Consumer welfare and market structure in a model of competition between open source and proprietary software

Gaudeul, Alexia

Department of Economics, University of East Anglia, ESRC Centre for Competition Policy

29 October 2008

Consumer Welfare and Market Structure in a Model of Competition between Open Source and Proprietary Software

by

Alexia Gaudeul
Dept. of Economics and ESRC Centre for Competition Policy
University of East Anglia

CCP Working Paper 08-31

The support of the Economic and Social Research Council is gratefully acknowledged.

ISSN 1745-9648

Consumer welfare and market structure in a model of competition between open source and proprietary software*

Alexia Gaudeul†

October 28, 2008

Abstract

I consider a Vickrey-Salop model of spatial product differentiation with quasi-linear utility functions and contrast two modes of production, the proprietary model where entrepreneurs sell software to the users, and the open source model where users participate in software development. I show that the OS model of production may be more efficient from the point of view of welfare than the proprietary model, but that an OS industry is vulnerable to entry by entrepreneurs while a proprietary industry can resist entry by OS projects. A mixed industry where OS and proprietary development methods coexist may exhibit large OS projects cohabiting with more specialized proprietary projects, and is more efficient than the proprietary model of production from the point of view of welfare.

Keywords: Open Source; Proprietary; Software Industry; Copyright; Non-Profit Organization; Mixed Market; Welfare; Spatial Product Differentiation

JEL codes: D23, H44, L17, L22, L33, L86, O34, O38

There is a variety of mixed industries; industries where for-profit and non-profit coexist. In the present model, I will focus on the decision by consumers between contributing to collective projects, or buying goods that are produced by independent entrepreneurs and sold for profit. The model will be particularly adapted to the study of competition in the software industry between

*I would like to thank Yossi Spiegel and other participants at the F/LOSS'2007 Workshop in Nice Sophia-Antipolis for their comments and suggestions. The support of the ESRC is gratefully acknowledged. This paper is forthcoming in the International Journal of Open Source Software & Processes (IJOSSP), edited by Stefan Koch.

†School of Economics and ESRC Centre for Competition Policy, University of East Anglia, NR4 7TJ, Norwich, United Kingdom. Email: a.gaudeul@uea.ac.uk. Website: http://agaudeul.free.fr

software developed under proprietary license terms and software developed under open source ('OS') license terms. However, it is also adapted to the wide range of areas in which open development methods are used concurrently with proprietary methods. This includes genetic and biological research, which among other research areas has benefited from the use of open source methods. For example, the publicly financed and open Human Genome Project competed with the privately financed Celera Genomics. Other projects have been set up to further the use of OS methods in this area (www.bioforge.net, www.bioinformatics.org, www.genome.wustl.edu). This also includes blogs (Raynes-Goldie, 2004, Gaudeul, Mathieu, and Peroni, 2008), which link spontaneously generated independent contributions together in a complex system of relations and have emerged as an alternative and/or complement to established news media. Online databases that can be openly edited by anybody (Wikis) generate alternative repositories of knowledge that compete with established dictionaries and encyclopedias. Online communities and other group communication mechanisms, such as Facebook or LinkedIn, compete with established search and coordination infrastructures provided by firms and in markets. Knowledge production in the academia, which can be described as an open source process (Bezroukov, 1999, Raymond, 2001), competes with the research output produced from within firms. Finally, non-profits in the health sector, the provision of services to the poor, museums, job training, etc... compete with their private counterparts. The domains of application of this article are therefore extensive.

The current literature on the patterns of competition and cooperation between open source and proprietary projects and methods is already rather well developed. However, there is little work on how the cohabitation of open source and proprietary models of production affects consumer welfare and market structure at the industry level. I find that an industry where only the OS development model is used is more efficient from the point of view of welfare than an industry where proprietary development is used to the exclusion of other development methods. However, an OS industry will be vulnerable to entry by entrepreneurs that use proprietary development methods, while an industry that uses the proprietary mode of development will be able to resist entry by OS projects. A mixed industry where OS and proprietary development methods coexist may exhibit large OS projects cohabiting with more specialized proprietary projects, and this pattern of coexistence will improve on a proprietary industry from the point of view of welfare.

Context: I have conducted two main empirical studies to analyze the patterns of cohabitation between open source and proprietary software. In the first project (Gaudeul, 2003, Gaudeul, 2007), I consider the evolution of the patterns of competition between LaTeX, an open source typesetting software, and its proprietary alternatives and complements. In a second project (Gaudeul, 2008), I consider empirical data to explain when and why open source software does not attract users. There are very few other empirical studies of the patterns of competition between OS and proprietary software ('PS') and of the influence of open source software ('OSS') on PS production and vice-versa. Bitzer and Schröder (2006) assert from case studies that OSS encourages innovation from proprietary software – even though Klincewicz (2005) reports that OSS itself may not be very innovative, and Harison and Koski (2008) report that firms that develop OSS as part of their business model are less productive than those that provide only proprietary applications. Franke and von Hippel (2003) consider the development of security functionalities in Apache, the web server, and show its success comes from being able to address the heterogeneous needs of those users who are dissatisfied with the proprietary offering. Mockus, Fielding, and Herbsleb (2005) compare the development of Apache and Mozilla and underline how Mozilla may have benefited from maintaining much of the machinery of its past commercial development.

Literature review: The above empirical studies have inspired a number of models of competition between OS and PS. I will neglect in this review all the papers that compare the performance of OS and proprietary development methods from the point of view of efficiency (see for example Johnson (2002) or Kuan (2002)). The models of competition that have been devised up to now essentially consider mixed market duopolies, where one OS software/platform competes with one proprietary alternative. In Bitzer (2004), proprietary developers can survive entry by OSS as long as they differentiate enough from the OS offering. In Darmon, le Texier, and Torre (2006), proprietary firms can adopt a "low price and/or high quality" strategy and dominate the market, or a "high price and/or low quality" strategy and risk losing the market to OSS. In Lambardi (2007), the proprietary developer may want to lower the price charged to users-developers in order to deter entry and prevent the development of an open source alternative. In Gaudeul (2005), I show that competition favors the use of the BSD license terms rather than the GPL. This is because potential competitors will contribute to BSD software with the hope of integrating it in their own proprietary software. In Economides and Katsamakas (2006), proprietary developers can base their development on an open-source or a proprietary platform. Proprietary platforms are shown to attract a bigger share of users. Sen (2007) considers the choice between developing proprietary software and developing a commercial version of OSS. Casadesus-Masanell and Ghemawat (2006) consider a dynamic mixed duopoly model and analyze the dynamics of diffusion of OSS in a market dominated by proprietary software. PS is shown to resist entry by OSS by exploiting network effects strategically. Leoncini, Rentocchini, and Marzetti (2008) consider a technology diffusion model and confirm coexistence is possible.

Quite apart from those models of competition, another strand of literature, presented in a special issue of Organization Science (von Hippel and von Krogh, 2003) offers a number of reflections on the emergence of a 'private-collective' innovation model that combines the strengths of both OS and proprietary production models. Hybrid development models are analyzed in Mustonen (2005) where firms may support OS development if network effects are small in order to encourage compatibility of OSS with their product, but also in Bonaccorsi and Rossi (2006) and Wichmann (2002) which survey the motivations for proprietary involvement in OS development. Finally, O'Mahony (2005) considers the role of non-profit foundations in fostering collaboration between the OS community and firms.

Darmon, le Texier, and Torre (2006), along with Schmidt and Schnitzer (2003) or Bessen (2006), are among the few articles to consider explicitly the consumers' welfare issues that are brought about by the adoption of OSS. Darmon, le Texier, and Torre (2006) argue that the development of OSS can lead to consumers mis-coordinating on the wrong sort of software if the objectives of commercial developers are not aligned with those of consumers. Schmidt and Schnitzer (2003) argues that OSS lower profits and therefore incentives for innovation from the part of proprietary developers. Comino and Manenti (2005), however, show that governments would be well advised, if not to subsidize OSS, at least to advertise its availability. Bessen (2006) argues that in the case of complex products, concurrent provision of software as OSS improves welfare as it saves on development costs for specialized needs. von Engelhardt and Swaminathan (2008) show that coexistence of OS and PS spurs innovation and growth.

