilibrium under such conditions are offered, and some comparative static and welfare effects are examined.

Key words: open source software; demand-side learning; network effects; Linux; mixed duopoly; competitive dynamics; business models
History: Accepted by Eric von Hippel and Georg von Krogh, guest editors; received September 1, 2004. This paper was with the authors 3 months for 1 revision.

1. Introduction

Open source software (OSS), particularly Linux, has attracted considerable interest recently.[1] What has perhaps sparked the most interest amongst researchers is the observation that since the provision of OSS and its continued improvement is a costly activity, it is made available publicly at a price (typically zero) that does not reflect the economic costs of the resources that developers and others expend on it. What, then, motivates the resource contributions to Linux and other OSS development efforts?

But conditional on the requisite resource contributions being made, there *is* a second question that is of interest as well: What competitive dynamics are introduced to markets by not-for-profit OSS initiatives and, in particular, how are they likely to fare relative to their for-profit proprietary competitors (e.g., Microsoft's Windows in the case of Linux)? Unfortunately, there has been relatively little analysis of this second question, despite its importance. This paper makes a start at remedying this state of affairs by analyzing a simple model that captures two key features of the competition between Linux (or, more generally, OSS) and Windows (or Microsoft and other for-profit providers of proprietary software): the dynamics of learning on the demand side and Windows' initial market power.

That said, the model is meant not to supply a literal representation of the interactions between Windows and Linux but, instead, to provide a stripped-down theoretical structure for weighing various claims about competition between these two models of software development. Common assertions about OSS that our theoretical analysis questions or qualifies will be discussed at greater length below but include the following notions:

1. Given faster demand-side learning with OSS development efforts, they will overtake or even oust proprietary development efforts from markets.
2. OSS is inherently incompatible with the protection of property rights.
3. Governmental commitments to promote OSS increase social welfare by increasing the viability of such development efforts and thereby lowering prices.
4. Forward-looking buyers tip market outcomes toward OSS and away from proprietary software. Some empirical evidence is also brought to bear on the identification or analysis of these claims.

Section 2 discusses prior studies on OSS. Section 3 sets forth the benchmark model of competition between OSS priced at zero and for-profit proprietary software. Section 4 extends the analysis to look at

[1] For a detailed discussion of Linux in 2004, see Ghemawat et al. (2004).

the effects of piracy, governmental commitments to promote OSS, and forward-looking buyers. Section 5 concludes.

2. Prior Research on OSS

Much of the prior literature on Linux/OSS focuses on how OSS development efforts are organized, particularly (as noted above) the satisfaction of the individual rationality or participation constraints of user-developers who are critical to learning on the demand side. For compactness, we will refer to this as the organizational strand of research on OSS.

The most focused substrand of the organizational research on OSS has assumed utility-maximizing (potential) users/developers and tried to derive various sorts of comparative static predictions about their developmental contributions. In early work of this sort, Thorn and Connolly (1987) used theories of the economics of public goods to argue that the rates and effectiveness of discretionary information sharing among employees in an organization would tend to decrease as participation costs increased and the size of the overall group increased, and the lower the value of information to participants and the greater the asymmetries in information values and benefits across participants.

More recent work in this line has pushed further with formalizing these insights and developing new ones. Thus, Kuan (1999) framed consumer choice between OSS and proprietary software as a make-or-buy decision, with the former option entailing a further decision about how much effort to exert contributing to the quality of software (a public good); Kuan concluded that the advantages of OSS were higher for programmers than nonprogrammers. She also inferred that if most high-paying users were also programmers, open source or community organization would be more likely (e.g., engineering tools or utilities), whereas if most high-paying users were nonprogrammers, proprietary or closed organization would be more probable (e.g., word processing programs, spreadsheets, and other products with a broad nonengineering market). Bessen (2002) also analyzed a self-selection model that had consumers helping to test and debug different variants of a complex product with many (interacting) features, of which only a fraction might be valuable to any particular user. He concluded that given OSS, individual users who placed a high enough value on the product would test and debug their own use product and that as long as costs were sufficiently low and product complexity sufficiently high, more use-product combinations would be tested with open source than under the proprietary case, a larger market would be served, and social welfare would be higher. Johnson (2002) analyzed a self-selection model with various informational imperfections and concluded that whether OSS development would increase when applications had a modular structure depended on whether the developer base exceeded a critical size; he also provided some finite and asymptotic results of effects of changing the population size of user programmers and tried to explain why certain useful programs do not get written. Xu (2002), with a variant of the same basic model, showed that decreasing development costs need not necessarily increase the amount of OSS software development.

A second substrand of the organizational literature has looked somewhat more broadly at whether OSS development efforts can be explained as the outcome of private cost-benefit analysis by user-developers or whether other, less conventionally economic motivations—e.g., altruism, participation in a gift economy/culture in which social status depends on what one gives away, or even a visceral dislike of Microsoft—need to be invoked to explain private provision of the public good of improved software quality. Thus, Lerner and Tirole (2002) argued that conventional cost-benefit analysis may be sufficient once one accounts for benefits related to career concerns and ego gratification (stemming from peer recognition) that induce an incentive for an individual to signal high quality through participation in OSS development. They also suggested that signaling incentives might be strengthened in OSS environments by better performance measurement (given the care with which individual contributors tend to be credited), full initiative by (empowered) programmers, and greater labor market fluidity/knowledge portability, and that other factors favorable to open source include modularity and the existence of "fun" challenge and credible leadership. In a similar vein, Lakhani and von Hippel (2002) looked—in the context of Apache—at the performance of the mundane but essential task of providing high-quality field support (to overcome either defects in the product or deficiencies in the user's understanding) and concluded that the need for explanations such as altruism and even (delayed) signaling benefits was limited by the inference that most of the effort information providers expended could be understood in terms of the direct rewards they derived immediately, that is, in terms of learning for themselves.

A third, more miscellaneous substrand of the organizational literature on OSS has taken the even broader, more inductively oriented approach of describing the actual organization of such software development efforts. Thus, three of the core contributors to the development of Apache (Mockus et al. 2000) built on their experience of that project, as well as the history of others, including Linux, to offer some rough numerical requirements for their organization: a core group of developers, no larger than 15 people, to control the approval and integration of new/modified code into the ongoing stream of "official" releases—a process more centralized than most others in OSS development—and to create more than 80% of new functionality, strict code ownership policies to disaggregate OSS efforts that would otherwise be too large, a group larger by an order of magnitude than the core group to repair defects, and a group another order of magnitude larger yet to report problems. The governance of such projects and, specifically, the legal tactics employed to protect the public property that they create, are discussed by O'Mahony (2003).

But these and other organizational issues surrounding open source, while interesting, are far from the only ones of interest. A second distinct set of issues concerns the outcomes from and implications of competition between OSS priced at zero (e.g., Linux) and for-profit proprietary software (e.g., Windows). This competitive strand of research on OSS is much less developed than the organizational strand discussed above, even though the rhetoric about it—open source as innovation savior versus destroyer—can get quite heated. While papers that focus on the relative efficiency of open and proprietary development models are obvious reference points, they generally neglect interactions between the two models and the effects of moving late versus early in determining competitive outcomes (Kogut and Metiu 2001). Still, some specific analytical contributions *are* worth noting. Bitzer (2000) proposed a simple model to make the point that the less the heterogeneity in product space between OSS and proprietary software, the more likely competition is to collapse prices below the levels necessary to support the proprietary software developer's (higher) average costs and lead it to abandon its development efforts. Dalle and Jullien (2002) employed a simulation approach to establish that increasing returns associated with creativity and their (re)distribution toward end users could create global and local positive externalities strong enough to help Linux reverse current standardization on Windows 2000. And Schmidt and Schnitzer (2002) set up a simple, essentially static model of Hotelling-like horizontally differentiated competition between OSS and for-profit proprietary software and showed that within that setup, mandated procurement of the open source product would unambiguously reduce welfare. None of these papers, however, really embeds the competition between Linux and Windows in a dynamic model with demand-side learning.

