MANAGEMENT SCIENCE
Articles in Advance, pp. 1–19
ISSN 0025-1909 | EISSN 1526-5501

DOI 10.1287/mnsc.1110.1353

Mixed Source

Ramon Casadesus-Masanell
Harvard Business School, Harvard University, Boston, Massachusetts 02163, casadesus@gmail.com

Gastón Llanes
Escuela de Administración, Pontificia Universidad Católica de Chile, Vicuna Mackenna 4860, Macul, Santiago, Chile, gaston@llanes.com.ar

We study competitive interaction between a profit-maximizing firm that sells software and complementary services, and a free open-source competitor. We examine the firm's choice of business model between the *proprietary* model (where all software modules are proprietary), the *open-source* model (where all modules are open source), and the *mixed-source* model (where some—but not all—modules are open). When a module is opened, users can access and improve the code, which increases quality and value creation. Opened modules, however, are available for others to use free of charge. We derive the set of possibly optimal business models when the modules of the firm and the open-source competitor are compatible (and thus can be combined) and incompatible, and show that (i) when the firm's modules are of high (low) quality, the firm is more open under incompatibility (compatibility) than under compatibility (incompatibility); (ii) firms are more likely to open substitute, rather than complementary, modules to existing open-source projects; and (iii) there may be no trade-off between value creation and value capture when comparing business models with different degrees of openness.

Key words: open source; user innovation; business models; complementarity; compatibility; value creation; value capture
History: Received September 14, 2009; accepted January 31, 2011, by Bruno Cassiman, business strategy. Published online in *Articles in Advance*.

1. Introduction

As is well understood by now, commercial firms may benefit from participating in open-source software development by selling complementary goods or services. For example, IBM sells consulting services and proprietary software complementary to the open-source software it develops, Red Hat sells subscription services, and Sun sells complementary hardware such as servers. In particular, the combination of open-source software and proprietary extensions has grown into an important phenomenon. In fact, the expressions *mixed source* and *hybrid source* refer to a business model whereby a software firm releases an open-source version of its software and derives revenue from selling proprietary complementary code. Examples include JasperSoft (business intelligence software), Zimbra (server software for e-mail and collaboration), SugarCRM (customer-relationship-management software), Hyperic (systems monitoring, server monitoring, and IT management software), xTuple (enterprise resource planning software), Zenoss (enterprise IT management software), Talend (data integration software), and Groundwork (IT management and network-monitoring software).

Even fervent advocates for proprietary software have jumped on the bandwagon. Having stated in 2001 that "open source is an intellectual property destroyer . . . I can't imagine something that could be worse than this for the software business," Microsoft has recently switched course to embrace the notion of mixed source.[1] Among other initiatives, it has partnered with Novell to put some of Microsoft's technologies on Linux and other open platforms. For example, the *Mono* project consists of porting the .Net framework onto Linux, and the *Moonlight* project provides an offer of Silverlight for Linux.[2] And in July 2009, Microsoft agreed to contribute some of its technology to Linux under a licensing agreement that allows developers outside Microsoft to modify the code.[3]

Open source has the potential to improve value creation because it benefits from the efforts of a large community of developers. Proprietary software, on the other hand, results in superior value capture because the intellectual property remains under the control of the original developer. Industry observers,

[1] James Allchin, Microsoft's executive in charge of Windows 2000 rollout (see Charny 2001).

[2] Silverlight is a Web-based digital video technology by Microsoft. It is a plug-in for delivering media and interactive applications for the Web.

[3] Microsoft announced the release of 20,000 lines of device driver code to the Linux community (see Letzing 2009).

however, point out that strict open-source and proprietary approaches to software development "don't work in a world where innovators have to innovate, investors need a profit, employees need to eat, and customer needs must be met" (Thomas 2008). The reasoning is that proprietary development leads to little innovation and open source leads to little profits. According to Horacio Gutierrez (Microsoft's Deputy General Counsel for IP Licensing): "striking a balance between [embracing open-source software and brandishing patents] is one of the key things every commercial technology company must do in order to compete effectively" (see Letzing 2009).

As a recent phenomenon, mixed source has given rise to a number of questions of interest to strategy scholars researching the design of optimal business models. Specifically, when will a profit-maximizing firm adopt a mixed-source business model? And how is the desirability of mixed source affected by the quality of the firm's software compared to that of competitors? For firms considering the adoption of a mixed-source business model, which technologies will be open and which ones will remain closed/proprietary? Moreover, how do these decisions depend on the compatibility regime between the products of the firm and those of its competitors? The purpose of this paper is to present a formal model to address these questions.

We set up a model where a profit-maximizing firm that sells software and complementary goods (such as training or support services) must choose whether to open all or part of its software and the price at which to sell its product. Software is composed of two modules: a base program (the *core* code) and a set of extensions (the *edge* code). The base may be used without the extensions. The extensions, on the other hand, are valueless unless used in conjunction with a base, i.e., the base is a one-way essential complement to the extensions (as defined by Chen and Nalebuff 2006).[4] Firms may open the base, the extensions, or both. We assume that base, extensions, and service are complements: an increase in the value of any one of them raises the returns of increases in value in the other two. We capture complementarity formally by assuming that the function that maps the quality of the core, edge, and service to overall product quality exhibits increasing differences (see Topkis 1998).

The trade-off we consider is as follows. When a module is opened, users can access and improve the source code, which increases quality and value creation.[5] Opened modules, however, are available for others to use free of charge. Thus, when opening a module the firm must consider that it may be adopted by other players, which may strengthen competitive pressure. As competitive pressure intensifies, the firm must lower prices, hampering its ability to capture value. We study how this trade-off is affected by (i) the value of the firm's modules relative to those of an existing open-source competitor, (ii) the strength of user innovation, and (iii) the compatibility regime between the firm's software and that of the open-source competitor.

Figure 1 **Examples of the Four Business Models**

	Extensions	
Base	Open	Closed
Open	Open source • MySQL • OpenSolaris • Red Hat Linux • Eclipse	Open core • SugarCRM • Zimbra • JasperSoft • Mac OS X
Closed	Open edge • MSFT .Net • Mathematica • Stata • Facebook	Proprietary • MS Windows • Oracle llg • MS Office • SAP

Consistent with the terminology in the software development community, we refer to the different degrees of openness as *business models*. Thus, we consider three business models. In the *proprietary* model, all the software is closed, and in the *open-source* model, both modules are open. In the *mixed-source* model, one module is open and the other one is closed. The mixed-source model has two variants. In the *open-core* model, the base program is open and the extensions closed, and in the *open-edge* model, the base program is closed and the extensions open. Figure 1 shows examples of the business models we consider.

We find that business model choice is heavily affected by the relative quality of the firm's modules to those of the open-source competitor. Under compatibility, the firm will always adopt modules from the open-source competitor if those are of higher quality. We also find that the proprietary business model may only be optimal when all of the firm's modules are of higher quality than those of the open-source competitor.

When some (but not all) of the firm's modules are of substantially higher quality than those of the open-source competitor, we find that a mixed-source business model is optimal. In addition, we find that when considering which mixed-source business model to employ, the firm will always prefer the one that adds less value to the product of the open-source competitor. Thus, the firm is more likely to open substitute, rather than complementary, modules to open-source competitor.

Under incompatibility, the firm is more willing to open high-quality modules than under compatibility. The reason is that when modules of different

[4] More generally, the extensions represent not only software modules, but all those technologies and protocols that have value only if used with the firm's base software.

[5] This effect was dubbed *demand-side learning* by Casadesus-Masanell and Ghemawat (2006).

providers are incompatible, it is less likely that the open-source competitor will adopt modules opened by the firm. When the firm's modules are of lower quality than those of the open-source competitor, however, the firm is more closed under incompatibility. The reason is that open-source business models combining modules of different developers (which may be desirable in this case and are available under compatibility) are not available under incompatibility.

We are able to map the different business models on the space of value creation and value capture and find an efficient frontier composed of undominated business models. We show that there are parameter values for which the frontier is composed of an open-source business model only. In other words, there are cases in which open source provides both higher value creation and value capture than alternative business models (proprietary or mixed). This finding questions the conventional wisdom that open source leads to higher value creation at the expense of lower value capture.

We end with some considerations about the choice of compatibility regime. We find that compatibility always provides higher value creation, but may not be optimal for the firm from a profit standpoint. We also find that incompatibility is optimal when the firm's modules are of substantially higher quality. When the firm has only one module of higher quality, then compatibility is generally best (because business models that combine modules from different developers are possible). Finally, when the firm's modules are both of lower quality, both compatibility regimes lead to the same profitability.

1.1. Related Literature

Our paper contributes to the literature on the economics of open source. For the most part, early papers on open source were concerned with explaining why individual developers contributed to open-source projects, allegedly for free (for excellent surveys, see Lerner and Tirole 2005, von Krogh and von Hippel 2006). The most common explanations were altruism, personal gratification, peer recognition, and career concerns. Bagozzi and Dholakia (2006), for example, demonstrate that participation in open-source development is partly explained by social and psychological factors, and Roberts et al. (2006) find that status and career concerns motivations significantly influence developers' levels of participation. Baldwin and Clark (2006) argue that the architecture of the code may affect the developers' incentives to contribute. Specifically, they show that a modular codebase mitigates free riding in open-source development.

Although the contributions of individual developers have played a crucial role in the growth of open-source software, the same is true of contributions by commercial firms. In fact, in a carefully executed empirical piece, Bonaccorsi et al. (2006) show that fully proprietary and fully open software firms are rare. Instead, firms typically adopt a hybrid business model. The presence of complementarities has been documented by Fosfuri et al. (2008). The authors perform an econometric analysis and find that firms with a larger stock of hardware patents and trademarks are more likely to participate in open source. Shah (2006) investigates the effects of sponsorship of open-source projects by commercial firms and finds that voluntary developers tend to contribute less, have different motivations for contributing, and take on fewer code maintenance tasks than in the absence of such sponsorship.

On the theory front, the first papers that studied competition between the open-source and proprietary paradigms considered duopoly models of a profit-maximizing, proprietary firm and a community of not-for-profit/nonstrategic open-source developers selling at zero price (Mustonen 2003, Bitzer 2004, Gaudeul 2005, Casadesus-Masanell and Ghemawat 2006, Economides and Katsamakas 2006, Lee and Mendelson 2008). In these papers, however, open-source firms had no profits, and the decision to open technologies was not endogenous.

More recently, a handful of papers have introduced profit-maximizing open-source firms (Henkel 2004, Bessen 2006, Schmidtke 2006, Haruvy et al. 2008, Llanes and de Elejalde 2009, von Engelhardt 2010). In particular, Llanes and de Elejalde (2009) present a model where profit-maximizing firms decide whether to be open source or proprietary, and where open-source firms profit from selling goods and services that are complementary to the software. One particular possibility is that firms develop open-source software and sell complementary proprietary software. However, in this case the determination of which products are open source and which products are proprietary is exogenous. In our paper, in contrast, we endogenize the process by which a for-profit firm decides which software programs to open and which programs to keep proprietary. Moreover, previous literature has not analyzed the effects of compatibility/incompatibility on the decision to open technologies.

Our paper builds upon this literature and presents a novel approach to the study of optimal business models and equilibrium market structure. Specifically, our modeling contributions are as follows: (a) firms have multiple software modules and must decide which modules to open and which to keep proprietary; (b) firms choose not only between the two *pure* business models (open source and proprietary), but may also compete through a *mixed*-source model; and

(c) modules developed by different players may be compatible or incompatible.

