Open Innovation:
The Paradox of Firm Investment in Open Source Software

Joel West*
Associate Professor, College of Business, San José State University
One Washington Square, San José, CA 95192-0070 USA
+1-408-924-7069; fax: +1-408-924-3555
Joel.West@sjsu.edu

Scott Gallagher
Assistant Professor, College of Business, James Madison University
Harrisonburg, VA 22807 USA
+1-540-568-8792; fax: +1-540-568-2754
gallagsr@jmu.edu

*Corresponding author

5 October 2004

Keywords: open innovation, technology commercialization, open source, software industry

Abstract

We identify three fundamental challenges in applying the concept of "open innovation": finding creative ways to exploit internal innovation, incorporating external innovation into internal development, and motivating outsiders to supply an ongoing stream of external innovations.

To illustrate these points, we show how firms use open source software to support innovation strategies. From this, we identify four approaches — pooled R&D, spinouts, selling complements and attracting donated complements — and discuss how they address the three key challenges of open innovation. From this, we offer suggestions as to how these principles can be applied to other industries, and directions for future research.

Open Innovation:
The Paradox of Firm Investment in Open Source Software

1. INTRODUCTION

Models of open innovation offer the promise that firms can achieve a greater return on their innovative activities and their resulting intellectual property (IP). Open innovation models stress the importance of using a broad range of sources for a firm's innovation and invention activities, including customers, rivals, academics, and firms in unrelated industries while simultaneously using creative methods to exploit a firm's resulting IP (Chesbrough, 2003a). While the use of external sources of innovation is nothing new, recently the most successful high-tech firms have been those that "free ride" on the basic research of others.

The open innovation paradigm is often contrasted to the traditional vertical integration model where internal R&D activities lead to internally developed products that are then distributed by the firm. However, saying what "open innovation" isn't doesn't tell managers what it is. Here we define open innovation as systematically encouraging and exploring a wide range of internal and external sources for innovation opportunities, consciously integrating that exploration with firm capabilities and resources, and broadly exploiting those opportunities through multiple channels. The open innovation paradigm therefore goes beyond simply the externalization of research and development and is as much a change in the use and management of IP as it is in technical and research driven generation of IP.

Over the past 20 years, an increasingly popular example of open innovation has been open source software, which involves collaboration between firms, suppliers, customers and makers of related products to pool software R&D to produce a shared technology such as the Linux operating system. Here we consider patterns of open source innovation as an exemplar for more general forms of open innovation, and how open source addresses three challenges of open innovation.

2. Research on Exploiting Innovation

The pace of technological advance has often been subdivided into two phases: invention (a scientific breakthrough) and innovation (commercialization of the invention). Nelson & Winter (1982: 263) attribute the distinction to Schumpeter (1934). This split parallels a similar bifurcation between research and development, where inventions come from basic research and innovations from the development group. Many organizations, however, define additional phases between the two extremes, as with Intel's "advanced development" step (Tennenhouse, 2003). Others have attempted to subdivided innovation into "radical" and "incremental", where the former more closely resembles the "invention" concept (e.g. Leifer et al, 2000).

Like Nelson and Winter, we use "innovation" in its broadest sense to refer to the entire process by which technological change is deployed in commercial products. Such innovation may incorporate formally protected intellectual property (such as patents or copyrights) that is difficult to imitate, or it may reflect tacit knowledge that is readily imitated and at best provides a transient competitive advantage.

2.1 Proprietary R&D as a Source of Innovation

The source of innovation activities has interested economists for many years (e.g. Schumpeter, 1942). One of the most successful models to emerge was a "proprietary" model where large enterprises internalized their R&D activities. Historical accounts suggest that early R&D activities grew out of the need in many industries to maintain and improve production activities (Chandler, 1990). Because activities were frequently unique for each firm, investments in R&D were firm-specific and thus (as predicted by transaction cost economics) firms internalized their R&D expenditures (Williamson, 1991). From such R&D, firms naturally moved to exploit their accumulated knowledge to develop new products, thereby enhancing their economies of scope; in many industries large scale dedicated R&D functions emerged, providing a barrier to entry through economies of scale (Chandler, 1990).

