Increasing Modularity of Production in Computer Business: Common Strategies and the Case of IBM Eclipse Project

Mehmet Gençer

Istanbul Bilgi University, Department of Computer Science

mgencer@cs.bilgi.edu.tr

Abstract

Market demands are driving computer software and hardware products towards increased variety and modularity. This study first reviews major developments in production regimes in the computer business and highlights common patterns in strategic management choices in attempt for accommodating market demands. Then the inter-organizational structures and business processes in IBM's Eclipse project are examined in further detail. In this example the open source licensing model and standardized interfaces are increasingly employed to tap on external resources, allowing to better utilize strategic alliances, achieve increased modularity and economies of substitution.

Finally implications for organizational design are discussed.

Keywords: innovation, modular production, business alliance networks, computer markets

1 Introduction

Computer hardware and software products are proliferating rapidly, inducing shorter product life-cycles and requiring continuous innovation in these markets. These developments are increasingly rendering traditional strategies of large firms based on economies of scale and vertical integration inapplicable, and forcing firms to externalize some of their innovation costs in various ways (Langlois 1990). Garud and Kumaraswamy propose the term 'economies of substitution' to suggest substitution of certain components of a technological system while reusing others in order to leverage market opportunities and improve overall innovative abilities; in the process, moving away from economies of scale and scope and vertical integration towards a mode of production based on strategic partnerships through alliance networks (Garud & Kumaraswamy 1995). Increasing influence of these processes and alliance networks on high technology businesses has been addressed widely in the management literature (Castilla, Hwang, Granovetter & Granovetter 2000, Oliver 2001, West 2003).

What are the implications of these developments for the management in general, and strategic management in the Information Technology sector

in particular? My conviction is that these developments confirm emergence of a holistic conceptualization, in the management practice, of production activity as a complex and organic system which departs from traditional mechanistic conceptualizations that reigned in early decades of large scale production of computer products. In other words, vertical disintegration in production activities of firms is not merely a result of increasing resource dependence with the managerial mindsets staying the same, but also a shift in the very core of management practice to cope with, and make best of increasing interdependency. Although limited to only certain areas of business activity, the shift is analogous to emergence of quantum physics concepts after centuries with Newtonian program(Byrne 2002). Therefore the practice once again is leading the way to a territory for which the academic research is relatively under-equipped to cope with.

Yet, there are certain establishments in management and organization theory literature we can consult. Aldrich's evolutionary program(Aldrich 1999) for example proposes wider application of biological concepts to study of inter-organizational contexts. Such perspectives appear to be a sensible response to criticisms of reductionism and representationalism voiced since early years of organizational and inter-organizational research(Selznick 1948, Bittner 1965). Similarly studies in social psychology are providing valuable abridgments to management studies to account for complex organizational contexts. Weick's conceptualization of organizations(business or otherwise) as interpretation systems, for example, potentially allows one to transcend

limitations of formalism in administrative design(Daft & Weick 1984). We are yet to reach a congruence of cutting edge quantitative methods such as social network analysis(e.g. (Burt 2005)) and population perspective with qualitative aspects such as organizational knowledge and interpretation, but we can still apply multiple lenses to various organizational contexts successfully, provided that our choices are informed with foundational inadequacies of each.

In this study we first review the trends towards modularity of production in the computer business with conditions of the business environment that stimulate these developments and fitness of various strategies to cope with these environmental conditions. Section 2 presents business strategies of some key players in computer hardware and software markets, highlighting the general trend towards inter-firm product modularity(Schilling 2000) and loosely coupled organization designs(Sanchez & Mahoney 1996). Increasing costs of innovation within the vertical integration has forced large businesses such as Sun Microsystems and Apple Computers to externalize these costs through various mechanisms such as employing open standards to achieve inter-firm product modularity or appealing to open source communities and their licensing regimes(West 2003, von Hippel & von Krogh 2003); in each case reclaiming an appropriate pace of innovation with the cost of increased interdependence on other firms or professional communities.