Our paper also contributes to an emerging literature in strategy that explores competitive interactions between organizations with different business models. Although there are several formal models of asymmetric competition that exist in strategy (differences in costs, resource endowments, or information, mainly), the asymmetries that this literature wrestles with are of a different nature: firms with fundamentally different objective functions, opposed approaches to competing, or different governance structures. The papers mentioned above on competition between open-source and proprietary firms belong to this literature. In addition, Casadesus-Masanell and Yoffie (2007) study competitive interactions between two complementors, Microsoft and Intel, with asymmetries in their objectives functions stemming from technology—software *versus* hardware. Casadesus-Masanell and Zhu (2010) study competitive interaction between a high-quality incumbent that faces a low-quality ad-sponsored competitor. Finally, Casadesus-Masanell and Hervas-Drane (2010) analyze competitive interactions between a free peer-to-peer file-sharing network and a profit-maximizing firm that sells the same content at positive price and that distributes digital files through an efficient client-server architecture. For the most part, this literature has studied interactions between firms with exogenously given business models.[6] We contribute by endogenizing the choice of business model by a profit-maximizing firm.

2. The Model

In this section, we present a model to study the decision to open technologies by a for-profit firm, which competes against a nonprofit open-source project to sell software to consumers. We set up a mixed duopoly where players have different objective functions. Whereas the firm seeks to maximize profits, the open-source competitor seeks to maximize the value of the software that it provides. For example, the firm could be Microsoft or Red Hat, whereas the open-source competitor could be Richard Stallman's Free Software Foundation or the Apache Software Foundation.

In addition to software, the firm also offers service complementary to the software. For simplicity, we assume that only the firm offers service. Red Hat, for example, offers tailoring of Linux and other open-source software that is not offered by decentralized, open-source communities. However, all our results hold if the open-source competitor also offers service, as long as its quality is lower than that of the firm.

[6] The exception is Casadesus-Masanell and Zhu (2010), who allow the incumbent (but not the entrant) to choose the business model.

2.1. Preferences

Preferences are based on a variant of Gabszewicz and Thisse (1979) and Shaked and Sutton (1982) models of vertical product differentiation. There is a continuum of consumers of mass 1, who differ in their valuations of the available products. Consumers are indexed by ρ, where $\rho \sim U[0,1]$. Consumer ρ's indirect utility from consuming good i is

$$u_{\rho i} = \rho V_i - p_i,$$

where V_i and p_i are the quality and price of product i. Given the list of qualities and prices for all products available, each consumer will choose the product that maximizes his indirect utility.

2.2. Technology

Consumers derive utility from consuming packages composed of software modules and a complementary service, denoted by z. There are two types of modules: base modules and extensions. Let a and b denote the base and extensions of the commercial firm, respectively, and α and β denote the corresponding modules of the nonprofit open-source project. With a slight abuse of notation, we use the same symbol to refer to a software module or service, and to the value of that software module or service. Thus, $a \geq 0$ and $b \geq 0$ are the values of modules a and b, $\alpha \geq 0$ and $\beta \geq 0$ are the values of modules α and β, and $z \geq 0$ is the value of service z. We assume zero marginal cost of production and distribution for all modules and service.

The value of a package is given by

$$V = V(x, y, z),$$

where x, y, and z are the values of the base, extensions, and service, respectively, and V is increasing in all its arguments. If a package is missing a particular component, then the imputed value is 0. For example, a package formed only by the base module and service of the commercial firm has value $V(a, 0, z)$.

Software modules and service are complementary: the effect on V of an increase in any one of its arguments is larger the larger are the values of the other two arguments. Complementarity is captured by assuming that V has increasing differences in its three arguments,[7] which are defined as follows:

Definition 1 (Vives 2001, p. 24). Let X be a lattice and T a partially ordered set. The function $g\colon X \times T \to R$ has (strictly) *increasing differences* in its two arguments (x, t) if $g(x, t) - g(x, t')$ is (strictly) increasing in x for all $t \geq t'$ ($t \geq t'$, $t \neq t'$).

[7] We note that increasing differences is a weaker condition than supermodularity; i.e., every supermodular function has increasing differences, but the reverse implication is not true.

There is a fundamental asymmetry between the two types of modules: the extensions have no value unless they are used with a core, i.e., $V(0, y, z) = 0$ for all $y \geq 0$ and $z \geq 0$. In the terminology of Chen and Nalebuff (2006), the core module is a one-way essential complement of the extensions. It is important to stress that although for concreteness we refer to b or β as software extensions, they may also represent any technologies, protocols, documentations, or ideas that have value only if used in conjunction with the core software a or α. For instance, file format protocols are of little value if there is no core software with which to use them.

Modules may be opened or kept closed. If a module is kept closed, users cannot improve on it, so the module does not increase in value through use in this case. On the other hand, when a module is opened, users may implement improvements that increase its value. Specifically, we assume that base module x increases value to $x' = f(x, q_x; \sigma_A) > x$ when opened, where q_x is the endogenous measure of consumers using the module; $\sigma_A > 0$ is an exogenous parameter indicating the strength of user innovation for the core module; and f is a nondecreasing function. Likewise, the extensions module increases in value to $y' = g(y, q_y; \sigma_B) > y$ when opened.[8]

In summary,

Assumption 1. (a) *Software modules and service are complements, i.e.,* $V(x, y, z)$: $\mathbb{R}^3_+ \to \mathbb{R}$ *is increasing and has increasing differences in its three arguments.*

(b) *The base module x is a one-way essential complement of y, i.e.,* $V(0, y, z) = 0$ *for all* $y \geq 0$ *and* $z \geq 0$.

(c) *User innovation improves the value of opened modules, i.e., when opened, the value of x and y become* $x' = f(x, q_x; \sigma_A)$ *and* $y' = g(y, q_y; \sigma_B)$, *where* $f, g : \mathbb{R}^3_+ \to \mathbb{R}$ *are nondecreasing.*

Note that our setting is quite general. For example, we do not assume continuity or differentiability of V, f, or g. Moreover, we do not assume any form of symmetry. Likewise, we make no assumptions on the behavior of these functions in the limit as their arguments grow to infinity or fall to zero. Moreover, we assume only a weak form of complementarity in that V does not even need to be supermodular.

2.3. Expectations

When a module is opened, the firm, the open-source competitor, and the users form expectations on the measure of users that will adopt it. Suppose, for example, that module a is opened. Before users decide whether or not to adopt it, they form expectations on the value of a package including that module, which depends on how many users are *expected* to use the module in equilibrium, q_a^e.

To determine the relation between expected and actual quantities, we follow Katz and Shapiro (1985) and assume fulfilled expectations. The expected number of users of module x, q_x^e, is taken as given by consumers and firms when they make their decisions. What Katz and Shapiro's criterion requires is that those expectations be fulfilled in equilibrium: $q_x = q_x^e$.

2.4. Business Models

We consider four business models, two pure models and two mixed models. The pure models are the proprietary and the open-source business models. These are pure models because all modules are either open or closed:

Proprietary (P)**:** The firm sells products based on closed modules. The firm does not benefit from user innovation.

Open Source (O)**:** The firm sells products based on open modules. User innovation is maximal.

The mixed models have one open module and one closed module:

Open Core (M_a **or** M_α)**:** The firm sells products based on an open base module (a or α).

Open Edge (M_b **or** M_β)**:** The firm sells products based on open extensions (b or β).

The most common definition of *business model* is "the logic of the firm, the way it operates to create and capture value for its stakeholders" (see Baden-Fuller et al. 2008, p. 1; Casadesus-Masanell and Ricart 2010, p. 196). A firm's *real* business model includes a broad range of organizational and competitive elements such as products and markets, sources of revenue, incentive systems, hiring policies, information technologies, and so on. Detailed descriptions of business models are often too complex to be amenable to mathematical treatment. We follow Casadesus-Masanell and Zhu (2010) and represent business models through profit functions. Thus, in our formal development the firm's choice of business model is represented through its choice of a profit function.

2.5. Timing

The sequence of decisions is in the spirit of leader-follower models and proceeds as follows:

1. The commercial firm decides which business model to adopt.
2. The open-source competitor decides whether to adopt any module opened by the firm.
3. Expectations are formed on the measure of users that will adopt opened modules, and the firm decides where to price its product.
4. Given expectations, business models, and prices, consumers pick their preferred product.

[8] We note that in our model user innovation works similarly to direct network externalities: the more consumers use the module, the larger its value. In contrast to network externalities, however, if the firm decides not to open the module, user innovation is absent.

We solve for subgame-perfect equilibria with fulfilled expectations.

3. Optimal Pricing

We begin by deriving the demand functions faced by the commercial firm and the open-source project.

3.1. Consumer Demands

Consumers can either buy the commercial product at price p, consume the product of the open-source competitor for free, or stay out of the market. Because the open-source competitor's offering gives utility $u = \rho V_o > 0$, consumers never choose to stay out, and the relevant comparison is always between the firm's and the open-source competitor's products. Therefore, demand for the commercial product and the open-source alternative are, respectively,

$$q_c = 1 - \frac{p}{V_c - V_o} \quad \text{and} \quad q_o = \frac{p}{V_c - V_o}.$$

3.2. Pricing Decision

The firm chooses price to maximize profits, which, given our assumption of zero costs (§2.2), are given by $\pi = p\, q_c$. The inverse demand function faced by the commercial firm is $p = (V_c - V_o)(1 - q_c)$, and the optimal price is half the choke price. At this price, half of the market is served by the firm and profits are $\pi = (V_c - V_o)/4$. The open-source project serves the other half of the market. The precise optimal price and profits depend on the functional forms of V_c and V_o, which, as discussed above, depend on the business model chosen by the firm.[9]

The result that the commercial firm and the open-source project always cover two different halfs of the market implies that every module (a, b, α, β) is either adopted by half the market, the entire market, or nobody.

To simplify notation, we let x_o represent the value of module x when it is open and adopted by the entire market, and $\hat{x}_o$ represent its value when it is open and adopted by half the market. Formally,

Notation 1. *Let* $x_o = f(x, 1; \sigma_A)$, $\hat{x}_o = f(x, \frac{1}{2}; \sigma_A)$, $y_o = g(x, 1; \sigma_B)$, *and* $\hat{y}_o = g(x, \frac{1}{2}; \sigma_B)$.

For example, $\alpha_o = f(\alpha, 1; \sigma_A)$ is the value of the base module of the open-source project when it is adopted by the entire market, and $\hat{b}_o = g(b, \frac{1}{2}; \sigma_B)$ is the value of the set of extensions of the firm, when the firm opens this module and it is adopted by half the market.

Finally, because the two software modules and service are available for separate commercialization, the firm could consider offering more than one commercial product. However, it can be shown that a profit-maximizing firm will only offer the product that combines the base, extensions, and service, even if the cost of new product development is zero.[10]

4. Compatibility

In this section, we assume that software modules from different developers are fully compatible: either a or α may be combined with b or β, and z. Therefore, when choosing its business model in the first stage (see §2.5), the commercial firm may embed an outside open-source module in its commercial software if this leads to higher profits. The implication is that the business models available to the commercial firm come in different "flavors." For example, the open-core business model may take two forms: M_a and M_α, likewise for the open-edge model. Similarly, there are four variants of the open-source business model: O_{ab}, $O_{\alpha b}$, $O_{a\beta}$, and $O_{\alpha\beta}$. Thus, the total number of business models available to the commercial firm is *nine*.

In the second stage, the open-source competitor may combine any one of its modules with any one of the modules opened by the incumbent. Obviously, the open-source competitor will always adopt modules opened by the commercial firm if these are of higher quality than its own. For example, if the commercial firm opens module $a > \alpha$, then the open-source competitor will adopt a and combine it with extensions β. Thus, the "flavors" of the open-source business model available to the competitor are endogenously determined because they depend on the commercial firm's business model choice in the fist stage of the game.

The following proposition narrows down, to the maximal possible extent, the set of business models that may be optimal for different values of the parameters in our general formulation. It shows that the set of optimal business models may be reduced substantially through arguments that rely solely on Assumption 1.

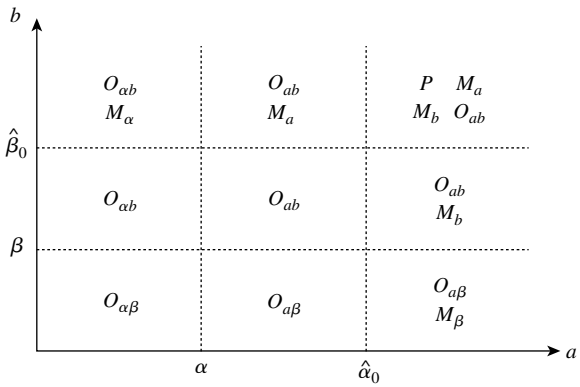
Proposition 1 (Optimal Business Models Under Compatibility). *Under compatibility, the optimal business model depends on the relative qualities of the available software modules* (a, b, α, *and* β), *as shown in Figure* 2.