Less directly related to this paper are a few papers that analyze competition between for-profit and non-profits. However, those papers focus on different aspects of this coexistence, such as ownership effects (Ballou, 2005), motivations and objectives in non-profit provision (Brhlikova, 2004, Brhlikova, 2006), or the taste of some consumers for non-profit provision (Besley and Ghatak, 2001). In Sloan (2000), consumers might prefer non-profit provision of services such as health care because they fear that for-profit provision might lead to inadequate incentives for hospital managers and physicians. In Glaeser (2002), competition from for-profits will discipline managers in non-profit firms. Rose-Ackerman (1990) analyzes more generally the patterns of coexistence between non-profits and for-profits and considers when non-profits will successfully enter the commercial sector. The influence of government policies is analyzed in the context of the competition between public (State) and private (Public) schools (Epple and Romano, 1998),[1] and in the context of public financing for the arts, which can crowd out private provision (Frey, 1999). It is important to notice that there is no role for the State or charitable institutions in the present paper, as I focus on a process of private provision of a public good (Bergstrom, Blume, and Varian, 1986). I will however be able to draw out some public policy implications from this paper.

The model: I adapt a model offered in Alesina and Spolaore (1997) ('A&S') to consider a circular city location model where consumers differ in their preferences over one dimension of a specific good (the software they use), and have the choice between a variety of competing products located along the preference circle. Each product costs K to produce, and has value g to the consumer that is closest to its location. A consumer i that is located at distance l_i from the location of the product derives utility $g(1 - al_i)$ from the product, with a measuring the extent to which products are differentiated – the higher is a, the more important it is for the good to be close to the consumer's preferred location.

The model is thus a Vickrey-Salop (circular city) model of spatial product differentiation with quasi-linear utility functions – see Vickrey (1964), which was reprinted in Vickrey (1999), and see Salop (1979). As in most of the literature using that type of model, I will consider equilibria where firms are evenly spaced, even though other types of equilibria may exist in theory (Braid, 2004). I consider free-entry equilibrium such that there is no profit to be made by an additional firm entering the market when firms do not react strategically to entry by either changing location or changing prices.

I consider two modes of production. Under the open source model, portion s of consumers decide to produce the good collectively, with each consumer contributing equally to the cost K of production. Under the proprietary model, an entrepreneur, j, decides to produce the good, at cost K, and then sells the good at price p_j, the same for all consumers. In a first section, I reproduce the results by A&S and extend them to consider potential entry by a proprietary entrepreneur. I show that this potential entry puts additional constraints on how big OS projects can become, and I also show that if OS contributors anticipate entry, then OS projects cannot resist entry by proprietary developers. In a second section I apply A&S's model to the context of proprietary production. I compare that situation with that obtained in the open source ('OS') production model and show that welfare obtained in a proprietary industry is lower than that obtained in an OS industry, and worse, that it is robust to entry by the potentially more efficient OS model. After having analyzed those two situations, where a single production method, collective or entrepreneurial, is used, I analyze a mixed industry model where both methods are used in the same market for different products. The analysis of this specific context allows me to determine the share of users of OS vs. proprietary software when all users are also potentially developers, and whether a mixed industry provides higher or lower welfare than an industry with a single production model. This mixed industry context also leads me to offer a number of ideas for extensions to the model at the end of the paper.

1 OS INDUSTRY

In this part, I give out and interpret the main results of interest from Alesina and Spolaore (1997), and examine the effect of potential entry by proprietary software in an OS industry.

A world population of mass 1 is composed of a continuum of individuals whose ideal points are distributed uniformly over a circle of circumference 1. Each individual may contribute to the production of a public good, an Open Source Software ('OSS') project. There may be a number of such projects, $N \geq 1$, and it costs K for any such project to be completed. This cost may be expressed in monetary terms, but does not necessarily reflect monetary contributions, but rather the money equivalent of the time and effort devoted by each contributing individuals to the project. An individual i that contributes t_i to a project that is located at distance l_i from her ideal point derives from that project utility $U_i = g(1 - al_i) - t_i$.

The location of the project is determined by the average of the locations of individual contributors to the project. I will assume that each individual does not consider the impact of her contribution on the location of the project (a contributor 'to the right' of the project would presumably change its position to the right, even if by an infinitesimal amount); that is, contributors take the position of the projects as given.

The contributions of the developers must sum up to K for the project to exist, that is, if s is the mass of contributors, and they all contribute t to the project, then I must have $st = K$ for the project to be achieved successfully. I will assume all developers contribute equally to the project they participate in and they contribute to only one project.

The proposition below sets out the optimal size s^* and number N^* of software projects from the point of view of a benevolent social planner who seeks to maximize social welfare.

Proposition 1 *A social planner seeking to maximize welfare will choose to establish N^* projects of equal size s^*, such that $s^* = 2\sqrt{K/ag}$ and $N^* = \sqrt{ag/K}/2$.*

Proof. Consider N projects equally spaced in the location circle. The social planner maximizes the sum of all individual utilities. Denoting project numbers as $x = 1, ..., N$ and $\bar{l}_x$ the average distance from the project for each of its individual contributors, then social welfare SW will be

$$SW = \sum_{x=1}^{N} s_x[g(1 - a\bar{l}_x) - t_x] \tag{1}$$

In a symmetric equilibrium, each project x covers space $s_x = 1/N$ and therefore, the farthest a consumer can be from her preferred location is $1/2N$, so that on average consumers will be located at distance $1/4N$ from their preferred location. Furthermore, the contribution that is required of each consumer must be such that $t_x = K/s_x$, which is equal to KN by the expression of s_x above. Therefore, the expression for social welfare can be simplified to

$$SW = g - KN - ag/4N \tag{2}$$

which is concave and is maximized for $N^* = \sqrt{ag/4K}$. SW will then be:

$$SW^* = g - \sqrt{Kag} \tag{3}$$

A detailed proof including the cases where N^* is not an integer is in A&S, proposition 1. ■

The following proposition sets out the number N_{OS} and size s_{OS} of projects in a symmetric equilibrium that occurs as the result of the competition between projects organized along open source principles. Projects must be at least larger than a minimum size for contributors not to defect to the adjoining project where they would be asked for a lower contribution.

Proposition 2 *If developers are free to choose the project they want to contribute to, but are not able to coordinate their decisions with other developers, then a stable equilibrium of the contribution game must be such that there will be N_{OS} projects of equal size such that $s_{OS} > \sqrt{2K/ag}$ and $N_{OS} < \sqrt{ag/2K}$.*

Proof. Consider project 1 and 2, of size s_1 and s_2 respectively. The user developer at the border between the two projects is located at distance $s_1/2$ of project 1 and $s_2/2$ of project 2 and must be indifferent between the two for the projects' border to be stable. Therefore, I must have:

$$g(1 - as_2/2) - K/s_2 = g(1 - as_1/2) - K/s_1 \tag{4}$$

which is solved in two cases, either $s_1 = s_2$ or $s_1 s_2 = 2K/ag$. The case where $s_1 = s_2 \equiv s$ is stable only if, supposing project 2 increases in size by ε at the expense of project 1, then the individual at the new border at distance $(s - \varepsilon)/2$ of project 1 prefers contributing to project 1 rather than project 2, that is:

$$g(1 - a(s + \varepsilon)/2) - K/(s + \varepsilon) < g(1 - a(s - \varepsilon)/2) - K/(s - \varepsilon) \tag{5}$$

which simplifies to $s^2 > 2K/ag$ as ε tends towards 0.

The other case, where $s_1 s_2 = 2K/ag$, will not be stable according to the logic above. There is therefore a unique stable symmetric equilibrium as exposed in the proposition. More details are provided in A&S, proposition 2. ■

The above proposition sets a minimum size for an OS project. The threat of forking, that is, developers within a project striking out on their own and establishing a separate project, will put an upper limit to how big a project can be. This is because an OS project that becomes too big will see a significant number of marginal contributors (contributors far from the center of the project) who, while not being required to put up high contributions to the project, will be dissatisfied with its central direction. Those developers may then decide to 'fork' and set up their own project.[2] The following proposition states that any equilibrium of the OS contribution game such that $s_{OS} \leq (\sqrt{6}+2)\sqrt{K/ag}$ will be robust to forking.