3. A Model of Open Source vs. Proprietary Software

The most direct antecedents of our modeling effort are to be found in the literature on "mixed duopolies"—duopolies in which competitors have heterogeneous objective functions. This line of work began in the 1960s (Merrill and Schneider 1966), with De Fraja and Delbono's (1990) survey still providing a useful guide to the earlier literature. Since then, there has been substantial additional work in this area, much of it spurred by (partial) privatization (e.g., Anderson et al. 1997, Matsumura 1998). However, much of this work either focuses on homogenous goods markets (which, Cremer et al. 1991 point out, are likely to be particularly incompatible with mixed-market structures) or takes a static perspective on small-numbers interactions. Neither assumption is appealing in the context of Linux versus Windows.

To capture some of the stylized features of the interactions between Linux and Windows, we set up and analyze a model of vertically differentiated mixed duopoly in which the evolution of levels of vertical differentiation is driven, at possibly asymmetric rates, by demand-side learning.

This formulation implies an important link back to the literature on competitive learning by doing (Spence 1981) and, less directly, to the literature on network externalities. One of the distinctive features of this paper in the context of these two literatures is its focus on mixed rather than symmetrically profit-maximizing competitors. Another relatively unusual feature is the focus on demand-side learning, which increases benefits to users, instead of supply-side learning, which decreases costs. OSS development is thought to harness demand-side learning more effectively than traditional "closed" models by compressing development cycles, leading to the testing of more use combinations, and providing more of an incentive for users to report problems or fixes than "closed" models. Such quality-enhancing, demand-side learning effects have been compared to conventional cost-reducing supply-side learning curves, with their traditional industrial logic of cutting price, gaining scale, and reducing costs particularly rapidly (e.g., by the Boston Consulting Group). In the context of OSS, the virtuous cycle involves giving source code away, attracting users through performance advantages as well as zero prices, and drawing on users' learning and contributions to increase product quality rapidly. What is similar in the two contexts is the focus on increasing returns to scale; what is different is whether scaling up decreases costs or increases benefits to users. Thus, the model in this paper can be read as an attempt to extend the literature on competition in the presence of demand-side learning by doing to a duopoly structure in which objectives are mixed rather than symmetric.

We begin by specifying the demand side of the model. In each period t, a new cohort of potential users enters the market. We normalize the size of this cohort to 1. Let $y_i(t) \in \mathbb{R}_+$, $i \in \{W, L\}$ be the cumulative market share (or installed base) of operating system (OS) i at time t. Parameter $\delta \in [0, \infty]$ is the "rate of decay" or "death rate" of past sales. If $q(\tau)$ is the portion of individuals in time τ's cohort who buy Windows, then

$$y_W(t) = \int_0^t e^{-\delta(t-\tau)} q(\tau)\, d\tau. \tag{1}$$

Differentiating (1) with respect to t, we obtain

$$\dot{y}_W = q(t) - \delta \int_0^t e^{-\delta(t-\tau)} q(\tau)\, d\tau = q(t) - \delta y_W(t).$$

Notice that regardless of the value of δ, $y_W(t) \geq 0$ because $q(t) \geq 0$ for all t.

We assume that every individual in each cohort uses one and only one OS; he or she either buys Windows or downloads Linux for free. Thus, if $1 - q(\tau)$ is the portion of individuals in time τ's cohort who download and use Linux, then

$$y_L(t) = \int_0^t e^{-\delta(t-\tau)} (1 - q(\tau))\, d\tau$$

and

$$\dot{y}_L = 1 - q(t) - \delta y_L(t).$$

Let $y(t) \equiv y_W(t) - s y_L(t)$, where s is a scalar greater than 1. Let $\alpha_i(y(t))$ denote OS i's value by the cohort entering at time t. We refer to $\alpha_i(y(t))$ as OS i's technological trajectory. OS i's trajectory is a function of its cumulative market share, $y_i(t)$, and the competing OS's cumulative market share, $y_{-i}(t)$. While technological trajectories are exogenously given in the model, how far each OS travels down its trajectory is endogenous—the result of the dynamics of competition.

When $\delta = 0$ the *effect of past purchases* on the current value of Linux and Windows ($\alpha_i(y(t))$) never fades. Clearly, if $\delta = 0$ and $q(t) \geq a > 0$ for all t, then $y_W(t) \to \infty$. At the other extreme, when $\delta = \infty$ the effect of past purchases on the current value of Linux and Windows fades immediately. In this case we have $y_W(t) = q(t) \leq 1$ for all t, and market shares never cumulate.

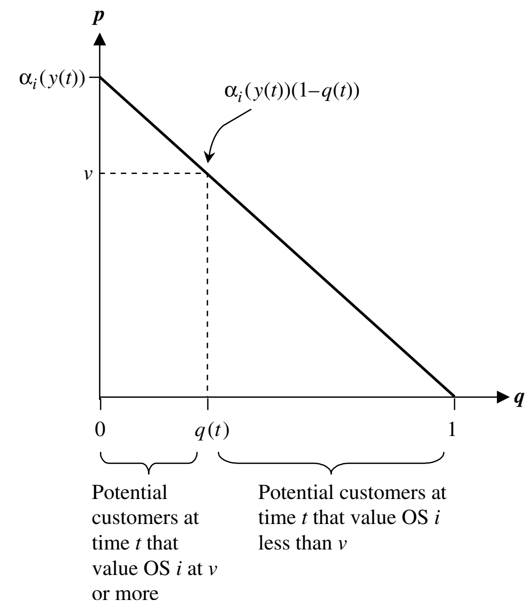
Assumption 1. *We assume linear demand functions: Windows' value to customer $q \in [0, 1]$ is*

$$\alpha_W(y)(1 - q). \tag{2}$$

Similarly, let the value of Linux be

$$\alpha_L(y)(1 - q).$$

Figure 1 **OS i Demand at Time t**



See Figure 1 for a graphic representation.

Assumption 2. $\alpha_i(y) \geq 0$; *that is, OS i's value is always positive or zero. Thus, technological trajectories are assumed to be bounded below. This guarantees that the demand functions are well defined. We further assume that* $\lim_{y\to-\infty} \alpha_W(y) = \lim_{y\to\infty} \alpha_L(y) = 0$; *that is, OS i's value approaches zero if virtually everybody uses the other OS. We do* not *assume that there is an upper bound on the maximum value created by each operating system.*

Assumption 3. $\partial\alpha_i(y)/\partial y_i > 0$; *that is, OS i's value increases with OS i's cumulative market share. This captures two kinds of effects: The more people use (or have used) a given OS, the more feedback is likely to have been provided for improvement. In the case of open source projects, users can make improvements directly on the code. In the case of proprietary software, users can call up or e-mail the software developer with suggestions. In addition, the larger $y_i(t)$ is, the more complements are likely to be available for OS i. Complements' availability raises the value of the OS.*

The assumption also implies that $\partial\alpha_i(y)/\partial y_{-i} < 0$; that is, OS i's value decreases as the cumulative market share of the competing OS increases. Again, this is related to the attention and effort that third-party developers devote to creating new and improving old software and hardware. An OS is more likely to get developers' attention if its cumulative market share is relatively large. The larger $y_{-i}(t)$ is (holding $y_i(t)$ constant), the less effort is devoted to developing complements for OS i, and vice versa. This reduction jeopardizes the value of OS i because bugs are not fixed as often, programs are not updated, new software and hardware may not work/communicate as well with existing software and hardware, unforeseen compatibility issues are more likely to arise, etc.

