PUTTING A VALUE ON OPENNESS: THE EFFECT OF PRODUCT SOURCE CODE RELEASES ON THE MARKET VALUE OF FIRMS[1]

OLIVER ALEXY

Technische Universität München
TUM Business School
Arcisstr. 21, D – 80333 Munich, Germany
Tel: +49-89-28925741
Fax: +49-89-28925742
E-mail: alexy@wi.tum.de

WORKING DRAFT (5 OCTOBER 2007)

PLEASE DO NOT CITE OR CIRCULATE

FEEDBACK IS HIGHLY APPRECIATED!

[1] The author would like to thank Jörn Block, Joachim Henkel, and Simone Käs for valuable feedback on this paper. Special thanks go to Hans Zischka for valuable assistance in data collection and to Josef Waltl for help with earlier versions of this paper. All remaining errors are of course my own.

PUTTING A VALUE ON OPENNESS: THE EFFECT OF PRODUCT SOURCE CODE RELEASES ON THE MARKET VALUE OF FIRMS

ABSTRACT

This study examines the effect of releasing the source code of commercial software products as open source software on the market value of firms. Using a sample of 30 software companies in the time span from 1 January 1999 to 30 April 2007, I find that market valuation is influenced by investor sentiment—abnormal returns take a curvilinear shape over time—and the business model firms choose for their OSS efforts—non-existence of an explicit revenue model is punished by the capital market. From my findings, I deduce several implications for IT-related event studies and research on open innovation processes.

Keywords: open source software; event study; investor sentiment; business model;

PUTTING A VALUE ON OPENNESS: THE EFFECT OF PRODUCT SOURCE CODE RELEASES ON THE MARKET VALUE OF FIRMS – AN EVENT STUDY

INTRODUCTION

Does open source software (OSS) pay off? Over the last years, researchers have put a lot of effort in explaining the open source phenomenon, its impact on the innovation process within and outside firms, and the economic effects of open source strategies of commercial firms (e.g., Dahlander, 2005; Lerner & Tirole, 2002; Stewart & Gosain, 2006; von Hippel & von Krogh, 2003; West, 2003; West & Gallagher, 2006). They found that, in many cases, patterns of contributing code are consistent with profit maximization.

Still, the current research on open source software does not yet answer the question whether open source activities in general and the releasing of product source code in particular encompass an effect on the value of firms—and, more specifically, a *positive* effect. In this paper, I thus tackle the question of whether product source code releases have a significant impact on the market value of firms and which factors might influence or drive this effect. After identifying a set of business models firms may choose to generate and appropriate value from releasing proprietary software as OSS, namely using OSS as a competitive weapon, cost or risk reduction, dual licensing, and the sale of complementary goods or service, I analyze how the capital market reacts to firms announcing a release of proprietary software as OSS and which factors are influential to this. I maintain that the time of the announcement in relation to the general market perception of OSS and the choice of business model affect the abnormal returns that firms may generate.

The event study method is an approved technique to measure market reaction to specific events. It is widely used in research to measure the impact of managerial decisions (McWilliams

& Siegel, 1997). Nevertheless, to my knowledge, no event study measuring the effect of OSS or open innovation (Chesbrough, 2003) efforts on the value of firms has been conducted up till now. There is one event study in the related field of open vs. proprietary Extensible Markup Language (XML) standardization where Aggarwal et al. (2006) discovered that capital markets react negatively on open XML standardizations in years from 1999 to 2003.

During the time span from January 1, 1999, to April 30, 2007, I find a curvilinear (u-shaped) trend in the market reaction to firms announcing the release of source code as OSS which can be ascribed to negative investor sentiment after the burst of the dot.com bubble. Furthermore, I maintain that the capital market specifically reacts to the choice of business model of the firm announcing the release of a (potential) product under an OSS license. In particular, investors punish firms engaging in OSS without a revenue model.

The rest of this paper is organized as follows. I will first review the literature on advantages and disadvantages of OSS for firms as well as business models they might choose when releasing their proprietary software and l deduce the hypotheses for the event study, also including the market and investor perspective. Next, I introduce data and methods, and, thereafter, present the results. Finally, I discuss the implications of my findings and limitations of the study and give recommendations for future research.

THEORY AND HYPOTHESES DEVELOPMENT

Definition of Open Source Software

Since its inception in April 1998, the term OSS is exactly specified by the open source initiative (OSI). Following their definition, OSS is software that is licensed under an OSS license approved by the OSI.[2] OSS does not necessarily mean that the software is gratis (although this is

[2] As of September 13, 2007, the OSI had approved 61licenses.

very common). For a license to be OSI compliant, users of the respective software have to be provided access to the source code (upon request, at least), the distribution of derived work must be allowed, and no discrimination against persons, groups or fields of endeavor is allowed (OSI, 2001).

Advantages of Releasing Proprietary Source Code as Open Source Software

By releasing their software under an open source license, companies can, first of all, benefit from external developers' support and the latter can make improvements and further developments to the software. As these developers will be making adaptations for other companies, in most cases the feedback received is direct user input to better tailor existing and future products to existing markets. Overall, it is highly likely that a piece software will develop faster than if kept proprietary (Dalle & Jullien, 2003; Henkel, 2004; Lakhani & von Hippel, 2003). In addition to this, by having customers make further developments to a piece of software, the company might not only be able to get a gratis stream of innovations to its product, it can also achieve higher rates of customer satisfaction. By allowing the customers to make changes and additions to the software on their own, they will be more likely to fully commit to the product, and make the improvements needed and hoped for (Goldman & Gabriel, 2005; Morrison, Roberts, & von Hippel, 2000; von Hippel, 2001). In this way, the company might also be able to receive knowledge that might have been difficult to find, transfer, or acquire otherwise, so-called "sticky information" (von Hippel, 1994, 1998). Furthermore, the community might not only help the company with the actual program, they might also engage in more mundane tasks such as user support or documentation. Again, this will reduce the amount of developer resources the company will have to spend for activities unrelated to core business that also do not generate any revenue (Goldman et al., 2005; Lakhani et al., 2003; Shah, 2006).

Standard setting and compatibility issues also play important roles. Releasing a piece of software gives the company the opportunity to make the software the standard in a certain area if none has existed there before, to tip the standard race in favor of its piece of software, or to prevent a proprietary standard. All of this is useful to the company and actually an opportunity to increase profits: if the company defines (parts of) the standard – even if it is an open one – it is highly likely that it includes parts that are only beneficial to that one company, may it be because only the company itself knows of them or because they are already optimally realized in an existing product (Goldman et al., 2005).

If one strong standard already exists, releasing a piece of software can make it part of this standard, or, at least, it will increase compatibility of the software to others (Harhoff, Henkel, & von Hippel, 2003; Hecker, 1999; Henkel, 2004; Raymond, 2001a). In any case this increased compatibility will create network effects that not only encourage distribution and adoption of the software, but also related innovations and second generation innovation built on the software (Bonaccorsi, Giannangeli, & Rossi, 2006; Farrell & Gallini, 1988; Farrell & Saloner, 1985; Harhoff et al., 2003; Henkel & von Hippel, 2005; Katz & Shapiro, 1985, 1986; Shepard, 1987). The attempt to set a standard or to at least influence the standard race might also be motivated by strategic reasons. If there is already a dominant standard or a dominant competitor in the market that the company has trouble keeping up with, building on OSS or releasing the software as OSS might be a valuable option.

Potential Downsides of Releasing Proprietary Software as OSS

While the possibility of free bugfixes and new features seems promising at first glance, a company has to bear in mind that releasing software under an OSS license does not automatically attract lots of developers who will do all the work and cost nothing. On the contrary, if the code is not modular enough, people will simply not be able to grasp the nature of

the software and will only be able to make minor contributions – the company will basically still have to do all the further developments by itself (Baldwin & Clark, 1997; Goldman et al., 2005). Thus, the source code needs to either be modular from the beginning or modified accordingly before being released. In addition to this, the source code needs to be sanitized, e.g. all inappropriate comments need to be removed, business logic extracted, and so on, which can bring a significant amount of start-up cost (Hecker, 1999).