In section 3 the case of IBM led Eclipse project is taken in perspective, which is framed as 'an ecosystem of innovation' by its founders. IBM's strat-

egy in the Eclipse project is informed by its previous experience with products of the open source community and rests on opening up one of its proprietary software products through open source licenses in exchange for proliferation of compatible components in the ecosystem produced by a vivid community of partnering firms and individuals(des Rivieres & Wiegand 2004). The overall process has resulted in a product architecture with increased modularity, rich features, and improvements in the adoption of product by users. Use of common standards and open source licenses accompanying this modular production regime underlines increased interdependence for accommodating demands for modularity in newly emerging strategic configurations in the computer markets.

Building on this line of presentation from the more general trends to structure and workings of specific communities, section 4 discusses some ideas on implications for organizational design. I argue that a loosely coupled systems perspective(Orton & Weick 1990) is fruitful in making sense of the business environment where a purely rational and monolithic perspective fails, such as the case of software innovation where companies are forced to externalize costs of product feature development and testing while simultaneously seek ways to manage the process and appropriate the outcomes. While purely evolutionary perspective is useful for inter-organizational research and predicting environmental trends, organizational design strategies can be better informed when conceptualizations make sense at the borderline of the organizational and the inter-organizational. The discussion also draws on

Sanchez and Mahoney's notion that products design organizations, although ostensibly it is the opposite(Sanchez & Mahoney 1996), and present views on situations where the strategic recipes in perspective may be applicable. Finally section 5 summarizes conclusions of the study.

2 Modularity trends in computer business

Most people in my generation have first saw a computer in the form of a game computer which was available in late 1970s with affordable prices. There were famous brands which we still remember today, such as Amiga and Sinclair, and they have produced the whole product from its chip design to its software. However the real breakthrough in widespread use of computers in home and offices came a few years later with IBM's introduction of the personal computer (PC) architecture. The move was revolutionary because for the first time it introduced a hardware standard called ISA (Industry Standard Architecture) which allowed pieces of hardware such as hard disks produced by other firms (usually from a far eastern country) to be 'plugged in' to the computer. The software platform for IBM PCs was made by a seperate, then very small, company called Microsoft. Many other computer hardware producers followed suit and the ISA became a standard which was in wide use for a long time and formed basis for subsequent standards.

This architectural design paved way for a rich marketplace of pluggable components for millions of computers worldwide. Furthermore, and equally

importantly, IBM has moved to using third party software extensively. The stability of modular computing architecture was supported by increasing involvement of independent bodies like IEEE (Institute of Electrical and Electronics Engineers) and IETF (Internet Engineering Task Force) and helped to improve interoperability of various pieces, of both hardware and software. Considering costs of developing increasingly complex software for their hardware platforms on the one hand(Brooks 1995), and difficulties in coping with varying demands for hardware components, the move was a total success for IBM; although some say that IBM's later decision to hand on their OS/2 technology to Microsoft, leading to development of Windows platform, was a big mistake (Today failure of Windows platform to comply with standards and Microsoft's aggressive marketing strategies are often accused for a reversal in the openness and standards compliance of software market for the platform, resulting in buggy software with security flaws and shorter product lifetimes.)

Today a product interoperability of this sort in the computer and electronics market is widely visible. Despite their monolithic design, mobile phones, for example, use software platform which usually comes from a company other than the hardware brand. Same is true for notebook computers which come with a great number of standardized interfaces such as USB, network, screen, sound outputs, wireless and bluetooth devices, etc., which allow for a great level of interoperability between various electronic devices. Studies by Langlois(Langlois 1990) and Sanchez and Mahoney (Sanchez &

Mahoney 1996) point to mechanisms of unplanned coordination and role of products and standardized interfaces in enabling loosely coupled and modular production to function effectively. Similar developments of 'opening-up' were reported in the literature, about workstation market, operating system platforms, etc., involving big players such as Sun Microsystems or Apple (Garud & Kumaraswamy 1995, West 2003).

An important influence for the software market came from academia in early 1990, by the appearance of Linux operating system and open source licensing. Although open source licensing existed for several years by then, having its roots in the increasing skepticism of academics towards originally publicly owned operating system called Unix, the so called open source movement became more visible after Linux' appearance and managed to mobilized thousands of programmers to contribute to computer programs which were common property. This core feature, we now understand, allows for a level of collective innovation which is quite hard to achieve with proprietary software products(von Hippel 2001, von Hippel & von Krogh 2003, von Krogh & von Hippel 2006). With widespread use of small electronics and inherent difficulty of adapting existing proprietary operating system platforms such as Microsoft's, many firms turned to Linux for its flexibility and lower cost. Today various private enterprises develop business models and product lines based on essentially publicly owned pieces of software.