Assumption 4. *Let y^0 be the value of y for which both Linux and Windows are perceived as equally valuable. That is, $\alpha_W(y^0) = \alpha_L(y^0)$. (Assumptions 2 and 3 imply that y^0 exists and is unique.) Then $d^2\alpha_W(y)/dy^2 \le 0$ and $d^2\alpha_L(y)/dy^2 \ge 0$ for $y > y^0$. This assumption says that increases in y_W when $\alpha_W(y) > \alpha_L(y)$ have a decreasing marginal effect on the value of Windows and that increases in y_L have an increasing marginal effect on the value of Linux.*

As mentioned above, we assume that everything else being constant, as y_i grows, the vertical intercept in the demand function of OS i ($\alpha_i(y)$) also increases. Thus, as the accumulated market share of OS i grows (holding constant y_{-i}), the value of OS i grows for *all* new entrants.

Within this context, we capture Linux's superior demand-side learning with the assumption that $s > 1$. Recall that $y(t) \equiv y_W(t) - sy_L(t)$. Formally, s is the absolute value of the derivative of y with respect to y_L. Because $s > 1$, increases in y_L have a more positive impact on perceived quality of Linux than the negative impact of comparable increases in y_W. Parameter s has two complementary interpretations: the differential in demand-side learning between Linux and Windows and the differential strength of network externalities due to the availability of complementary software. Thus, for a given level of network externalities due to complements, increases in s correspond to a strengthening in Linux's demand-side learning. In what follows, we analyze the dynamics of competition and its effects on cumulative market shares y_W and y_L.

The model is meant to supply a stripped-down structure with which to work rather than a more or less faithful representation of the actual interactions between Linux and Windows. For instance, while the model does pay explicit attention to learning from demand-side interactions and the possibility that learning rates might vary across open source and traditional development models, somebody immersed in the details of learning in open source environments will probably still find its treatment here underdeveloped. Maintaining (near) tractability is the most compelling reason for not further complicating the treatment of demand-side learning in particular and the model in general. And despite the resultant disjunction between the real circumstances of the interactions between Linux and Windows and the set-up of the model, we think that the model's sparseness brings into sharp focus certain effects that should, at a minimum, be kept in mind in the real-world context (as opposed to not being recognized at all).

We distinguish between two cases: one in which Microsoft is a monopolist and another in which it is a duopolist, competing to sell Windows against Linux.

A Monopoly Benchmark

In a monopolistic market structure, there is no substitute for Windows, and all members of the entering cohort are willing to pay something (even only a small amount) for Windows. Inverse demand follows directly from Equation (2).

Let r be Microsoft's discount rate and assume that marginal cost of an extra copy of Windows is zero. Then the monopolist's problem is

$$\begin{aligned} \max_{p(t)} \quad & \int_0^\infty e^{-rt} q(t)p(t)\,dt \\ \text{subject to} \quad & \dot{y}_W = q(t) - \delta y_W(t) \\ & p(t) = \alpha(y_W(t))(1 - q(t)) \\ & \alpha_W(0) > 0 \\ & p(t) \ge 0. \end{aligned}$$

Lemma. *The unique stable steady-state y_W^{ss} satisfies*

$$\frac{\alpha'_W(y_W^{ss})}{\alpha_W(y_W^{ss})} = \frac{(2\delta y_W^{ss} - 1)(r+\delta)}{\delta y_W^{ss}(1 - \delta y_W^{ss})}.$$

Furthermore, $1/(2\delta) \le y_W^{ss} \le 1/\delta$.

Proof. The proof involves using phase diagram analysis to graphically represent the path leading to the steady state and show that the path is optimal by checking Mangasarian's sufficient conditions.[2]

The phase diagram in Figure 2 summarizes the analysis (m is the Hamiltonian multiplier—see proof):

Because the value of Windows increases with the installed base y_W, the monopolist sets prices below the myopic profit-maximizing price ($p = \alpha_W(y_W)/2$) to enlarge the accumulated market share. The proportion of individuals in every cohort who buy Windows is always larger than 1/2. This allows the monopolist to charge higher prices in the future.

The following comparative statics are of interest:

Proposition 1. (a) $dy_W^{ss}/d\delta < 0$; (b) $dy_W^{ss}/dr < 0$; (c) $\lim_{\delta\to 0^+} y_W^{ss} = \infty$; (d) $\lim_{\delta\to\infty} y_W^{ss} = 0$; (e) $\lim_{r\to\infty} y_W^{ss} = 1/(2\delta)$.

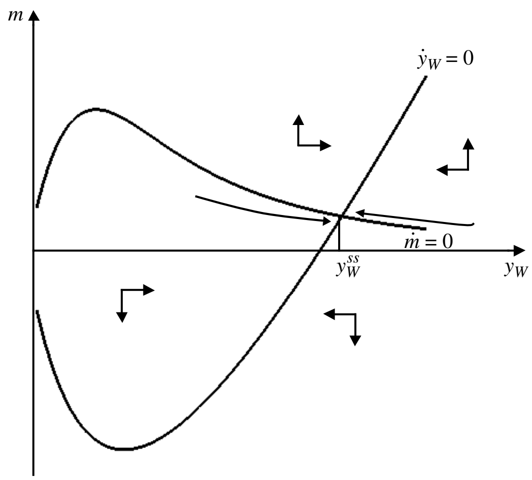
That is,

(a) *An increase in the rate of decay of past sales has a negative effect on Microsoft's ability to build the installed base. Thus, as δ increases, the steady-state value of Windows $\alpha_W(y_W^{ss})$ is reduced.*

(b) *The more myopic Microsoft is, the lower the long-term steady-state installed base is.*

[2] All proofs can be found in the online supplement to this article on the *Management Science* website at http://mansci.pubs.informs.org/ecompanion.html.

Figure 2 **Phase Diagram for Monopoly Benchmark**



(c) *As the decay rate approaches* 0, *the steady-state installed base increases without bound*.

(d) *As the death rate approaches infinity, it is harder and harder for Microsoft to build an installed base. As a consequence, the steady-state installed base approaches zero*.

(e) *If Microsoft were only concerned about the present, then prices would be set at the short-term profit-maximizing level* $\alpha_W(y_W)/2$. *In this case,* $q(t) = 1/2$ *for all t and accumulated market share at the steady state would be*

$$\lim_{t\to\infty}\int_0^t \frac{1}{2}e^{-\delta(t-\tau)}\,d\tau = \frac{1}{2\delta}.$$

We turn now to the case where both OSs are available.

