Paper ID: W317

A Transaction Cost Perspective of Corporate Involvement in OSS

Mingzhi LI[1], Kai REIMERS[2]

[1] Tsinghua University, Beijing, 100084, P.R. China
Tel: +86 10 6277 1778, Fax: + 86 10 6278 5562 , Email: limzh@em.tsinghua.edu.cn

[2] RWTH Aachen University, Johanniterstr. 22-24, Aachen, 52064, Germany
Tel: +49 241 40923-22, Fax: + 49 241 40923-18, Email: reimers@wi.rwth-aachen.de

Abstract

In this paper we develop a transaction cost perspective of the computer industry evolution and standardization processes in order to evaluate the future of the open source software (OSS) movement, particularly with regard to how the involvement of firms--as opposed to individuals--will shape its direction. For this purpose we define two types of transaction costs related to coordinating the development of complementary products and services on the one hand and to communication of knowledge necessary for evaluating products incorporating new technology from a buyer's point of view on the other hand. We then apply these concepts to the analysis of the computer industry evolution and its standardization processes. Building on a discussion of the differences of OSS from other traditional standardization processes in terms of the two types of transaction costs, we depict two possible future scenarios for OSS and its impact on the whole computer industry. This allows us to identify the conditions under which OSS will not just provide an alternative standardization mechanism for the computer industry but a governance structure for organizing innovative activity as an alternative to the market.

Key Words:

Transaction Costs, OSS, Computer Industry, Industry Evolution, Technological Development

Topic Area:

IPR, INDUSTRIAL DYNAMICS & MARKETS FOR KNOWLEDGE

1. Introduction

The open source software (OSS) movement in the development of the computer industry can be regarded as a natural outcome of two related trends of IT industry development: the evolution of the computer industry into a stage of trying to break up the Wintel regime and a dissatisfaction of the IT community with the traditional standards setting processes (Takahashi and Namiki, 2003; Cargill and Bolin, 2004).

With the fashionable diffusion of OSS projects we see a trend of more corporate involvement which will have a huge impact on the future direction of OSS. Considering the grassroots character of OSS and its image as a brand-new model of user driven innovation in software development, the influence of firms as commercial organizations in the OSS movement clearly poses a dilemma for businesses and academics alike.

After an introduction of two threads of literature on the OSS movement--the historical perspective of the computer industry development and the perspective of OSS as another form of standard setting organization--we develop a transaction cost perspective of the future of the OSS movement. We argue that the success of OSS will lie in the effective addressing of the problems posed by the transaction costs of (1) conveying information about new technology incorporated into products to customers and (2) of coordinating the suppliers of system components which we term Type I and Type II transaction costs respectively. In proprietary standards, there is a trade-off between adoption and appropriability (West, 2003) which is related to our distinction between Type I and Type II transaction costs. We will attempt to evaluate how this trade-off will change in OSS. We also apply these concepts to the analysis of the pros and cons of different modes of governing the information technology industry standardization process: trade associations and standards developing organizations (SDOs); consortia and alliances; OSS. In the case of the computer industry, up to the Wintel era the second mode has been the dominant one with the POSIX case as an example of the first mode's infeasibility (Isaak, 2005). The most important distinguisher of OSS from the traditional proprietary software development mechanism is the high level of user participation which will have a determining impact on the two types of transaction costs and thus on the coordination of software development and the computer industry standardization process. Building on the differences of OSS from other traditional standardization processes in terms of the two types of transaction costs, we depict two future scenarios for OSS and its ultimate impact on the whole computer industry. In particular we are interested in how the involvement of commercial organizations will shape the direction of the OSS movement. Out of their own interest, deep involvement of firms in the OSS movement will be inevitable which might help create truly innovative business models and breakthroughs in technology. However, the potential of business interests' capturing of the OSS development poses a dilemma to the original intention of OSS advocates. It is also an interesting and open question whether corporate involvement symbolizes the come back of the SDO-mode as a computer industry standardization process, albeit with some unique variations from the traditional SDO-mechanism. Based on our transaction cost framework we will attempt to specify the conditions und which the one or the other scenario will become more likely.