The managerial approach used for this proprietary model was summed up by Harvard president James Bryant Conant as "picking a man of genius, giving him money, and leaving him alone" (Conant, 2002). Edison's Menlo Park, AT&T's Bell Labs, and Xerox's PARC were exemplars of this type of innovation model and brought about many inventions and innovations during the 20th century. Laboratory researchers in these labs generated IP that engineers could develop into commercial products.

However, sometimes the production-line innovation model broke down when research generated IP that could not be internally commercialized. In some cases, such IP would be licensed to others, but in many other cases it "sat on a shelf" waiting either for internal development or its research proponents to leave the firm and develop it on their own. This led to considerable knowledge spillovers to other firms who were able to capture the benefits of the innovation; the best known contemporary example of such spillovers is Xerox PARC (Smith and Alexander, 1998; Chesbrough & Rosenbloom, 2002).

2.2 Absorbing External Knowledge

As even proponents of the proprietary model will concede, considerable knowledge is generated beyond the confines of the firm. In addition to internally generated knowledge, von Hippel (1988) identified four external sources: 1) suppliers and customers; 2) university, government and private laboratories; 3) competitors; and 4) other nations. Firms that fail to exploit such external R&D may be at a severe competitive disadvantage (Rosenberg and Steinmueller, 1998).

Many models have been developed to explain how firms can exploit external knowledge. Perhaps the simplest method is to imitate a competitor: such free riding on the product and market investments of rivals is a common way for firms to overcome a first mover strategy (Lieberman and Montgomery, 1998). Consulting with customers can provide firms ideas about discovering, developing, and refining innovations (von Hippel, 1988). Public sources are also an important source of knowledge, for example government R&D spending was identified almost 50 years ago as an important stimulus for private R&D (David, Hall, and Toole, 2000). Similarly, university research is often explicitly funded to generate external spillovers (Colyvas et al, 2002). The opportunities for firms to benefit from these latter two sources of innovation are aided by the strong professional norms and desires of researchers to publish the results of their work to as wide an audience as possible.

The challenge for firms then is how to find and exploit these external sources of innovation. The nature of the knowledge, including its complexity, explicitness, and similarity to extant knowledge are important considerations (Afuah, 2003). A firm's ability to tap external R&D knowledge frequently seems to depend on its investment in internal R&D capabilities that facilitate the development of "absorptive capacity" (Cohen and Levinthal, 1990). If firms cannot (or don't wish to) develop sufficient absorptive capacity they may utilize strategic alliances in order to gain such knowledge or utilize complementary resources to exploit that knowledge (see Gulati, 1998 for a review of alliances). This alliance or "network" approach is particularly common in technology intensive industries such as biotechnology (Powell, Koput and Smith-Doerr, 1996; Mowery et al 1996). Finally, location has also been shown to result in knowledge spillovers between firms and from university research in many industries, especially high-tech (Porter, 1990; Baptista & Swann, 1998; Kenney, 2000).

The managerial challenges of utilizing external knowledge then center around identifying useful external knowledge, and then integrating that knowledge with the firm. For example, for new products there are significant trade-offs between innovation speed, development costs, and competitive advantage for relying on external rather than internal learning (Kessler, Bierly, and Gopalakrishnan, 2000). Environmental scanning, competitive intelligence, sponsored research, and membership in relevant trade organizations is a way to uncover external knowledge opportunities. Developing absorptive capacity, via *internal* R&D investments appears to be an important prerequisite for converting external knowledge into internal innovation (Cohen and Levinthal, 1990).