The wedge between α and $\hat{\alpha}_o$ is the range of values of a such that the firm's base module is of higher quality than that of the open-source competitor, but where the open-source competitor, would leapfrog the firm (through the effect of user innovation) if the firm kept module a proprietary. (The same applies to the wedge between β and $\hat{\beta}_o$.)

[9] We note that all of the paper's results go through for any structure of demand such that maximized profits are an increasing function of $V_c - V_o$.

[10] A detailed proof can be obtained from the authors upon request.

Figure 2 **Optimal Business Models Under Compatibility**



We read Figure 2 as follows: when $a > \hat{\alpha}_o$ and $\beta < b < \hat{\beta}_o$, the only business models that may be optimal are O_{ab} and M_b (therefore, the firm may disregard the other seven business models P, M_a, M_α, M_β, $O_{a\beta}$, $O_{\alpha b}$, and $O_{\alpha\beta}$ in this case); or when $a < \alpha$ and $b < \beta$ the optimal business model is $O_{\alpha\beta}$, regardless of all other parameter values (therefore, the firm may disregard the other eight business models in this case); and so on.

It is surprising that complementarity alone, as captured by the assumption of increasing differences, allows us to simplify the cardinality of the strategy set so considerably. In eight of nine quadrants in Figure 2, there is either one single comparison to be made or no comparison at all. In the top-right quadrant, six comparisons must be made. Without the proposition, there are always $((9^2 - 9)/2{=})36$ comparisons.

Proposition 1 is complex in that there are many results embedded in Figure 2. Indeed, the proof of this proposition is several pages long, and it is composed of six intermediate lemmas. To present the intuition, we break the result into the five smaller remarks, which are easier to digest.[11]

Remark 1. The optimal business model always embeds the highest-quality modules available.

For example, when $a > \alpha$ and $b < \beta$, only business models that use a and β may be optimal. This result is a consequence of complementarity between the software modules and service. Suppose, for example, that $a > \alpha$ and that the commercial firm is considering an open-core business model. In this case, the firm is comparing M_a and M_α. The firm might be tempted to choose M_α because opening a (when $a > \alpha$) implies that the open-source competitor will adopt a and will end up being of higher quality (and, as a consequence, a more formidable competitor) than if the firm chose M_α. However, a's complementarity with b and z implies that the quality of the firm's product (V_c) increases relatively more than the quality of the open-source product (V_o) when a is opened. And because profit is $\pi = (V_c - V_o)/4$, the firm will prefer M_a to M_α.

Remark 1 shows that the firm will find it optimal to replace its own software with that of the open-source competitor if the latter has better quality. An example is IBM's support of Linux. IBM had several competing operating systems (like Z/OS), but it began supporting Linux because it was of higher quality, had a growing user base, and it could profit from selling Linux-related support and consultancy services. Currently, IBM provides support for over 500 software products running on Linux and has more than 15,000 Linux-related customers worldwide.[12]

Remark 2. If $a < \hat{\alpha}_o$, then the firm will always adopt a business model with open a or α (a business model in the set $\{M_a, M_\alpha, O_{ab}, O_{a\beta}, O_{\alpha b}$, or $O_{\alpha\beta}\}$). If $b < \hat{\beta}_o$, then the firm will always adopt a business model with open b or β (a business model in $\{M_b, M_\beta, O_{ab}, O_{a\beta}, O_{\alpha b}$, or $O_{\alpha\beta}\}$).

Suppose that the firm does not open module a or that it does not adopt α when $a < \hat{\alpha}_o$. In this case, the open-source competitor winds up with a higher-quality core module than the commercial firm. By the same argument to that following Remark 1, the commercial firm gains more than the open-source competitor by either opening a (if $\alpha < a < \hat{\alpha}_o$) or by adopting α (if $a < \alpha$). (The same applies to module b.)

Remark 3. When $a < \hat{\alpha}_o$ and $b < \hat{\beta}_o$, the firm will adopt an open business model (O_{ab}, $O_{a\beta}$, $O_{\alpha b}$, or $O_{\alpha\beta}$).

The intuition is analogous to that for Remark 2. The set $\{O_{ab}, O_{a\beta}, O_{\alpha b}$, or $O_{\alpha\beta}\}$ is the intersection of the set of business models that may be optimal when $a < \hat{\alpha}_o$ and the set that may be optimal when $b < \hat{\beta}_o$.

Remark 4. The proprietary business model P may only be optimal when $a > \hat{\alpha}_o$ and $b > \hat{\beta}_o$.

By Remark 2, when $a < \hat{\alpha}_o$ or $b < \hat{\beta}_o$, P may not be optimal because the firm is better off adopting one of the open-source modules of its competitor or by opening a or b. When $a > \hat{\alpha}_o$ and $b > \hat{\beta}_o$, however, the better value capture allowed by the proprietary model may more than compensate for the loss in value creation that results from P not taking advantage of user innovation.

Remark 5. If $\alpha < a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$, then $\pi(M_a) > \pi(M_b)$. If $a > \hat{\alpha}_o$ and $\beta < b < \hat{\beta}_o$, then $\pi(M_b) > \pi(M_a)$.

Remark 5 delivers a powerful message: if the firm chooses a mixed-source business model, then it will

[11] The remarks follow from direct inspection of Figure 2.

[12] See http://www.ibm.com/linux/ (accessed September 1, 2010).

open the module that *substitutes* the highest-quality module of the open-source competitor, and never the module that *complements* it.

To understand this result, note that when $\alpha < a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$, profits are $\pi(M_a) = V(a_o, b, z) - V(a_o, \hat{\beta}_o, 0)$ and $\pi(M_b) = V(a, b_o, z) - V(\hat{\alpha}_o, b_o, 0)$. Suppose that the firm is currently using the open-edge business model M_b and is considering the profit implications of switching to the open-core business model M_a. With this switch, the quality of its product will move from $V(a, b_o, z)$ to $V(a_o, b, z)$, whereas that of the outside open-source project will move from $V(\hat{\alpha}_o, b_o, 0)$ to $V(a_o, \hat{\beta}_o, 0)$. Note that the quality of the commercial firm's product will increase because $a_o > a$, but it will also decrease because $b < b_o$. Likewise, the quality of the open-source project's product will increase because $a_o > \hat{\alpha}_o$, but it will also decrease because $\hat{\beta}_o < b_o$. The key is to notice that $|a_o - a| > |a_o - \hat{\alpha}_o|$ and $|b_o - b| < |b_o - \hat{\beta}_o|$. In words, when $\alpha < a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$, the commercial firm's increase in value in the core module from opening a is larger than the increase in value to the open-source competitor (who will now adopt a), and the commercial firm's decrease in value in the extensions module from closing b is lower than the increase in value to the open-source competitor. With this, moving from M_b to M_a implies

$$\underbrace{V(a_o, b, z) - V(a, b_o, z)}_{\text{Difference in value in firm's product}} > \underbrace{V(a_o, \hat{\beta}_o, 0) - V(\hat{\alpha}_o, b_o, 0)}_{\text{Difference in value in OS's product}}$$

and, therefore, $\pi(M_a) > \pi(M_b)$. (A similar argument applies when $a > \hat{\alpha}_o$ and $\beta < b < \hat{\beta}_o$ to show that $\pi(M_b) > \pi(M_a)$.)

Having discussed the most important features of Proposition 1, we now present one important additional result. Figure 2 shows that when $a < \hat{\alpha}_o$ and $b < \hat{\beta}_o$ the firm will adopt an open-source business model. When $a > \hat{\alpha}_o$ and $b < \hat{\beta}_o$ or when $a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$, the firm may adopt a mixed business model. And only when $a > \hat{\alpha}_o$ and $b > \hat{\beta}_o$, the proprietary business model may be optimal. Therefore, the following applies:

Corollary 1. *The larger the quality difference between the firm's modules and those of the open-source competitor, the less the likelihood that the firm will open modules.*

Intuitively, the larger the quality of the firm's modules relative to those of the open-source competitor, the more the jump up in value of the open-source competitor if it adopts the firm's modules. Thus, when $a - \alpha$ and $b - \beta$ are large, and a and/or b are opened and adopted by the competitor, the positive effect from stronger user innovation is outweighed by the negative effect of increased competitive pressure.

4.1. Decreasing Returns to Complementarity and User Innovation

Proposition 1 shows all we can derive by only assuming that V has increasing differences. We now show that we can obtain tighter results by imposing further structure on V. Specifically, we present a condition that guarantees the following:

(i) There exist values of the parameters for which M_a, M_b, M_α, M_β, and P are guaranteed to be optimal.

(ii) If a mixed-source business model is preferred to an open-source business model for given values of a and b, it is still preferred for $a' \geq a$ and $b' \geq b$. Likewise, if P is optimal for a and b, it is still optimal for $a' \geq a$ and $b' \geq b$.

Let D_x and D_y be defined as follows:

$$\begin{aligned} D_x(x, y, z) &= V(F(x), y, z) - V(x, y, z) \\ &\quad - (V(F(x), y, 0) - V(x, y, 0)), \\ D_y(x, y, z) &= V(x, G(y), z) - V(x, y, z) \\ &\quad - (V(x, G(y), 0) - V(x, y, 0)), \end{aligned}$$

where $F(x) = f(x, q_x; \sigma_A)$ and $G(x) = g(y, q_y; \sigma_B)$.[13] Our assumption is that there are "decreasing returns" to increasing differences:

Assumption 2. *D_x and D_y are decreasing in x and y.*

The condition places a restriction on the strength of complementarity between the modules, and between the modules and service. Increasing differences means that the returns of increasing one variable rise when another variable increases. This effect is magnified in our model by the assumption of user innovation because the value of a module increases by $f(x, q_x; \sigma_A) - x$ or $g(y, q_y; \sigma_B) - y$ when opened. What Assumption 2 requires is that the combined effect of complementarity and user innovation decreases as the value of the modules increases.

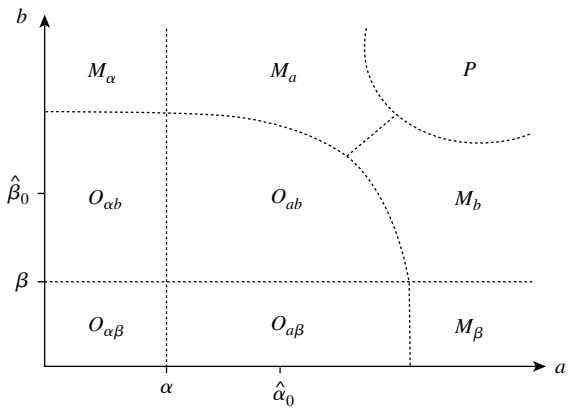
Proposition 2. *Suppose Assumption 2 holds. The optimal business model under compatibility depends on the relative qualities of the available software modules (a, b, α, and β), as shown in Figure 3.*[14]

To see the role of Assumption 2, note that without it we cannot guarantee that a mixed model is ever optimal. If complementarity or user innovation

[13] Note that increasing differences (which we maintain) implies that $D_x > 0$ and $D_y > 0$.

[14] We have represented the boundaries between the regions corresponding to the different business models as smooth lines. Our theoretical results imply that the lines between M_α and O_{ab} and between M_β and $O_{a\beta}$ are straight lines as shown in the figure. The exact shape of the other lines depends on the specific functional forms under consideration. However, we know that all the regions will exist and that their relative locations with respect to α, $\hat{\alpha}$, β, and $\hat{\beta}$ will be as shown in the figure.

Figure 3 **Optimal Business Models Under Compatibility with Assumption 2**



strengthen as a or b increase, the value of open business models grows with a and b, and therefore a mixed model may never be optimal.