The most obvious risk of releasing the source code of proprietary software under an OSS license is of course a loss of intellectual property, and, consequently of competitive advantage. The idea is that the company's competitors are able to right away start working with all ideas contained in the product at no additional cost. However, a caveat is in order. While spillovers to a competitor may be problematic, the firm may nevertheless profit from going OSS if the added value – lower costs of development, new features and fixed bugs, and new sources of revenue – outweighs the losses as needs to be addressed by a preceding business case (West et al., 2006). Still, the threat of losing intellectual property hampers the release of proprietary software. As, with OSS, customers will also have direct access to the source of the company's products, they may see less of a reason to pay for it. Indeed, with OSS it will no longer be possible to demand license fees for the product, so it is likely than no direct revenue stream will come from the software any longer. Instead, the company will have to look for new, mainly indirect sources of revenues, such as the sale of complementary goods or services (Raymond, 2001a).

From a technical point of view, several dangers arise. In case the source code is incomplete or of low quality, the outcome of an OSS project could range from the public merely ignoring the OSS efforts to a serious damage of the company's technical reputation. However, in no case will the possible benefits of an OSS strategy be reaped (Goldman et al., 2005; Henkel, 2004).

If a company has managed to successfully establish an OSS project based on formerly proprietary software, the danger of forking remains:[3] if people are unhappy with the way the company manages the OSS project they may simple take the existing code base and start their own project as is permitted by many OSS licenses (FSF, 2006; OSI, 2001).

Open Source Software-based Business Models for Proprietary Software Firms

> ***"How do I make money on software if I can't sell my code?***
> *You can sell your code. Red Hat does it all the time. What you can't do is stop someone else from selling your code as well. That just says that you need to add extra value to your code, by offering service, or printed documentation, or a convenient medium, or a certification mark testifying to its quality."* [4]

Despite the advantages presented above, at first glance, it may seem hardly if at all possible to directly earn money with OSS. One even has to reveal the source code of one's product to the customer which seemingly dries up all revenue streams possibly coming from this piece of software! Still, there are ways to make money with OSS, may they be indirect ones in most cases, and it is possible to build up a business around open source software. Indeed, it is explicitly allowed to demand money for the software, as was already stated in the introductory quote to this chapter. Independent of what a company is selling, it can charge every amount of money it wants to. However, what it cannot do is barring its customers from giving away the software they purchased for free, by for example putting the source code to the program on the web (FSF, 2006). According to the rights given to customers by the OSS license the company decided to use, they might also be able to create new proprietary software based on the company's existing OSS product. Yet, in most situations, the customers will either have no

---

[3] "A *fork* is a *competing project* based on a version of the pre-existing project's source code. All OSS/FS projects can be 'forked'; the ability to create a fork is fundamental to the definition of OSS/FS." (Wheeler, 2002, formatting copied from original source)

[4] http://www.opensource.org/advocacy/faq.php, retrieved March 10, 2006, 2:50 p.m.

interest in doing that, or the further distribution of the software might even be beneficial to the company, as in the case of standard setting.

Using the least common denominator found in the literature, I define a business model as the way in which the firm creates and delivers value for the customer, whereas the revenue model focuses on how the firm appropriates the rents from the business. The revenue model, consequently, is an important part of a business model (Amit & Zott, 2001; Chesbrough & Rosenbloom, 2002; Richardson, 2005). From the existing literature on OSS and the advantages of OSS specified before, four different but non-exclusive business models can be deduced: using OSS as a competitive weapon, cost or risk reduction, dual licensing, and the sale of complementary goods or assets. Indeed, oftentimes, a combination of several of these models seems more promising than choosing a single one.

In Table 1, I have briefly summarized these business models and given examples of firms using them in the past showing that the four business models not only differ with respect to the goals that firms may reach when applying them, but also with respect to time horizon and appropriability, that is, an easily applicable revenue model.

Insert Table 1 about here

As is shown in Table 1, there are important differences in the revenue models accompanying the four business models. When thinking about investors valuating those differently, it is important to understand that releasing proprietary software as OSS using the strategies of cost and risk reduction, dual licensing, and sale of complementary goods and services can be easily expressed as a business model, and a revenue model in particular, whereas the use of OSS as a competitive weapon shows little to no clear short-term benefits, and the hard-

to-quantify long-term benefits might be ignored by the capital market (Dos Santos, Peffers, & Mauer, 1993; Oh, Gallivan, & Kim, 2006). On average, I thus expect that firms who announce a release of proprietary software as OSS for merely strategic reasons will see this more negatively evaluated than firms choosing one of the three other strategies.

> *H1: Firms following the competitive weapon strategy will be evaluated less positively by investors than those firms releasing proprietary software as OSS using another business model.*

Investor Sentiment

Time does not only play a role with respect to the effects, that is, the *time horizon*, of the decision to go OSS, the *timing* of the announcement, too, might be of vital importance. When looking at the period of time since when OSS activities of firms can be observed on the capital market, it is important to note that this started before the burst of the dot.com-bubble in mid-2000. The importance of this observation lies in the fact that, sometimes, capital markets may be inefficient with respect to investors not behaving rationally (De Long, Shleifer, Summers, & Waldmann, 1990; Lee, Shleifer, & Thaler, 1991): investors will value stocks more positively in a time of positive investor sentiment and more negatively in a time of negative investor sentiment. As an example for investor sentiment with respect to the IT market around the dot.com bubble burst, Lee (2001) and Cooper, Dimitrov and Rau (2001) show that IT markets react favorably to firms changing their name to include '.com' before mid-2000, whereas Cooper et al. (2005) show a positive effect of the removal of '.com' from the firm name after mid-2000. Similarly, Dehning et al. (2004) report positive effects on stock price caused by the announcement of e-commerce initiatives in 1998 and—depending on the method they use—negative effects or indications for those for the forth quarter of 2001.

Concerning the valuation of firms announcing "openness", Aggarwal, Dai, and Walden (2006) compare the effect of the announcement of open vs. proprietary XML standards by firms

on their market value between 1999 and 2003. They find an overall negative effect which they argue is due to the fact that the firm may obtain a small monopoly with the proprietary standard, which is supposed to be evaluated more highly by the capital market than the potential of getting a small share of a larger market through open XML standardization. Yet, in line with the above argumentation, they find a negative effect of the release of an open XML standard for the year 2001 and, what is more, they find a decrease of both total announcements and announcements of open standards over time, both culminating in 2000 and then steadily decreasing.

With a normalization of investor sentiment with respect to IT investments in general and to the announcement of the release of (potential) commercial products under an OSS license in particular over time, I expect this trend to stabilize, too, and, eventually, to turn positive, that is, both the number of such announcement and their valuation by investors. In the following, I will give several reasons why this may be the case.

While customers did not expect to get access to source code to products at all at the beginning of this decade, for some segments of the IT industry, this has developed so far as that openness has even become a competitive factor, thereby making offering one's source code as OSS the rule rather than the exception in these areas (Henkel, 2006). Moreover, at the time of the burst of the dot.com-bubble, the market only had a rather small number of objects of comparisons showing the positive effects of OSS—especially with respects to their long-term sustainability of such a strategy. In the time of skepticism thereafter, consequently, investors most likely evaluated the potential of the idea of releasing valuable intellectual property into the open skeptically (Dehning, Richardson, & Zmud, 2003; Dos Santos et al., 1993; Im, Dow, & Grover, 2001; Oh et al., 2006). When positive effects of IT investments and in particular of going OSS became visible *over time* (Brynjolfsson, 1993; Dehning et al., 2003), however, the market would understand the rational behind and the positive effects of the initial decision, and

consequently reacted more positively to similar future decisions. Again, it is important to note that "market reaction" not only includes the behavior of investors observing the market, but also other firms observing the success of their competitors' OSS efforts (Cooper et al., 2005; Im et al., 2001).

> *H2: Over time, there will be a curvilinear (u-shaped) development of the effect on stock price of firms announcing the release of proprietary software as OSS.*

METHOD AND DATA

Event Study Methodology

Event studies have been widely used in the IT and IS literature (see, e.g., Dehning & Richardson, 2002; Dehning et al., 2003; Dos Santos et al., 1993; Im et al., 2001; Oh et al., 2006) but are new to the field of OSS. The research design used in this study is mainly based on the papers of McWilliams and Siegel (1997), MacKinlay (1997), McWilliams and McWilliams (2000) and Campbell, Lo, and MacKinlay (1997).