Duopolistic Competition

When both Windows and Linux are available and Windows is sold at price p, the customer who is indifferent between the two, q, is found by solving the following equation:

$$\alpha_W(y)(1-q) - p = \alpha_L(y)(1-q).$$

Thus, inverse demand is

$$p = \beta(y)(1-q),$$

where $\beta(y) \equiv \alpha_W(y) - \alpha_L(y)$ indicates the value difference between Windows and Linux.

Notice that in this model, customers in every new cohort are assumed to be myopic in the sense that they buy the OS that is immediately most valuable to them (after subtracting price). This can also be interpreted as users thinking that the product has little durability (because of rapid innovation) and that they are too insignificant to affect the future value of an OS with their purchasing decisions. The extension to the case of forward-looking buyers is developed in §4.

We assume that at time $t = 0$ Windows is perceived as more valuable than Linux. That is, $\beta(y(0)) > 0$. Note that when Microsoft sets $p = 0$ and $\beta(y) > 0$, demand for Windows is 1 (the size of the entering cohort). Thus, as long as $\beta(y) > 0$, Microsoft can capture the entire new cohort by setting $p = 0$. However, in this case profit is also 0. The customer who most values Windows is willing to pay no more than $\beta(y)$, an increasing function of market share y_W, for it. In contrast, if market shares are such that $\beta(y) \le 0$ (Linux's perceived quality is at least as large as that of Windows), then nobody is willing to pay anything for Windows.

Because

$$y_W(t) = \int_0^t e^{-\delta(t-\tau)}q(\tau)\,d\tau,$$

$$y_L(t) = \int_0^t e^{-\delta(t-\tau)}(1-q(\tau))\,d\tau$$

and

$$y(t) \equiv y_W(t) - s y_L(t),$$

we have

$$\dot{y} = q(t) - s(1-q(t)) - \delta y.$$

Let r be the discount rate and assume that marginal cost of an extra copy of Windows is zero. Microsoft's problem is:

$$\begin{aligned} \max_{p(t)}\ & \int_0^\infty e^{-rt}q(t)p(t)\,dt \\ \text{subject to}\ & \dot{y} = q(t) - s(1-q(t)) - \delta y \\ & p(t) = \beta(y(t))(1-q(t)) \\ & \beta(y(0)) > 0 \\ & p(t) \ge 0. \end{aligned}$$

We can therefore use standard phase diagram analysis to examine the long-run dynamics of competition. Recall that y^0 is the unique y such that $\beta(y) = 0$.

Proposition 2. *When* $y^0 < 1/\delta$, *Microsoft fights for market share and both Linux and Windows coexist in the long-run, steady-state equilibrium, regardless of the value of demand-side learning s. When* $y^0 > 1/\delta$, *Microsoft is forced to leave the market*.

The two steady states y^0 and y^{ss} are characterized by

$$\beta(y^0) = 0$$

and

$$\frac{\beta'(y^{ss})}{\beta(y^{ss})} = \frac{(r+\delta)(s-1+2\delta y^{ss})}{(s+\delta y^{ss})(1-\delta y^{ss})}.$$

When $y^0 < 1/\delta$, there are two steady states: y^0 and y^{ss}. Steady state y^0 is unstable, while y^{ss} is a saddle point. When $y^0 > 1/\delta$, the unique steady state y^0 is a saddle point.

Because Windows's period t demand is $q(t) = 1 - p(t)/\beta(y(t))$, Microsoft can guarantee as large a period market share as it desires by setting $p(t)$ sufficiently close to zero. However, when $y^0 > 1/\delta$, the death rate δ is so large relative to y^0 that even if Microsoft set the price at zero in every period, it would not be able to sustain a sufficiently large installed base to maintain the quality difference necessary to remain a viable competitor for the long term. Formally, if Microsoft sets $p(t) = 0$ in every period, its *period* market share is 1. If it follows this pricing policy in every period, its accumulated market share in the long-run steady state will be

$$\lim_{t\to\infty} \int_0^t 1e^{-\delta(t-\tau)}\, d\tau = \frac{1}{\delta}.$$

Notice that if δ is large, the long-run installed base will be small. In fact, if the rate of decay of past sales δ is sufficiently large, Microsoft cannot maintain (in the long term) a sufficiently large accumulated market share to remain viable. Thus, if y^0 is large relative to $1/\delta$, Microsoft will be *eventually* forced to exit. Notice that Linux's differential demand-side learning s has no bearing on this result.

When the decay rate δ is low, Microsoft is *not* pushed out of the market by the free, open source operating system, *regardless* of the speed of demand-side learning (the value of s) and *regardless* of the difference in potential maximum values (i.e., $\alpha_W(1/\delta) - \alpha_L(-s/\delta)$).[3] In particular, when $y^0 \le 0$, Windows never exits. For the rest of the paper, we will assume that $y^0 < 1/\delta$; that is, we will assume that we are in the (more interesting) case in which Microsoft can fight to stay in if it wishes to do so.

The following comparative statics are of interest:

Proposition 3. (a) $dy^{ss}/d\delta < 0$ *and* (b) $dy^{ss}/dr < 0$. *That is,*

(a) *The larger the decay rate is, the smaller the steady-state difference in accumulated market shares is between Windows and Linux.*

(b) *The more myopic Microsoft is, the more similar are the steady-state perceived qualities of Windows and Linux. And the more patient Microsoft is, the greater the long-term perceived quality advantage of Windows.*

Because a monopolist can always set prices at the duopoly level, Microsoft's steady-state profit is lower in a duopolistic industry structure. Clearly, at monopoly prices some customers prefer to get Linux free. Microsoft takes this into account and lowers prices.

[3] If in every period the entire cohort buys Windows, we have $\lim_{t\to\infty} y_W(t) = \lim_{t\to\infty} \int_0^t 1e^{-\delta(t-\tau)}\, d\tau = 1/\delta$ and $\lim_{t\to\infty} y_L(t) = 0$. Therefore, $\lim_{t\to\infty} y(t) = 1/\delta$. As a consequence, the *maximum potential value of Windows* is $\alpha_W(1/\delta)$. Likewise, if the entire new cohort in every period uses Linux, we have $\lim_{t\to\infty} y_L(t) = \lim_{t\to\infty} \int_0^t 1e^{-\delta(t-\tau)}\, d\tau = 1/\delta$ and $\lim_{t\to\infty} y(t) = -s/\delta$. As a consequence, the *maximum potential value of Linux* is $\alpha_L(-s/\delta)$.

The following proposition is central to our inquiry on the long-run competitive dynamics between Linux and Windows.

Proposition 4. *Windows stays in the market even if Linux's learning rate increases without bound. Formally,*

$$y^0 < \lim_{s\to\infty} y^{ss}.$$

That is, even if the sensitivity of y to y_L grows to infinity, there is a lower bound on y^{ss} that is strictly greater *than y^0. As a consequence, if Windows is ahead, it will stay ahead* regardless *of the value of s.*

Contrary to earlier results on competition with network externalities, in our model the failure of Linux to replace Windows is not due to switching or search costs (see, for example, David 1985). Furthermore, the failure of a higher potential quality OS to eventually win the market is not related to demand-side coordination issues (as in Farrell and Saloner 1985, 1986) because demand coordination does not raise the instantaneous value of the OS on which buyers coordinate. In our model, without Microsoft's forward-looking pricing strategy, Windows would inevitably wind up being replaced by Linux (whenever $s > 1$). Instead, it is Microsoft's strategic actions that generate the result. The market does not fully tip to Linux because Microsoft's strategic decisions prevent that from happening.