Non-rival use of software promoted by open source licenses had considerable effects in the market. Several companies such as Sun, IBM and

Netscape has voluntarily opened up their products with expectations to tap on resources from the computing community otherwise not available. Today products such as OpenOffice, Firefox web browser, and Apache web server both compete successfully with their proprietary counterparts and provide a basis for services and software offered by these and other firms. Although open source licensing is not the only strategy for success in the software market, its increasing popularity can not be neglected.

3 The Eclipse project

The Eclipse community came into existence in 2001 and consists of several dozen firms and thousands of individuals gathered around the common interest of producing a piece of software, also called Eclipse, which is used for software production; i.e. Eclipse is a computer program for computer programmers to help them manage various aspects of increasingly complex software production activity more easily. The Eclipse community is not the oldest, largest, or most important one in the world of computer software, but in many ways it is a successful representative of the cutting edge core. The product is quite large and complex, its sleek and flexible design is widely appreciated and helped Eclipse establish a considerably large user base in a highly competitive market, and its production model based on open source licenses in one which receives increasing attention for its potential to accommodate flexibility and innovation in large scale production of complex goods.

In 2004 Eclipse Foundation was created to support the growing community of firms and individuals contributing to and using Eclipse software platform.

The key promise of Eclipse was to fix the lack of interoperability among software tools which has been a major problem for developers. The modular, 'plug-in' architecture of Eclipse is open and allows anyone to develop pluggable software components which can not only be integrated easily with the rest of the system but also interacts with other components in a safe way which reduces crashes of the overall software. Today the Eclipse *ecosystem*, as it is called by the founders, contain 157 member companies from 47 different industries endorsed by the Eclipse foundation, 30 sub-projects, and over 10.000 people including staff from the member firms and independent developers or users. Not all members are equal as it is the case in rather 'purely' open source projects. The Eclipse community is somewhat semi-hierarchical. Member firms are of three types: strategic members, add-in providers, and associates, in the decreasing order of status.

Figure 1 is a visualization of relations between the firms based on their partnering in the sub-projects. Among all 157 members, only 33 were involved in projects, and the graph shows only those members. The structure of the community reflects central position of IBM who has contributed the original software code and driven the community process for the large part. There are only two small groups which develop sub-projects without involvement of IBM.