The organization of the paper is as follows. In Section 2 we look at the standards and standardization process of the computer industry and try to put the OSS movement in a historical perspective. The major conclusion is that OSS is a natural outcome given the trend of industry development and the technology revolution in the network age. Section 3 extends the transaction cost framework of new product diffusion developed in Reimers and Li (2005) into the context of computer industry evolution and standardization processes. The fact that most OSS projects are mainly substitutes of existing proprietary software applications clearly demonstrates their limitation. In order to make OSS a user driven innovation model of computer industry development, the issues related with Type II transaction costs, i.e. the costs of attracting complementary products and services, need to be dealt with carefully. The other problem with OSS is that its early adopters are mostly technologically sophisticated. How to promote it to a broader audience lies in how to solve the problem posed by Type I transaction costs, i.e. the costs of communicating evaluation knowledge about new products to buyers. In Section 4 we depict two possible scenarios for the future of OSS according to the analysis of the two types of transaction costs presented in Section 3. Section 5 concludes the paper.

2. The OSS movement in the context of the computer industry evolution and standardization processes

There has been a rich literature on the evolution of the computer industry, such as Flamn (1988), Langlois (1990), Bresnahan and Greenstein (1999), Bresnahan and Malerba (1999), Malerba et. al. (1999).

According to Malerba et. al. (1999), the evolution of the computer industry divides rather naturally into four periods. The first began with the early experimentation with computers which culminated in designs sufficiently attractive to induce their purchase by large firms with massive computational tasks as well as by scientific laboratories. This opens the era of the mainframe computer. The second era began with the introduction of integrated circuits and the development of minicomputers. The third era is that of the personal computer, made possible by the invention of the microprocessor. We are now in the era of networked computers and the increasing use of the Internet.

Concurrent with the technological development was the vertical disintegration of the computer industry. The dominant mainframe era monopoly firm IBM not only designed and produced most of the critical components for its mainframes, but also wrote most of the basic software. For a time, IBM also designed and produced a significant fraction of the integrated circuits that were employed in its mainframe computers. In the minicomputer industry there was, from the beginning, more vertical specialization than in mainframe production, with most minicomputer companies buying their integrated circuits and a number of other key components on the open market. A striking characteristic of the firms producing personal computers is that they are primarily assemblers, buying most of their components on the open market. (Malerba et. al., 1999).

Bresnahan and Greenstein (1999) examine the computer industry evolution from the perspective of technological competition between computer “platforms”, such as IBM System/360 or the Apple Macintosh, which are defined as bundles of standard components around which buyers and sellers coordinate efforts. They demonstrate that young platforms serving newly founded segments eventually challenged established platforms across segment boundaries through a process of indirect entry. Vertically disintegrated platforms have led to divided technical leadership in important segments. In the case of mainframe computers, there was a single platform offered by a single firm with a high level of vertical integration (IBM). In personal computers, the IBM PC platform was controlled at first by a single firm, but later decentralization led to the “Wintel” platform controlled by Microsoft and Intel.

West and Dedrick (2001) extend the evolution of the computer industry into the network age and also put OSS in this context. They believe the success of the open source movement reflects a confluence of three factors in the mid-1990s: (1) users seeking an inexpensive Unix implementation free of AT&T licensing restrictions; (2) a philosophical movement rejecting the idea of software ownership and appropriability; (3) emergence of the Internet as both an enabler and objective for collaborative software development.

In a similar vein, Takahashi and Namiki (2003) see the OSS movement as a "public good approach" of de-Wintelization. The advent of freeware in the form of Linux, a Unix-based operating system developed on the Internet, challenged not only Wintel's monopoly but also the way of carrying out innovation within a single-mission, commercial organization. Linux is unique in that innovation occurred through a "forum" model. Torvalds made his software's design public on the Internet and since then programmers from around the world, who were strangers to each other, contributed to its development without pay.