More generally, much of the network externalities literature focuses on one profit-maximizing firm versus another or, if looking at demand-side issues, assumes competitive supply. Instead, we look at asymmetric/mixed-mode competition. In particular, the interaction between for-profit and not-for-profit entities seems particularly interesting in the context of knowledge development/innovation. In addition, our model features explicit dynamics, not a two-period abstraction as in much of the literature on network externalities.

Proponents of open source have long argued that the learning advantage of the "swarm" will prove decisive in its interactions with commercial software in many categories. This perspective can be traced back to Raymond's (1997) famous assertion in *The Cathedral and the Bazaar* that "with enough eyeballs, all bugs are shallow." More recently, it has been elevated to best-seller status in Surowiecki's (2004) *The Wisdom of Crowds*, which celebrates Linux (among others) and argues that companies should adopt similar decisionmaking models: "Any major decision should be taken by as large a group of managers as is logistically feasible" (*The Economist* 2004). Our model suggests that the swarm hypothesis is flawed: For a wide range of reasonable parameterizations, Windows stays in the market indefinitely and continues to lead Linux even though the latter benefits from lower (zero) prices that boost market share and therefore experience, as well as faster learning from a given amount of experience as a result of different patterns of demand-side interactions. One might think of this result as exposing one of the competitive disadvantages of the swarm or a bazaar-based approach: A company can strategize in terms of thinking nonmyopically (with myopic behavior, Windows would quickly be forced out in the model), whereas it is not reasonable to expect a swarm to do so.

The focus on modeling the effects of learning on the demand side has induced some simplifications on the cost side. The baseline model in this paper assumes that marginal costs are zero for both competitors. Cases with symmetric constant marginal costs are structurally equivalent. The effects of cost asymmetries are less trivial. In particular, if Microsoft incurs a per period fixed cost of development, that does increase the likelihood that it will eventually be forced out of the market. However, it is also easy to check that Microsoft stays in the market longer than it would if it were behaving myopically, that is, like Linux. So strategizing does continue to increase the viability of Windows relative to Linux.

Welfare

The result that Microsoft's steady-state profit is lower in a duopolistic industry structure suggests that a duopoly is likely to dominate Microsoft's monopoly in terms of total welfare generation. Also, if Linux's potential quality is above that of Windows ($\alpha_L(-s/\delta) \geq \alpha_W(1/\delta)$, see footnote 3), one would expect that Linux's monopoly should dominate Windows' monopoly and the duopoly. In this section we analyze the welfare implications of each industry structure and show that neither of these claims is necessarily true.

We begin by analyzing welfare for the new cohorts entering after the steady state has been reached. That is, we look at the *flow* of total welfare at the steady state. Using the equation of motion ($\dot{y}_W = q_W^{ss} - \delta y_W^{ss}$) and the demand function, the steady-state quantity, price, and total surplus are derived immediately:

$$q_W^{ss} = \delta y_W^{ss}, \qquad p = \alpha_W(y_W^{ss})(1-\delta y_W^{ss}),$$

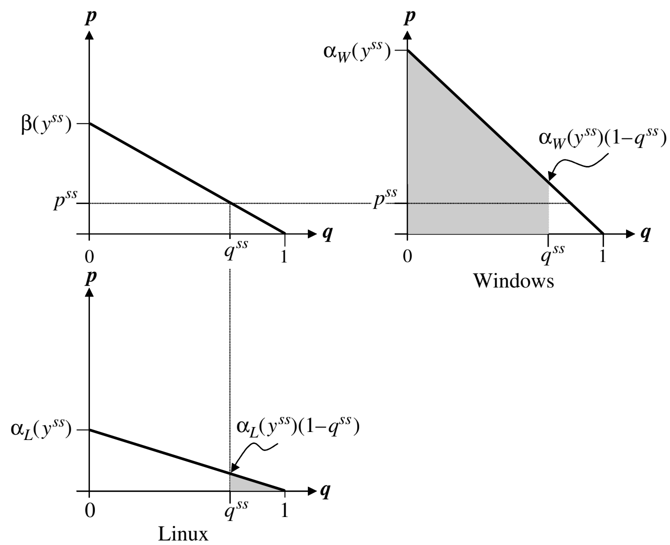
and

$$TS_W^{Monopoly} = \frac{\alpha_W(y_W^{ss})}{2}\delta y_W^{ss}(2-\delta y_W^{ss}).$$

Similarly, if Linux was a monopolist, we would have that $y_L(t) \to 1/\delta$ as $t \to \infty$. Therefore, $\alpha_L(y) \to \alpha_L(-s/\delta)$ and $TS_L^{Monopoly} = \alpha_L(-s/\delta)/2$.

Because $y_W^{ss} < 1/\delta$, a sufficient condition for Linux's monopoly to be socially more desirable than Windows' monopoly is $\alpha_L(-s/\delta) > \alpha_W(1/\delta)$. That is, if the maximum potential value of Linux is larger than that of Windows, then a Linux monopoly is preferable.

Figure 3 **Consumer Welfare in Duopoly**



More interesting is the comparison between a Windows monopoly and a Windows-Linux duopoly. Figure 3 summarizes the computation of total surplus generated by the duopoly (total surplus is the sum of the shaded areas).

Computing the shaded areas, we see that total surplus generated by Windows and Linux is

$$TS_W^{Duopoly} = \frac{1}{2}\alpha_W(y^{ss})\frac{(s+\delta y^{ss})(2+s-\delta y^{ss})}{(1+s)^2}$$

and

$$TS_L^{Duopoly} = \frac{1}{2}\alpha_L(y^{ss})\frac{(1-\delta y^{ss})^2}{(1+s)^2}.$$

Total duopoly surplus is

$$TS^{Duopoly} = TS_W^{Duopoly} + TS_L^{Duopoly}.$$

Proposition 5. *Steady-state total surplus may be larger under Windows' monopoly than under a duopoly.*

There are two reasons that duopoly can enhance surplus. First, because Linux is available, Microsoft is induced to set lower prices. Second, those individuals in the cohort who do not buy Windows are not left empty handed; they can download and use Linux for free, and this raises total surplus. However, the fact that part of the population uses Linux lowers the value of Windows (because there is some substitution of third-party complement development from Windows to Linux). If this effect is large, monopoly (where all developers produce complements for Windows) may result in larger total surplus.

We conclude this section with an observation about welfare on the path to the steady state. As mentioned above, when $\alpha_L(-s/\delta) > \alpha_W(1/\delta)$, steady-state total welfare under Linux monopoly is larger than steady-state total welfare under Windows monopoly or duopoly. However, if it will take a long time for Linux to build market share and Windows' initial advantage is large, Linux's monopoly may still not maximize the net present value of total surplus.

4. Extensions

We now present three simple extensions to the basic model: piracy of Windows, strategic commitment to Linux by governments, and forward-looking buyers. Piracy helps Windows increase its steady-state quality difference and may, therefore, result in increased profits for Microsoft. Government commitments to promote OSS can actually reduce social welfare. And forward-looking buyers may tip market outcomes toward proprietary software.

Piracy of Windows

Microsoft executives have asserted on several occasions that OSS is inherently inimical to property rights and therefore threatens the incentive to innovate and even "the American way" (Jaffe 2001). In this subsection we look both theoretically and empirically at this accusation through the lens of software piracy rates.