In an event study, the stock market reactions on the public announcement of specific information affecting a firm are investigated. According to the efficient market hypothesis (Fama, Fisher, Jensen, & Roll, 1969), the semi-strong form of which is underlying an event study, the market reacts to the announcement of new information. Basically all publicly available information goes into the stock price of firms. An event is anything that results in new relevant information which may have an impact on the future cash flows of a firm (McWilliams and Siegel 1997). Thus, when software or technology companies announce the release of source code to the public the information of releasing code can be expected to be included in the stock price shortly after the announcement (Dann, Mayers, Rafts, & Jr, 1977; Mitchell & Netter, 1989).

An event study works with the daily returns of stocks in a time series. There are two time windows in this time series (see Figure 1). The estimation window W1 is the time slot prior to

the event where the typical return of the stock is calculated. The event window W2 has an anticipation component prior to the event and a market reaction component after the event.

Insert Figure 1 about here

Data Collection

Event definition and selection. In this paper, I define an event as follows.

> *An event is the announcement of the release of proprietary source code—that could have been sold (or has been)—under an OSI compliant license either to an existing public open source project or by setting up a new public OSS project.*

Using Lexis Nexis, I searched the PR-Newswire, Business Wire, and Market Wire database using the search term "open source AND (contribute OR release OR reveal) AND code" from January 1999 to April 2007. After checking whether the company that released code was listed on the AMEX, NYSE, or NASDAQ and whether the event fit to the event definition, 111 events by 58 distinctive firms were identified. As expected, descriptive statistics of the identified events show a curvilinear trend in the number of events over time (see Figure 2).

Insert Figure 2 about here

It is important to note that the total number of release announcements, that is, taking both publicly listed and non-listed firms, for both 2005 and 2006 was higher than that of 2000, implying that OSS activity of those years has surpassed that before the burst of the dot.com bubble. However, during the search, I found that a large number of firms engaging in OSS are not listed on a stock exchange, but rather small, privately-owned companies. Furthermore, larger,

more renowned OSS firms, while sometimes already VC-backed, are also often still far from an IPO stage and some of those few close to an IPO may well get acquired before they are ever publicly traded.

Estimation window, event window and check for confounding events. After creating a sample of events the time windows had to be specified. The length of the windows was strongly orientated on the window length found in the related literature (Brown & Warner, 1985; Campbell et al., 1997; MacKinlay, 1997; McWilliams et al., 1997). The estimation window was defined as 125 trading days and the event window as two days including the event day and the day prior to the event. The inclusion of the day prior to the event was to take anticipation effects into account. I then checked for confounding events within this period of time. A confounding event was defined as an announcement within the event window that might overshadow the effect of the actual event on the stock price of the company. Confounding events were for example the announcement of new products, information about pending lawsuits, or the release of quarterly or annual reports.

The check for confounding events eliminated 69 of the 111 events. A possible reason for the mass of confounding events may be that many software companies tend to publish information on new products, strategic partnerships, etc. in bundles on conferences and other mass events.[5] In addition, four more events had to be removed from the sample; three events because the IPO of the respective firms had only happened recently, so that the available stock price data would not have sufficed for the 125-day estimation window. Another event was removed because the respective firm was considered for delisting at the time of the event. A list of all remaining 38 events by 30 firms with no confounding events is given in Table 2. As is

[5] An example of such a conference is the JavaOne conference organized by Sun Microsystems Inc. (SUN, 2007).

depicted in Figure 2, the curvilinear trend in the number of releases per year still holds for the final dataset. More detailed event descriptions can be found in Table A. 1 in the appendix.

Insert Table 2 about here

Calculation of abnormal returns. Abnormal returns are returns achieved by one stock that are significantly higher or lower than the returns of the market. To calculate them, first the expected return for each event i on day t is approximated and then the difference between the real returns of the market and the expected returns R_{it} leads to abnormal returns. The return of the market is estimated using the market model.[6] The model shown in equation 1 includes α_i as intercept, β_i is the slope parameter and ε_i is the error term which is expected to be zero in case of no event specific influence. The variable R_{mt} is the return on a comparable market on day t (McWilliams et al., 1997).

$$R_{it} = \alpha_i + \beta_i R_{mt} + \varepsilon_{it} \quad \textbf{(1)}$$

From the market model the abnormal return AR_{it} of event i is calculated. Equation 2 shows that AR_{it} is the difference between the real day-to-day return R_{it} calculated from actual stock prices and the expected return $a_i + b_i R_{mt}$ based on the market model prediction from equation 1 (MacKinlay, 1997). The parameters a_i and b_i are calculated with an ordinary least squares (OLS) regression of R_{it} on R_{mt} over the estimation window. As is done in most IT-related event studies, I used the NASDAQ Composite Index as comparable market index. The time series data of the NASDAQ Composite Index and the securities were taken from Thompson Financial Datastream. It has to be noted that the time series data from Thompson Financial

[6] A detailed description of the market model can be found in the paper of MacKinlay (1997). Binder (1998) shows that it has advantages compared to the capital asset pricing model, the mean adjusted returns model, and the market adjusted returns model. According to both Strong (1992) and Park (2004) the market model is the most popular model used in event studies.

Datastream excludes all weekends but not the official holidays and other non-trading days (e.g. the days after September 11, 2001). These days were excluded manually from the dataset.

$$AR_{it} = R_{it} - \left(a_i + b_i R_{mt}\right) \quad \textbf{(2)}$$

To get evidence of the impact that a specific event has on the stock price the abnormal returns in the event window are aggregated. They can be aggregated over time and across several events (MacKinlay, 1997). In case the aggregation through time is carried out the abnormal returns AR_{it} for a single security are aggregated across the event window. In equation 3, CAR_i is the cumulated abnormal return for a specific security and T_2 the first and T_4 the last day of the event window. This variable also was the dependent variable in the multivariate analysis.[7]

$$CAR_i = \sum_{t=T_2}^{T_4} AR_{it} \quad \textbf{(3)}$$

For an aggregation across events the average return for each day t in the event window is calculated. $\overline{AR_t}$ is the average abnormal return for one day t in the event window across all events N.

$$\overline{AR_t} = \frac{1}{N}\sum_{i=1}^{N} AR_{it} \quad \textbf{(4)}$$

The average cumulated abnormal return $\overline{CAR}$ for all events N shown in equation 5 is derived from equations 3 and 4.

$$\overline{CAR} = \sum_{t=T_2}^{T_4} \overline{AR_t} = \frac{1}{N}\sum_{i=1}^{N} CAR_i \quad \textbf{(5)}$$

Dependent variables for multivariate analysis. In order to measure the effect of time, I calculated the number of days that had passed since the press conference on April 8, 1998

[7] As I am only using a two-day event window in this study, this implies that T_3 and T_4 were in fact the same day (see also Figure 1). For the sake of consistency, I will nevertheless refer to the end of the event window as T_4.

(Tiemann, 2006) in which "open source" was coined and the day of the event. For ease of interpretation, I divided this figure by 365.25 to arrive at the number of years.

The business model was coded based on the categories introduced before. In the press releases, I looked for information which of the above business models the firm was primarily intending to follow. In order to ensure reliability of this variable, multiple coders completed such a categorization independently of one another. Differences were resolved in discussions afterwards, and the final categorization was unanimously accepted.

Control variables for multivariate analysis. Henkel (2006) found that small firms tend to release more code since they lack development resources and benefit from external development support. In addition, I included the firm's R&D-to-sales and sales-per-employee ratio to account for research intensity of the firm and employee productivity. Sales per employee were transformed to sales per 1000 employees to ease interpretability of the results.

Descriptive statistics on both dependent and control variables can be found in Table 3, correlations are given in Table 4.