Proposition 3 *Symmetric equilibria of the OS contribution game are robust to forking only s.t.* $s_{OS} \leq (\sqrt{6}+2)\sqrt{K/ag}$.

Proof. In A&S, proposition 5. ■

I can therefore conclude from this section that in a stable symmetric equilibrium of the contribution game, the size of OS projects will be anywhere between $\sqrt{2K/ag}$ and $(\sqrt{6}+2)\sqrt{K/ag}$. I will show in the following that the threat of forking is not as strong as the threat of entry by proprietary software, and thus imposes additional limits on the size of OSS projects. In the discussion of welfare, I will show that the threat of entry by PS may improve welfare by preventing the emergence of welfare inefficient 'overgrown' OSS projects.

Proprietary entry: Consider the possibility of entry by proprietary firms in a situation where there are only OS projects. This is a situation of interest when OSS has been the development method of choice in one development area, and proprietary developers try to enter that area with their software. A situation where this would hold would for example be proprietary developers providing specialized applications that cater for the needs of those who are not satisfied with such OSS as Apache, TeX or Sendmail.[3]

I will analyze here whether the pure OS equilibrium is robust to entry by proprietary firms. I will consider entry as follows: a proprietary firm sets up between two OS projects and seeks to attract a portion of those who would have been developers of one or the other project – that is, those developers, instead of participating in OS production, choose to buy from the proprietary developer. As stated previously in the exposition of the model, I assume that OS location does not change as a strategic reaction to the entry of PS.

The timing is as follows:

1. OS projects, who do not anticipate entry, set their location and recruit contributors.

2. Before planned contributors start contributing, a proprietary project locates between two projects and offers software at price p.

3. Contributors choose between fulfilling their commitment to the OS project or buying the proprietary software.

There are then two cases, **case A**,with 'naïve users', where OS contributors assume all other contributors will fulfill their commitment to the OS project, and **case B**, with 'savvy users', where OS contributors assume that some contributors will buy proprietary software instead.

Case A: Consider **case A** first. Under case A then the size of projects will be limited further than with the threat of forking, as only a subset of symmetric equilibria of the OS contribution game that are robust to forking are robust to entry under case A:

Proposition 4 *Only symmetric equilibria such that* $s_o \leq (\sqrt{2}+2)\sqrt{K/ag}$ *will be robust to entry under **case A** where users are naïve.*

Proof. In appendix A. ■

Case B: Consider now **case B** where a contributor anticipates that if she decides not to contribute to an OS project and buys proprietary software instead, then other likely contributors that are closer to the proprietary project than she is herself are likely to do so as well. This makes entry by a proprietary firm easier than in case A. Indeed, those developers who would not have bought PS if others kept on contributing to OSS will be tempted to buy PS as the contribution required of them to sustain the project in the absence of those who fled will be higher. This is confirmed in the following proposition:

Proposition 5 *The equilibrium of the development game played by competing open-source projects is not robust to entry by a proprietary firm under **case B** where users are savvy.*

Proof. In appendix B ■

This means that what makes entry possible for the entrepreneur under case B is the fact that any OS user taken away from the OS project increases contributions needed from other OS developers remaining in the project. Proprietary entry thus works by a domino effect.

In the following, I discuss how proprietary entry under case A limits the size of OS projects in a way that is beneficial for welfare compared to the case where the only threat to OS projects was the possibility of forking:

Welfare: The optimum of the social planner is achievable under the OS system and is robust to forking. However, there is a wide range of equilibria that may occur under the OS mode of production, and there is therefore no guarantee that the optimum will be achieved. In the following proposition, I define lower bounds for welfare under the open source model.

Proposition 6 *Welfare in an OSS industry may be decreased by* 33.71% *of* $\sqrt{Kag}$ *compared to the welfare optimal market structure.. Threat of entry by proprietary software will limit the loss in welfare to* 14.64% *of* $\sqrt{Kag}$ *when user-developers are naïve.*

Proof. In appendix C ■

Intuitively, the threat of forking is not sufficient to prevent inefficient overgrowth of OS projects, while the threat of entry by proprietary software, when it does not threaten the existence of an OS equilibrium (case B), but rather limits the size of OS projects (case A), will limit potential overgrowth of OS projects in a way that is beneficial to social welfare.

In conclusion to this part, potential competition by proprietary software imposes additional constraints for the existence and stability of the OS model of production. In cases where OS developers anticipate correctly the effect of entry by proprietary firms, then proprietary firms will enter. This will justify later on the study of mixed models where OS and PS coexist. From a policy point of view, as we will see, the OS model of production can achieve higher social welfare than the proprietary model, but open source projects are at risk of overgrowth, which might lead to an OS industry being very inefficient. However, potential entry by proprietary software can limit the size of OSS in such a way that the loss from overgrowth of OS projects can be limited. All this means that a social planner might want to encourage the use of the OS model but leave open the possibility for proprietary software to enter by maintaining copyright protection for software.

2 PROPRIETARY INDUSTRY

In this part, I consider the situation where all projects are proprietary. As stated in the introduction, A&S reduces to a standard Vickrey-Salop model in that case, with well known results. What is original in this part is to analyze whether the proprietary equilibrium is robust to OS entry, and also to compare welfare under the proprietary system and that under the OS system.

Consider thus an individual (the entrepreneur) who develops a project j at cost K and sells it to consumers at price p_j. I consider a free entry equilibrium whereby no new projects can be set up between other projects and turn up any strictly positive profit (as before, I consider entry without strategic reaction from established firms).

Proposition 7 *The free entry equilibrium of the development game played by competing profit motivated entrepreneurs is such that there will be* N_p *projects of equal size* s_p *such that* $N_p = \sqrt{ag/K}$ *and* $s_p = \sqrt{K/ag}$.

Proof. In appendix D. ■

There will thus be more proprietary firms than in even the equilibrium with the smallest stable projects of the OS development game. There is excess entry compared to the welfare optimal market structure because size is limited by the zero profit condition. Since the sum of all payments made by consumers to a specific entrepreneur must cover development costs K, and less consumers use a specific proprietary software than would use an OS software, then each consumer has to contribute more, in terms of price to pay, than they would contribute in kind in terms of development in an OS industry.

OSS entry: Consider now if OSS can make a successful entry in a domain that is dominated by proprietary software. A situation of interest would be that of GPL software such as Linux entering the field of operating systems vs. Apple or Microsoft.

The timing of entry is as follows:

1. Proprietary developers, who do not anticipate entry, set their location and price p.

2. Before consumers choose which software to buy, an open source project locates between two projects and seeks contributors.

3. Consumers choose between contributing to the open source project or buying proprietary software.

Proposition 8 *The equilibrium of the development game played by competing proprietary entrepreneurs is robust to entry by an open source project.*

Proof. In appendix E. ■

Contrary to the case of entry by a proprietary firm into an OS equilibrium, the proprietary developers can choose to maintain their price p at the same level as without entry by the OS project; therefore, the contribution needed from consumers to obtain the proprietary good does not increase in the same way as contributions required from OS developers increased when there was entry by a proprietary firm in the OS equilibrium and contributors were savvy. This makes entry by OSS in a proprietary equilibrium more difficult than entry by PS in an OS equilibrium. Indeed, while entry by PS was always possible when OS contributors were savvy, entry by OSS is never possible in a proprietary industry.