Piracy has an obvious effect on Microsoft's profits: Users of pirated software do not pay for the license to run Windows; therefore, Microsoft's profits deteriorate as piracy increases. This argument, however, ignores that pirates add to Windows' installed base. Because the pirated software is available at zero (or very low) price, the increase in the installed base due to piracy may be substantial. And because the value of Windows (relative to Linux) is positively correlated with the size of the installed base, piracy of Windows makes it harder for Linux to remain competitive. In fact, the theoretical model suggests that high piracy rates will reduce Linux's share of the market and increase Windows' share. Put differently, Linux is predicted to do better in institutional environments where the quality of software property rights protection is also good. And a very basic empirical analysis suggests that the estimated piracy rates do have a significant negative association with Linux market share. So there is at least one sense in which one can question the claim that OSS somehow doesn't fit with good property rights.

We use an extension of the basic model to analyze the effects of piracy. Suppose that every period, a portion ρ of the entering cohort pirates Windows. We assume that the portion of pirates is small (positive but sufficiently close to zero so that an interior solution exists). It is important to identify who these customers are. Suppose a portion $\mu \in (0, 1)$ comes from individuals who would have bought Windows (high-value customers) and the rest, $1-\mu$, would have gotten Linux (low-value customers).

Proposition 6. *The larger the piracy rate ρ, the larger the steady-state difference in installed bases $(y_W^{ss} - y_L^{ss})$.*

To test this proposition empirically, we use data from the Gartner Group on the ratio of shipments of Linux and Microsoft server OSs in 45 countries in 2001, from the Business Software Alliance on piracy rates in those countries and from the Economist Intelligence Unit on gross domestic product (GDP) per capita (as a control variable). An ordinary least squares (OLS) regression of the *Linux/Windows ratio* on the two other variables yields the estimates in Table 1.

Note that while per capita income does not appear to affect Linux penetration relative to Windows, piracy does have a significant negative effect—as predicted by the theoretical model. Steve Ballmer, CEO of Microsoft, may have had this effect in mind when he said in 2002 that he had seen Windows selling "in funny packages" for \$2.50 at a Beijing bookstore and added, "I don't mean to sound facetious, that's not where we want to be, but really, for most people in China, Linux and Windows cost the same amount of money" (Schafer 2004, p. 48).

Strategic Commitment to Linux

Perhaps unsurprisingly, Microsoft's conflicting claims about OSS and intellectual property rights do not seem to have had much of an effect on public policy. In contrast, proponents of OSS appear to have helped induce measures mandating various types of public preference for OSS in many jurisdictions. In this subsection we analyze effects of government procurement of OSS instead of proprietary software.

The theoretical model provides a structure for examining this issue. Suppose that in each cohort, a measure $\varepsilon > 0$ of customers (who may or may not be governmental buyers) is committed to using Linux. It is useful to distinguish between two polar sub-

Table 1 Regression Output. Dependent Variable: Linux/Windows Ratio

	Coefficients	Standard error	t stat	p-value
Intercept	0.041603882	0.129983312	0.320070952	0.750501449
Piracy rate	−0.085058641	0.049258797	−1.726770564	0.091558265
ln (GDP per head)	0.007707245	0.011396057	0.676308011	0.502552684

cases. In the first subcase, the potential customers represented by ε would have used Linux even if they were not committed to Linux. In this case, there is no change to the outcomes previously identified.

The second subcase is more interesting. Suppose that these ε customers would all have bought Windows had they not been committed to Linux. These are individuals that value Windows above Linux (after subtracting p), but they use Linux instead. The following proposition shows that with the presence of strategic buyers, if demand-side learning on the part of Linux is sufficiently swift, Windows is in fact pushed out.

Proposition 7. *Given a portion of customers committed to Linux, ε, if demand-side learning differential s is sufficiently large, Microsoft is pushed out of the market. Equivalently, given Linux's demand-side learning $s > 1$, if the portion of customers committed to Linux ε is sufficiently large, Microsoft is pushed out of the market.*

Intuitively, when s is large, Microsoft has to make sure that Linux's share $(1 - q(t))$ remains very small. However, the presence of a portion of potential customers who will never buy Windows jeopardizes Microsoft's ability to capture (current cohort) market share. If such ability is sufficiently damaged by these strategic buyers, Windows is eventually pushed out.

Therefore, if s is large and $\alpha_L(-s/\delta) \gg \alpha_W(1/\delta)$ (Jaffe 2001), strategic commitment to Linux induces efficient push-out of Windows by Linux. However, if s is small (say, $s < 1$) and $\alpha_L(-s/\delta) \ll \alpha_W(1/\delta)$ so that without strategic commitment, Linux would be efficiently pushed out by Microsoft, with strategic commitment, Linux may prevail. Note that in this scenario, strategic commitment to Linux actually decreases total welfare.

To empirically validate the proposition, we gathered public information on 22 countries and uncovered 10 in which government organizations appeared to explicitly promote the use of OSS. (We present the data in the appendix: Data.) An analysis of the data suggests that such public preferences do appear to have an effect on market outcomes.

Although the data set is small, regressing the Linux/Windows ratio for this subsample on piracy rates, per capita income, and a dichotomous variable that takes the value of 1 in the presence of government organizations that promote the use of OSS and 0 otherwise, yields the results shown in Table 2.

The data show once more that per capita income does not affect Linux penetration relative to Windows. However, there is a strong positive relationship between Linux penetration and this form of public preference for OSS. Similarly strong results are obtained when we capture public preference in terms of reported deployments of OSS within government organizations.

Forward-Looking Buyers

Both opponents and proponents of OSS have exhibited some uncertainty about how much of Linux's success is really due to the support of corporations that have an interest in breaking the stranglehold Microsoft has on them as a supplier, and with correspondingly deep pockets, that are willing to contribute development time and even some of their own software to open source as well as developing complementary offerings as opposed to the intrinsically greater efficiency of the Linux model. Implicit in this way of framing the question is the notion that strategic buyers are bad news for the profit-maximizing leader in its interactions with its zero-priced competitor. In this subsection we model strategic buyers by allowing them to be forward-looking rather than myopic in choosing between the duopolists' offerings. The analysis indicates that such strategic buyers aren't necessarily harmful to Microsoft, so the usual presumption that they always are must be resisted.

Recall that buyers in the baseline model are assumed to be myopic. In any given cohort, each individual buyer makes a comparison based on current characteristics between Linux and Windows when deciding which OS to purchase/use: If the value (net of price) of operating system i is greater than that of operating system j (regardless of how that value may evolve in the future), then operating system i is chosen. We now relax this assumption and allow buyers to take into consideration the future value of each alternative when making their purchase/use decisions.

We first derive Windows' demand function when buyers are forward looking (they foresee that they will use the OS for a long time) but believe that they

Table 2 **Regression Output. Dependent Variable: Linux/Windows Ratio**

	Coefficients	Standard error	t stat	p-value
Intercept	0.083395978	0.137145757	0.608082814	0.55072889
Piracy rate	−0.13985573	0.053460214	−2.61607126	0.017497549
ln (GDP per head)	0.004502488	0.012212869	0.368667499	0.716676008
Gov't. organizations that promote use of open source	0.031236786	0.012339577	2.531430905	0.020894609

are so insignificant that their individual purchasing decisions will not affect the long-run steady state. Let ϕ be the (common) discount rate buyers use to evaluate future utility. Suppose that present time is t, that the state variable has value $y(t)$, and that price is $p(t)$.