4.2. Comparative Statics

Having narrowed down to the maximal possible extent the set of possibly optimal business models in our setting with general V, f, and g, we now study how business model choice varies with the value of the complementary service z and the extent of user innovation σ_x and σ_y.

Lemma 1. *As z increases, the region of parameters for which open models are optimal becomes larger, as does the region of parameters for which mixed models are preferred to proprietary models.*

This result follows from complementarity. Suppose that we are in a region where an open-source or a mixed-source model may be optimal. In this case, it is always true that the open-source model maximizes the value of the firm's product. The only reason why the firm may choose not to use it is that at the same time, this model creates relatively too much value for the open-source competitor. However, as z increases, the complementarity between z and the software modules implies that the difference between values for the open-source model increases more than the difference in values for the mixed model, thereby making it more likely that the optimal model is the open-source model.

Note also that as z increases, the region of parameters for which open models are optimal becomes larger, and the region of parameters for which proprietary models are optimal becomes smaller. However, the effect on the region of parameters for which mixed models are optimal depends on the relative size of the changes in the other two regions.

Lemma 2. *As σ_A increases, the regions for which $\pi(O_{ab}) > \pi(M_b)$ and $\pi(O_{\alpha\beta}) > \pi(M_\beta)$ become larger. Likewise, as σ_B increases, the regions for which $\pi(O_{ab}) > \pi(M_a)$ and $\pi(O_{\alpha\beta}) > \pi(M_\alpha)$ become larger. Finally, as σ_A or σ_B increase, the region of parameters for which P is optimal becomes smaller.*

Lemma 2 shows an interesting result. When the extent of user innovation increases for one of the modules, the mixed model based on the *other* module becomes less attractive compared to the open-source business models. Notice, however, that based solely on the assumption that V has increasing differences, we cannot establish a similar result for the comparison of profits between the mixed model based on the module for which user innovation increases and open business models.

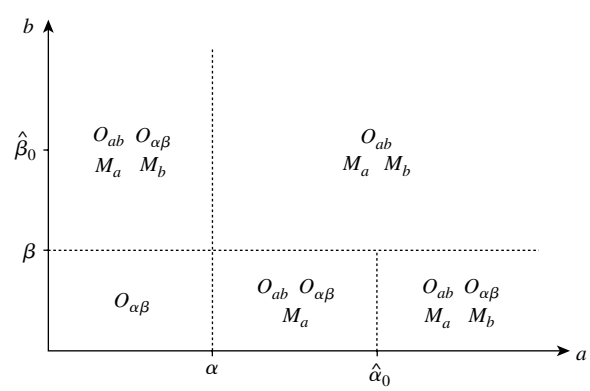
5. Incompatibility

We now solve the model under the assumption that modules from different developers are incompatible and, thus, may not be combined. We begin by noting that, for technological reasons, there are two business models that are unavailable: $O_{a\beta}$ and $O_{\alpha b}$. For example, $O_{\alpha b}$ would require the combination of α and b, but this is impossible when the modules are incompatible. Therefore, there are seven business models available to the commercial firm when modules are incompatible. This means that there are potentially $((7^2-7)/2{=})21$ business model comparisons to be made in the case of incompatibility. We begin the analysis with a result that narrows down to the maximal possible extent the set of business models that may be optimal in a setting with general V, f, and g.

Proposition 3 (Optimal Business Models Under Incompatibility). *Under incompatibility, the optimal business model depends on the relative qualities of the available software modules (a, b, α, and β), as shown in Figure 4.*

Direct inspection of Figures 2 and 4 reveals that for most combinations of parameters a, b, α, and β there are fewer business models that we may discard when modules are incompatible. To understand

Figure 4 **Optimal Business Models Under Incompatibility**



why, consider, for example, the quadrant $\alpha < a < \hat{\alpha}_o$ and $b < \beta$. If the firm wanted to compete through an open business model and could choose any one of the four open models (O_{ab}, $O_{\alpha b}$, $O_{a\beta}$, and $O_{\alpha\beta}$), it would choose $O_{a\beta}$. Under incompatibility, however, $O_{a\beta}$ is technologically impossible because it combines modules from different developers. Thus, the firm must settle with either O_{ab} or $O_{\alpha\beta}$. However, with the general value function V that we have assumed, it is impossible to tell whether $\pi(O_{ab})$ is larger or smaller than $\pi(O_{\alpha\beta})$ when $\alpha < a < \hat{\alpha}_o$ and $b < \beta$. Either one may happen for Vs that satisfy Assumption 1 (§2.2).

The following remarks help us unpack the content of Proposition 3.

Remark 6. Business models P, M_α, and M_β are never chosen.

Clearly, the proprietary model is dominated by the open-edge model. Because under incompatibility b may not be combined with α, M_b always leads to higher profits than P because it takes advantage of user innovation without strengthening the open-source competitor's product.[15] For the firm to adopt M_α when modules are incompatible, it must give up on its b module. With this, $\pi(M_\alpha) = (V(\alpha_o, 0, z) - V(\alpha_o, \hat{\beta}_o, 0))/4$. Increasing differences imply that the firm will earn more by also adopting β. In this case, $\pi(O_{\alpha\beta}) = (V(\alpha_o, \beta_o, z) - V(\alpha_o, \hat{\beta}_o, 0))/4$. Finally, for the firm to adopt M_β, it must give up on its a module. In this case, the firm's product has no value as $V(0, \beta_o, z) = 0$. Because the firm can always guarantee positive profit for itself by choosing $O_{\alpha\beta}$, M_β is never chosen.

Remark 7. The firm's product may not embed the highest-quality modules available in equilibrium.

Contrary to compatibility, when $a > \alpha$ but $b < \beta$ (or vice versa), it is technologically impossible for the firm to sell a product that embeds all of the highest-quality modules. However, O_{ab} or $O_{\alpha\beta}$ may result in higher profit in this case (this happens, for example, when σ_A, σ_B, and z are large).

Remark 8. Opening b has substantially different implications than opening a. Thus, contrary to Figure 2, Figure 4 is not symmetric.

When b is opened, it is never adopted by the open-source competitor because b may not be combined with α and cannot be used standalone. When a is opened, however, it may be adopted when the value of a to the open-source competitor $V(a_o, 0, 0)$ is larger than the value of its own open-source system $V(\hat{\alpha}_o, \hat{\beta}_o, 0)$.

As it turns out, the firm can take advantage of the possibility that an open a is adopted by the open-source competitor when $V(a_o, 0, 0) > V(\hat{\alpha}_o, \hat{\beta}_o, 0)$. Specifically, when $a < \alpha$ and $\beta < b < \hat{\beta}_o$, the open-source competitor may adopt a and renounce β. This boosts user innovation on the core a and the firm benefits substantially given the complementarity between a, b, and z. Obviously, this tactic cannot be applied to b given that it is worthless to the open-source competitor without a, $V(0, b_o, 0) = 0$. Thus, the open-source competitor will never renounce to α to adopt b (alone). We conclude that the asymmetry between the core and the edge modules plays an important role under incompatibility.

Remark 9. The two mixed business models, M_a and M_b, are more desirable than in the case of compatibility.

Clearly, if the firm chooses to compete through M_b, it is protected from the adoption of b when modules are incompatible. Under compatibility, b may be adopted and combined with α to increase the quality of the open-source competitor. When the firm chooses M_a and modules are incompatible, a will be adopted only if $V(a_o, 0, 0) > V(\hat{\alpha}_o, \hat{\beta}_o, 0)$ (because the open-source competitor has to disregard β if a is adopted). Under compatibility, however, a will be adopted if $V(a_o, \hat{\beta}_o, 0) > V(\hat{\alpha}_o, \hat{\beta}_o, 0)$ and, thus, it is more likely that a and b will be adopted in this case. Combined with the fact that P is never optimal (Remark 6), this is why we see that M_a and M_b are more prevalent in Figure 4 than in Figure 2.

Our findings show the relevance of the open-edge business model, which is an inexpensive way to become open. Microsoft's .Net framework and Stata are two examples. In the first case, Microsoft is committed to opening the languages that can be compiled with .Net (Visual Basic, C#, J#, etc.), even to the point of promoting open standards. Users of those languages, however, need .Net (which is kept proprietary) to compile the code that they develop. Likewise, StataCorp has opened hundreds of ado files, which are programs that implement econometric techniques used to perform specific tasks (such as maximum-likelihood estimations for particular econometric models). Whereas the ado files are open, users need to use Stata (which is kept proprietary) to compile those programs.

Having discussed the most important features of Proposition 3, we now present one important additional result. Figure 2 reveals that the firm always adopts an open business model when $a < \hat{\alpha}_o$ and $b < \hat{\beta}_o$ under compatibility. Under incompatibility, however, mixed models may be optimal for these

[15] More generally, the firm can open all those technologies, protocols, and ideas which have no value unless used with the base module, instead of remaining completely closed.

parameter values. Therefore, the following applies:

Corollary 2. *When the firm's modules are of low quality ($a < \hat{\alpha}_o$ and $b < \hat{\beta}_o$), the firm is more closed under incompatibility that under compatibility.*

As noted above, a technological constraint precludes the firm from choosing open business models such as $O_{a\beta}$ or $O_{\alpha b}$. In choosing among the remaining business models, the firm takes into account that if it opens b it will not be adopted, and that if it opens a it will rarely be adopted because it is of low or intermediate quality and the open-source competitor would need to give up its extensions module β if it adopted the firm's base. Given this, the firm is protected from imitation while benefiting from user innovation.

5.1. Decreasing Returns to Complementarity and User Innovation

We now explore the effects of imposing Assumption 2 on the optimal choice of business model under imcompatibility. Proposition 4 shows that we can further restrict the parameter sets over which business models are possibly optimal.

Proposition 4. *Under Assumption* 2, *there is a region of parameters where M_a or M_b are optimal. Also, if a mixed model yields higher profits than an open-source model at (a', b'), then it yields higher profits at (a'', b''), where $a'' > a'$ and $b'' > b'$.*

Although Proposition 4 is helpful to further narrow down the set of possibly optimal business models, contrary to the case of compatibility, imposing the additional Assumption 2 under incompatibility is insufficient to fully determine regions of parameters for which one, and only one, business model is optimal.

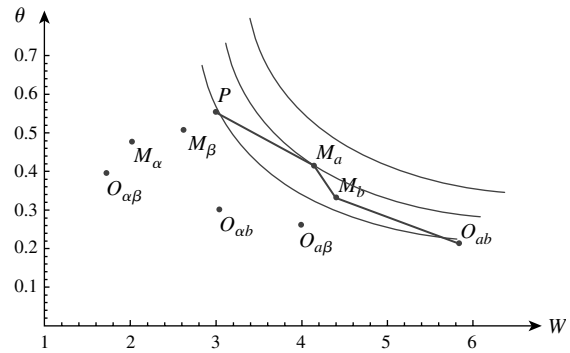
5.2. Comparative Statics

The comparative statics results for incompatibility are essentially the same as in the case of compatibility: Lemmas 1 and 2 hold unchanged.

6. Value Creation and Value Capture

The profit-maximizing choice of business model depends on the resolution of a trade-off between value creation and value capture, defined as follows:[16]

Figure 5 **The Nine Business Models, Efficient Frontier, and Isoprofits**



Value Creation (W)**:** Value creation is the sum of the firm's profits and consumer surplus: $W = CS + \pi$.

Value Capture (θ)**:** Value capture is the proportion of the value created that is appropriated by the firm (in the form of profits): $\theta = \pi/(CS + \pi)$.

