



Dynamics of Institutions and Markets in Europe is a network of excellence of social scientists in Europe, working on the economic and social consequences of increasing globalization and the rise of the knowledge economy.
<http://www.dime-eu.org/>

DIME Working Papers on INTELLECTUAL PROPERTY RIGHTS



Sponsored by the
6th Framework Programme
of the European Union

<http://www.dime-eu.org/working-papers/wp14>

Emerging out of DIME Working Pack:

'The Rules, Norms and Standards on Knowledge Exchange'

Further information on the DIME IPR research and activities:

<http://www.dime-eu.org/wp14>

This working paper is submitted by:

Joachim Henkel^{*} and Simone Käs^{}**

^{*}Technische Universität München and CEPR
Email: Jhenkel@hbs.edu

^{**}McKinsey & Company.
Email: Simone_kaes@mckinsey.com

*The emergence of openness: How firms discover
free revealing as a means to appropriate value*

**This is Working Paper
No 37 (April 2008)**

The Intellectual Property Rights (IPR) elements of the DIME Network currently focus on research in the area of patents, copyrights and related rights. DIME's IPR research is at the forefront as it addresses and debates current political and controversial IPR issues that affect businesses, nations and societies today. These issues challenge state of the art thinking and the existing analytical frameworks that dominate theoretical IPR literature in the fields of economics, management, politics, law and regulation- theory.

The emergence of openness:

How firms discover free revealing as a means to appropriate value

Joachim Henkel¹, Simone Käs²

Abstract

In open innovation processes, and notably in the commercial development of open source software, firms increasingly make some of their intellectual property freely and publicly available. This paper extends existing research on this topic by taking a dynamic perspective. We analyze what triggers the transition from ‘closed’ to ‘open,’ how openness evolves over time, how firms manage this shift, and what its implications for firm strategy are. For our study, we focus on driver software for computer components to be used with the Linux operating system. We use interviews, document analysis, and a survey to study this industry’s move to a higher level of openness. It turns out that Linux being open source software was not so much the reason for this increased openness, but rather a trigger to rethink engrained industry practice of overly strong protection. A feedback loop between increasing customer demand for openness and learning on the sellers’ side then set in. As a result, both value creation and value capture by the focal firm are potentially increased, and openness has emerged as a dimension of product quality in which firms compete.

¹ Technische Universität München and CEPR, currently visiting at Harvard Business School. Jhenkel@hbs.edu.

² McKinsey & Company. Simone_kaes@mckinsey.com.

Extended abstract

Background

In order for innovation to be profitable, the innovator must be able to appropriate a sufficient share of the value the innovation creates. One established criterion for such appropriation is the strength of the “appropriability regime.” However, recent phenomena in the field of intellectual property management suggest to qualify this criterion. Closer analysis shows that Teece’s (1986) notion of the appropriability regime in fact refers to *excludability* rather than *appropriability*. The two are related, but differ in important cases.

It turns out that firms may appropriate more of the value they create by waiving available means of exclusion and practicing openness instead. As we show in this study, firms may use openness actually as a dimension of product quality in which to compete. Such behavior, though, is in stark contrast to established practices of intellectual property management, which favor strong protection and exclusion. Thus, a learning process is required, which we analyze empirically in this paper. By thus taking a dynamic perspective, we go beyond existing research. We analyze what triggers the transition from ‘closed’ to ‘open,’ how openness evolves over time, how firms manage this shift, and what its implications for firm strategy are.

Approach

We analyze the emergence of openness empirically, using interview and survey data. As our subject of study we chose computer component manufacturers, focusing on three market segments of components relevant for the Linux operating system. These are the two most important types of components in the Linux area, embedded processors and embeddable single board computers (SBC), plus one additional type of component, embeddable data acquisition (DAQ) cards.

Components require certain software programs—drivers—which are specific to the operating system used. For drivers dedicated to the Linux open source operating system, the license which governs the use of Linux (the GPL) suggests to put the driver, too, under the GPL. However, this is not mandatory, and openness of driver source code indeed exhibits a strong variation between firms. It is in some cases made openly available under an open source license, in other cases kept secret (i.e., they are given out in compiled, or “binary,” form only). Essentially, it is thus a decision between “proprietary” and “open” to be taken by the firm selling the component. This decision is at the center of our analysis. In more detail, we analyze three aspects of openness: the way in which it is practiced, its level, and the change of this level over time.