Customer q derives net present value of utility

$$\int_t^\infty e^{-\phi(\tau-t)}\alpha_W(y(\tau))(1-q)\,d\tau - p(t)$$

if he or she buys Windows at time t and uses it from that moment onward. If he or she downloads and uses Linux for free, the utility is

$$\int_t^\infty e^{-\phi(\tau-t)}\alpha_L(y(\tau))(1-q)\,d\tau.$$

If we assume that everybody in cohort t has the same expectations regarding the future value of Windows and Linux ($\alpha_W(y(\tau))$ and $\alpha_L(y(\tau))$ for $\tau \geq t$), the demand function for Windows at time t is

$$q(t) = 1 - \frac{p(t)}{\int_{\tau=t}^\infty e^{-\phi(\tau-t)}\beta(y(\tau))\,d\tau}. \tag{3}$$

Comparing (3) and the demand function in the benchmark model ($q = 1 - p/\beta(y(t))$), we see that whether the threshold q with forward-looking buyers is larger or smaller than that with myopic buyers depends on the buyers' view on which OS will be more valuable in the future. In particular, when $\beta(y(t))$ ($\equiv \alpha_W(y(t)) - \alpha_L(y(t))$) is large and positive and buyers' discount rate is not too low, the presence of forward-looking buyers plays to the advantage of Microsoft, because for a given p, demand for Windows q is now larger (compared to the case when buyers are myopic). However, if buyers expect $\beta(y(t))$ to eventually turn negative and their discount rates are low, Microsoft will be forced to price lower than in the case where buyers are myopic.

Microsoft can once again ensure a 100% market share of every new cohort by pricing sufficiently low, as long as

$$\int_{\tau=t}^\infty e^{-\phi(\tau-t)}\beta(y(\tau))\,d\tau > 0.$$

In particular, Microsoft can guarantee that Linux gets zero market share by setting $p(t) = 0$. Therefore, the result that Microsoft is not forced out by Linux (regardless of the intrinsic advantages of Linux) remains intact when buyers are forward looking but believe that they are so insignificant that their purchasing decisions will not affect the long-run steady state.

Equation (3) shows that it may be worth it for Microsoft to influence the value of

$$\int_{\tau=t}^\infty e^{-\phi(\tau-t)}\beta(y(\tau))\,d\tau$$

by infusing "fear, uncertainty, and doubt" into the OS user and developer communities. Such emotions were stirred in the Linux community by, among other things, SCO, a small Swiss-based "vulture" firm that had bought up the intellectual property rights to a particular version of Unix and was threatening Linux users with lawsuits over infringement of those rights unless they agreed to pay it substantial licensing fees. IBM, which was one of the prime corporate sponsors of Linux as well as the target of a lawsuit by SCO that sought \$1 billion in damages, alleged in mid-2003 that SCO was in cahoots with Microsoft (Dolley 2003).

In addition, Microsoft uses its Web site to assert its superiority over Linux and to warn users that the deployment of Linux can be very costly (Microsoft). To add credibility to its claims, Microsoft exhibits reports by independent sources. In a recent interview, Ballmer stated:

> Innovation is not something that is easy to do in the kind of distributed environment that the open source/Linux world works in. I would argue that our customers have seen a lot more innovation from us than they have seen from that community. Linux itself is a clone of an operating system that is 20-plus years old. That's what it is. That is what you can get today, a clone of a 20-year-old system .... Customers will never really know who stands behind this product [Linux]. If the lead developer for this component chooses to do something else with his life, who will carry on the mantle for that? (Kanellos and Shankland 2003)

In terms of the model, with these statements Ballmer is trying to increase $\int_{\tau=t}^\infty e^{-\phi(\tau-t)}\beta(y(\tau))\,d\tau$ by shaping users' expectations about the future value of Windows and Linux. Notice, however, that Microsoft's efforts to instill fear are to some extent offset by the users' and developers' concern that Microsoft may decide to raise prices in the future if it succeeds in derailing the progression of Linux. Assertions of this kind would be vacuous in the context of the baseline model of §3 with myopic users.

5. Conclusions

This paper used the case of Linux versus Windows as a point of departure for specifying a simple model of competition between OSS priced at zero and for-profit proprietary software in the presence of demand-side learning effects. The analytical set-up, while not game theoretic in the usual sense of interdependent strategy choices, *was* strategic in the sense of requiring Microsoft Windows to take a deep look into the future that recognized intertemporal linkages in its profit function (e.g., between past or current choices and future profits) as opposed to acting myopically. As Arrow (1964) and others have stressed, such intertemporal linkages and the commitment or irreversibility underlying them are the key reasons that the optimal *intertemporal* investment program may not coincide with the *instantaneous* equation of the marginal productivity and the marginal cost of capital (of whatever sort), that is, the myopic investment program (Shapiro 1989, Ghemawat 1991).

Embedding irreversibility in a formal analytical model of this sort yielded some arguably surprising conclusions. Thus, Microsoft Windows' persistence exceeded our preanalytic intuitions because of the effects of Microsoft's strategic management of its position relative to Linux. Other effects/possibilities that were somewhat unexpected—or at least sounded some cautions about common assertions concerning competition between OSS and proprietary software—included the positive association between the enforcement of property rights and the relative viability of OSS, the possibly welfare-reducing effects of government promotion of OSS, and the possibility that forward-looking buyers might tip market outcomes away from rather than toward OSS.

Our baseline model and the various extensions to it all relied on various simplifying assumptions that could be generalized, although we suspect that in many cases, this would force a shift from analytical to numerical methods. In addition, this paper flags some topics on which additional empirical work would be particularly welcome. First, a better understanding of the drivers of adoption could guide modeling efforts by identifying strategic variables other than price that Microsoft (Linux) has at its disposal to decelerate (accelerate) the progress of OSS. Second, some (more) of the ongoing work on the organization of OSS development could explicitly focus on studying effort coordination in ways designed to shed light on issues such as "code forking" that stand out as critical to the competitive dynamics between OSS and proprietary software. Third, better estimates of the differential in demand-side learning between OSS and proprietary software could guide the development of more educated guesses about their long-run viability, improve managerial decision making, and direct the design of public policies.

Most broadly, this paper intended to advance the analysis of heterogeneity in competitors' objective functions—a structure that we refer to as mixed duopoly or, more generally, mixed oligopoly. It focused on what is arguably the most obvious modification to the standard assumption of symmetric profit maximization by analyzing interactions between a not-for-profit player that prices its product at zero (or at marginal cost) and a for-profit player. This stylization evokes not only interactions between OSS development efforts (of which Linux is one of many) and their for-profit competitors (of which Microsoft is one of many) but aspects of a number of other types of interactions as well. These include interactions between a profit maximizer and a competitor pursuing volume or market share by pricing at marginal cost, between profit maximizers and much more patient competitors, between private and state-owned/supported enterprises (e.g., Boeing versus Airbus, in the official U.S. view), and between for-profit firms and nonprofits or even the social sector, broadly defined (e.g., between pharmaceutical firms and universities in the life sciences—although those relationships involve complementarities and side payments as well as somewhat fragmented competition). While there has been some work on the (comparative) organizational correlates of objectives other than profit maximization, interactions between conventional profit-maximizing firms and organizations with other objectives remain largely uncharted.