Using these definitions, firm profits can be expressed as $\pi = \theta W$. It is straightforward to compute W and θ:

$$W = \frac{3}{8}V_c + \frac{1}{8}V_o, \quad \theta = \frac{2(V_c - V_o)}{3V_c + V_o}.$$

It is helpful to illustrate W and θ through a numerical example. Consider the case of compatibility and the following functional forms: $V = (a + b + z)^\gamma$, $f(x, q_x; \sigma_A) = x \cdot (1 + q_x \sigma_A)$ and $g(y, q_y; \sigma_B) = y \cdot (1 + q_y \sigma_B)$, with $a = 1.1$, $b = 1$, $\alpha = 0.3$, $\beta = 0.5$, $z = 1$, $\sigma_A = \sigma_B = 0.5$, and $\gamma = 1.8$.[17] Figure 5 presents the business models available to the firm on $\{W, \theta\}$ space. The figure reveals that five business models ($O_{a\beta}$, $O_{\alpha b}$, $O_{\alpha\beta}$, M_α, and M_β) are dominated by business models that provide both greater value creation and greater value capture (P, M_a, M_b, and O_{ab}). Considering the four undominated models, we can draw an "efficient business models frontier" that illustrates the trade-off between value capture and value creation: as we move on the frontier from business models with low value creation toward those with higher value creation, value capture decreases. Which business model is optimal depends on the resolution of this trade-off. Figure 5 also shows isoprofit curves corresponding to different profit levels.[18] Obviously, the

[16] The expressions *value creation* and *value capture* have become mainstream in strategy since the publications of Brandenburger and Stuart (1996) and Brandenburger and Nalebuff (1997). See also MacDonald and Ryall (2004). In Brandenburger and Stuart (1996), all transactions with positive value take place. Therefore, value created equals potential value. However, Brandenburger and Stuart also recognize that there may exist "frictions" that may introduce a wedge between potential and realized value. In our model, for example, the monopolist cannot price discriminate. Hence, there are transactions with positive potential value that do not occur in equilibrium (i.e., there is a welfare loss). Our definition of value creation corresponds with actual value created, which is more relevant for studying the choice of an optimal business model. For an insightful discussion of value creation and value capture and endogenous business models, see Salas-Fumás (2009).

[17] It is trivial to conduct the corresponding calculations for incompatibility.

[18] Combinations of θ and W which yield the same level of profit. Note that, given our definitions, profit is simply $\pi = \theta W$, and thus it is trivial to compute the isoprofit curves in $\{W, \theta\}$ space.

farther away the curves are from the origin, the higher the profit level that they represent. Thus, the optimal business model is the one that reaches the highest isoprofit curve. In this particular example, M_a is the profit-maximizing business model.

Having illustrated the notions of value creation and value capture through numerical examples, we now turn to the general results. What follows applies to both compatibility regimes. Proposition 5 shows that open business models provide maximal value creation (although, as we noted above, they do not necessarily maximize profit).

Proposition 5. *Value creation is always maximal with the open business model that combines the highest-quality modules available.*

In terms of the efficient business model frontier, Proposition 5 implies that there will always be an open-source business model on the frontier, which will be located on the extreme right of the frontier.

Our second result follows directly from Proposition 5.

Corollary 3. *If $a < \hat{\alpha}_o$ and $b < \hat{\beta}_o$, then value creation under compatibility is always greater than or equal to value creation under incompatibility.*

For general V, the profit-maximizing business model may not be the model that maximizes value capture (defined as the portion of value created W that is appropriated in the form of profit). Proposition 6 shows that if the complementarities between modules and service are strong (as in the value function V being log-supermodular), maximizing value capture is equivalent to maximizing profits.[19]

Proposition 6. *If V is log-supermodular, the ranking of business models in terms of value capture is the same as the ranking in terms of profits.*

The proposition says that if V is log-supermodular, the optimal business models as given in Figures 2 and 4 are also the business models that maximize the portion of total value that the firm captures. Thus, when complementarity is strong, all that firms must do to ensure maximal profits is to choose the business model that maximizes the "portion of the pie" that the available models generate, regardless of "size of the pie" that they produce.

One important implication of Propositions 5 and 6 is that whenever Figures 2 or 4 prescribe that an open business model is optimal, then there is no trade-off between value creation and value capture: Proposition 5 says that the open business model maximizes value creation and Proposition 6 says that if V is log-supermodular, value capture is also maximized with that same business model.

Corollary 4. *Becoming more open may lead to higher value capture.*

The corollary goes against the conventional wisdom that "open source always creates more value at the expense of reducing the fraction of value captured by the firm." In some cases, openness goes hand-in-hand with more value capture, and the open-source business model is unambiguously optimal from both the private and the social point of view.

7. Choice Between Compatibility and Incompatibility

As we have seen, the choice of business model depends on whether the firm's modules are compatible with those of the open-source project or not. This raises the question of whether the firm will choose to make its programs compatible in the first place whenever this decision is possible.[20]

In our setting, the firm choosing a compatibility regime means that it considers 13 business models in its competition against the outside open-source project: P, O_{ab}, $O_{\alpha\beta}$, $O_{a\beta}$, $O_{\alpha b}$, M_a^c, M_b^c, M_a^i, M_b^i, M_α^c, M_β^c, M_α^i, and M_β^i, where superscript c stands for compatibility and i for incompatibility. Notice that P, O_{ab}, and $O_{\alpha\beta}$ yield the same profits under compatibility and incompatibility, and that $O_{a\beta}$ and $O_{\alpha b}$ are only available under compatibility.

Incompatibility affects profits in three ways vis-à-vis compatibility. First, incompatibility implies that some potentially desirable combinations of modules of the firm with those of the open-source project are not possible. Second, it implies that at times, user innovation is not maximized (sometimes the firm may open a module that is not adopted by the open-source project, and thus only half of the users improve on it). Third, incompatibility reduces the likelihood that the open-source project adopts the modules of the firm under mixed models or minimizes the value of the open-source product when the open-source project adopts them. The first two effects make it less likely for the firm to choose incompatibility, and the third one makes it more likely. Depending on the shapes of

[19] Function V is log-supermodular if $\log V$ is supermodular. Log-supermodularity implies supermodularity if V is monotone (as is our case), and therefore, it implies increasing differences. The reverse implications, however, are not true. It is important to remark that all we need for the result in Proposition 6 is that $\log V$ has increasing differences, which is a weaker condition than log-supermodularity.

[20] In some cases, this decision is external to the firm (for example, Linux was designed to be compatible with proprietary Unix systems, and this decision was taken by the open-source developers), or it may be difficult to retrofit existing modules to make them compatible/incompatible with new open-source software that may become available.

V, f, and g, the net result of the three effects may be positive or negative. Nevertheless, the following three results are general.

Lemma 3. *If it is optimal for the firm to choose an open-source model under incompatibility, then the firm may gain (but never lose) by choosing compatibility.*

To understand this lemma, recall that when $a < \alpha$ and $b < \beta$, or $a > \alpha$ and $b > \beta$, the optimal open-source business model under compatibility and incompatibility coincide (see Figures 2 and 4). Therefore, in this case compatibility is indifferent to incompatibility. When $a < \alpha$ and $b > \beta$, on the other hand, the optimal business model under incompatibility is either O_{ab} or $O_{\alpha\beta}$, whereas the optimal business model under compatibility is $O_{\alpha b}$, which, by Proposition 1, yields higher profit than O_{ab} or $O_{\alpha\beta}$.[21]

The following lemma compares mixed-source business models under compatibility and incompatibility.

Lemma 4. (a) *If* $V(a_o, 0, 0) > V(\hat{\alpha}_o, \hat{\beta}_o, 0)$, *then* $\pi(M_a^i) > \pi(M_a^c)$.
(b) *If* $a < \alpha$ *and* $b > \beta$, *then* $\pi(M_\alpha^c) > \pi(M_a^i)$.
(c) *If* $a > \alpha$ *and* $b < \beta$, *then* $\pi(M_\beta^c) > \pi(M_b^i)$.

The condition in part (a) says the open-source project adopts module a if the firm chooses to open it. When this is the case, $\pi(M_a^i) = \frac{1}{4}(V(a_o, b, z) - V(a_o, 0, 0))$ and $\pi(M_a^c) = \frac{1}{4}(V(a_o, b, z) - V(a_o, \hat{b}_o, 0))$. Therefore, $\pi(M_a^i) > \pi(M_a^c)$ because $V(a_o, 0, 0) < V(a_o, \hat{b}_o, 0)$. In words, the value of the product offered by the open-source competitor is lower under compatibility in this case.[22]

To understand part (b), note that when $a < \alpha$ and $b > \beta$, if the firm chooses to compete with business model M_a^i, then module a is not adopted by the open-source competitor and only half of the market ends up using it. If the firm chooses M_α^c, the base module is of the same quality than that of the open-source competitor, but the firm benefits more from complementarities because $b > \beta$. Therefore, $\pi(M_\alpha^c) > \pi(M_a^i)$. The intuition for part (c) is analogous.

The final lemma says that when both modules of the firm are of much larger quality than those of the open-source project, under Assumption 2 incompatibility is preferable.

Lemma 5. *Under Assumption* 2, *incompatibility is always optimal when a and b are large.*

From Proposition 2 we know that under Assumption 2, P is optimal under compatibility for a and b large enough. By Proposition 3 we also know that under incompatibility $\pi(M_b^i) > \pi(P)$. Therefore, incompatibility is preferred when a and b are large.

[21] The case $a > \alpha$ and $b < \beta$ is analogous.

[22] Note that an analogous argument to (a) cannot hold for the extensions b because $V(0, b_o, 0) = 0 < V(\hat{\alpha}_o, \hat{\beta}_o, 0)$.

8. Concluding Remarks

We have analyzed a model where a profit-maximizing firm with modular software must decide which modules to open and which to keep proprietary. We have allowed the firm to choose between two "pure" business models (open source and proprietary) and two "mixed" models (open core and open edge). The analysis has produced several novel findings that can be applied to the design of optimal business strategies. Indeed, Figures 2–4 yield helpful insights that may be used directly by managers in the strategy formulation process. Given the recent outburst of attention on hybrids between open-source and proprietary business models, the topic deserves further study and the obvious next step is relaxing two convenient, though limiting, assumptions: the absence of competition between profit-maximizing firms and the absence of horizontal differentiation.[23]

Our model has clear empirical implications, among which the most important are the following: (i) as the quality differential between the software of commercial firms and that of open-source competitors increases, it is more likely that the firm will adopt a mixed-source business model instead of an open-source business model; (ii) firms are more likely to open modules that substitute, rather than complement, modules offered by existing open-source projects; (iii) as the value of the goods and services complementary to the software increases, we should see more adoption of open-source and mixed-source business models; (iv) incompatibility makes it more desirable to adopt mixed-source and proprietary business models; and (v) firms choosing an open-source business model are more likely to pursue compatibility.

In addition to the literature on open source, we aimed to contribute to recent work in strategy that analyzes competitive interactions between organizations with different business models. Interest in this line of research has increased in the past decade as new technologies, regulatory changes, and new customer demands have allowed firms to implement new approaches to competing in a wide range of industries spanning from airlines (e.g., Ryanair) to furniture (e.g., IKEA) and from the circus (e.g., Cirque du Soleil) to betting (e.g., Betfair). In fact, many of the fastest-growing firms in the recent past appear to have taken advantage of opportunities sparked by

[23] An early version of this paper tackled the issue of competition between profit-maximizing firms that choose business models. Given the complexity of the game, the analysis was conducted through numerical simulations. Although the results were suggestive, they were derived by use of particular functional forms and, contrary to the analysis herein, were not general.

globalization, deregulation, or technological change to "compete differently" and to innovate in their business models (for additional examples, see Kim and Mauborgne 2005, Markides 2008).

A central question that this literature wrestles with is as follows: Where do the business models that we observe come from? Our contention is that, to a large extent, the configuration of business models in an industry is the equilibrium outcome of a search process for higher profits. We have proposed and illustrated a methodology for the study of endogenous business models; a two-period game where in the first period business models are chosen and in the second period firms interact in the marketplace to attract users.

Our approach to studying business model choice has similarities to the biform games introduced by Brandenburger and Stuart (2007) (for applications to strategy, see Chatain and Zemsky 2007, Adner et al. 2010). A biform game is a two-period game in which players first make noncooperative strategic choices that determine the cooperative subgame played in the second period. The difference between our approach and biform games is that we model the second period as noncooperative. Therefore, the size of the final "pie of value" in our setting depends on the first-period business model choices *and* the second-period pricing choices. In the biform game formulation, the size of the pie remains fixed in the second period and is determined by the first-period choices.