In an interesting article by Cargill and Bolin (2004) the OSS movement is viewed as a new mode of standardization in the computer industry. They put the OSS movement as a natural stage of standard setting in the ICT industry following "SDOs" and "consortia & alliances":

> "Consortia replaced SDOs as a preferred venue because they responded to the needs of business better; Commercial Joint Ventures (a variant form of consortia) emerged to compete with the older, large, multidiscipline consortia, and open source reflected yet another market requirement" (pp. 4-5)

In the case of the computer industry, clearly the dominant mode of standardization has been "consortia and alliances", with dominant firms defining the directions of industry evolution: IBM in the mainframe era, DEC in the minicomputer era and Intel and Microsoft in the PC era.

Regarding the future of the IT industry standardization processes, Cargill and Bolin propose a governmental solution:

> "If standards are an impure public good, then the government has not only the right but the duty to intervene when the private sector fails. We believe that the beginning of this failure – as evinced by either chaos or monopoly – are already beginning to be seen. So the question really comes down to whether or not the private sector, with help from the government, can correct itself, or if it is willing to risk that no one will notice until the entire systems collapses. It is a bet that we will see played over the next five years" (p. 8)

In our opinion, the argument of Cargill and Bolin is too pessimistic, the rise of OSS is a dynamic force to challenge the dominance of a few giant firms in the computer industry per se, it is too early to predict this will result in chaos. Whether the "governmental" solution suggested by Cargill and Bolin will be effective in developing IT standards remains to be seen. We believe it will be a trend that the second and third mechanisms of standardization--consortia and OSS--converge due to the deep involvement of business communities in the OSS movement which may eventually even share some characteristics of SDOs.

In summary, all of these researchers have shown the importance of standards and standardization processes in the computer industry. The history of the computer industry evolution has been marked by shifts of the power centre of standard setting. As noted by Bresnahan and Greenstein (1999), after years of gradual evolution computing in the early 1990s saw a 'competitive crash' in which seller rents were drastically reallocated across market segments. Putting it into historical perspective, the OSS movement is an effort of some computer enthusiasts to break up the entrenched Wintel standards enabled by the commodification of computer hardware and by the proliferation of the Internet. This movement will have the potential of shaking up the whole computer industry landscape. In order to predict the possible future scenarios of OSS in the computer industry, in the next section we introduce a transaction cost theoretical framework for the analysis of platform success and of future scenarios of the OSS movement.

3. Theoretical framework: Transaction cost perspective of the computer industry evolution and standardization processes

Reimers and Li (2005) propose to distinguish between two types of transaction costs in the diffusion of a new product. Any firm offering a new product on the market must communicate knowledge about how to assess its new product properly or rely on potential buyers acquiring that knowledge by themselves, i.e. without the help of vendors. In addition to these "vertical" transaction costs which we term Type I transaction costs, there is a second type of transaction costs which needs to be considered when analyzing the special case of IT markets. IT markets are increasingly characterized by systems competition, meaning that the products offered by IT vendors are but components of a larger system which has to be assembled by the buyer or a buyer's agent. Thus, a firm offering new technology on a market characterized by systems competition (systems markets) has not only to communicate new evaluation knowledge to buyers--an instance of Type I transaction costs--but also to persuade other firms to offer complementary products. This requirement constitutes a second type of transaction costs for vendors offering information technology which we call Type II transaction costs. In the following, we will apply this distinction to the analysis of the computer industry evolution as depicted in the preceding section and then attempt to outline possible future scenarios for the OSS movement based on this analysis in the subsequent section.