2.3 The Open Innovation Paradigm

In contrast to earlier models and "fully integrated innovators" like AT&T (now Lucent) Bell Labs and IBM which do basic research through commercial products, open innovation celebrates success stories like Cisco, Intel and Microsoft, which succeed by leveraging the basic research of others (Chesbrough, 2003a). Under this paradigm, firms exploit both internal and external sources of innovation, while maximizing the returns that accrue from both sources (Table 1).

Tactics that embody an open innovation approach include exploiting knowledge spillovers and consulting with venture capitalists, while also using both inbound and outbound licensing of key technologies. Although earlier frameworks acknowledged the role of external knowledge and "accidental" internal discoveries, it is the systematic encouragement and integration of these issues coupled with creative exploitation of IP that distinguishes open innovation from earlier innovation models.

While leveraging the innovations of others might seem the equivalent of “found money,” in practice the integration of internal and external innovation entails three challenges:

- *maximization*. Firms need a wide range of approaches to maximize the returns to internal innovation — not just feeding the company’s product pipeline, but also outbound licensing of IP, patent pooling and even giving away technology to stimulate demand for other products.
- *incorporation*. The existence of external knowledge provides no benefits to the firm if the firm cannot identify the relevant knowledge and incorporate it into its innovation activities. This requires scanning, absorption and also the political willingness to incorporate external innovation.
- *motivation*. Open innovation assumes an ongoing stream of external innovation, but such “found money” spillovers require that the stream be replenished.

2.3.1 Maximizing Returns to Internal Innovation

A central concern to open innovation is how to best use the internal R&D capabilities of the firm to maximum advantage. Those capabilities can be used for

- generating innovations to be internally commercialized (the traditional model);
- building absorptive capacity and using that capacity to identify external innovations;
- generating innovations that generate returns through external commercialization (e.g. licensable patent portfolios); and
- generating IP that does not produce direct economic benefit, but indirectly generates a return through spillovers or sale of related goods and products.

Successful approaches will often combine a variety of these approaches. For example, to identify promising technologies Intel establishes research labs near top university research groups, with open flows of information in both directions. If an innovation proves promising, Intel recruits the top academic researchers to help commercialize the technology and see it through to production (Tennenhouse, 2003).

This approach can be used cooperatively as well, as with the GSM patent pool assembled by European telephone makers in the early 1990s. While the patents were often the result of basic research, contribution of a patent to the patent pool allowed firms to have favorable access to all of the IPR of the GSM standard, creating a cost advantage for European pool participants over potential Asian rivals (Bekkers et al, 2002).

2.3.2 Incorporating External Innovations

To benefit from external innovations, organizations need to identify such innovations, maintain the absorptive capacity to understand them, and be able to combine such spillovers with firm-specific internal innovation to produce a product tailored to the firm's specific needs.

Even if external innovations are identified, that does not mean they will be incorporated into the firm's product strategies. A firm that was once highly successful at the integrated innovation model will tend to believe its innovations superior to any competing ideas from outsiders. For example, flush from its successful user interface innovations of the 1980s, engineers at Apple Computer rejected external ideas in areas such as handheld computers, adopting the phrase "not invented here" to describe such rejection (Kaplan 1996: 156).

2.3.3 Motivating Spillovers

With external innovation, there is often an unstated assumption that the supply will continue. But what happens if everyone tries to be a "free rider"? Will "innovation benefactors" — such as government and nonprofit research sponsors be as fertile a field (Chesbrough, 2003b)? If commercial firms do not realize a return on their innovative activities, they will tend to under-invest in innovative activities that are either highly risky (e.g. basic research) or that are easily imitated by free-riding competitors. Therefore, we consider the incentives for generating the knowledge spillovers at two levels: the individual and the organizational.

Motivating individuals to generate and contribute their IP in the absence of financial returns is a significant management challenge for an open innovation approach. One of the simplest models of motivation is expectancy theory that posits that individuals are motivated when both valence, the attractiveness of a reward, which can be either intrinsic (e.g. happiness) or extrinsic (e.g. fame or fortune) and instrumentality, the path to that reward, are present (Lawler, 1971). The proprietary innovation model solved this challenge though extrinsic compensation coupled with adherence to traditional professional scientific norms, e.g. freedom. The external model relied upon others, e.g. universities, to partially or wholly provide motivation.