The generic two-period game that we have presented can be applied to all sorts of competitive settings, such as strategies to fight ad-sponsored rivals, strategies to fight low-cost entrants, strategies to fight platform players, and the like. We hope to have provided a solid first step toward a general framework for the study of competition through business models.

Acknowledgments

The authors thank the editors and referees for comments that helped improve the paper substantially. They specially thank Josh Lerner and Andres Hervas-Drane for detailed comments on an early draft. The authors also thank Lynda Applegate, Bhaskar Chakravorti, Ramiro de Elejalde, Joan Enric Ricart, Christian Ruzzier, Jorge Tarziján, Stefano Trento, and seminar participants at the Harvard Business School (HBS), Massachusetts Institute of Technology, IESE Business School, Pontificia Universidad Católica de Chile, and Universitat Autònoma de Barcelona for useful comments and suggestions. G. Llanes thanks the Entrepreneurship Unit at HBS where he conducted part of this research as a postdoctoral fellow. The authors gratefully acknowledge financial support from the HBS Division of Research and the NET Institute (http://www.netinst.org). R. Casadesus-Masanell also thanks the Public-Private Sector Center at IESE Business School for financial support.

Appendix. Proofs of Propositions and Lemmas in Text

Proof of Proposition 1

This proof follows several steps.

Lemma A1. *If $a < \alpha$, then the firm will include α in its commercial product. If $b < \beta$, then the firm will include β in its commercial product.*

Proof. We start by assuming that $a < \alpha$. Suppose the firm bases its software on a. We will show that in this case, there is always some other business model in which the firm adopts α and increases its profit.

When the firm bases its software on a, profits depend on whether the firm opens a or keeps it closed. Let x_c represent the value of the base module of the firm. Then, $x_c = a$ if a is closed, and $x_c = \hat{a}_o$ if a is open. In any case, given that $a < \alpha$, it is always true that $x_c < \hat{\alpha}_o$.

Let π_a represent the profits of the firm if it uses base a:

$$\pi_a = (V(x_c, y_c, z) - V(\hat{\alpha}_o, y_o, 0))/4.$$

If the firm adopts α instead,

$$\pi_\alpha = (V(\alpha_o, y_c, z) - V(\alpha_o, y_o, 0))/4,$$

where y_c and y_o represent the value of the extensions of the commercial firm and the open-source competitor, which depend on b, β, and on the decisions to open and adopt modules.

Rearranging the difference between the profits, we obtain

$$\begin{aligned} &4(\pi_\alpha - \pi_a) \\ &\quad = V(\alpha_o, y_c, z) - V(x_c, y_c, z) - (V(\alpha_o, y_o, 0) - V(\hat{\alpha}_o, y_o, 0)) \\ &\quad \geq V(\alpha_o, y_c, z) - V(x_c, y_c, z) - (V(\alpha_o, y_o, 0) - V(x_c, y_o, 0)). \end{aligned}$$

There are two possibilities. If $y_c \geq y_o$, then $\pi_\alpha - \pi_a > 0$ by increasing differences, and we already have the result. If, on the other hand, $y_c < y_o$, the proof requires an aditional step. In particular, the only cases in which $y_c < y_o$ are $y_c = b$, $y_o = \hat{\beta}_o$, with $b < \hat{\beta}_o$, and $y_c = \hat{b}_o$, $y_o = \hat{\beta}_o$, with $\hat{b}_o < \hat{\beta}_o$. Consider now the case in which the firm adopts *both* α and β. Profits become

$$\pi_{\alpha\beta} = (V(\alpha_o, \beta_o, z) - V(\alpha_o, \beta_o, 0))/4.$$

The difference in profits is now

$$\begin{aligned} &4(\pi_{\alpha\beta} - \pi_a) \\ &\quad = V(\alpha_o, \beta_o, z) - V(x_c, y_c, z) - (V(\alpha_o, \beta_o, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)) \\ &\quad > V(\alpha_o, \beta_o, z) - V(x_c, \hat{\beta}_o, z) - (V(\alpha_o, \beta_o, 0) - V(x_c, \hat{\beta}_o, 0)), \end{aligned}$$

where the last inequality arises because $x_c < \hat{\alpha}_o$ and $y_c < \hat{\beta}_o$. Then, $\pi_{\alpha\beta} - \pi_a > 0$ by increasing differences. Therefore, we have shown that if $a < \alpha$ there is always a business model in which the firm adopts α and increases its profit, and thus, in equilibrium, the firm will adopt α in its software.

The proof for $b < \beta$ follows the same steps. □

Lemma A2. *If $a > \alpha$, then the firm will use a in its commercial product. If $b > \beta$, then the firm will use b in its commercial product.*

Proof. We start by assuming that $a > \alpha$. Suppose the firm bases its software on α. We will show that in this case, there is always some other business model in which the firm uses a instead and increases its profit.

When the firm uses α its profits are

$$\pi_\alpha = (V(\alpha_o, y_c, z) - V(\alpha_o, y_o, 0))/4,$$

where y_c and y_o represent the value of the extensions of the commercial firm and the open-source competitor, which depend on b, β, and on the decisions to open and adopt modules.

Suppose now that the firm uses a in its product, and at the same time opens it. Given that $a > \alpha$, the open-source competitor will adopt a. Profits become

$$\pi_a = (V(a_o, y_c, z) - V(a_o, y_o, 0))/4.$$

Rearranging the difference in profits, we get

$$\begin{aligned} 4(\pi_a - \pi_\alpha) = {} & V(a_o, y_c, z) - V(\alpha_o, y_c, z) \\ & - (V(a_o, y_o, 0) - V(\alpha_o, y_o, 0)). \end{aligned}$$

There are two possibilities. If $y_c \geq y_o$, then $\pi_a - \pi_\alpha > 0$ by increasing differences, and we already have the result. If, on the other hand, $y_c < y_o$, the proof requires an additional step. In particular, the only cases in which $y_c < y_o$ are $y_c = b$, and $y_o = \hat{\beta}_o$, with $b < \hat{\beta}_o$, and $y_c = \hat{b}_o$, and $y_o = \hat{\beta}_o$, with $\hat{b}_o < \hat{\beta}_o$. Consider now the case in which the firm uses a and β. Profits become

$$\pi_{a\beta} = (V(a_o, \beta_o, z) - V(a_o, \beta_o, 0))/4.$$

The difference between profits becomes

$$\begin{aligned} & 4(\pi_{a\beta} - \pi_\alpha) \\ & \quad = V(a_o, \beta_o, z) - V(\alpha_o, y_c, z) - (V(a_o, \beta_o, 0) - V(\alpha_o, \hat{\beta}_o, 0)) \\ & \quad > V(a_o, \beta_o, z) - V(\alpha_o, \hat{\beta}_o, z) - (V(a_o, \beta_o, 0) - V(\alpha_o, \hat{\beta}_o, 0)). \end{aligned}$$

Then, $\pi_{a\beta} - \pi_a > 0$ by increasing differences. Therefore, we have shown that if $a > \alpha$ there is always a business model in which the firm uses a and increases its profit, and thus, in equilibrium, the firm will always include a in its software.

The proof for $b > \beta$ follows the same steps. □

Corollaries A1 and A2 follow from the above lemmas.

Corollary A1. *If it is optimal for the firm to use an open-source business model, then it will use the combination of base and extensions that maximizes the value of software.*

Corollary A2. *The comparison between the values of the modules a, b, α and β determines four regions of equilibria*: (i) *if $a < \alpha$ and $b < \beta$, then the optimal business model is $O_{\alpha\beta}$*; (ii) *if $a < \alpha$ and $b > \beta$, then the optimal business model is either M_α or $O_{\alpha b}$*; (iii) *if $a > \alpha$ and $b < \beta$, then the optimal business model is either M_β or $O_{a\beta}$*; *and* (iv) *if $a > \alpha$ and $b > \beta$, then the optimal business model is either P, M_a, M_b, or O_{ab}.*

In the following lemmas, we compare a and b with $\hat{\alpha}_o$ and $\hat{\beta}_o$.

Lemma A3. *Suppose that $a > \alpha$ and $b > \beta$. A necessary condition for $\pi(M_a) > \pi(O_{ab})$ is that $b > \hat{\beta}_o$, and a necessary condition for $\pi(M_b) > \pi(O_{ab})$ is that $a > \hat{\alpha}_o$.*

Proof. The difference in profits between M_a and O_{ab} is

$$\begin{aligned} & 4(\pi(M_a) - \pi(O_{ab})) \\ & \quad = (V(a_o, b, z) - V(a_o, \hat{\beta}_o, 0)) - (V(a_o, b_o, z) - V(a_o, b_o, 0)) \\ & \quad = (V(a_o, b_o, 0) - V(a_o, b, 0)) - (V(a_o, b_o, z) - V(a_o, b, z)) \\ & \qquad + (V(a_o, b, 0) - V(a_o, \hat{\beta}_o, 0)). \end{aligned}$$

Increasing differences imply that $V(a_o, b_o, z) - V(a_o, b, z) > V(a_o, b_o, 0) - V(a_o, b, 0)$. Therefore, for $\pi(M_a) > \pi(O_{ab})$ it is necessary that $V(a_o, b, 0) - V(a_o, \hat{\beta}_o, 0) > 0$, which is only possible if $b > \hat{\beta}_o$.

The proof for $a > \hat{\alpha}_o$ is analogous. □

Lemma A4. *If $\alpha < a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$, then $\pi(M_a) > \pi(P)$ and $\pi(M_a) > \pi(M_b)$. If $a > \hat{\alpha}_o$ and $\beta < b < \hat{\beta}_o$, then $\pi(M_b) > \pi(P)$ and $\pi(M_b) > \pi(M_a)$.*

Proof. Suppose $\alpha < a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$. Then, the difference in profits between M_a and P is

$$\begin{aligned} 4(\pi(M_a) - \pi(P)) = {} & (V(a_o, b, z) - V(a_o, \hat{\beta}_o, 0)) \\ & - (V(a, b, z) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)). \end{aligned}$$

Rearranging the above difference, we get

$$\begin{aligned} & 4(\pi(M_a) - \pi(P)) \\ & \quad = (V(a_o, b, z) - V(a, b, z)) - (V(a_o, \hat{\beta}_o, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)) \\ & \quad = (V(a_o, b, z) - V(a, b, z) - (V(a_o, b, 0) - V(a, b, 0))) \\ & \qquad + (V(a_o, b, 0) - V(a_o, \hat{\beta}_o, 0) - (V(a, b, 0) - V(a, \hat{\beta}_o, 0))) \\ & \qquad + (V(\hat{\alpha}_o, \hat{\beta}_o, 0) - V(a, \hat{\beta}_o, 0)). \end{aligned}$$

The first and second terms on the right-hand side of the previous equation are positive by increasing differences, and the third term is positive because $a < \hat{\alpha}_o$ by assumption. It follows that $\pi(M_a) - \pi(P) > 0$.

The difference in profits between M_a and M_b is

$$\begin{aligned} & 4(\pi(M_a) - \pi(M_b)) \\ & \quad = (V(a_o, b, z) - V(a_o, \hat{\beta}_o, 0)) - (V(a, b_o, z) - V(\hat{\alpha}_o, b_o, 0)) \\ & \quad = (V(a_o, b, z) - V(a, b_o, z) - (V(a_o, b, 0) - V(a, b_o, 0))) \\ & \qquad + (V(a_o, b, 0) - V(a_o, \hat{\beta}_o, 0)) + (V(\hat{\alpha}_o, b_o, 0) - V(a, b_o, 0)). \end{aligned}$$

The first term of the right-hand side of the previous equation is positive by increasing differences. The second and third terms are positive because $a < \hat{\alpha}_o$ and $b > \hat{\beta}_o$ by assumption. This means that $\pi(M_a) - \pi(M_b) > 0$, which proves the result.