According to Bresnahan and Greenstein (1999), there are two striking empirical regularities in the history of the computer industry. First, each new platform was introduced by new firms, not by previously successful computer companies. Although the creation of these new platforms represented technologically impressive innovation, in general these events did not disrupt the market structure of established platforms in established market segments. The rise of new segments of the computer industry originated from new demands which could not be satisfied by the old service providers, then some firms sensed this need and built a new platform. Later on these new business segments gained more popularity with the maturity of the platform. One advantage of this form of indirect entry is that fierce competition with powerful incumbents can be avoided, which allows the new platform to gain space to survive. Second, each new platform was first used by technical users in a new segment. Complexity and complementarity of platform components imply potentially long 'gestation' before platforms obtain mature designs, before users realize the full benefit of a new platform and before vendors reduce the cost of producing it. These two regularities show the roles played by the two types of transaction costs in the adoption and flourishing of a new platform. The fact that new platforms are adopted first by technical users shows the difficulty of overcoming the obstacle posed Type I transaction costs for the former to be accepted by the market. For technical users, the benefits of acquiring evaluation knowledge with regard to new products often exceeds the costs because new products potentially better fit their special requirements which are part of their professional activities and careers; thus they become the first adopters of a new platform. Only with these users as the seeds for the new platform will complementary products and service providers see some potential in the new platform thus reducing Type II transaction costs of deciding whether or not to develop complementary products and/or persuading other firms to do so. As a result, an increasing number of complementary products and services will be offered on the market starting a virtuous circle of increasing numbers of users of a new platform and an increasing variety of available complementary products and services. As interfaces become more user friendly--this itself being an instance of a complementary product/service--these newly developed complementary products and services make the platform more attractive also for non-technical users.

While the computer industry standardization process has been driven by dominant firms, Cargill and Bolin (2004) describe the institutional aspects of ICT standardization in a broader context. As mentioned above, they classify the efforts of ICT standardization into three stages: SDOs, consortia and alliance, and OSS. We can use our two types of transaction costs to analyze the pros and cons with regard to these mechanisms. While SDOs are good at addressing Type II transaction costs (coordinating among vendors of complementary products) they are not as good at innovation (i.e. with regard to Type I transaction costs) because of their very nature (ex post standardization). This problem is mitigated by consortia who, however, tend to fragment thus faring worse with regard to Type II transaction costs relative to SDOs (Markus et al., 2005). Since OSS facilitates ex ante coordination of complementary product and services development through revealing program code and thus possible interfaces with complementary products and services it seems to be a better way of dealing with Type II transaction costs relative to consortia; but the question, of course, is how it will be able to handle the problem posed by Type I transaction costs, i.e. innovation. Thus, one conclusion could be that, rather than evolving towards ever better ways of standardization, the process is torn between the problems posed by these two types of transaction costs and thus constantly moves between two partial sub-optimal solutions with tipping points reached when dissatisfaction with one mode reaches a critical threshold (as organization structure seems to do with regard to centralized and decentralized forms). The fact that the OSS movement's lack of marketing capability is a barrier to their success clearly demonstrates the importance of Type I transaction costs. As elaborated in more details in the next section, involvement of commercial organizations in OSS has the potential to reduce the problems caused by Type I transaction costs but poses risks with regard to OSS as a computer industry standardization mechanism.

The most relevant mechanisms of computer industry standardization thus far have been consortia and alliances. According to Cargill and Bolin (2004), in the consortia and alliances mechanism of ICT standardization there are two schools of thought on when and what to standardize: the "current practice school" believes it is preferable to standardize current practice by abstracting an interface specification from existing products, and the "future technology school" permits a group design, combining the best of breed (at times), but is usually slower and can produce a specification that is filled with compromise. In terms of our transaction cost framework, the "current practice school" focuses on Type II transaction costs by eliminating the problem posed by Type I transaction costs from their agenda while the "future technology school" addresses both types of transaction costs simultaneously. As discussed above, this approach faces a possible trade-off between the problems posed by these two types of transaction costs. In the next section we will discuss the conditions under which OSS will be able to effectively cope with this trade-off and thus to provide an alternative to the consortia and alliances mechanism.

4. Standardization versus innovation: Two scenarios for the future of the OSS movement

By its nature, the success of the OSS movement depends on contributions from "hackers" from around the world. Lerner and Tirole (2002) summarize the motivation of the freelance contributors into two reasons which they believe are consistent with traditional labor economics theory: (1) the immediate benefits obtained by OSS developers in "scratching an itch", namely solving a problem they face, and (2) the existence of a "signaling incentive" that derives from the gratification associated with peer recognition or from delayed benefits in the form of better job offers relative to non-OSS developers. While it is an interesting and open question whether the enthusiasm and altruism of these contributors will be sustainable for the prosperity of OSS projects--and what types of projects will thrive and under what governance structure--we believe the interaction between OSS communities and software companies will be critical for the shaping of the direction of the OSS movement.