Welfare: The following proposition underlines how excess entry impacts social welfare in a proprietary industry:

Proposition 9 *Welfare in a proprietary industry will be decreased by* 25% *of* $\sqrt{Kag}$ *compared to the welfare optimal market structure.*

Proof. As seen previously in the calculation of the social welfare generated when a social planner chooses N optimally to maximize welfare, I found in equation 2 that the expression for social welfare could be simplified to $SW = g - KN - ag/4N$. In the equilibrium of the proprietary industry, I will have $N_p = \sqrt{ag/K}$, which is double the optimal number of projects, so $SW_p = g - \frac{5}{4}\sqrt{Kag}$, to be compared with optimal welfare of $SW^* = g - \sqrt{Kag}$. Welfare is thus reduced by 25% of $\sqrt{Kag}$ compared to the optimum. ■

As a matter of interpretation, and since $\sqrt{Kag}$ is the price of PS in the proprietary equilibrium, welfare is reduced by 25% of the total sales of proprietary firms in the proprietary equilibrium. This loss in welfare is more than in any of those equilibria of the OS industries that are robust to entry by proprietary entrepreneurs when contributors are naïve in their expectations (case A). This would mean the OS model is more efficient than the proprietary model as long as it is subject to entry by PS but contributors do not anticipate entry. As seen previously however, OS production will not be robust to entry by PS in case B, where OS contributors anticipate entry, which means that the OS model may not be sustainable and I need to examine the situation where both production models coexist.

3 MIXED INDUSTRY

I will now consider competition between OS and proprietary projects. Competition will be considered at the level of the user-developer, who chooses either to develop software or buy software developed by someone else. The mixed industry case corresponds to a number of market configurations (Gaudeul, 2008) where OS and PS coexist. The model applies specifically to some areas in software development where users are or can be also developers (for example, professional applications such as web server software). I will speak later on in this paper of how the model could be adapted to correspond to the type of markets where most users are not developers, and may not even know how to use OSS.

I will consider the case where OS and proprietary projects alternate in location. This is not the only possible configuration, but it is the simplest to consider. I will be interested in the relative size of open source and proprietary projects – how many users they attract – and I will make the usual robustness checks: robustness to entry to either OS or proprietary competitors, and robustness to forking for OS projects. Figure 1 below illustrates the type of configuration I will be considering. Note that in a symmetric equilibrium, open source and proprietary software will be evenly spaced and equal in number, so if there are N projects in total, half of them will be open source and half of them will be proprietary, and they will be at distance $1/N$ of each other.

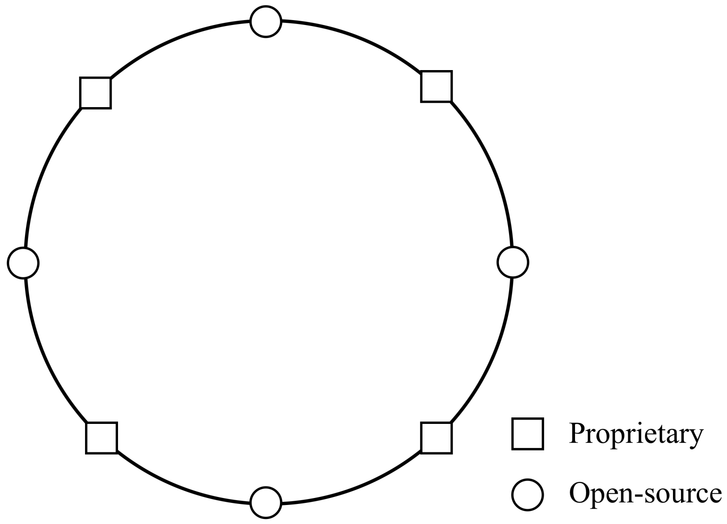

Figure1: Symmetric equilibrium with open-source and proprietary software alternating in the location circle.

The following proposition outlines the properties of the two possible equilibria of the mixed industry where OS and PS alternate in the preference space:

Proposition 10 *There are two possible equilibria of the mixed industry model where open source and proprietary projects alternate in the consumer's preference space:*

- *The first equilibrium is such that open source and proprietary projects are of the same size, that size being the size of proprietary projects in a proprietary industry. Each type of software gains half of the market.*
- *The second equilibrium is such that open source projects are twice as big – attract more consumers/developers – than proprietary projects, and therefore gain two thirds of the market. Open source projects are of the optimal size from the point of view of welfare, while proprietary projects are the same size as in a proprietary industry.*

Proof. In appendix F. ■

The proposition above can be expanded further as follows:

- In the first equilibrium with equally sized open source and proprietary software, the proprietary projects will take up $N_p s_p$ of the space, with $N_p = \frac{1}{2}\sqrt{ag/K}$ and $s_p = \sqrt{K/ag}$, so the space will be divided half and half between PS and OS. Since OS projects will be of the same size as proprietary projects, then each OSS user will contribute the same to OS development as each PS user pays for PS. Total expenses for private and public provision of goods will be equally shared in the economy (half will be made of in-kind contributions by OS users, half will be made of monetary contributions by proprietary users). Total expenses and contributions in kind will be $\sqrt{Kag}$, the same as was expended in a proprietary industry.

- In the second equilibrium, with asymmetrically sized proprietary and open source projects, proprietary projects will take up $N_p s_p$ of the space with $N_p = \frac{1}{3}\sqrt{ag/K}$ and $s_p = \sqrt{K/ag}$, so the space will be divided one third/two thirds between PS and OS projects respectively. This means there will be twice as many users of OSS as of PS. Since the size of OSS projects will be $s_o = 2\sqrt{K/ag}$, that is, twice the size of PS projects, then each OSS user will contribute to OS development half of what each PS users pays for PS. However, total expenses and contributions for private and public provision of goods will be equally shared in the economy, in the same way as it was equally shared in the first equilibrium. However, expenses and contributions will total only $\frac{2}{3}\sqrt{Kag}$, which is one third less than what was expended in a proprietary industry.

In the second equilibrium, with unequally sized projects, proprietary software is more specialized than OS software, in the sense that its users will be closer to their ideal points, than the average user in the OS projects. In the first equilibrium, where OS and PS projects are of the same size, they are of the equilibrium size in the proprietary industry, while in the second equilibrium, where OS and PS projects are of different sizes, then the size of the OS projects will be equal to the size that would have been chosen by a social planner. This leads to an analysis of welfare:

Welfare: If OS and proprietary projects are of the same size in a mixed industry model, then total welfare will be the same as in the pure proprietary model, which thus represents a lower bound to welfare.

In the equilibrium with asymmetrically sized proprietary and open source projects, where OS and proprietary projects alternate in the preference space, the average size of a project will be $(s_o+s_p)/2 = \frac{3}{2}\sqrt{K/ag}$, which is more than $\sqrt{2K/ag}$, the minimum stable size of an open source project in the pure OS model. The difference is small, but would indicate that overall welfare might be higher in a mixed industry model with different sized OS and proprietary projects, than in some pure OS industries.

Proposition 11 *Total welfare in the mixed industry model with asymmetric sized open-source and proprietary projects may be higher than in some equilibria of the open source model, and will be significantly higher than welfare in the pure proprietary model.*

Proof. In appendix G. ■

This means that an industry where both production models coexist is more efficient than an industry where only proprietary software is developed. Competition from OSS thus improves the situation for the society as a whole. To give an idea of how much of an improvement mixed industries can represent compared to pure proprietary models, one can take as a reference point the total sales of firms in a pure proprietary industry. The loss of welfare compared to the optimum is 25% of those sales in the proprietary model, compared to 8.33% of those sales in the mixed industry with asymmetrically sized open source and proprietary projects.

Figure 2 below represents total welfare as a function of project sizes in the economy:

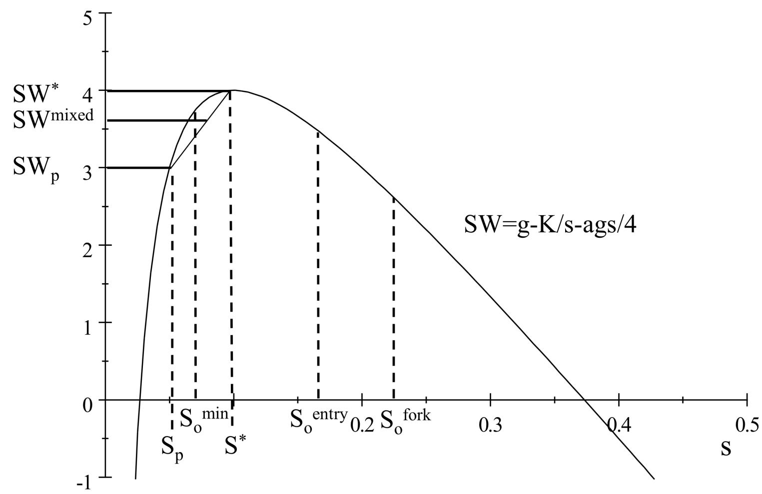

Figure 2: Social welfare under different industry configurations.