An online supplement to this paper is available on the *Management Science* website (http://mansci.pubs.informs.org/ecompanion.html).

Acknowledgments
The authors thank two anonymous referees and the editors for several comments that substantially improved the paper. They also thank Bharat Anand, Antonio Cabrales, Ken Corts, Yves Doz, Miguel Angel García Cestona, Josh Lerner, Fabio Maneti, Patrick Moreton, Vicente Salas Fumás, Ana Sotelo, Dennis Yao, Peter Zemsky, and seminar participants at HBS; NYU; INSEAD; IESE; Universitat Autònoma de Barcelona; Universitat Pompeu Fabra; Universidad de Navarra, Instituto de Empresa; Universitat de les Illes Balears; the 2003 Strategy Research Forum held at Washington University, St. Louis; the 1st Meetings of the International Industrial Organization Society, held in Boston; the 2003 Society for the Advancement of Economic Theory (SAET) conference in Rhodes (Greece); and the 2004 Academy of Management Meetings in New Orleans, for useful comments on an earlier version of this paper. Christina B. Pham provided superlative research assistance. Ramon Casadesus-Masanell is grateful to the HBS and IESE Divisions of Research, *Fundación BBVA*, and Programa *Ramón y Cajal* for financial support, and Pankaj Ghemawat to the HBS Division of Research.

Appendix. Data

Our efforts at gathering data on governmental policies toward Linux involved an otherwise open-ended search over the period from early 2000 to early 2003. A research assistant with work experience in the software industry, Christina Pham, consulted several hundred articles, market studies, official reports et cetera and prepared a 63-page summary which she then used to code policies in each country. While it is not feasible to include a compact characterization of the materials consulted, Table A1 summarizes the coding that resulted, in a way that should provide a basis for reexamination and, if necessary, reanalysis.

Table A1 Governmental Policies Toward Linux, 2000–2003

Country	Gov't. organizations that promote use of open source	Deployed open source in gov't. systems
Austria	1	0
Belgium	1	0
China	0	0
Denmark	0	0
Finland	0	0
France	1	0
Germany	1	1
Greece	1	0
India	0	0
Ireland	1	0
Italy	1	0
Netherlands	1	0
Norway	0	0
Philippines	0	0
Portugal	1	0
Singapore	0	0
South Africa	0	0
Spain	0	1
Sweden	0	0
Taiwan	0	0
United Kingdom	1	1
United States	0	1

References

Anderson, S. P., A. de Palma, J.-F. Thisse. 1997. Privatization and efficiency in a differentiated industry. *Eur. Econom. Rev.* **41** 1635–1654.

Arrow, K. J. 1964. Optimal capital policy, the cost of capital and myopic decision rules. *Ann. Inst. Statist. Math. Tokyo* **16** 21–30.

Bessen, J. 2002. Open source software: Free provision of a complex public good. Working paper, Boston University School of Law, Harpswell, ME, http://opensource.mit.edu/online_papers.php.

Bitzer, J. 2004. Erosion of monopoly power due to the emergence of Linux. *Econom. Systems* **28**(4) 369–381.

Cremer, H., M. Marchand, J.-F. Thisse. 1991. Mixed oligopoly with differentiated products. *Internat. J. Indust. Organ.* **9** 43–53.

Dalle, J.-M., N. Jullien. 2002. "Libre" software: Turning fads into institutions? *Res. Policy* **32**(1) 1–11.

David, P. 1985. CLIO and the economics of QWERTY. *Amer. Econom. Rev.: Papers Proc.* **75** 332–337.

De Fraja, G., F. Delbono. 1990. Game theoretic models of mixed oligopoly. *J. Econom. Surveys* **4** 1–17.

Dolley, A. 2003. IBM claims SCO conspiring with Microsoft over Linux. *ZDNet Australia* (July 30) http://www.zdnet.co.nz/newstech/enterprise/story. Accessed on August 6, 2003.

Economist, The. 2004. Collective intelligence—When the many know best. (May 27).

Farrell, J., G. Saloner. 1985. Standardization, compatibility, and innovation. *RAND J. Econom.* **16** 70–83.

Farrell, J., G. Saloner. 1986. Installed base and compatibility: Innovation, product preannouncements, and predation. *Amer. Econom. Rev.* **76** 940–955.

Fuller, T. 2003. How Microsoft warded off rival software. *New York Times* (May 15) C1.

Ghemawat, P. 1991. *Commitment*. Free Press, New York.

Ghemawat, P., B. Subirana, C. B. Pham. 2004. Linux in 2004. Case 705407, Harvard Business School Publishing, Boston, MA.

Jaffe, S. 2001. Why Linux is giving Microsoft a migraine. *Bus. Week* (February 22), http://ww.businessweek.com/dnflash/feb2001/nf20010222_369.htm.

Johnson, J. P. 2002. Open source software: Private provision of a public good. *J. Econom. Management Strategy* **11**(4) 637–662.

Kanellos, M., S. Shankland. 2003. Ballmer: No sleep lost over Linux. *CNET News.com.* http://news.com.com/2008-1082-998297.html.

Kogut, B., A. Metiu. 2001. Open-source software development and distributed innovation. *Oxford Rev. Econom. Policy* **17**(2) 248–264.

Kuan, J. 1999. Understanding open source software: A nonprofit competitive threat. Working paper, Sloan Software Industry Center, Carnegie Mellon University, Pittsburgh, PA.

Lakhani, K. R., E. von Hippel. 2002. How open source software works: "Free" user-to-user assistance. *Res. Policy* **32**(6) 923–943.

Lerner, J., J. Tirole. 2002. Some simple economics of open source. *J. Indust. Econom.* **50**(2) 197–234.

Matsumura, T. 1998. Partial privatization in mixed duopoly. *J. Public Econom.* **70** 473–483.

Merrill, W., N. Schneider. 1966. Government firms in oligopoly industries: A short-run analysis. *Quart. J. Econom.* **33** 400–412.

Microsoft. Get the facts. Retrieved May 2006, http://www.microsoft.com/windowsserversystem/facts/default.mspx.

Mockus, A., R. T. Fielding, J. D. Herbsleb. 2000. A case study of open source software development: The Apache server. *The ICSE 2000, Proc. 22nd Internat. Conf. on Software Engrg., Limerick Ireland, June 4–11*.

O'Mahony, S. 2003. Guarding the commons: How community managed software projects protect their work. *Res. Policy* **32**(7) 1179–1198.

Raymond, E. 1997. *The Cathedral and the Bazaar*. http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/.

Schafer, S. 2004. Microsoft's cultural revolution. *Newsweek Internat. Ed.* (June 21) 46–49.

Schmidt, K. M., M. Schnitzer. 2002. Public subsidies for open source? Some economic policy issues of the software market. Working paper, SSRN, http://www.ssrn.com.

Shapiro, C. 1989. The theory of business strategy. *Rand J. Econom.* **20**(1) 125–134.

Spence, A. M. 1981. The learning curve and competition. *Bell J. Econom.* **12**(1) 49–70.

Surowiecki, J. 2004. *The Wisdom of Crowds*. Doubleday-Random House, New York.

Thorn, B. K., T. Connolly. 1987. Discretionary data bases: A theory and some experimental findings. *Comm. Res.* **14**(5) 512–528.

Xu, Xiaopeng. 2002. Development costs and open source software. Working paper, University of California, Berkeley, CA.