The incentives for organizations to contribute spillovers fall into two categories. In the one case, the innovation benefits the innovator and nothing is reduced by sharing that benefit. Customers often share their innovations with their vendors if it means improved products in the future (von Hippel, 1988). And of course suppliers invest in innovations to sell more products, as when Intel increases the performance of microprocessors that it sells to Dell.

Spillovers to a direct competitor are more problematic, but still are economically rational under conditions of "co-opetition". Firms in the same industry complement each other in creating markets but compete in dividing up markets (Brandenburger & Nalebuff, 1996: 34). So if a firm stands to benefit from an innovation that grows the market, it will accept spillovers if the return from its share of market growth is attractive enough.

3. Research Design

Our three challenges led to three related research questions:

- What circumstances motivate firms to embrace open innovation approaches as part of their R&D efforts?
- Why would for profit firms commit their intellectual property as well as *ongoing* human resources to an effort that they know will benefit others, including competitors?
- Why do individuals contribute their IP to a project that benefits firms without receiving financial remuneration?

We chose to study the use of external innovation in the software industry, in particular the "open source" movement. Given the comparative newness of the open innovation paradigm, we chose to use a theory-building approach grounded in the context of rich data. This draws on established procedures for generating theory from qualitative data (Glaser and Strauss, 1967), as well as management studies that employ the inductive method to draw theory from a set of case studies (Harris and Sutton, 1986; Eisenhardt, 1989).

From 2002 to 2004, we conducted 5 interviews with 41 informants representing 26 organizations. Of the 26 organizations, 14 were for-profit companies in the I.T. industry. The interviews also included 7 major open source projects, as well as other professionals indirectly involved in the industry. Most interviews ranged from 45-90 minutes, and most were tape recorded for later consultation. This was supplemented by observation of (and, in some cases, participation in) four Silicon Valley industry conferences and seminars from August 2003 through March 2004 that focused solely on open source software.

In addition to these primary data sources, we also incorporated secondary data. During the observation period, one of the authors reviewed approximately 800 news articles from trade journals, business press and websites related to Linux and other open source topics. Finally, we reviewed of published literature in technology management, sociology and computer science on open source collaboration.

4. Open Source as Open Innovation

Open source and other collaborative development techniques in the software industry offer examples of how the three key challenges of open innovation can and have been addressed by commercial firms. Open source also offers an approach to address what West (2003) refers to as an "essential tension" in information technology innovation: appropriating the returns from an innovation versus winning adoption of that innovation.

The term "open source" was coined in 1997 to refer a category of software where the source code is freely distributed. The term is defined to require free redistribution of the software, its source code, or any derived works thereof, while at the same time acknowledging the original author's ownership and contribution (Perens 1999; Raymond 2004). In popular usage, the term "open source" overlaps (and largely subsumes) the "free software" category as launched by Richard Stallman in 1984.[1]

While "open source" is explicitly an IP policy, it often refers to a development methodology where geographically dispersed programmers collaborate in the joint production of a software good through a set of virtual collaboration tools. Such programmers were originally hobbyist users, but increasingly have been professionals paid by employers that either intend to use the software internally or sell related products and services. To be effective, such paid collaborators must internalize the ethos of both their project and the employer (O'Mahony, 2003).

---

[1] However, the "free" software contains IP restrictions intended to force sharing of any derivative works, while other forms of "open" software (such as the Apache license) allow private commercialization of related innovations (West 2003).

Meanwhile, the widespread dissemination of the necessary tools have both fueled the rise of specific open source projects but also enabled other forms of open innovation in the software industry, such as user scenario development for PC games.