I take as an example $g = 8$, $a = 10$ and $K = 0.2$. In that setting, the optimal number of projects is 10, in which case social welfare is $SW^* = 4$ and the size of projects is $s^* = 0.1$. OS equilibria achieve a range of outcomes, from s_o^{min} which prevents contributors from switching projects, to s_o^{fork} which prevents contributors from forking and establishing their own project within a project. This range includes s_o^{entry}, which denotes the maximum size for OS that are robust to entry by PS when contributors are naïve and do not anticipate flight from other developers. Proprietary projects in both a proprietary and a mixed industry are of size s_p and are too small to achieve efficiency so a proprietary industry achieves social welfare of $SW_p = 3$ only, that is, 25% less than optimal. In a mixed industry with asymmetrically sized OS and PS, total welfare $SW^{\text{mixed}} = \frac{2}{3} * 4 + \frac{1}{3} * 3 = 3.67$ is the average from two thirds of consumers using optimally sized OS projects and from one third of consumers using software from proprietary projects that are too small. As seen graphically, this equilibrium achieves higher welfare than most OS equilibria and than the equilibrium in a proprietary industry.

From a public policy point of view, and as long as one can assume that all software users can be developers, the State may want to encourage the emergence of large open source software projects that would be used by the majority of users, while proprietary entrepreneurs would fulfill more specialized needs. This type of ideal market structure would become more relevant as software users become more experienced and more able to make their own contributions to the development of software they use.

A further possible public policy, to be explored in future versions of the paper, would be to encourage the use of BSD licenses so entrepreneurs can make use of OS code into specialized applications for ‘marginal’ consumers of OSS – consumers that have priorities (preferences) that differ from the average user of a specific OSS project.

4 CONCLUSION AND EXTENSIONS

I analyzed in this paper the patterns of competition between open source and proprietary software by adapting standard models of product differentiation and public good production in a novel way. This paper provides some insights on how open source and proprietary production systems can be expected to cohabit, and why cohabitation of OS and proprietary systems of production is desirable. In an industry where consumers can choose between contributing to open source software development or buying proprietary software, situations may occur where a majority of users would choose to contribute to open source projects, while a minority would buy from proprietary projects. Each of the proprietary projects would be more specialized than OS projects in so far as they would cover the needs of a smaller range of users.

There is some indication that this would be what is indeed observed in markets in which users can choose between a private and a public sector. For example, Apache dominates the market for web server software, and $\TeX$ was the dominant typesetting software in the early days of its development. More generally, OSS will gain a large share of the market when most of its users are also developers. The conclusions of the article do not hold however when most users are not or cannot be developers.

The conclusions from the article in its present state could also be used to analyze cohabitation of private and public provision of health care. Propper and Green (2001) report that the public share of total expenditure on health has been stable at around 75% in OECD countries since the 1970s. However, in this case, users of private services usually also have to contribute to the public services, which differs from the model in this paper. The case is different again in education. The UNESCO Institute for Statistics (2007) compares public and private expenditure on education in a variety of countries and shows that public provision is highest in primary and secondary education, but is much lower in tertiary education. For example, only 70% of expenditures in tertiary education in the UK are publicly funded, compared to 87% in primary and secondary education. Without paying too much attention to the precise share of private vs. public funding, it is however apparent that a number of countries find it optimal to combine a dominant public sector with a marginal private sector. This would seem to confirm the efficiency properties of a mixed model of production in some industries.

The model will need to be adapted to correspond to a wider variety of settings, and I indicate below several ways in which this could be done in the case of open source software provision:

Users and developers: There is no central authority that can require equal contributions from each participant in OSS development. In effect, Mockus, Fielding, and Herbsleb (2005) show that development of the code base is characterized by a high concentration of contributions among a few, while other tasks such as defect repair are more equally distributed among contributors. Lakhani and von Hippel (2003) emphasize further how disparate the type of contributions to OSS projects can be, ranging from programming to user-assistance. Fershtman and Gandal (2004) show that contributions will depend on the type of license used, while Krishnamurthy (2002) and Healy and Schussman (2003) show that most OSS projects are the work of an individual rather than of a community. The model would therefore need to be adapted to take account of those empirical findings.

An easy way to do this would be to examine what happens when a portion of the users are not able to develop software on their own and/or are not able to use OSS. In this paper, I considered that all users were able to develop and use OSS, while in reality, only a fraction may be able to use OSS, and a fraction of that fraction would be able to develop. Considering this extension may bring about ambiguous results. On the one hand, users who cannot develop OSS (or do not want to and prefer to free-ride) would essentially get OSS for free and thus obviously prefer it to PS. On the other hand, OS projects would have to extend further in order to attract sufficient contributions, and would thus possibly become even more fragile and vulnerable to competition by PS than before. In future extensions to this paper, I will argue that PS may improve welfare if PS is more accessible to some users than OSS is.

Imperfect competition: It is probably not reasonable to assume as in this paper that there is free entry into a proprietary industry. Sunk costs are important in the software industry. Branding is very important, which allows Norton, Microsoft or Apple to cover a wide range of applications. Consumer inertia is also a factor, not only due to network but also learning effects – consumers may understandably be loathe to switch to possibly better alternatives because of the trauma of their early experience with learning how to use Microsoft software! One would therefore have to consider the effect of introducing sunk costs in the model. Total profit in the market would then be equal to the value of the sunk costs an entrant needs to incur to enter it. This would reduce welfare as there would be less firms, each charging more than in a perfectly competitive market, so consumers would have to pay more and would obtain software that is further from their own preferences.

Core and extensions: Open source development occurs over time, and developers may be involved into either the development of the core or of extensions of the program that are designed to adapt it to specific uses. Core and extensions may be developed simultaneously, but the development of the core will usually determine what extensions of the program can be made. The development of the core could be modeled as in the present paper as a collective enterprise, involving many entrepreneurs into a common project. The development of the extensions on the other hand could be modeled as a private enterprise, whereby a developer can choose to develop an extension to the program at some cost, and the extension would allow the developer to move the point of gravity of the project from its original point to a point closer to the developer's own preferred point. Under the GPL, a developer could borrow from an OS project and develop on it for her own private uses and goals without contributing back to it, as long as she does not sell it for profit, while under the BSD, she would be able to sell software that includes OS code. This leads me to a further possible extension:

Proprietary extensions: What if a BSD license was used in OS production, so proprietary developers could borrow from OS software for use in their own proprietary applications? Consider an OS project that is under the Berkeley Software Distribution ('BSD') licensing scheme. In that case, proprietary developers can extend the software and sell the extension to the public. I could suppose as above that there is a cost to developing an extension to the OS program, that this allows the developer to move the location of the project to some extent and that the proprietary developer may sell the extension. The extension would have to be developed on an element of the core, that is, its location would have to be within the boundaries of the original project. The proprietary extension would extend the number of users of the open source software and may also increase motivation in participating in the open source project since proprietary exploitation of the project may be made further down the line.

Further work on that line would examine competition between independently developed proprietary projects and proprietary extensions on open source products. This would differ from the previously explored competition between proprietary projects and open source projects that were developed under the General Public License ('GPL'). This extension would be relevant as there is ongoing debate over whether to use restrictive (GPL) or permissive (BSD) licenses.[4]

Dynamics: A mixed industry was shown to possibly occur when proprietary software enters an industry dominated by OS software, but the opposite was shown not to be possible. Which equilibrium will occur in the mixed-industry model thus depends on the way the OS equilibrium will converge to the mixed industry equilibrium after entry by proprietary software. This will be the subject of further inquiry.

References

ALESINA, A., AND E. SPOLAORE (1997): "The number and size of nations," *The Quarterly Journal of Economics*, 112(4), 1027–1056.

BALLOU, J. P. (2005): "An Examination of the Presence of Ownership Effects in Mixed Markets," *Journal of Law, Economics, & Organization*, 21(1), 228–255.

BERGSTROM, T., L. BLUME, AND H. VARIAN (1986): "On the Private Provision of Public Goods," *Journal of Public Economics*, 29(1), 25–49.