Most research on the relationship between software firms and OSS communities has focused on how firms can leverage the new phenomenon of OSS for their strategic advantage. As documented in Krishnamurthy (2003A, 2003B), Microsoft's relationship with the OSS movement is a good example of how to take advantage of the OSS movement. On the one hand, it has engaged in a bitter battle with Linux. On the other hand, Microsoft has built a cooperative relationship with Ximian—an open source community. The company is now working with Ximian to build a Linux-based version of its .Net platform. Microsoft's flagship products, Windows 2000 and XP have both used a leading open-source operating system (Berkeley System Distribution or BSD).

Such corporate interest in exploiting the OSS movement is, of course, not limited to Microsoft. Many major players in the computer industry are now actively involved in a diverse set of OSS projects, most prominently the Linux project. As a result, this project has become professionalized as well as dependent upon corporate support (Hamm, 2005). Increasingly, contributors to the code base are professional programmers employed by major computer firms thus potentially crowding out "hackers" as well as changing the character of the project from a grassroots movement to a professional organizations (ibid.). While many early participants and observers may resent this development, we feel that it represents a necessary step if OSS is to play a continuing role in the evolution of the computer industry. However, the question is what this role will be. Based on our previous analysis, we depict two scenarios and attempt to identify the conditions which will make the one or the other more likely.

In the first scenario, OSS-based products will provide a common core of basic functionality which computer firms use to develop their proprietary products and services. While this common core functionality will facilitate integration and inter-operability across different vendor's products and services, significant hurdles towards inter-operability and integration of individual products and services will remain because vendors will continue to build their competitive advantage on proprietary products and services arguing that this is a necessary by-product of innovation. In this scenario, OSS will resemble current consortia and alliances based on the "current practice school". OSS governance structures will reflect market structures in terms of market power of individual firms, including the possibility of breaking up into competing alliances as has happened in the case of UNIX (Isaak, 2005).

In contrast, the second scenario would be akin to alliances and consortia following the "future technology school". Here, OSS would be used not only for coordinating basic functionality among competing and complementary products and services but also for developing new products and services. Thus, OSS would be a governance structure for innovation as well as standardization. As is true for the case of traditional consortia and alliances, projects would likely take more time than those focused on standardization alone; however, competing and complementary products would be inter-operable to a much higher degree facilitating integration and collaboration among users. Vendors would lose competitive advantage based on proprietary products and services and have to compete on the basis of cost, quality, service, and speed. While this scenario may sound much more futuristic than the previous one and even utopian, it reflects a core argument that is put forward in favor of the OSS model of developing software, namely the similarity of OSS and the mode of production of scientific knowledge (David et al., 2001).

Clearly, the second scenario would have far-reaching implications for the way innovative activity would be organized, including a rather different view of the function of markets than prevails today. According to that view which is closely related to the thinking of von Hayek (1945, 1973), markets are the main motor of innovative activity in modern societies. This school of thought also provides the justification for extensive use of intellectual property rights which has recently received some criticism with regard to some industries which are heavily dependent on IPRs such as the pharmaceutical industry (Anonymous, 2004). However, such considerations are outside the scope of this paper. Rather, we want to focus on the question under which conditions OSS as a governance mechanism would support the one or the other of the two scenarios depicted above. In addition, we think that the first scenario is reasonably likely so that we can limit our analysis to the second one.

Clearly, OSS as a governance mechanism would not only have to address the problem posed by Type II transaction costs, i.e. coordination of complementary products and services, but also Type I transaction costs which are related to innovative activity. Apart from the problem of coming up with innovative products--a problem which the OSS model might or might not address effectively--the main problem firms offering new products on a market is to educate potential buyers about the costs and benefits associated with a new technology. While a new product's net-benefit might be obvious to the prospective user who is aware of these benefits and costs, most users will shy away from obtaining this knowledge on their own unless specifically motivated, as we have argued with regard to technically sophisticated users in Section 3. Thus, firms offering new products would have to shoulder this burden of educating prospective users which poses a main hurdle for offering new products on a market. This is the condition that we have tried to capture with the notion of Type I transaction costs.