Figure 2 shows the same network with colors to distinguish between dif-

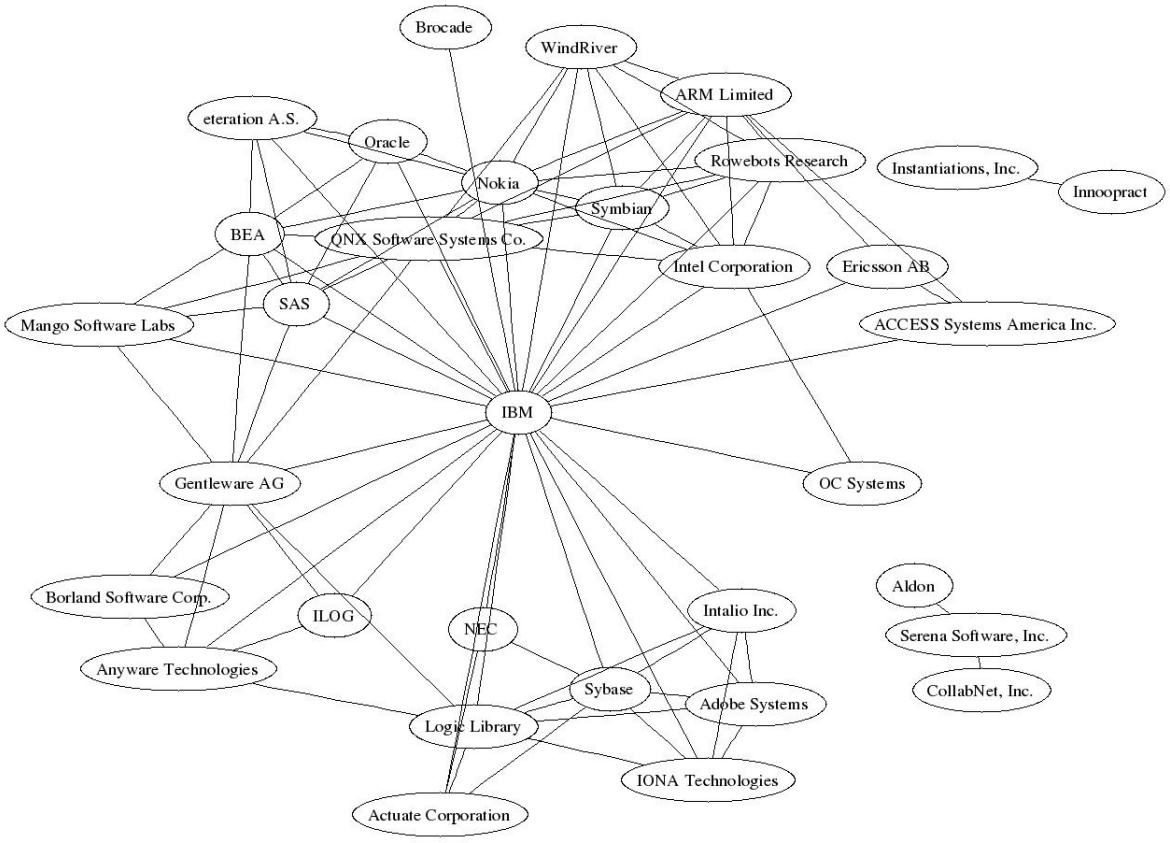

Figure 1: Partnership relations between the Eclipse project member firms.

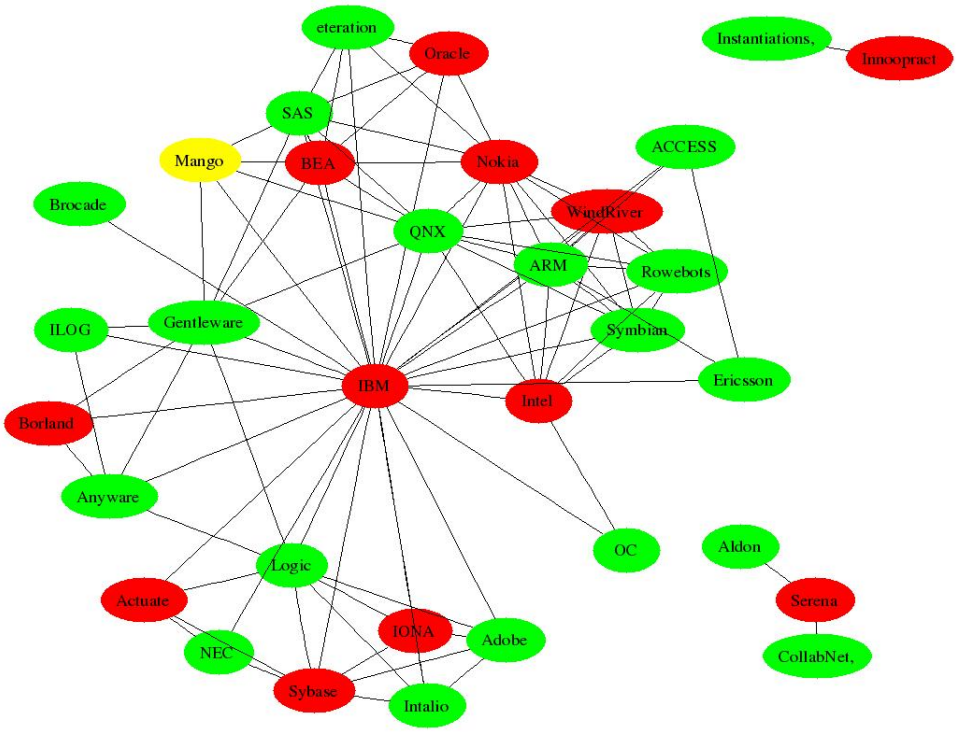

Figure 2: Partnership relations between different member types in the Eclipse project. Red: Strategic members, Green: Add-in providers, Yellow: Associate members

ferent types of members. In this visualization we observe that strategic members have more central positions compared to add-in providers, whereas only one associate member appears in the network. It is reasonable to expect this member to eventually achieve a higher rank formally. Also we observe that in the two small independent groups, a strategic member seems to be the initiator in group formation.