Insert Table 3 and Table 4 about here

RESULTS

In the multivariate analysis, I take a look at factors influencing the CAR for individual events. Conducting a regression analysis with CAR_i as dependent variable using the independent and control variables introduced before, I arrive at a model with an R^2 of 44% (see Table 5). Regarding H1, I compare the use of the business model "competitive weapon" with the other three, that is, I used competitive weapon as a reference group and include a dummy variable

measuring whether this business model was chosen (value = 1) or not (value = 0). I find that the business model competitive weapon is indeed valued more negatively on a level of significance ($p\text{-value}_{\text{one-sided}} = 0.03$). As both a univariate (see Table 6) and a multivariate analysis show this effect, H1 is, consequently, also fully confirmed.

With respect to the second hypothesis, I find that both measures for time—years since the inception and OSS and years squared—carry the expected sign and are highly significant ($p_{\text{one-sided}} < 0.01$ for both) resulting in the predicted curvilinear trend. Consequently, the first hypothesis H2 is fully confirmed.

Insert Table 5 and Table 6 about here

To control for possible autocorrelation effects between, that is, events originating from the same firm, in the regression, I reduced the sample to include only one event by one firm (N = 30) by using only the first or only the last event by the respective firms in two separate regression. In both analyses, both the effect of time and the business model remain significant (first event: $p_{\text{time}} = 0.02$, $p_{\text{time-squared}} = 0.02$, $p_{\text{business model}} = 0.07$; last event: $p_{\text{time}} = 0.02$ $p_{\text{time-squared}} = 0.02$, $p_{\text{business model}} < 0.01$; all p-values from one-sided tests). Similar tests where conducted for events having happened shortly after one another only to find identical results.

Seeing the curvilinear trend confirmed and combining this finding with the descriptive statistics, it would come at no surprise if, over all events, I cannot find an average CAR significantly different from zero. As this result has been found in many an event study on the impact of IT-related announcements on the market value of firms, I also performed such a check of significance of the average CAR in this study. The non-parametric rank test introduced by Corrado (1989) has been reviewed to be the most powerful test in the scientific literature on

event studies (Campbell & Wasley, 1993).[8] In short, the test statistic of the Corrado test is the ratio of the mean deviation of the event day ranks to the estimated standard deviation of the mean abnormal rank over all events (Campbell et al., 1993; Corrado, 1989). For the sample, the Corrado test results in a test statistic of -0.51 ($p = 0.61$), clearly indicating that, on average, the firms in the sample did not achieve abnormal returns significantly different from zero.

DISCUSSION AND IMPLICATIONS

In this study, I have shown two drivers influencing the reaction of the capital market on firms announcing the release of source code to (potential) products as OSS, namely the time of the announcement and the underlying business model.

Concerning time, I find that there is a curvilinear trend that is most likely explained by investor sentiment. Initially, that is, during the dot.com hype, I see a positive valuation of OSS by the capital market that, however, turns negative with the burst of the bubble. In fact, if I use the regression coefficients from Table 5 and insert mean values for all other variables but time (see Figure 3), the date value of the first root of the resulting quadratic equation turns out to be January 10, 2001, which is right after the dot.com crash. The second root is August 13, 2005, coming shortly before the NASDAQ price increased to and then stabilized around 2,300 points—the highest value since 2001—and during a series of source code releases in 2005. Moreover, I see a nearly identically shaped trend for the number of announcements per years, which could both indicate deliberate management action or learning about OSS and its positive effects over time. Of course, I do not expect this evaluation to become infinitively more positive but rather to

[8] Typically, the parametric t-test is used for hypothesis testing in almost every event study. An important assumption underlying the t-test is that of a normal distribution of the abnormal returns. However, as the latter may in fact show *any* kind of distribution, rank tests, which do not assume any underlying distribution—thus being non-parametric—should preferably be employed for statistical significance checks in event studies (MacKinlay, 1997).

level off at zero—at which point in time the release of source code under an OSS license will have truly become mainstream firm behavior.

Insert Figure 3 about here

I think that the curvilinear relationship observed in this study may well hold for more segments of the IT sector and related industries. Henkel and Käs (2007) have for example observed that the number of firms releasing source code in several areas of embedded computing such as single-board computers has been steadily increasing over the last years. I would assume market reaction to the respective announcements to follow a similar u-shape.

What is more, I think that this study has generally illustrated the effect that investor sentiment towards a certain action has on its valuation on the market and how investor sentiment and management reaction to it can change over time. First, during the time of the dot.com bubble, OSS—as probably any other Internet-related IT investment—was heartily embraced by the capital market and firms announcing an OSS strategy saw their stock price increase abnormally. Managers observing and understanding this would consequently think about if and how their firms could also release product source code as OSS. With the bubble burst and investor sentiment turning negative, both the valuation and the number of OSS strategies decreased. When the stock market, especially the NASDAQ, stabilized and began to again steadily increase in value, firms making OSS-related announcements could again achieve positive abnormal returns and, again, the number of firms doing so consequently increased. I would thus expect studies analyzing the development of stock prices or IPO performance in the IT sector and maybe also related industries such as telecommunications to find an underlying curvilinear trend over time with respect to market performance caused by investor sentiment on

the one hand and, on the other hand, with respect to the number of announcements caused by managers' deliberate reaction to the perceived investor sentiment. Maybe such a curvilinear trend can also be found in the datasets that have been analyzed in previous event studies on the effect of IT investments on the market value of the firm.

The second factor influencing market valuation of firms deciding to go OSS I have analyzed is the business model the firm chooses in doing so. When looking at the business model the firm chooses, I find that firms that use OSS merely as a competitive weapon see a significantly worse investor reaction than those firms following one of the other three business models. I maintain that the reason behind this negative market reaction is that the business models cost or risk reduction, dual licensing, and sale of complementary goods or services are usually introduced together with a clear revenue model, which enables the company to appropriate value from its OSS engagement, and that the capital market values this more highly than the usually uncertain long-term benefits of the competitive weapon model. The correlation table also hints in the direction that R&D-intensive firms tend to rather use the competitive weapon model, whereas firms with lower R&D-to-sales ratio rather choose the sale of complementary goods or services. This could again be an indication for the more long-term nature of the competitive weapon model as opposed to the other three. Should the competitive weapon business model, however, turn out to have positive effects over the long run as I have argued before, this could be another indication of myopic investor behavior. An example for this could be IBM's release of Eclipse that was evaluated negatively by the capital market at the time of the announcement ($CAR_{Event\ 17}$ = -1.36%) yet is widely considered a success for IBM nowadays.

Again considering the idea that the business model competitive weapon has long term positive effects, I controlled for an improvement of the market evaluation of firms choosing this

business model over time, with the underlying assumption that the market will come to understand the rational behind and the positive effects of the initial decision, and consequently react more positively to similar future decisions. For this, I included an interaction term between the business model competitive weapon and time in the regression (see Table 7). I find that, while all other factors keep or improve their level of significance, the interaction term is positive and significant ($p_{one\text{-}sided} = 0.05$), indicating that the market, over time, has come to better appreciate this business model, too. Another possible explanation might be that firms have come to understand that the competitive weapon model alone is not sustainable—or at least not accordingly valuated by the capital market—and over time learned to combine it effectively with one of the other models.

Insert Table 7 about here

I think that these findings can valuably contribute to the research on business models as they highlight that the capital market is able to distinguish between differences in the choice of business model and the consequences on value appropriation by the firm. Business models that include a clearly articulated and comprehensible—to the capital market, that is—revenue model indicating short-term appropriability of generated value are viewed more positively by investors than business models that only promise uncertain positive effects in the long run—at least until the uncertainty has been reduced, that is. The findings also show that value may well be created by firms who decide to open their knowledge in the form of OSS—under the condition that they have a valuable business model for this. This line of research might be extended by further analyzing the differences between the other three revenue models—dual licensing, cost or risk reduction, and the sale of complementary goods and services, or other business models—where I

would for example expect mere “outsourcing” deals falling in the cost or risk reduction category to be valuated less positively by the capital market than more innovative or transformative IT investments especially inherent in the sale of complementary goods and services category (Dehning et al., 2003; Dos Santos et al., 1993).

Like any other event study this paper faces some limitations. One limitation may be the rather small sample size. Limiting the event window to one day so that the confounding check eliminates fewer events might seem as a solution to this problem, however, as I expect rather strong anticipation effects on the day preceding the event, this does not seem to be a wise choice. Rather, one might think about expanding the search terms and applying those to more and different data sources as the ones used in the study, or recomposing the study in a couple of years: since the number of qualified events has been steadily increasing over the past view years, redoing the study in a few years time should produce a much larger sample size. In such a study, learning effects of firms over time might be analyzed, too, that is, such studies could look at whether the success or failure in previous efforts to release source code under an OSS license has an impact on the market valuation of another such announcement.