In other words, even if the OSS mode of developing software would be effective with regard to producing truly innovative products and services, the question of how to address the problem posed by Type I transaction costs would still have to be solved. This problem implies two related sub-problems, a classical problem of collective action in the form of free-riding--the smaller firms might rely on the larger ones for the task of educating prospective buyers--and a coordination problem. The coordination problem results from the fact that firms would have to agree on specific (new) technologies in order to jointly develop the common software code basis. In a market context, such ex ante agreement is not necessary and it may even be said that this is one of the main strengths of market coordination with regard to the development of new technologies. As a result, a coordination mechanism would be necessary if firms were to jointly develop new software incorporating new technologies as a basis for their individual products and services. Apart from a benevolent dictator, there are no immediately obvious solutions for this problem.

However, there is a historical precedence for such a coordination mechanism which we want to briefly draw upon in order to conclude this discussion. As described in Reimers and Li (2004), the process of standardizing the so-called Third Generation (3G) technology for mobile communication systems was, in fact, a process of joint development of a new technology (Gessler, 2002). Apart from specifying the required interfaces, the 3G standardization process also resulted in a major shift in the underlying wireless technology, namely from TDMA to CDMA. This shift was initiated by one newcomer firm that used the global standardization infrastructure to convince the major players to shift from the established technology towards a new one (developed by this firm). In a way, it can be said that this firm educated its competitors and complementors about costs and benefits of a new technology so that, in turn, it could rely on them for educating the whole market. After the new technology was agreed upon, all firms participating in that process had an interest in educating prospective buyers about its advantages (of course, such interest does not solve the potential problem of free-riding; due to space restrictions, we will not address this problem here).

The main precondition for this to happen was the existence of a “forum” on which ideas about new technologies could be shared. This forum was provided by the close-knit network of national, regional and international standardization organizations in the field of telecommunications. However, such a forum is conspicuously absent from most OSS projects, most prominently the Linux project which is coordinated according to the model of the benevolent dictator. We therefore conclude that for the second scenario to become more likely, existing OSS governance structures would have to be extended by communication platforms for exchanging and discussing ideas about new technologies. The study of the complex structure of organizations involved in the 3G standardization process may help in understanding the difficulty of setting up such a structure but it may be futile to attempt to transfer this structure which has emerged in the specific context of regulating international telecommunications to the world of software development.

5. Conclusions

In this paper we have developed a transaction cost perspective of the OSS movement. In particular we were interested in how the OSS movement will impact the computer industry evolution and its standardization processes under the assumption of increased corporate involvement in the OSS movement. For this purpose, we have drawn on existing literatures on computer industry evolution and standardization processes and shown how these can be explained by a simple theoretical concept that distinguishes between two types of transaction costs related to the coordination of complementary products and services on the one hand and to the communication of knowledge necessary for evaluating new products and services on the other hand. We have then used this concept to outline and evaluate two possible scenarios for the future development of OSS and its impact on the computer industry evolution and standardization processes. In the first scenario, OSS will become a mechanism for coordinating diverse activities related to IT products and services through providing a common set of basic functionalities and definitions. In the second scenario, OSS will, in addition to coordinating activities in the industry, also serve as a governance structure for coordinating innovation. We have found that, in order for this second scenario to be viable, existing OSS governance structures would have to be extended to include a communication platform on which ideas for new technologies could be discussed and evaluated. This analysis has assumed that OSS is indeed an effective mechanism for stimulating innovative activity and that problems of possible free-riding behavior can be effectively addressed. However, these two problems were not discussed in our analysis.

We do not want to make a prediction about the future of the OSS movement in this paper. Rather, the purpose was to demonstrate the usefulness of a simple theoretical concept for understanding the evolution of the computer industry and related standardization processes. We think that this concept’s usefulness is not only established through our analysis of past developments but also of our finding regarding its future, namely that one of the promises which is frequently mentioned with regard to the OSS phenomenon, namely that it provides a novel governance structure for organizing innovation, hinges on a crucial condition which, so far, is absent from existing OSS governance structures. Thus, we conclude that if the OSS movement is to fully or partially replace the market as the dominant governance structure for organizing innovative activity, it needs to be mended in significant ways which have been outlined in this paper.