Figure 3 shows all the firms in the Eclipse project. Most firms are not involved in the sub-projects which are 'officially' endorsed by the foundation.

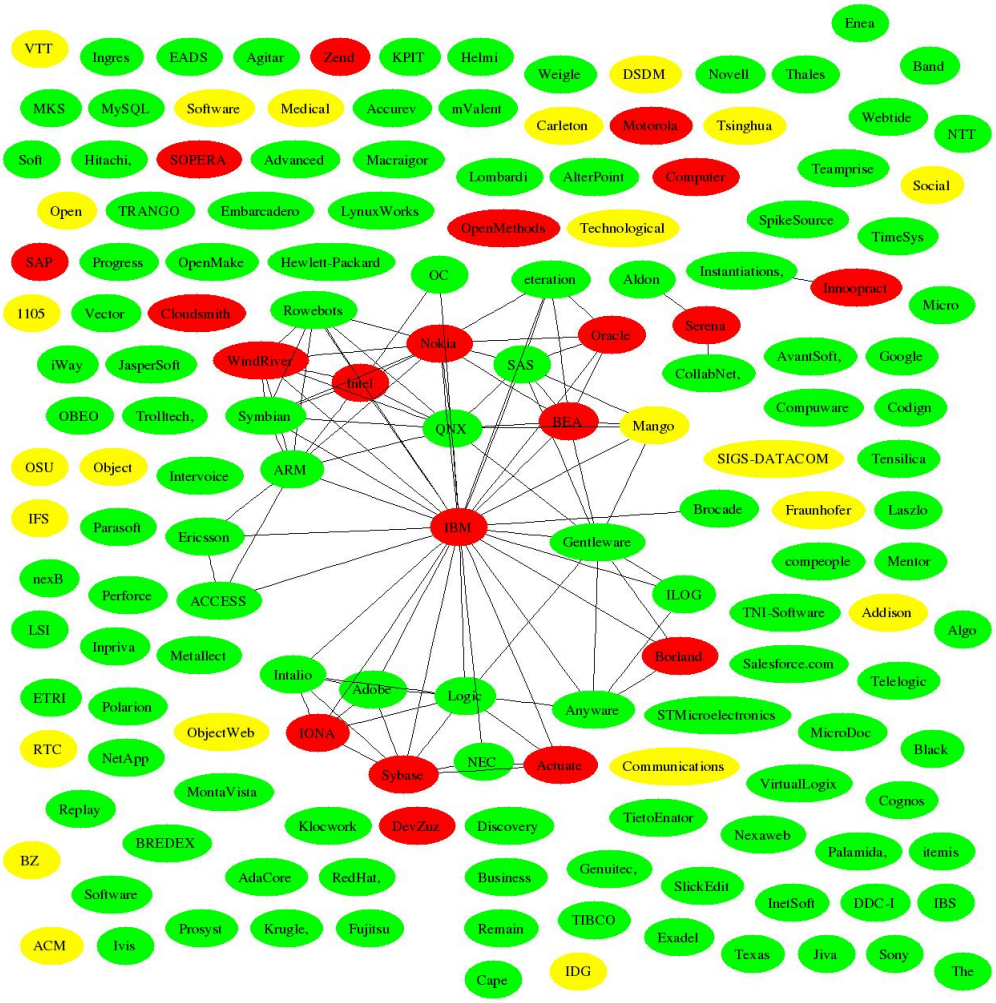

Figure 3: All members and their relations in the Eclipse project. Red: Strategic members, Green: Add-in providers, Yellow: Associate members

Therefore some of the firms which are isolates in this figure are possibly producing software, but nevertheless the network shown reflects relative positions of firms within the ecosystem. More precisely the average nodal degree for strategic members is 4.4 whereas it is 0.85 for add-in providers and 0.22 for associate members.