BESLEY, T., AND M. GHATAK (2001): "Government Versus Private Ownership Of Public Goods," *The Quarterly Journal of Economics*, 116(4), 1343–1372.

BESSEN, J. (2006): "Open source software: Free provision of complex public goods," in *The economics of open source software development*, ed. by J. Bitzer, and P. J. H. Schröder, pp. 57–81. Elsevier: Amsterdam.

BEZROUKOV, N. (1999): "Open Source Software Development as a Special Type of Academic Research (Critique of Vulgar Raymondism)," *First Monday*, 4(10).

BITZER, J. (2004): "Commercial versus open source software: the role of product heterogeneity in competition," *Economic Systems*, 28, 369–381.

BITZER, J., AND P. J. H. SCHRÖDER (2006): "The impact of entry and competition by open source software on innovation activity," in *The Economics of Open Source Software Development*, ed. by J. Bitzer, and P. J. H. Schröder, pp. 219–246. Elsevier: Amsterdam.

BONACCORSI, A., AND C. ROSSI (2006): "Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement: From Community to Business," *Knowledge, Technology, and Policy*, 18(4), 40–64.

BRAID, R. M. (2004): "Uneven spacing in free-entry equilibrium for spatial product differentiation," *Economics Letters*, 84(2), 155–161.

BRHLIKOVA, P. (2004): “Models of competition between one for profit and one non profit firm,” *CERGE-EI Working Paper no. 240*.

——— (2006): “Mixed Competition and Welfare Under Various Nonprofit Objectives // Mixed Competition Under Various Cost Configurations,” *CERGE-EI Working Paper No. 310*.

CASADESUS-MASANELL, R., AND P. GHEMAWAT (2006): “Dynamic mixed duopoly: A model motivated by Linux vs. Windows,” *Management Science*, 52(7), 1072–1085.

COMINO, S., AND F. M. MANENTI (2005): “Government Policies Supporting Open Source Software for the Mass Market,” *Review of Industrial Organization*, 26(2), 217–240.

DARMON, E., T. LE TEXIER, AND D. TORRE (2006): “Commercial or Open Source Software? Winner-Takes-All Competition, Partial Adoption and Efficiency,” Discussion paper, University of Nice-Sophia Antipolis, CREA and University of Rennes.

ECONOMIDES, N., AND E. KATSAMAKAS (2006): “Two-sided competition of proprietary vs. open source technology platforms and the implications for the software industry,” *Management Science*, 52(7), 1057–1071.

EPPLE, D., AND R. E. ROMANO (1998): “Competition Between Private and Public Schools, Vouchers and Peer-Group Effects,” *American Economic Review*, 88(1), 33–62.

FERSHTMAN, C., AND N. GANDAL (2004): “The Determinants of Output per Contributor in Open Source Projects: An Empirical Examination,” Discussion paper, Tel-Aviv University, Michigan State University, and CEPR.

FRANKE, N., AND E. VON HIPPEL (2003): “Satisfying heterogeneous user needs via innovation toolkits: the case of Apache security software,” *Research Policy, Special Issue on Open Source Software*, 32(7), 1199–1215.

FREY, B. S. (1999): “State Support and Creativity in the Arts: Some New Considerations,” *Journal of Cultural Economics*, 23(1), 71–85.

GAUDEUL, A. (2003): “The LaTeX project: A case study of open-source software,” *TUGBoat*, 24(1), 66–79.

——— (2005): “Public Provision of a Private Good: What is the Point of the BSD License?,” *SSRN eLibrary*, http://ssrn.com/paper=933631.

——— (2007): “Do Open Source Developers Respond to Competition?: The LaTeX case study,” *Review of Network Economics*, 6(2), 239–263.

——— (2008): "Open Source Licensing in Mixed Markets, or Why Open Source Software Does Not Succeed," *CCP Working Paper No. 08-2*.

GAUDEUL, A., L. MATHIEU, AND C. PERONI (2008): "Blogs or the economics of reciprocal (in)attention," Discussion paper, School of Economics and ESRC Centre for Competition Policy, University of East Anglia, Norwich.

GLAESER, E. L. (2002): "The governance of not-for-profit firms," *Harvard Institute of Economic Research, Paper No. 1954*.

HARISON, E., AND H. KOSKI (2008): "Does open innovation foster productivity? Evidence from open source software firms," Discussion paper 1135, University of Groningen and ETLA.

HEALY, K., AND A. SCHUSSMAN (2003): "The Ecology of Open-Source Software Development," Discussion paper, Department of Sociology, University of Arizona.

JOHNSON, J. P. (2002): "Open Source Software: Private Provision of a Public Good," *Journal of Economics & Management Strategy*, 11(4), 637–662.

KLINCEWICZ, K. (2005): "Innovativeness of open source software projects," Discussion paper, Tokyo Institute of Technology and Warsaw University.

KRISHNAMURTHY, S. (2002): "Cave or Community?: An Empirical Examination of 100 Mature Open Source Projects," *First Monday*, 7(6).

KUAN, J. (2002): "Open Source Software as Lead User's Make or Buy Decision: A study of Open and Closed Source Quality," Discussion paper, Stanford Institute for Economic Policy Research.

LAKHANI, K., AND E. VON HIPPEL (2003): "How open source software works: "free" user-to-user assistance," *Research Policy*, 32(6), 923–943.

LAMBARDI, G. D. (2007): "Code development: Open *vs.* proprietary source," Discussion paper, GREMAQ, University of Toulouse.

LEONCINI, R., F. RENTOCCHINI, AND G. V. MARZETTI (2008): "You won the battle. What about the war? A model of competition between proprietary and open source software," University of Trento, Discussion paper 2008.11.

MOCKUS, A., R. T. FIELDING, AND J. D. HERBSLEB (2005): "Two case studies of open source software development: Apache and Mozilla," in *Perspectives on free and open source software*, ed. by J. Feller, B. Fitzgerald, S. A. Hissam, and K. R. Lakhani, chap. 10, pp. 163–210. The MIT Press: Cambridge, MA.

MUSTONEN, M. (2005): “When Does a Firm Support Substitute Open Source Programming?,” *Journal of Economics & Management Strategy*, 14(1), 121–139.

O’MAHONY, S. (2005): “Non-Profit Foundations and their Role in Community-Firm Software Collaboration,” in *Perspectives on Free and Open Source Software*, ed. by J. Feller, B. Fitzgerald, S. A. Hissam, and K. R. Lakhani, chap. 20, pp. 393–414. The MIT Press: Cambridge, MA.

PROPPER, C., AND K. GREEN (2001): “A Larger Role for the Private Sector in Financing UK Health Care: the Arguments and the Evidence,” *Journal of Social Policy*, 30(4), 685–704.

RAYMOND, E. (2001): *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary.* O’Reilly Media, Inc.

RAYNES-GOLDIE, K. (2004): “Pulling sense out of today’s informational chaos: LiveJournal as a site of knowledge creation and sharing,” *First Monday*, 9(12).

ROSE-ACKERMAN, S. (1990): “Competition between non-profits and for-profits: entry and growth,” *Voluntas*, 1(1), 13–25.

SALOP, S. (1979): “Monopolistic competition with outside goods,” *Bell Journal of Economics*, 10(1), 141–156.

SCHMIDT, K. M., AND M. SCHNITZER (2003): “Public Subsidies for Open-Source? Some Economic Policy Issues of the Software Market,” *Harvard Journal of Law & Technology*, 16(2), 473–505.

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

SLOAN, F. A. (2000): “Not-for-profit ownership and hospital behavior,” in *Handbook of Health Economics*, ed. by A. Culyer, and J. Newhouse, chap. 21, pp. 1141–1174. Elsevier Science B.V.

UNESCO INSTITUTE FOR STATISTICS (2007): “What do societies invest in education? Public versus private spending,” UIS/FS/07/04.

VICKREY, W. (1964): *Microstatics.* Harcourt, Brace and World: New York, USA.

VICKREY, W. S. (1999): “Spatial competition, monopolistic competition, and optimum product diversity,” *International Journal of Industrial Organization*, 17(7), 953–963, with foreword by S.P. Anderson and R.M. Braid.