References

Anonymous (2004): An Open-Source Shot in the Arm? The Economist, June 10, 2004.

Bresnahan, T.F., Greenstein, S., 1999. Technological competition and the structure of the computer industry. Journal of Industrial Economics 47, 1-40.

Bresnahan, T.F., Malerba, F., 1999. Industrial dynamics and the evolution of firms' and nations' competitive capabilities in the world computer industry. In: Mowery, D., Nelson, R. (Eds.), The Sources of Industrial Leadership. Cambridge University Press.

Cargill, C.; Bolin, S., 2004. Standardization: A Failing Paradigm. Paper presented on the conference "Standards and Public Policy", May 13-14, 2004, Federal Reserve Bank of Chicago.

David, P.A.; Arora, S.; Steinmueller, W.E., 2001. Economic Organization and Viability of Open Source Software--A Proposal to the National Science Foundation. Submitted by the Stanford Institute for Economic Policy Research, Stanford University, 22 January 2001.

Flamm, K., 1988. Creating the Computer. The Brookings Institution, Washington.

Gessler, F., 2002. The Development of Wireless Infrastructure Standards. Dissertation, Royal Institute of Technology, Stockholm, March 2002.

Hamm, S., 2005. Linux Inc. In: BusinessWeek Online, January 31, 2005.

Isaak, J., 2005. POSIX—Inside: A Case Study. Proceedings of the 38th Hawaii International Conference on System Science, 2005.

Krishnamurthy, S., 2003A. A managerial overview of open source software. Business Horizon September—October (2003), 47-56.

Krishnamurthy, S., 2003B. An analysis of open source business models. Working Paper, University of Washington, Bothell, February 2003.

Langlois, R.N., 1990. Creating external capabilities: innovation and vertical disintegration in the microcomputer industry. Business and Economic History 19, 93–102.

Lerner, J., Tirole, J., 2002. Some simple economics of F/OSS, Journal of Industrial Economics 52, 197–234.

Malerba, F., Nelson, R., Orsenigo, L., Winter, S., 1999. History friendly models of industry evolution: the computer industry. Industrial and Corporate Change. 1, 3–41.

Markus, M. L.; Steinfield, C.W.; Wigand, R.T.; Minton, G., 2004. Standards, Collective Action and IS Development--Vertical Information Systems Standards in the US Home Mortgage Industry. Discussion Paper, November 2004.

Reimers, K.; Li, M.Z., 2004. Effectiveness of Governance Structures for 3G Standardization and Implications for China's 3G Policy. In: P. Cunningham and M. Cunningham (eds.): eAdoption and the Knowledge Economy: Issues, Applications, Case Studies. Amsterdam et al.: IOS Press, 256-263.

Reimers, K.; Li, M.Z., 2005. Should Buyers Try to Shape IT Markets Through Non-Market (Collective) Action? Antecedents of a Transaction Cost Theory of Network Effects, Journal of IT Standards & Standardization Research 3, 44-67.

Takahashi, T.; Namiki, F., 2003. Three Attempts at "De-Wintelization": Japan's TRON Project, the US Government's Suits Against Wintel, and the Entry of Java and Linux, Research Policy 32, 1589-1606.

von Hayek, F.A., 1945. The Use of Knowledge in Society, American Economic Review 35, 519-530.

von Hayek, F.A., 1973. Economics and Knowledge. A Presidential Address to the London Economic Club, 10 November 1936, in J.M. Buchanan, G.F. Thirlby (eds.): L.S.E. Essays on Cost. London.

West, J., Dedrick, J., 2001. Open source standardization: The rise of Linux in the network era. Knowledge, Technology, and Policy 14, 88–112.

West, J. 2003. How open is open enough? Melding proprietary and open source platform strategies. Research Policy 32, 1259—1285.