Moreover, event studies only take events for public companies into account. Thus, privately held companies are not in the sample. In this study in particular, a lot of events from private companies were excluded. Future research efforts might look at venture capital or corporate venture capital investments in such firms, OSS-based firm acquisitions by publicly-traded companies, or, potentially, at the IPO performance of OSS-based firms, since I expect all of these actions to rise in number in the future.

This study has emphasized the need for a view on the capital market in OSS and open standards research. Both have long become relevant for business world—this study has shown that there is well reason to believe that this relevance will further increase in the future. Whereas

the capital market had a negative impression of firms deciding to engage in OSS for most of the past years, as I have shown, this valuation has been gradually improving recently. Managers of proprietary software firms thinking about releasing some of their software as OSS thus need not fear a negative reaction of the capital market—that is, if they have carefully selected a business that allows them to both create *and* appropriate value with their OSS efforts. While it is obvious that OSS is not the right choice for every firm in the IT industry, the results clearly indicate that its overall importance will most likely increase further over the coming years. For many firms, thus, the interesting question will no longer be *whether* to actively engage in OSS, but rather *how*, that is, the processes firms need to follow and implement when transforming from closed to open source.

FIGURES & TABLES

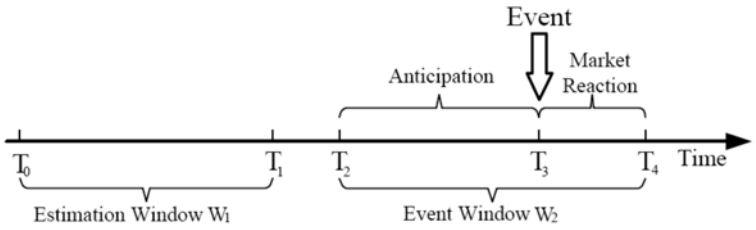

Figure 1: Time windows in event studies (based on MacKinlay (1997))

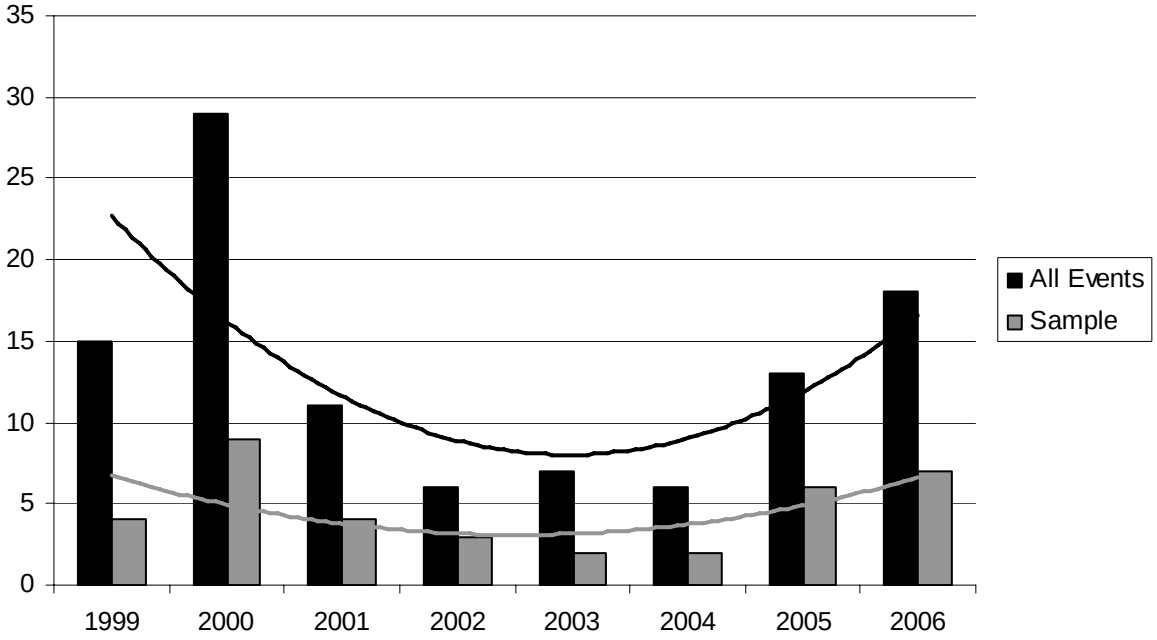

Figure 2: Distribution of events over time

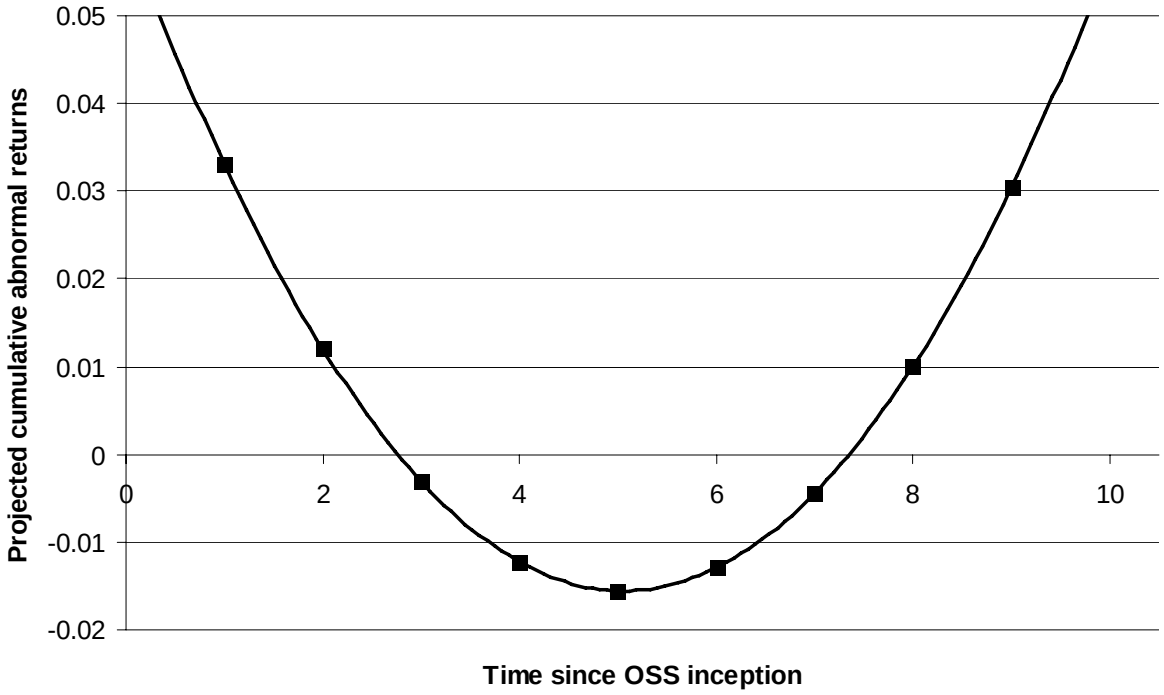

Figure 3: Projected CARs using only time and time squared

Business Model	**Potential goals**	**Examples**	**Revenue source**
Competitive weapon	- preventing a competitor from establishing or keeping a dominant standard ("preventing a choke hold") - tipping the standard race in favor of oneself - commoditizing a layer of the software stack that is of little to no value to the company - advertising oneself as software-developing company on the job market	- Netscape releases source code of Navigator to prevent Microsoft from locking up HTTP and HTML (Raymond, 2001a) - IBM releases Eclipse to replace Sun's or Microsoft's "native" software development products with its own standard cross-development framework (Koenig, 2004) - IBM's support of Linux provided a common set of APIs across IBM's entire product line, it changed the area of competition so that IBM was able to show its traditional strengths in services, availability, and reliability (West, 2003) - Dresdner Kleinwort Wasserstein releases openadaptor in January 2001 because, in late 2000, "it was very difficult to employ competent developers, because everybody was interested in joining start-ups and getting their share options." Making openadaptor an OSS project "acted as a kind	No direct source of income. Benefits will only become visible in the long run.