The network of Eclipse member firms contain several cliques, ie. groups of firms which have more relations between themselves than with others. Such patterns of network development is observed in many other social networks(Burt 2005, Gençer, Oba, Özel & Tunalıoğlu 2006*b*) and indicate a subgroup in the business ecosystem getting specialized on a market niche and tightening ties among themselves. In an analysis of communication patterns in the Eclipse project, Kidane and Gloor found a positive correlation between number of ties in the group and its performance and creativity(Kidane & Gloor 2007). Therefore we see emergence of specialized groups with the capability to combine with other firms or groups, through modular structure based on open interfaces and interoperability they provide. However little is understood regarding what early features and actions lead to success in such networks.

By maintaining a balanced community structure which on the one hand lowering barriers for entry through openness of community and openness of core product for plugging-in new components, and maintaining a hierarchical arrangement which controls evolutionary paths from lower to higher status on the other, IBM and Eclipse Foundation manages to maintain a position

at the core of a productive and innovative ecosystem. For other firms, the Eclipse project presents a business opportunity to take place in a promising enterprise and more importantly establish new business alliances and pursue business opportunities safely with the help of non-rival licensing policy.

It is worth noting that there is nothing utterly substantial about open source licensing regime regarding the workings of Eclipse ecosystem. However it does facilitate enormous reductions in the transaction costs between the parties involved and comes in as a natural choice of governance for many software development problems(Demil & Lecocq 2006, Gençer, Oba, Özel & Tunalıoğlu 2006*a*). IBM has been one of the first firms to see the opportunities the open source model presents and build successful profit models based on this ostensibly business hostile model. But the actual success in the case of Eclipse project comes from the ability of IBM in harvesting the innovations of the community while at the same time decoupling itself from its operation and costs associated. The strategy is also amenable to scaling since the costs and level of interaction increases rapidly in cases of monolithic organizing where community grows in size(Gençer 2007).

4 Discussion: implications for organizational design strategies

When is an 'opening-up' in the vein of Eclipse community desirable? Most businesses, especially in the consumer products market, pursue various means

to include consumers in feature development. These means vary from approaching core communities such as in what Hippel describes for high-performance windsurfing(von Hippel 2001), to award campaigns in which companies distribute prizes for best feature suggestions from consumers, or process improvement suggestions from workers. Therefore product and process development is a top candidate for these types of inclusive strategies.

In the case of open source software product features are 'asserted rather than elicited'(Noll 2007). These asserted features survive if there is a demand for them. This natural inclusiveness of open source model was found promising, for example, in production of off-the-shelf software components(Ayala, rensen, Conrad, Franch & Li 2007), in ways very similar to what ISA architecture has achieved in the case of hardware. Therefore from an evolutionary perspective, the key contribution of open source model in cases like the Eclipse project is its ability to foster variation in a scalable way. This was perhaps the key reason when Sun Microsystems has switched to open sourcing one of their key platforms, the Java, after years of being criticized for moving slowly in feature propagation under their strict control, although they worked hard to be inclusive in the process.

The key to a successful opening-up strategy seems to lie in an intentional and carefully managed loose coupling of processes(Sanchez & Mahoney 1996). In the case of IBM's software strategy, open source model provides a starting template for such a strategy. By tweaking the licenses and managing the process through a semi-hierarchical control, IBM's strategy brings to-

gether an improved variation in product features and controlled selection in its adoption of the community innovations.

In the case of software and hardware products the design of products needs to reflect the business strategic choice towards economies of substitution. This means product designs based on interfaces and pluggable components rather than monolithic designs which is an impediment for scaling. On the other hand recipes for managing the loosely coupled processes can only be suggested, but their exact application is likely to be quite domain specific.

5 Conclusion

Many businesses increasingly feel the need for modularity and flexibility in production, and enactment of external resources for innovation. Examples of business strategies from computer hardware and software sector reviewed in this study, together with a more extensive analysis of IBM's strategy in the Eclipse software platform, underlines certain patterns in building successful organizational designs to achieve these ends. The key to success in applications of such strategies lies in balancing the managerial control while loosening the coupling between processes. In the case of Eclipse and many other software products, open source model has been used successfully to strike this balance, provided that the managerial control is exerted without disturbing the capabilities of the model for fostering innovation. In general a more organic conceptualization of innovation processes based on loosely

coupled system is a necessary component of an orientation which provides decision makers with alertness necessary to harness potentials of modular production without losing competitive edge, and improve strategic flexibility of their business under fluctuations in business environment.