The proof for $a > \hat{\alpha}_o$ and $\beta < b < \hat{\beta}_o$ is analogous. □

Lemma A5. *Suppose $a > \alpha$ and $b > \beta$. If $V(a, b, 0) < V(\hat{\alpha}_o, \hat{\beta}_o, 0)$, then $\pi(O_{ab}) > \pi(P)$. (A sufficient condition for $V(a, b, 0) < V(\hat{\alpha}_o, \hat{\beta}_o, 0)$ is $a < \hat{\alpha}_o$ and $b < \hat{\beta}_o$.)*

Proof. The comparison of profits is

$$\begin{aligned} & 4(\pi(O_{ab}) - \pi(P)) \\ & \quad = (V(a_o, b_o, z) - V(a_o, b_o, 0)) - (V(a, b, z) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)) \\ & \quad = (V(a_o, b_o, z) - V(a, b, z) - (V(a_o, b_o, 0) - V(a, b, 0)) \\ & \qquad + (V(\hat{\alpha}_o, \hat{\beta}_o, 0) - V(a, b, 0)). \end{aligned}$$

The first term on the right-hand side is positive by increasing differences, and the second term is positive by assumption. Therefore, $\pi(O_{ab}) - \pi(P) > 0$. □

Lemma A6. *If $\alpha < a < \hat{\alpha}_o$ and $b < \beta$, then $\pi(O_{a\beta}) > \pi(M_\beta)$. If $a < \alpha$ and $\beta < b < \hat{\beta}_o$, then $\pi(O_{\alpha b}) > \pi(M_\alpha)$.*

Proof. Suppose $\alpha < a < \hat{\alpha}_o$ and $b < \beta$. The comparison of profits is

$$\begin{aligned}
&4(\pi(O_{a\beta}) - \pi(M_\beta)) \\
&\quad = (V(a_o, \beta_o, z) - V(a_o, \beta_o, 0)) - (V(a, \beta_o, z) - V(\hat{\alpha}_o, \beta_o, 0)) \\
&\quad = (V(a_o, \beta_o, z) - V(a, \beta_o, z) - (V(a_o, \beta_o, 0) - V(a, \beta_o, 0))) \\
&\qquad + (V(\hat{\alpha}_o, \beta_o, 0) - V(a, \beta_o, 0)).
\end{aligned}$$

The first term on the right-hand side is positive by increasing differences, and the second term is positive by assumption. Therefore, $\pi(O_{a\beta}) - \pi(M_\beta)$.

The proof for $a < \alpha$ and $\beta < b < \hat{\beta}_o$ follows the same steps. □

Combining the profit comparisons in the lemmas we just proved, we obtain Figure 2. □

Proof of Proposition 2

The proof involves two steps.

Lemma A7. *If $\lim_{x\to\infty} V(x, y, z) < \infty$, or if $\lim_{x\to\infty} V(x, y, z) = \infty$, $\lim_{y\to\infty} V(x, y, z) = \infty$ and Assumption 2 holds, then* (i) *for any $b' < \beta$ there is $a' > \hat{\alpha}_o$ such that $\pi(M_\beta) > \pi(O_{a\beta})$; and* (ii) *for any $b' > \beta$ there is $a' > \hat{\alpha}_o$ such that $\pi(M_b) > \pi(O_{ab})$. Likewise, if $\lim_{y\to\infty} V(x, y, z) < \infty$, or if $\lim_{y\to\infty} V(x, y, z) = \infty$ and Assumption 2 holds, then* (iii) *for any $a' < \alpha$ there is $b' > \hat{\beta}_o$ such that $\pi(M_\alpha) > \pi(O_{\alpha b})$; and* (iv) *for any $a' > \alpha$ there is $b' > \hat{\beta}_o$ such that $\pi(M_a) > \pi(O_{ab})$. Finally, assume the previous two conditions hold. Then,* (v) *there is $a' > \hat{\alpha}_o$ and $b' > \hat{\beta}_o$ such that P is optimal.*

Proof. Given $a > \alpha$, consider the difference in profits between O_{ab} and M_a:

$$\begin{aligned}
&4(\pi(O_{ab}) - \pi(M_a)) \\
&\quad = V(a_o, b_o, z) - V(a_o, b_o, 0) - (V(a_o, b, z) - V(a_o, \hat{\beta}_o, 0)) \\
&\quad = V(a_o, b_o, z) - V(a_o, b, z) - (V(a_o, b_o, 0) - V(a_o, b, 0)) \\
&\qquad - (V(a_o, b, 0) - V(a_o, \hat{\beta}_o, 0)).
\end{aligned}$$

Let $\lim_{y\to\infty} V(x, y, z) = K < \infty$. Then, for b large enough, $V(a_o, b_o, z) - V(a_o, b, z) - (V(a_o, b_o, 0) - V(a_o, b, 0))$ will be very close to zero. On the other hand, $V(a_o, b, 0) - V(a_o, \hat{\beta}_o, 0) > 0$ for $b > \hat{\beta}_o$, so there is some $b > \hat{\beta}_o$ for which the difference in profits becomes negative.

Consider now $\lim_{y\to\infty} V(x, y, z) = \infty$. If Assumption 2 holds, then $V(a_o, b_o, z) - V(a_o, b, z) - (V(a_o, b_o, 0) - V(a_o, b, 0))$ is bounded as $b \to \infty$ (it is decreasing, but it always has to be greater than zero). On the other hand, $V(a_o, b, 0) - V(a_o, \hat{b}_o, 0) \to \infty$. Therefore, for b large enough, the difference in profits becomes negative.

The same steps can be followed to prove statements (ii) to (iv).

Finally, to see that (v) holds, consider the difference in profits between O_{ab} and P:

$$\begin{aligned}
&4(\pi(O_{ab}) - \pi(P)) \\
&\quad = V(a_o, b_o, z) - V(a_o, b_o, 0) - (V(a, b, z) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)) \\
&\quad = V(a_o, b_o, z) - V(a_o, b, z) - (V(a_o, b_o, z) - V(a_o, b, 0)) \\
&\qquad + V(a_o, b, z) - V(a, b, z) - (V(a_o, b, z) - V(a, b, 0)) \\
&\qquad - (V(a, b, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)).
\end{aligned}$$

Using the same arguments as before, it can be shown that the above assumptions imply that for a and b large enough, $\pi(P) > \pi(O_{ab})$. The same can be proved for the comparison of P with M_a and M_b. □

Lemma A8 (Monotonicity). *Under Assumption 2, if a mixed model yields higher profit than an open-source model for a', b', it still yields higher profit for $a'' > a'$, $b'' > b'$. Likewise, if the proprietary model yields higher profit than a mixed or open-source model for a', b', it still yields higher profit for $a'' > a'$, $b'' > b'$.*

Proof. Suppose that $\pi(M_a) > \pi(O_{ab})$ for a', b'. From the proof of Lemma A7, we know that

$$\begin{aligned}
&4(\pi(O_{ab}) - \pi(M_a)) \\
&\quad = V(a'_o, b'_o, z) - V(a'_o, b', z) - (V(a'_o, b'_o, 0) - V(a'_o, b', 0)) \\
&\qquad - (V(a'_o, b', 0) - V(a'_o, \hat{\beta}_o, 0)).
\end{aligned}$$

Consider now $a'' > a$ or $b'' > b'$. By Assumption 2, we know that $V(a_o, b_o, z) - V(a_o, b, z) - (V(a_o, b_o, 0) - V(a_o, b, 0))$ decreases as a or b increase. Also, $V(a_o, b, 0) - V(a_o, \hat{b}_o, 0)$ increases when a increases because of increasing differences, and increases when b increases because V is increasing. Therefore, profit decreases for $a'' > a$ or $b'' > b'$.

The same monotonicity result holds for the other profit comparisons between mixed and open-source models. Finally, to show monotonicity in the case of P, differences in profits can be rearranged as in the proof of Lemma A7, and the results follow using the same arguments as in the previous proof. □

The above lemma means that the line dividing the regions for which O_{ab} and M_a are optimal in the $\{a, b\}$ space is decreasing. The same is true for the line that divides the regions for which O_{ab} and M_b are optimal. The line dividing $O_{\alpha b}$ and M_α, on the other hand, is constant with respect to a because neither profit depends on a. Likewise, the line dividing $O_{a\beta}$ and M_β is constant with respect to b. Finally, we know that these lines have to be outside the box determined by $\hat{\alpha}_o$, $\hat{\beta}_o$ because of Proposition 1. This completes the proof. □

Proof of Lemma 1

Suppose that we are in a region where an open-source or a mixed-source model may be optimal, and let $\pi(O)$ and $\pi(M)$ represent the profits of the open and mixed models in that region. The difference in profits is

$$\begin{aligned}
&4(\pi(O) - \pi(M)) \\
&\quad = (V(x_o, y_o, z) - V(x_o, y_o, 0)) - (V(x', y', z) - V(x'', y'', 0)),
\end{aligned}$$

where x_o and y_o are the values of the open modules, and x', y', x'', and y'' depend on the specific mixed model used. What is unambiguously true is that $x_o \geq x'$ and $y_o \geq y'$, which is enough to prove our result. Rearranging terms, we get

$$4(\pi(O) - \pi(M)) = (V(x_o, y_o, z) - V(x', y', z)) - (V(x_o, y_o, 0) - V(x'', y'', 0)).$$

The second term on the right-hand side is constant with respect to z, whereas the first term is increasing in z by increasing differences. Therefore, as z increases, the region of parameters for which the optimal business model is open source increases. Equivalent theorems can be proved for the comparison between open-source and proprietary business models, and for the comparison between mixed-source and proprietary business models. □

Proof of Lemma 2

Suppose that $a > \hat{\alpha}_o$ and $b > \beta$, so that O_{ab} or M_b may be optimal. The difference in profits is

$$4(\pi(O_{ab}) - \pi(M_b)) = (V(a_o, b_o, z) - V(a_o, b_o, 0)) - (V(a, b_o, z) - V(\hat{\alpha}_o, b_o, 0)).$$

As σ_A increases, the first term in the right-hand side increases by increasing differences, whereas the second term decreases because the value of the firm's product does not change, whereas the value of the open-source good increases. Thus, the difference in profits increases. The proofs for the other cases stated in the lemma follow the same steps as the above proof. □

Proof of Proposition 3

This proof follows several steps. Figure 4 is a summary of the different partial results.

Lemma A9. *Under incompatibility, P, M_α, and M_β are never optimal.*

Proof. Under incompatibility, if the firm opens only b, this module cannot be used by the open-source project, so the value of the open-source good is the same with M_b and P. The difference in profits is $\pi(M_b) - \pi(P) = V(a, \hat{b}_o, z)/4 - V(a, b, z)/4$, which is always positive. Similar comparisons show that $O_{\alpha\beta}$ dominates M_α and M_β.

Lemma A10. *If $a < \alpha$ and $b < \beta$, then $\pi(O_{\alpha\beta}) > \max\{\pi(O_{ab}), \pi(M_a), \pi(M_b)\}$.*

Proof. From the proof of Lemma A1 in Proposition 1 we know that $O_{\alpha\beta} > O_{ab}$. The difference in profits between $O_{\alpha\beta}$ and M_a is

$$\begin{aligned}4(\pi(O_{\alpha\beta}) - \pi(M_a)) &= (V(\alpha_o, \beta_o, z) - V(\alpha_o, \beta_o, 0)) - (V(\hat{\alpha}_o, b, z) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)) \\ &= (V(\alpha_o, \beta_o, z) - V(\hat{\alpha}_o, b, z)) - (V(\alpha_o, \beta_o, 0) - V(\hat{\alpha}_o, b, 0)) \\ &\quad - (V(\hat{\alpha}_o, b, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)).\end{aligned}$$

Increasing differences imply that $V(\alpha_o, \beta_o, z) - V(\hat{\alpha}_o, b, z) > V(\alpha_o, \beta_o, 0) - V(\hat{\alpha}_o, b, 0)$, and $V(\hat{\alpha}_o, b, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0) < 0$ by assumption. Therefore, $\pi(O_{\alpha\beta}) > \pi(M_a)$.