		of advert […] we are a bank but we do really cool stuff" (Henkel, 2004).	
Cost or risk reduction	- OSS is cheaper than doing proprietary software development for components that provide basic functionality and little to no sales value, yet are highly critical - Software is not related to the core business, yet further existence of the software is necessary. Releasing the software as OSS helps by reducing the developers' amount of work on the non-business-relevant program and by securing a continuous stream of actualizations even if they have turned to other projects or left the company	- IBM replaces its own web server development with Apache (Hecker, 1999; Raymond, 2001b) - Two employees at Cisco had devised a clever solution to printer selection and management in the late 1990s. Cisco had no intention of ever selling the software, so they released it into open source. By 2005, one of the two developers had already left the company (Goldman et al., 2005; Henkel, 2004; Raymond, 2001a)	Usually, no source of income, but potential for cost reduction. Short-term to mid-term.
Dual licensing Precondition: firms owns 100% of copyrights[9]	- Company offers different licenses of the product to different customer groups, for example a free version to individuals and a commercial version under a company-friendly license to firms	- Both MySQL offers its database for free to individual developers and offers companies the possibility to purchase commercial licenses that allow for easy integration even into proprietary software products (Goldman et al., 2005; Raymond, 2001b)	Licensing revenue. Short-term to mid-term.

[9] Releasing software as open source does not mean giving away copyright. Under any OSS license, too, authors of a piece of code still have the copyrights to the parts they wrote.

Sale of comple-mentary goods or services	- Services such as consulting, implementation, training, and subscription-based models, as enterprise customers that choose OSS are in most cases also looking for services around the offer (most easily applicable model) - Goods such as software, hardware, documentation, books, or gadgets. In case of releasing OSS related to a certain hardware product, the software itself is usually not a profit center, for example drivers - Adapted model: instead of giving away an entire software product, the company keeps a proprietary core containing the most important functionalities of the software. The company then releases that part of the source code that either enables or shows interaction with the proprietary core, that is, how to best make use of it. To make using the core even easier, the company might think about releasing a software development kit (SDK)	- Most entirely OSS-based firms run this business model. OSS also allows start-ups to overcome entry barriers and liabilities of newness and smallness and they are typically faced with (Gruber & Henkel, 2006) - Apple open sourced the core of its new operating system Darwin; Creative is also providing support to the OSS developers trying to make Creative hardware work with Linux (Hecker, 1999; Raymond, 2001a). - Valve Inc. had developed a superior graphical engine for the game Half-Life. By revealing much of the source code of its game but keeping the engine proprietary, people were able to develop add-ons, so called mods, to Half-Life, but running these mods still required the user to own a copy of Half-Life. When the user-innovated mod Counter-Strike was introduced to the market, it immediately took off and became an enormous hit.(Jeppesen & Molin, 2003)	Revenue stream from sale of complementary good or services. Short-term to mid-term.

Table 1: Business models in OSS

ID	Firm	Date	CAR_i	Main OSS business model as stated
1	3DFX INTERACTIVE	1999-03-01	0.86%	Comp. goods/services
2	APPLIX	1999-03-02	10.86%	Comp. goods/services
3	SILICON GRAPHICS	1999-04-26	4.18%	Comp. goods/services
4	VISIO	1999-07-27	5.32%	Dual licensing
5	SILICON GRAPHICS	2000-02-14	-0.57%	Comp. goods/services
6	BINDVIEW DEV	2000-02-15	-0.84%	Competitive weapon
7	SUN MICROSYSTEMS	2000-03-13	-3.92%	Comp. goods/services
8	INTEL	2000-06-15	-3.27%	Comp. goods/services
9	SYBASE	2000-08-22	-1.80%	Cost/risk reduction
10	CADENCE DESIGN SYS.	2000-09-11	12.39%	Comp. goods/services
11	SAP	2000-10-04	-3.79%	Comp. goods/services
12	SANCHEZ COMPUTER ASSOCS.	2000-11-06	4.97%	Comp. goods/services
13	PROGRESS SOFTWARE	2000-12-11	0.02%	Comp. goods/services
14	ADAPTEC	2001-01-30	1.05%	Comp. goods/services
15	SUN MICROSYSTEMS	2001-04-25	-8.38%	Competitive weapon
16	ON2 TECHS.	2001-08-07	1.08%	Competitive weapon
17	IBM	2001-11-05	-1.36%	Competitive weapon
18	OPENWAVE SYS.	2002-05-30	-3.01%	Comp. goods/services
19	ORACLE	2002-08-14	1.12%	Comp. goods/services
20	APPLE	2002-09-25	-2.25%	Competitive weapon
21	COMMERCE ONE	2003-04-29	-7.57%	Competitive weapon
22	REALNETWORKS	2003-07-07	-0.48%	Dual licensing
23	BEA SYSTEMS	2004-05-19	1.26%	Comp. goods/services
24	TIPPINGPOINT TECHS.	2004-11-01	11.50%	Comp. goods/services
25	IONA TECHNOLOGIES	2005-06-20	-1.09%	Competitive weapon
26	EBAY	2005-06-21	-2.51%	Comp. goods/services
27	QUOVADX	2005-07-19	-2.97%	Competitive weapon
28	ORACLE	2005-08-09	0.88%	Comp. goods/services
29	IBM	2005-08-09	0.61%	Comp. goods/services
30	IBM	2005-08-15	0.42%	Competitive weapon
31	AUTODESK	2006-03-07	-1.04%	Comp. goods/services
32	WIND RIVER SYSTEMS	2006-07-31	1.46%	Comp. goods/services
33	SUN MICROSYSTEMS	2006-08-23	3.21%	Competitive weapon
34	TIBCO SOFTWARE	2006-10-02	0.67%	Dual licensing
35	QUALCOMM	2006-10-11	4.92%	Cost/risk reduction
36	ADOBE SYSTEMS	2006-11-07	2.75%	Competitive weapon
37	SUN MICROSYSTEMS	2006-11-13	-2.09%	Competitive weapon
38	ADOBE SYSTEMS	2007-04-26	0.07%	Dual licensing

Table 2: List of all events

	Observations	Median	Mean	Std. dev.	Min	Max
CAR_i	38	0.00	0.01	0.04	-0.08	0.12
Time	38	4.41	4.86	2.74	0.90	9.05
$Time^2$	38	19.43	30.96	27.78	0.80	81.88
ln(Total Assets)	38	14.40	14.37	2.58	8.28	18.46
$ln(Total\ Assets)^2$	38	207.26	212.91	72.38	68.49	340.74
R&D-to-sales	38	0.16	0.22	0.28	0.06	1.74
Sales per 1000 employees	38	0.28	2.30	12.48	0.00	77.21
Competitive Weapon	38	0.00	0.32		0	1
Cost/Risk reduction	38	0.00	0.05		0	1
Dual licensing	38	0.00	0.11		0	1
Compl. goods or services	38	1.00	0.53		0	1

Table 3: Descriptive statistics

	CAR_i	Time	$Time^2$	ln(Total assets)	$ln(Total\ assets)^2$	R&D-to-sales	Sales per 10000 employees	Competitive weapon	Cost/Risk reduction	Dual licensing	Compl. goods or services
CAR_i	1										
Time	-0.05	1									
$Time^2$	0.01	**0.98****	1								
ln (Total assets)	-0.12	0.20	0.18	1							
ln $(Total\ assets)^2$	-0.12	0.21	0.19	**0.99****	1						
R&D-to-sales	-0.04	-0.07	-0.10	**-0.56****	**-0.51****	1					
Sales per 1000 employees	**0.41***	0.10	0.07	0.22	0.23	0.02	1				
Competitive weapon	**-0.34***	0.22	0.18	-0.04	0.00	**0.28†**	-0.11	1			
Cost/Risk reduction	0.05	0.05	0.07	-0.25	-0.24	-0.04	-0.04	-0.16	1		
Dual licensing	0.06	0.15	0.19	-0.08	-0.10	-0.03	-0.06	-0.23	-0.08	1	
Compl. goods or services	0.26	**-0.32†**	**-0.31†**	0.19	0.17	-0.22	0.16	**-0.72****	-0.25	**-0.36***	1