VON ENGELHARDT, S., AND S. SWAMINATHAN (2008): “Open source software, closed source software or both: Impacts on industry growth and the role of intellectual property rights,” Discussion paper 799, FSU Jena and DIW Berlin.

VON HIPPEL, E., AND G. VON KROGH (2003): “Open Source Software and the “Private-Collective” Innovation model: Issues for Organization Science,” *Organization Science*, 14(2), 209–223.

WICHMANN, T. (2002): “Firms’ open source activities: Motivations and policy implications,” Free/Libre and Open Source Software Final Report, Part II, International Institute of Infonomics.

Notes

[1] ‘Public’ being private in the UK of course…

[2] Note that in practice, modular development can overcome part of the problem, as developers are then able to develop their own module without separating from the project altogether.

[3] I do not consider here those developers who extend BSD software with their own proprietary packages, in the way Scientific Workplace provides an interface to LaTeX. Rather, I consider the case of developers who develop proprietary applications from scratch to compete with a dominant OSS.

[4] See ‘controversy’ in http://en.wikipedia.org/wiki/Free-software_licence

A PROOF OF PROPOSITION 4

The proprietary project will set up in the middle of the distance between two OS projects, where the marginal contributor, who obtains the lowest utility of all contributors, is located. Suppose OS projects are of size s_o, which as explained above is the result of stage 1 where the size of each project is that of a symmetric equilibrium with no anticipated entry by proprietary software. Here (case A), I suppose developers are naïve and do not take into account that if they flee the OS project, then others are likely to do so too. Then the marginal consumer will be such that

$$g(1 - as_p/2) - p = g(1 - a(s_o - s_p)/2) - K/s_o \tag{6}$$

Indeed, $(s_o - s_p)/2$ is the distance of the marginal developer to the location of the OS project. Since the marginal consumer does not anticipate flight, then it anticipates that the required contribution will still be K/s_o.

Profit for the entrepreneur will be

$$\pi_p = ps_p - K \tag{7}$$

with $p = ag(s_o/2 - s_p) + K/s_o$ by the equation above.

For profit made through entry to be positive, I must have $p \geq K/s_p$ which translates through the expression of p into $ag(s_o/2 - s_p) + K/s_o \geq K/s_p$ (i.e. price must be more than average costs).

Denoting $T = ags_o/2 + K/s_o$, this translates in terms of s_p into

$$ags_p^2 - Ts_p + K \leq 0 \tag{8}$$

which is not possible if $\Delta = T^2 - 4agK < 0$ and is possible if $\Delta = T^2 - 4agK > 0$ s.t. $s_p \in [s_p^1, s_p^2]$ with $s_p^1 = (T - \sqrt{\Delta})/2ag$ and $s_p^2 = (T + \sqrt{\Delta})/2ag$.

Therefore, entry is not profitable s.t.

$$(ags_o/2 + K/s_o)^2 - 4agK \leq 0 \tag{9}$$

which translates in $s_o^2 \in [(6 - 4\sqrt{2})K/ag,(6 + 4\sqrt{2})K/ag]$.

In the symmetric equilibria, I will have $s_o^2 \in [2K/ag,(10 + 4\sqrt{6})K/ag]$, which means that only symmetric equilibria such that $s_o^2 \leq (6 + 4\sqrt{2})K/ag$ will be robust to entry under the case A. This translates in the condition expressed in the proposition.

B PROOF OF PROPOSITION 5

The proprietary project will set up in the middle of the distance between two OS projects, where the marginal contributor, who obtains the lowest utility of all contributors, is located. Suppose OS projects are of size s_o, which as explained above is the result of stage 1 where the size of each project is that of a symmetric equilibrium with no anticipated entry by proprietary software. Here (case A), I suppose developers are naïve and do not take into account that if they flee the OS project, then others are likely to do so too. Then the marginal consumer will be such that

$$g(1 - as_p/2) - p = g(1 - a(s_o - s_p)/2) - K/(s_o - s_p/2) \tag{10}$$

Indeed, $(s_o - s_p)/2$ is the distance of the marginal developer to the location of the OS project. Since the marginal consumer anticipates flight, then she anticipates that the required contribution will still be $K/(s_o - s_p/2)$ as half of the buyers of the PS will come from her own project.

Profit for the entrepreneur will be

$$\pi_p = ps_p - K \tag{11}$$

with $p = ag(s_o/2 - s_p) + K/(s_o - s_p/2)$ by the equation above.

For profit made through entry to be positive, I must have $p \geq K/s_p$ which translates through the expression of p into $ag(s_o/2 - s_p) + K/(s_o - s_p/2) \geq K/s_p$ (i.e. price must be more than average costs). Note already that this condition is easier to fulfill than in case A.

This translates in terms of s_o into $ags_p s_o^2 - (\frac{5}{2}ags_p^2 + 2K)s_o + ags_p^3 + 3Ks_p \geq 0$ (equation 1) Denote Δ the determinant of this function; $\Delta = 4K^2 - 2Kags_p^2 + \frac{9}{4}a^2g^2s_p^4$, which is a function of s_p^2. The determinant of Δ is always negative, which means that $\Delta > 0$ for any s_p so that for any s_p there are always solutions to the equation 1. For any s_o, the proprietary developer need only set s_p such that either $s_o \geq (\frac{5}{2}ags_p^2 + 2K + \sqrt{\Delta})/2ags_p$ or $s_o \leq (\frac{5}{2}ags_p^2 + 2K - \sqrt{\Delta})/2ags_p$ in order for entry to be successful. Therefore, entry is always possible.

One can check for example that in the pure OS equilibrium where $s_o = \sqrt{2K/ag}$ (the equilibrium with the smallest possible OS project size) then the proprietary entrepreneur can set $s_p = s_o$ and then the condition for entry is verified (indeed, in that case, $s_o \leq (\frac{5}{2}ags_p^2 + 2K - \sqrt{\Delta})/2ags_p$).

C PROOF OF PROPOSITION 6

The expression of social welfare in equation 2 can be expressed as a function of s by exploiting the fact that in a symmetric equilibrium, $N = 1/s$, so social welfare will be

$$SW = g - K/s - ags/4 \tag{12}$$

which is a concave function of s. Therefore, the minimum possible welfare in an OS industry that is not subject to entry by proprietary software will be either when $s = \sqrt{2K/ag}$ (minimum stable size) or when $s = (\sqrt{6} + 2)\sqrt{K/ag}$ (maximum size without forking).

In the first case, total welfare is

$$SW = g - \frac{3\sqrt{2}}{4}\sqrt{Kag} \tag{13}$$

to be compared to optimal welfare of $SW^* = g - \sqrt{Kag}$, so welfare is reduced by at most 6.07% of $\sqrt{Kag}$ in the smallest size OS equilibrium.

In the second case, total welfare is

$$SW = g - \frac{3\sqrt{6}-2}{4}\sqrt{Kag} \tag{14}$$

which is less than optimal welfare by 33.71% of $\sqrt{Kag}$. This is therefore the highest possible loss compared to optimal welfare that can occur under an OS equilibrium when there is no threat of entry by PS.

Now, when the OS industry is subject to entry by proprietary software, and one is under case A (naïve users), then s_o cannot be higher than $(\sqrt{2}+2)\sqrt{K/ag}$, which puts a new lower bound to the loss in welfare as this limits overgrowth in OS projects. From equation 12, social welfare will be

$$SW = g - (\frac{6-\sqrt{2}}{4})\sqrt{Kag} \tag{15}$$

so the loss of welfare compared to the social optimum will be limited in case A to 14.64% of $\sqrt{Kag}$, compared to 33.71% of $\sqrt{Kag}$ when the threat of forking was the only factor limiting the size of OSS projects. Threat of entry by PS under case A may thus improve welfare by limiting the size of OSS projects.

Under case B however the optimum of the social planner is not achievable as it will not be robust to entry by proprietary firms.

D PROOF OF PROPOSITION 7

Consider a configuration with firms i, j and k located in this order on the preference circle, with distance measured clockwise from an arbitrary point of reference. With a shortcut in notation, say that $i < j < k$ is the location of firm i, j and k respectively. The utility of individual A situated at distance l_{Aj} from firm j is $U_A = g(1 - al_{Aj}) + y - p_j$ when buying from firm j, situated between i and k.