Qualitative study

The key conclusions from the qualitative study can be summarized as follows. Linux, being OSS, did not so much *cause* the switch to openness, since binary-only drivers are legal and common. However, it *triggered a re-thinking* of industry practice, namely that of keeping driver source code closed. Then, a positive feedback loop set in between increasing demand for openness on the customer side, and learning/positive experiences on the side of the firms, which ultimately led to gradually increasing openness in the industry. This feedback loop is illustrated in Figure 1.

The logic behind this loop is that open driver source code provides additional value for the customers (customization, simplified bug fixing, vendor independence, quality assurance), a part of which can be appropriated by the component vendor in terms of higher willingness to pay and increased sales. At the same time, openness of driver code creates direct benefits for the firms releasing the drivers, as they are able to realize quality and efficiency gains due to external development support. Having tried out openness, many component sellers realize these advantages, and also realize that the suspected disadvantages due to a potential loss of

intellectual property are limited. On the other side, buyers realize that, after all, they do have the opportunity to receive driver source code, and make this an increasingly important purchasing criterion—keeping the feedback loop in motion. It is important to note that openness thus allows firms not only to create, but also to appropriate value.

Quantitative survey

For the quantitative survey, we manually compiled a list containing, to the best of our knowledge, all firms worldwide active in these market segments, and sent links to online questionnaires to 192 firms after establishing personal contact to a suitable respondent. 74 valid questionnaires were returned (39% response rate), of which due to item non-response 62 could be used for multivariate analysis.

As to the way of revealing, we identified two different approaches corresponding to distinct aims of revealing. For increasing the product's value for users, offering the code for download on the company website is sufficient. For instigating informal external development support, in contrast, a bona fide public open source project needs to be launched, with forum, mailing list, code repository, and an active maintainer. We perform a multivariate analysis of the level of openness that the responding firms have chosen for their Linux driver software, distinguishing between revealing in general (i.e., both on a website and in an OSS project) and revealing as part of an OSS project.

The dependent variable in the regressions refers to the percentage of code revealed, and is thus by definition double censored (lying between 0 and 100). Accordingly, a Tobit regression model is applied. In order to check robustness of results with respect to the model specification, an Ordered Probit model was used additionally. In this model, the dependent variable lies in the set {1, 2, 3} where “1” indicates that no code is revealed (0%), “2” indicates that “5-95%” are revealed, and “3” indicates that all code is revealed (100%), which leads to three groups of roughly similar size. To rule out multicollinearity, variance inflation

factors have been calculated for the explanatory variables. As all were below the critical value of 10, multicollinearity is not a concern.

With few exceptions, the results are consistent across different specifications. Most striking are the following findings. For revealing driver source code in general, we find that customer demand for source code (Cust) has a significant positive effect in all specifications, confirming our hypothesis H1 (not developed in this abstract). In contrast, the levels of importance that the respondents attach to marketing, technology improvements, and external development support as reasons to reveal (variables MARK, TECH, DEV) do not show any significant effect in the Tobit models, and only partly in the ordered probit model (these variables are derived using factor analysis, with each capturing the reported importance to the firm of a number of potential motives to reveal the source code related to, respectively, marketing, quality enhancement, and development support).

This picture is completely reversed when focusing on the share of code revealed in OSS projects. Now, customer demand (Cust) is insignificant, while the factors MARK, TECH, and DEV carry the expected positive sign in all specifications (with one exception out of 12).

Conclusions

Our results inform the theory of the role of innovation in competitive strategy in two respects. First, our study elucidates the use of openness—in our case, openness of driver source code—as a competitive weapon. Second, we show how openness as a new degree of freedom in managing intellectual property, and as a dimension of competition emerges over time. Starting out from a situation in which closedness of drivers was taken for granted by all market participants, an iterative interplay of increasingly articulated customer need for openness and growing experience of firms in how to handle openness has over time led to a

diversification between firms, and between products within firms, with respect to the extent of openness.

In the literature, various “open source business models” have been discussed. This term may suggest that firms can profit from openness mainly in the context of open source software, and only by adopting specific business models. Our results suggest otherwise. In our study, the fact that Linux is open source software under a particular license (the GPL) does not oblige firms to reveal their driver source code to the public. However, Linux being OSS did serve to trigger a process of re-thinking engrained industry practice, namely, the paradigm that more exclusion is better. After this initial triggering, a positive feedback loop between increasing demand for openness on the customer side and positive experiences on the side of the firms set in, which became largely decoupled from the open source issue. Thus, our results support the view that the strong “proprietary” stance towards intellectual property management, while widespread practice, is in general not rational: a selective approach balancing protection with openness appears superior. Once pushed to venture openness, firms actually realize that it can serve them as a competitive weapon, enhancing both value creation and value appropriation.

Figure 1: Linux as a trigger to rethink industry practice