Table 4: Correlation table

† $p < .10$

* $p < .05$

** $p < .01$ (p-values are two-sided)

Independent Variable	Coefficient value	Robust standard error
Time	**-0.030***	(0.011)
$Time^2$	**0.003***	(0.001)
ln(Total assets)	-0.040	(0.027)
$ln(Total\ assets)^2$	0.001	(0.001)
R&D-to-sales	-0.029	(0.025)
Sales per 1000 employees	**0.002****	(0.000)
Business model: comp. weapon	**-0.025†**	(0.012)
Constant	**0.379†**	(0.199)

Observations	38
R^2	0.438
F-statistic	**77.847****
Degrees of freedom	31

Table 5: Results of OLS regression on CAR using business model "competitive weapon" as reference group
† $p < .10$
* $p < .05$
** $p < .01$ (p-values are two-sided)

Business Model	Obs.	Mean	Std. Err.	Std. Dev.
Competitive Weapon	12	-0.016	0.010	0.035
Other	26	0.016	0.009	0.045
T-test	**0.018****	(p-value, one-sided)		

Table 6: Effect of business model choice (univariate analysis)

Independent Variable	Coefficient value	Robust standard error
Time	**-0.027***	(0.011)
$Time^2$	**0.002***	(0.001)
ln(Total assets)	**-0.046†**	(0.025)
$ln(Total\ assets)^2$	0.001	(0.001)
R&D-to-sales	-0.024	(0.025)
Sales per 1000 employees	**0.002****	(0.000)
Business model: comp. weapon	**-0.069***	(0.030)
Time x competitive weapon	**0.007†**	(0.004)
Constant	**0.419***	(0.186)

Observations	38
R^2	0.469
F-statistic	**129.900****
Degrees of freedom	30

Table 7: Results of OLS regression on CAR including the interaction term time x "competitive weapon"
† $p < .10$
* $p < .05$
** $p < .01$ (p-values are two-sided)

APPENDIX

ID	Event (Title of Release Announcement)
1	3Dfx Interactive Enables Cross-Platform Development Through Linux Community
2	Applix Launches Open Source Initiative With Applix SHELF; Embeddable, Graphical Programming Language Now Available for Applixware
3	SGI Contributes Key Storage Area Network Media Management Software to Open Source Community
4	Visio Announces Source Code Collaboration Initiative for IntelliCAD Technology
5	SGI Demonstrates High-Performance Itanium Processor Compilers at Intel Forum; Technology to Enhance Linux Application Performance to Be Released to Open Source Community
6	RAZOR Team Offers Free Utility to Put Zombies Back to Sleep
7	Sun Microsystems Announces Plans to Release Cross-Platform Java(TM) Development Tool Code to Open Source Community
8	Intel Software to Make Linux-Based Internet Access and Home Networking Devices Easier to Use
9	Sybase to Open Source Watcom C/C++ and Fortran Compilers
10	Cadence Offers Testbench Authoring Technology as Open Source Solution for Verification Challenges
11	SAP Drives Open-Source Database Development
12	Sanchez Offers GT.M Database as Open Source Freeware to GNU/Linux Users
13	Progress Software Announces Open Source Availability of Its Application Development Environment
14	Adaptec Embraces Open Source/Linux Community
15	Sun Unveils Project JXTA
16	On2 Technologies to Open Source VP3.2 Video Compression Technology; Full-Screen, High-Quality Video Technology Available to Software Community
17	IBM Donates $40 Million of Software to Open Source Community
18	Openwave Contributes Open Usability Interface to Open Source Community
19	Oracle Announces New Clustered File System for Linux
20	Apple 'Open Sources' Rendezvous
21	Commerce One Releases Open Source, Royalty Free DocSOAP XML Developer Kit for Document Style SOAP
22	RealNetworks Releases Industry Standard SMIL Source Code to Helix Community
23	BEA Plans Open Source Project to Accelerate Java Adoption and Provide Universal Framework for Enterprise Java Applications
24	TippingPoint Releases Open Source Code for First Intrusion Prevention Test Tool, Tomahawk
25	IONA to Introduce Open Source Java Enterprise Service Bus
26	eBay Introduces Community Codebase for Open Source Developers
27	Rogue Wave Software Donates Programming Code for Enterprise Software Development to Open Source Community
28	Oracle's Industry Leading Cluster File System Endorsed by Linux Community
29	IBM Unveils Services and Contributes Management Console Code for Apache Software Foundation Geronimo Project
30	IBM Contributes Open Source Code to Make FireFox Browser More Accessible
31	Autodesk Announces Availability of MapGuide Open Source Web Mapping Software Through the Open Source Geospatial Foundation
32	Wind River Contributes Over 300,000 Lines of Code to the Eclipse Foundation
33	Sun Open Sources Java Mobile Edition Development Tool
34	TIBCO to Open Source Best-Rated AJAX Rich Internet Application Toolkit
35	QUALCOMM Launches Project in Collaboration With Mozilla Foundation to Develop Open Source Version of Eudora Email Program
36	Adobe and Mozilla Foundation to Open Source Flash Player Scripting Engine
37	Sun Open Sources Java Platform and Releases Source Code Under GPL License Via NetBeans and Java.net Communities
38	Adobe to Open Source Flex

Table A. 1: Events

REFERENCES

Aggarwal, N., Dai, Q., & Walden, E. A. 2006. Do Markets Prefer Open or Proprietary Standards for XML Standardization? An Event Study. International Journal of Electronic Commerce, 11(1): 117-136.

Amit, R. & Zott, C. 2001. Value Creation in E-business. Strategic Management Journal, 22(6-7): 493-520.

Baldwin, C. Y. & Clark, K. B. 1997. Managing in the age of modularity. Harvard Business Review, 75(5): 84-93.

Binder, J. 1998. The Event Study Methodology since 1969. Review of Quantitative Finance and Accounting, 11(2): 111-137.

Bonaccorsi, A., Giannangeli, S., & Rossi, C. 2006. Entry Strategies under Competing Standards: Hybrid Business Models in the Open Source Software Industry. Management Science, 52(7): 1085-1098.

Brown, S. J. & Warner, J. B. 1985. Using Daily Stock Returns: The Case of Event Studies. Journal of Financial Economics, 14(1): 3-31.

Brynjolfsson, E. 1993. The Productivity Paradox of Information Technology. Communications of the ACM, 36(12): 66-77.

Campbell, C. J. & Wasley, C. E. 1993. Measuring Security Price Performance Using Daily NASDAQ Returns. Journal of Financial Economics, 33(1): 73-92.

Campbell, J. Y., Lo, A. W., & MacKinlay, A. C. 1997. The Econometrics of Financial Markets. Princeton, NJ: Princeton University Press.

Chesbrough, H. & Rosenbloom, R. S. 2002. The Role of the Business Model in Capturing Value from Innovation: Evidence from Xerox Corporation's Technology Spin-off Companies. Industrial and Corporate Change, 11(3): 529-555.

Chesbrough, H. W. 2003. Open Innovation: The New Imperative for Creating and Profiting from Technology. Boston: Harvard Business School Press.

Cooper, M. J., Dimitrov, O., & Rau, P. R. 2001. A Rose.com by Any Other Name. Journal of Finance, 56(6): 2371-2388.

Cooper, M. J., Khorana, A., Osobov, I., Patel, A., & Rau, P. R. 2005. Managerial Actions in Response to a Market Downturn: Valuation Effects of Name Changes in the Dot.com Decline. Journal of Corporate Finance, 11(1-2): 319-335.

Corrado, C. J. 1989. A Nonparametric Test for Abnormal Security-price Performance in Event Studies. Journal of Financial Economics, 23(2): 385-395.

Dahlander, L. 2005. Appropriation and Appropriability in Open Source Software. International Journal of Innovation Management, 9(3): 259-285.

Dalle, J.-M. & Jullien, N. 2003. 'Libre' Software: Turning Fads into Institutions? Research Policy, 32(1): 1–11.

Dann, L. Y., Mayers, D., Rafts, R. J., & Jr. 1977. Trading Rules, Large Blocks and the Speed of Price Adjustment. Journal of Financial Economics, 4(1): 3-22.