The profit of the firm j, if l_j and L_j are the distance to the right and to the left of the consumer that is indifferent between i and j and j and k respectively, is $\pi_j = p_j(L_j + l_j) - K$, with K the cost of development.

l_j will be such that $g(1 - al_j) + y - p_j = g(1 - aL_i) + y - p_i$ with $L_i = j - i - l_j$ while L_j is such that $g(1 - aL_j) + y - p_j = g(1 - al_k) + y - p_k$ with $l_k = k - j - L_j$.

I find that $\frac{1}{2}(j-i) - \frac{p_j - p_i}{2ag} = l_j$ while $\frac{1}{2}(k-j) - \frac{p_j - p_k}{2ag} = L_j$ so the firm maximizes

$$\pi_j = p_j(\frac{1}{2}(k-i) - \frac{2p_j - p_i - p_k}{2ag}) - K \tag{16}$$

This is maximized for $\frac{1}{4}(ag(k-i)+p_i+p_k) = p_j$. Consider a symmetric equilibrium, and denote p the equilibrium price for all firms. Then I have

$$p_j = \frac{1}{4}(ag(k-i) + 2p) \tag{17}$$

and $p_j = p$ s.t. $p = \frac{1}{2}ag(k-i)$.

Replacing the above into the zero profit condition, I obtain $k - i = 2\sqrt{K/ag}$, which is the space taken up by two symmetric firms, so any one firm is of size $s = \sqrt{K/ag}$, which is less than the maximum size of OS projects in a pure OS model. The number of firms is then the closest integer to $\sqrt{ag/K}$, which is more than the number of OS projects in a pure OS model. Note also that $p_j > t$, the contribution needed to sustain OS (assuming contributions are expressed in monetary terms). The situation will be robust to entry by an additional proprietary firm.

E PROOF OF PROPOSITION 8

I want to show that any entry of any size s_o by open source software is not possible. Suppose the entrant locates in the middle of two proprietary software (the best entry position as it targets the consumers who are most dissatisfied with the proprietary offering), and there is no strategic change in price or location as a competitive measure by the entrepreneur, then the marginal consumer will be such that:

$$g(1 - \frac{a}{2}(s_p - s_o)) - p \leq g(1 - a\frac{s_o}{2}) - \frac{K}{s_o} \tag{18}$$

For such a s_o to exist, I must have

$$gas_o^2 - (\frac{ag}{2}s_p + p)s_o + K \leq 0 \tag{19}$$

Remember that $p = \sqrt{Kag}$ in the proprietary equilibrium, while $s_p = \sqrt{K/ag}$. Replacing this in the above, this means I must find s_o such that:

$$gas_o^2 - (\frac{3}{2}\sqrt{Kag})s_o + K \leq 0 \tag{20}$$

but the determinant of this function is $\Delta = \frac{9}{4}Kag - 4Kag < 0$ so there is not solution to the equation. This means that there is no s_o that would make the marginal consumer prefer contributing to the OS project rather than buy from the proprietary entrepreneur in the equilibrium of the proprietary industry development game.

F PROOF OF PROPOSITION 10

First note that projects will be at distance $1/N$ of each other. Indeed, in equilibrium, proprietary projects will locate in the middle of two adjoining OS projects to maximize differentiation and thus profit. Similarly, in a symmetric equilibrium where all PS is of the same price, OS projects will locate in the middle of proprietary projects. In a symmetric equilibrium, projects will thus be equally spaced. Consider a consumer who has a choice between paying p for proprietary software that is at distance l from her preferred point, or contributing t to an OS project which is at distance $1/N - l$ from her preferred point.

She will be indifferent between the two if

$$g(1 - al) - p = g(1 - a(1/N - l)) - t \tag{21}$$

which means that l^*, the location of the indifferent consumer, will be such that

$$l^* = [1/N - (p - t)/ag]/2 \tag{22}$$

Profit for the proprietary developer will be

$$\pi_p = 2l^* p - K \tag{23}$$

and profit maximizing price will be such that

$$\partial\pi_p/\partial p = 0 \tag{24}$$

so optimal price will be $p^* = \frac{1}{2N}(ag + Nt)$ and therefore, from equation 22,

$$l^* = (ag + Nt)/4Nag \tag{25}$$

In a zero profit equilibrium, I will have $\pi_p = 0$, that is, $2l^*p^* = K$, which translates in $(Nt + ag)^2 = 4N^2agK$ which obtains only one solution with positive contributions, $t^* = 2\sqrt{agK} - ag/N$, so that from equation 25, $l^* = \frac{1}{2}\sqrt{K/ag}$.

This means that the size of the proprietary projects is $s_p = \sqrt{K/ag}$ in any mixed strategy equilibrium of the development game. This is equal to their equilibrium size in a proprietary industry.

Now, to find N^*, remember that the OS project will require contribution t such that $2(1/N - l^*)t = K$ (total contributions equal the cost of the project, so that replacing t with its optimal level t^*, that I find that N^* must be a solution to $3KN^2 - 5\sqrt{Kag}N + 2ag = 0$. The two solutions are $N^* = \left\{\sqrt{ag/K}, \frac{2}{3}\sqrt{ag/K}\right\}$.

If $N^* = \sqrt{ag/K}$, then $p^* = \sqrt{Kag}$ and $t^* = \sqrt{Kag}$

If $N^* = \frac{2}{3}\sqrt{ag/K}$, then $p^* = \sqrt{Kag}$ and $t^* = \frac{1}{2}\sqrt{Kag}$.

I have to check the *stability of the first equilibrium*, since OSS projects are smaller in the first equilibrium than was stable in the pure OS industry. In equilibrium, I have

$$g(1 - as_p/2) - p = g(1 - as_o/2) - K/s_o$$

(indifferent consumer condition). Here (first equilibrium), $s_o = s_p$. This case where $s_o = s_p \equiv s$ is stable only if, supposing the OS project increases in size by ε at the expense of proprietary projects, then the individual at the new border at distance $(s - \varepsilon)/2$ of the proprietary projects prefers paying the (unchanged) p rather than contributing to the OS project, that is:

$$g(1 - a(s + \varepsilon)/2) - K/(s + \varepsilon) < g(1 - a(s - \varepsilon)/2) - p \tag{26}$$

that is, $K - p\varepsilon - ps + ag\varepsilon^2 + ags\varepsilon > 0$ which simplifies to $K > ps$ as ε tends to 0, which is indeed verified in equilibrium since this is the free entry condition (zero profit for the proprietary developer).

G PROOF OF PROPOSITION 11

The expression for social welfare in equation 1 can be expressed as

$$SW = g - \sum_{x=1}^{N} ag\frac{s_x^2}{4} - NK \tag{27}$$

In the mixed equilibrium with asymmetrically sized OS and proprietary projects, $s_o = 2\sqrt{K/ag}$ and $s_p = \sqrt{K/ag}$ and there is an equal number of OS and proprietary projects. The total number of projects is $N = 2/(s_o + s_p) = \frac{2}{3}\sqrt{ag/K}$. Denote $N_o = N_p = N/2$ the number of open source and proprietary projects respectively.

Total welfare will then be

$$SW = g - \frac{N}{2}(ag\frac{s_o^2}{4} + ag\frac{s_p^2}{4}) - NK \tag{28a}$$

$$= g - \frac{13}{12}\sqrt{agK} \tag{28b}$$

Welfare in the mixed model is thus lower than welfare in the minimum sized stable equilibrium of the pure OS model ($13/12 > 3\sqrt{2}/4$) but higher than in the maximum sized forking-proof equilibrium of the pure OS industry ($13/12 < (3\sqrt{6}-2)/4$), and also higher than in the maximum sized entry-proof equilibrium of the OS industry ($13/12 < (6 - \sqrt{2})/4$). The mixed model is a clear improvement on total welfare in the pure proprietary equilibrium: There is a loss of 8.33% of $\sqrt{agK}$ in welfare compared to the optimum, compared with a loss of 25% of $\sqrt{agK}$ in the pure proprietary model.