The proof for $\pi(O_{\alpha\beta}) > \pi(M_b)$ follows the same steps as the previous proof. □

Lemma A11. *If $a > \alpha$ and $b > \beta$, then $\pi(O_{ab}) > \pi(O_{\alpha\beta})$.*

Proof. This proof follows from the proof of Lemma A2 of Proposition 1. □

Lemma A12. *If $\alpha < a < \hat{\alpha}_o$ and $b < \beta$, then $\pi(O_{\alpha\beta}) > \pi(M_b)$.*

Proof. The difference in profits between $O_{\alpha\beta}$ and M_b is

$$\begin{aligned}4(\pi(O_{\alpha\beta}) - \pi(M_b)) &= (V(\alpha_o, \beta_o, z) - V(\alpha_o, \beta_o, 0)) - (V(a, \hat{b}_o, z) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)) \\ &= (V(\alpha_o, \beta_o, z) - V(a, \hat{b}_o, z)) - (V(\alpha_o, \beta_o, 0) - V(a, \hat{b}_o, 0)) \\ &\quad - (V(a, \hat{b}_o, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)).\end{aligned}$$

Increasing differences imply that $V(\alpha_o, \beta_o, z) - V(a, \hat{b}_o, z) > V(\alpha_o, \beta_o, 0) - V(a, \hat{b}_o, 0)$, and $V(a, \hat{b}_o, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0) < 0$ by assumption. Therefore, $\pi(O_{\alpha\beta}) > \pi(M_b)$. □

Combining the different lemmas, we get to the results shown in Figure 4. □

Proof of Proposition 4

Suppose that we are above the line dividing the optimal regions of $O_{\alpha\beta}$ and O_{ab}, and let's compare O_{ab} with M_b. The difference in profits is

$$4(\pi(O_{ab}) - \pi(M_b)) = V(a_o, b_o, z) - V(a_o, b_o, 0) - (V(a, \hat{b}_o, z) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)).$$

We can rearrange this expression as follows:

$$\begin{aligned}4(\pi(O_{ab}) - \pi(M_b)) &= V(a_o, b_o, z) - V(a, \hat{b}_o, z) - (V(a_o, b_o, 0) - V(a, \hat{b}_o, 0)) \\ &\quad - (V(a, \hat{b}_o, 0) - V(\hat{\alpha}_o, \hat{\beta}_o, 0)).\end{aligned}$$

Then, by the same arguments of the proof of Lemma A7, we can see that $\pi(M_b) > \pi(O_{ab})$ as $a \to \infty$ or $b \to \infty$. Also, by the same arguments of the proof of Lemma A8, if $\pi(M_b) > \pi(O_{ab})$ for a', b', then $\pi(M_b) > \pi(O_{ab})$ for $a'' > a'$, $b'' > b'$. The same two results hold when comparing O_{ab} with M_a. □

Proof of Proposition 5

Given a, b, α, β, the open model (among those that are technologically feasible) that combines $\max\{a, \alpha\}$ with $\max\{b, \beta\}$ maximizes the value of software, and therefore maximizes V_c and V_o. Given that $V_c > V_o$, W is also maximized with this combination. □

Proof of Proposition 6

Let θ_1 and θ_2 correspond to the value captured by business models BM_1 and BM_2. BM_1 captures more value than BM_2 ($\theta_1 > \theta_2$) if and only if

$$\frac{2(V_{c1} - V_{o1})}{3V_{c1} + V_{o1}} > \frac{2(V_{c2} - V_{o2})}{3V_{c2} + V_{o2}}.$$

Working with this inequality, we get that the condition for $\theta_1 > \theta_2$ is

$$\frac{V_{c1}}{V_{o1}} > \frac{V_{c2}}{V_{o2}}.$$

Taking logs on both sides of the previous inequality,

$$\log(V_{c1}) - \log(V_{o1}) > \log(V_{c2}) - \log(V_{o2}).$$

The above inequality implies that if V is log-supermodular (i.e., its natural logarithm is supermodular), then all previous theorems comparing the profits of the different business models also hold when comparing the fraction of value captured by them. □

Proof of Lemma 3

If $a < \alpha$ and $b < \beta$, or $a > \alpha$ and $b > \beta$, the optimal open-source business model under compatibility and incompatibility coincide by Lemmas A1 and A2 in Proposition 1. If $a < \alpha$ and $b > \beta$, on the other hand, the optimal business model under incompatibility may be either O_{ab} or $O_{\alpha\beta}$, whereas the optimal business model under compatibility is $O_{\alpha b}$, which we know yields higher profit than O_{ab} or $O_{\alpha\beta}$ in that region of parameters by Lemmas A1 and A2. The case in which $a > \alpha$ and $b < \beta$ is analogous to this case. □

Proof of Lemma 4

If $V(a_o, 0, 0) > V(\hat{\alpha}_o, \hat{\beta}_o, 0)$, then the open-source competitor will adopt module a if the firm decides to open it. This means that the difference of profits is

$$4(\pi(M_a^i) - \pi(M_a^c))$$
$$= V(a_o, b, z) - V(a_o, 0, 0) - (V(a_o, b, z) - V(a_o, \hat{b}_o, 0)).$$

Then, $\pi(M_a^i) > \pi(M_a^c)$ because V is increasing.

Consider now M_α^c and M_a^i when $a < \alpha$ and $b > \beta$. In this case, with M_a^i, the firm's module is never adopted by the open-source competitor. If the firm chooses M_α^c, it will get a better-quality base, and it will benefit more from complementarities because $b > \beta$. Therefore, $\pi(M_\alpha^c) > \pi(M_a^i)$. The proof for the comparison between M_β^c and M_b^i when $a > \alpha$ and $b < \beta$ is analogous. □

Proof of Lemma 5

We know from Lemma A7 in Proposition 2 that under Assumption 2, P is optimal under compatibility for a and b large enough. However, by Lemma A9 in Proposition 3, we know that $\pi(M_b^i) > \pi(P)$. Therefore, incompatibility is preferred when a and b are large. □

References

Adner, R., F. Csaszar, P. Zemsky. 2010. Positioning on a multi-attribute landscape. Working paper, Dartmouth College, Hanover, NH.

Baden-Fuller, C., I. MacMillan, B. Demil, X. Lecocq. 2008. Special issue call for papers: Business models. *Long Range Planning*.

Bagozzi, R. P., U. M. Dholakia. 2006. Open source software user communities: A study of participation in Linux user groups. *Management Sci.* **52**(7) 1099–1115.

Baldwin, C. Y., K. B. Clark. 2006. The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Sci.* **52**(7) 1116–1125.

Bessen, J. 2006. Open source software: Free provision of complex public goods. J. Bitzer, P. Schröder, eds. *The Economics of Open Source Software Development*. Elsevier, Amsterdam, 57–82.

Bitzer, J. 2004. Commercial versus open source software: The role of product heterogeneity in competition. *Econom. Systems* **28**(4) 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.

Brandenburger, A., B. Nalebuff. 1997. *Co-Opetition: A Revolution Mindset That Combines Competition and Cooperation. The Game Theory Strategy That's Changing the Game of Business*. Doubleday, New York.

Brandenburger, A., H. Stuart. 1996. Value-based business strategy. *J. Econom. Management Strategy* **5**(1) 5–24.

Brandenburger, A., H. Stuart. 2007. Biform games. *Management Sci.* **53**(4) 537–549.

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

Casadesus-Masanell, R., A. Hervas-Drane. 2010. Peer-to-peer file sharing and the market for digital information goods *J. Econom. Management Strategy* **19**(2) 333–373.

Casadesus-Masanell, R., J.-E. Ricart. 2010. From strategy to business model and onto tactics. *Long Range Planning* **43**(2–3) 195–215.

Casadesus-Masanell, R., D. Yoffie. 2007. Wintel: Cooperation and conflict. *Management Sci.* **53**(4) 584–598.

Casadesus-Masanell, R., F. Zhu. 2010. Strategies to fight ad-sponsored rivals. *Management Sci.* **56**(9) 1484–1499.

Charny, B. 2001. Microsoft raps open-source approach. *CNET News* (May 3), http://news.cnet.com/2100-1001-257001.html.

Chatain, O., P. Zemsky. 2007. The horizontal scope of the firm: Organizational tradeoffs versus buyer-supplier relationships. *Management Sci.* **53**(4) 550–565.

Chen, M. K., B. J. Nalebuff. 2006. One-way essential complements. Discussion Paper 1588, Cowles Foundation, Yale University, New Haven, CT.

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.

Fosfuri, A., M. S. Giarratana, A. Luzzi. 2008. The penguin has entered the building: The commercialization of open source software products. *Organ. Sci.* **19**(2) 292–305.

Gabszewicz, J., J. Thisse. 1979. Price competition, quality and income disparities. *J. Econom. Theory* **20**(3) 340–359.

Gaudeul, A. 2005. Competition between open-source and proprietary software: The (La)TeX case study. Discussion paper, University of East Anglia, Norwich, UK.

Haruvy, E., S. Sethi, J. Zhou. 2008. Open source development with a commercial complementary product or service. *Production Oper. Management* **17**(1) 29–43.

Henkel, J. 2004. The jukebox mode of innovation—A model of commercial open source development. Discussion Paper 4507, Centre for Economic Policy Research, London.

Katz, M. L., C. Shapiro. 1985. Network externalities, competition, and compatibility. *Amer. Econom. Rev.* **75**(3) 424–440.

Kim, W. C., R. Mauborgne. 2005. *Blue Ocean Strategy: How to Create Uncontested Market Space and Make Competition Irrelevant*. Harvard Business School Press, Boston.

Lee, D., H. Mendelson. 2008. Divide and conquer: Competing with free technology under network effects. *Production Oper. Management* **17**(1) 12–28.

Lerner, J., J. Tirole. 2005. The economics of technology sharing: Open source and beyond. *J. Econom. Perspect.* **19**(2) 99–120.

Letzing J. 2009. Microsoft defends right to brandish open-source patents. *Market Watch* (July 22), http://www.marketwatch.com/story/microsoft-defends-nuanced-open-source-approach.

Llanes, G., R. de Elejalde. 2009. Industry equilibrium with open source and proprietary firms. Working Paper 09-149, Harvard Business School, Boston.

MacDonald, G., M. Ryall. 2004. How do value creation and competition determine whether a firm appropriates value? *Management Sci.* **50**(10) 1319–1333.

Markides, C. C. 2008. *Game-Changing Strategies: How to Create New Market Space in Established Industries by Breaking the Rules*. Jossey-Bass, San Francisco.

Mustonen, M. 2003. Copyleft—The economics of Linux and other open source software. *Inform. Econom. Policy* **15**(1) 99–121.

Roberts, J., I. Hann, S. Slaughter. 2006. Understanding the motivations, participation, and performance of open source software developers: A longitudinal study of the Apache projects. *Management Sci.* **52**(7) 984–999.

Salas-Fumás, V. 2009. Modelos de Negocio y Nueva Economía Industrial. *Universia Business Review*. Tercer Trimestre, Madrid, 122–143.

Schmidtke, R. 2006. Private provision of a complementary public good. Discussion Papers in Economics 2006-20, University of Munich, Munich, Germany.

Shah, S. 2006. Motivation, governance, and the viability of hybrid forms in open source software development. *Management Sci.* **52**(7) 1000–1014.

Shaked, A., J. Sutton. 1982. Relaxing price competition through product differentiation. *Rev. Econom. Stud.* **49**(1) 3–13.

Thomas, G. 2008. Microsoft and mixed source software. (October 22), http://www.infoworld.com/d/mobilize/microsoft-exec-touts-mixed-source-ventures-795.

Topkis, D. 1998. *Supermodularity and Complementarity*. Princeton University Press, Princeton, NJ.

Vives, X. 2001. *Oligopoly Pricing: Old Ideas and New Tools*. MIT Press, Cambridge, MA.

von Engelhardt, S. 2010. Quality competition or quality cooperation? License-type and the strategic nature of open source vs. closed source business models. Research Paper 2010-034, University of Jena, Jena, Germany.

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