De Long, J. B., Shleifer, A., Summers, L. H., & Waldmann, R. J. 1990. Noise Trader Risk in Financial Markets. Journal of Political Economy, 98(4): 703-738.

Dehning, B. & Richardson, V. J. 2002. Returns on Investments in Information Technology: A Research Synthesis. Journal of Information Systems, 16(1): 7-30.

Dehning, B., Richardson, V. J., & Zmud, R. W. 2003. The Value Relevance of Announcements of Transformational Information Technology Investments. MIS Quarterly, 27(4): 637-656.

Dehning, B., Richardson, V. J., Urbaczewski, A., & Wells, J. D. 2004. Reexamining the Value Relevance of E-Commerce Initiatives. Journal of Management Information Systems, 21(1): 55-82.

Dos Santos, B. L., Peffers, K., & Mauer, D. C. 1993. The Impact of Information Technology Investment Announcements on the Market Value of the Firm. Information Systems Research, 4(1): 1-23.

Fama, E. F., Fisher, L., Jensen, M. C., & Roll, R. 1969. The Adjustment of Stock Prices to new Information. International Economic Review, 10(1): 1.

Farrell, J. & Saloner, G. 1985. Standardization, Compatibility, and Innovation. RAND Journal of Economics, 16(1): 70-83.

Farrell, J. & Gallini, N. T. 1988. Second-sourcing as a Commitment: Monopoly Incentives to Attract Competition. Quarterly Journal of Economics, 103(4): 673-694.

FSF; Frequently Asked Questions about the GNU GPL; http://www.gnu.org/licenses/gpl-faq.html; May 10, 2006.

Goldman, R. & Gabriel, R. P. 2005. Open Source as Business Strategy: Innovation Happens Elsewhere. San Francisco, CA: Morgan Kaufmann.

Gruber, M. & Henkel, J. 2006. New Ventures Based on Open Innovation – an Empirical Analysis of Start-Up Firms in Embedded Linux. International Journal of Technology Management, 33(4): 356-372.

Harhoff, D., Henkel, J., & von Hippel, E. 2003. Profiting from Voluntary Information Spillovers: How Users Benefit by Freely Revealing their Innovations. Research Policy, 32(10): 1753-1769.

Hecker, F. 1999. Setting Up Shop: The Business of Open-Source Software. IEEE Software, 16(1): 45-51.

Henkel, J. 2004. Open Source Software from Commercial Firms – Tools, Complements, and Collective Invention. ZfB-Ergänzungsheft, 2004(4).

Henkel, J. & von Hippel, E. 2005. Welfare Implications of User Innovation. Journal of Technology Transfer, 30(1/2): 73-87.

Henkel, J. 2006. Selective Revealing in Open Innovation Processes: The Case of Embedded Linux. Research Policy, 35(7): 953-969.

Henkel, J. & Käs, S. 2007. Source Code Urgently Needed! How the Demand for Openness Redefines Competition, 5th International Workshop on User Innovation. Copenhagen Business School, Copenhagen, Denmark.

Im, K. S., Dow, K. E., & Grover, V. 2001. Research Report: A Reexamination of IT Investment and the Market Value of the Firm--An Event Study Methodology. Information Systems Research, 12(1): 103-117.

Jeppesen, L. B. & Molin, M. J. 2003. Consumers as Co-developers: Learning and Innovation Outside the Firm. Technology Analysis & Strategic Management, 15(3): 363-383.

Katz, M. L. & Shapiro, C. 1985. On the Licensing of Innovations. RAND Journal of Economics, 16(4): 504-520.

Katz, M. L. & Shapiro, C. 1986. Technology Adoption in the Presence of Network Externalities. Journal of Political Economy, 94(4): 822-841.

Koenig, J.; Seven Open Source Business Strategies for Competitive Advantage; http://management.itmanagersjournal.com/article.pl?sid=04/05/10/2052216; November 30, 2005.

Lakhani, K. & von Hippel, E. 2003. How Open Source Software Works: 'Free' User-to-User Assistance. Research Policy, 32(7): 923-943.

Lee, C. M. C., Shleifer, A., & Thaler, R. H. 1991. Investor Sentiment and the Closed-End Fund Puzzle. Journal of Finance, 46(1): 75-109.

Lee, P. M. 2001. What's in a Name.com?: The Effects of '.com' Name Changes on Stock Prices and Trading Activity. Strategic Management Journal, 22(8): 793-804.

Lerner, J. & Tirole, J. 2002. Some Simple Economics of Open Source. Journal of Industrial Economics, 50(2): 197-234.

MacKinlay, A. C. 1997. Event Studies in Economics and Finance. Journal of Economic Literature, 35(1): 13-39.

McWilliams, A. & Siegel, D. 1997. Event Studies in Management Research: Theoretical and Empirical Issues. Academy of Management Journal, 40(3): 626-657.

McWilliams, T. P. & McWilliams, V. B. 2000. Another Look at Theoretical and Empirical Issues In Event Study Methodology. Journal of Applied Business Research, 16(3): 1-11.

Mitchell, M. L. & Netter, J. M. 1989. Triggering the 1987 Stock Market Crash: Antitakeover Provisions in the Proposed House Ways and Means Tax Bill? Journal of Financial Economics, 24(1): 37-68.

Morrison, P. D., Roberts, J. H., & von Hippel, E. 2000. Determinants of User Innovation and Innovation Sharing in a Local Market. Management Science, 46(12): 1513-1527.

Oh, W., Gallivan, M. J., & Kim, J. W. 2006. The Market's Perception of the Transactional Risks of Information Technology Outsourcing Announcements. Journal of Management Information Systems, 22(4): 271-303.

OSI; The Open Source Definition (Version 1.9); http://www.opensource.org/docs/definition.php; May 10, 2006.

Park, N. K. 2004. A Guide to Using Event Study Methods in Multi-country Settings. Strategic Management Journal, 25(7): 655-668.

Raymond, E. S. 2001a. The Magic Cauldron. In E. S. Raymond (Ed.), The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, 2nd ed.: 113-191. Sebastopol: O'Reilly.

Raymond, E. S. 2001b. The Cathedral and the Bazaar. In E. S. Raymond (Ed.), The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, 2nd ed.: 19-63. Sebastopol: O'Reilly.

Richardson, J. E.; The Business Model: An Integrative Framework for Strategy Execution; http://ssrn.com/abstract=932998; September 12, 2007.

Shah, S. 2006. Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development. Management Science, 52(7): 1000-1014.

Shepard, A. 1987. Licensing to enhance demand for new technologies. RAND Journal of Economics, 18(3): 360-368.

Stewart, K. J. & Gosain, S. 2006. The Impact of Ideology on Effectiveness in Open Source Software Teams. MIS Quarterly, 30(2): 291-314.

Strong, N. 1992. Modelling Abnormal Returns: A Review Article. Journal of Business Finance & Accounting, 19(4): 533-553.

SUN; JavaOne Conference 2007 - OpenJDK: Now the Journey Starts for the Community; http://java.sun.com/javaone/sf/2007/articles/openjdk_sands.jsp; May 12, 2007.

Tiemann, M.; History of the OSI; http://www.opensource.org/history; August 20, 2007.

von Hippel, E. 1994. "Sticky Information" and the Locus of Problem Solving: Implications for Innovation. Management Science, 40(4): 429-439.

von Hippel, E. 1998. Economics of Product Development by Users: The Impact of "Sticky" Local Information. Management Science, 44(5): 629-644.

von Hippel, E. 2001. Innovation by User Communities: Learning from Open-Source Software. MIT Sloan Management Review, 42(4): 82-86.

von Hippel, E. & von Krogh, G. 2003. Open Source Software and the 'Private-Collective' Innovation Model: Issues for Organization Science. Organization Science, 14(2): 209-233.

West, J. 2003. How Open is Open Enough? Melding Proprietary and Open Source Platform Strategies. Research Policy, 32(7): 1259–1285.

West, J. & Gallagher, S. 2006. Challenges of Open Innovation: The Paradox of Firm Investment in Open Source Software. R&D Management, 36(3): 319-331.

Wheeler, D. A. 2002. Why Open Source Software / Free Software (OSS/FS)? Look at the Numbers!