References

Aldrich, Howard. 1999. *Organizations Evolving*. Sage.

Ayala, Claudia, Carl-Fredrik Sørensen, Reidar Conrad, Xavier Franch & Jingyue Li. 2007. *Open Source Developement, Adoption and Innovation*. Springer chapter Open Source Collaboration for Fostering Off-The-Shelf Components Selection.

Bittner, Egon. 1965. "The Concept of Organization." *Social Research* 32(3):239–255.

Brooks, Frederick P. 1995. *The mythical man-month : essays on software engineering*. Addison-Wesley.

Burt, Ronald S. 2005. *Brokerage and Closure*. Oxford University Press.

Byrne, David. 2002. *Interpreting Quantitative Data*. Sage.

Castilla, Emilio J., Hokyu Hwang, Ellen Granovetter & Mark Granovetter. 2000. *The Silicon Valley Edge : a habitat for innovation and en-*

trepreneurship. Stanford University Press chapter Social Networks in Silicon Valley, pp. 218–424.

Daft, Richard L. & Karl E. Weick. 1984. "Toward a Model of Organizations as Interpretation Systems." *Academy of Management Review* 9(2):284–295.

Demil, Benoît & Xavier Lecocq. 2006. "Neither Market nor Hierarchy nor Network: The Emergence of Bazaar Governance." *Organization Studies* 27(10):1447–1466.

des Rivieres, J. & J. Wiegand. 2004. "Eclipse: A platform for integrating development tools." *IBM Systems Journal* 43(2):371–383.

Garud, Raghu & Arun Kumaraswamy. 1995. "Technological and Organizational Designs for Realizing Economies of Substitution." *Strategic Management Journal* 16:93–109.

Gençer, Mehmet. 2007. "Structure and Collaborative Aspects of Internet Standards." IEEE Symposium on Computers and Communications.

Gençer, Mehmet, Beyza Oba, Bülent Özel & V. Sinan Tunalıoğlu. 2006*a*. "Forking: The GPL Coherent Technology for Flexible Organizing in FOSS Development." European Group of Organizational Studies 2006 Colloqium in Bergen, Norway.

Gençer, Mehmet, Beyza Oba, Bülent Özel & V. Sinan Tunalıoğlu. 2006*b*. *Open Source Systems*. IFIP Working Group 2.13 Foundation on Open

Source Software 2006 Springer chapter Organization of Internet Standards.

Kidane, Yared H. & Peter A. Gloor. 2007. "Correlating temporal communication patterns of the Eclipse open source community with performance and creativity." *Computational & Mathematical Organization Theory* 13:17–27.

Langlois, Richard N. 1990. "Creating External Capabilities: Innovation and Vertical Disintegration in the Microcomputer Industry." *Business and Economic History* 19:93–102.

Noll, John. 2007. *Open Source Developement, Adoption and Innovation.* Springer chapter Innovation in Open Source Software Development: A Tale of Two Features.

Oliver, Amalya L. 2001. "Strategic Alliances and the Learning Life-cycle of Biotechnology Firms." *Organization Studies* 22(3):467–489.

Orton, J. Douglas & Karl E. Weick. 1990. "Loosely Coupled Systems: A Reconceptualization." *The Academy of Management Review* 15(2):203–223.

Sanchez, Ron & Joseph T. Mahoney. 1996. "Modularity, Flexibility, and Knowledge Management in Product and Organization Design." *Strategic Management Journal* 17:63–76.

Schilling, Melissa A. 2000. "Toward a General Modular Systems Theory and its Application to Interfirm Product Modularity." *Academy of Management Review* 25(2):312–334.

Selznick, Philip. 1948. "Foundations of the Theory of Organization." *American Sociological Review* 13(1):25–35.

von Hippel, Eric. 2001. "Learning from open source software." *MIT Sloan Management Review* pp. 82–86.

von Hippel, Eric & Georg von Krogh. 2003. "Open Source Software and the "Private-Collective" Innovation Model: Issues for Organization Science." *Organization Science* 14(2):209–223.

von Krogh, Georg & Eric von Hippel. 2006. "The Promise of Research on Open Source Software." *Management Science* 52(7):975–983.

West, Joel. 2003. "How open is open enough? Melding proprietary and open source platform strategies." *Research Policy* 32:1259–1285.