What motivates individuals to contribute to open source projects? Consistent with expectancy theory, empirical researchers (Hars and Ours, 2001; Hertel et al, 2003; Lakhani & von Hippel, 2003) found three general categories of contributor motivations:

- *direct utility,* either to the individual or to one's employer;
- *intrinsic benefit* from the work, such as learning a skill or personal fulfillment;
- *signaling* one's capabilities to gain respect from one's peers or interest from prospective employers.

Thus, open source as an open innovation strategy has two key components: shared rights to use the technology, and collaborative development of that technology. Here we consider four approaches for external innovation in software, where firms have both invested in open innovation and benefited from them (Table 2).

4.1 Pooled R&D: Linux, Mozilla

A familiar model of open innovation is that of pooled R&D. While cooperative research often occurs to save costs, prior research also suggests that firms cooperate in cases where they cannot appropriate spillovers from their research (Ouchi and Bolton, 1988), in industries with strong vertical relationships (Sakakibara, 2001) and in areas that are highly risky or for industries most dependent on advanced science (Miotti and Sachwald, 2003).

Two highly visible open source examples are support for the Linux operating system through the Open Source Development Lab, and the Mozilla web browser. For both, firms donate R&D

to the open source project while exploiting the pooled R&D of all contributors to facilitate the sale of related products.

A simple example is the Mozilla open source project, a descendant of the Netscape Navigator browser offered for a wide range of systems — Windows, Macintosh, Linux and at least 7 Unix variants. This browser was the first commercial browser product (“Netscape Navigator”, 2004)., Navigator held more than two-thirds of the browser market until late 1997 — surpassing Microsoft’s Internet Explorer — but only two years later the shares were reversed due to IE’s bundling with Microsoft Windows (Bresnahan & Yin, 2004).

Netscape created the Mozilla open source project in 1998, and in July 2003 terminated all internal development, deferring further work to the open source community. Unix system vendors such as IBM, HP and Sun were left without a browser, which they needed to sell Internet-connected workstations. Thus, each of the major Unix systems vendors assigned software engineers to work with the Mozilla project, both to help keep the project moving forward and to assure that new releases are compatible with their respective systems (Dotzler, 2004).

The R&D cooperation in the Open Source Development Labs (OSDL) for Linux is more complex. Founded in 2000, the OSDL takes as its mission “To be the recognized center-of-gravity for the Linux industry; a central body dedicated to accelerating the use of Linux for enterprise computing” (“Corporate Overview,” 2004). In its first five years, the consortium began work on three projects: data center Linux, carrier grade Linux and desktop Linux.

The founders, sponsors and other members of OSDL and their motivations for supporting OSDL could be grouped into four broad categories: vendors of computer and telecommunications systems, producers of microprocessors, Linux distributors and support organizations, and developers of complementary software products (Table 3).

How do such projects address the three open innovation challenges? For firms participating in Mozilla, the *quid pro quo* is straightforward: systems vendors maximize the returns of their innovation by concentrating on their own needs (such as platform-specific customization), and then incorporated the shared browser technology into their integrated systems. However, the motivation problem is not completely solved, in that the systems vendors assume a pool of individual open source contributors that sustain innovation in the core product.

For the OSDL, firms contributed their specialized knowledge (e.g. in telecommunications operations or microprocessor architecture) to build a common platform. OSDL resembles other self -supporting industrial research consortia, where firms pool interests towards a common goal, and assume they can both cooperate in supporting that goal and compete in selling their respective products.

However, both Mozilla and OSDL differ from typical consortia in two ways:

- *Spillovers are not controllable*. Many consortia incent membership by limiting direct access to the consortium's research output to member-participants, limiting access to indirect spillovers. Open source licenses typically make it impossible to limit even direct access, allowing non-members to accrue many of the same benefits as members.
- *Contributions from non-participants*. The engineering contributions to these open source projects extend beyond the sponsoring companies to include user organizations, academics, individual hobbyists and other interested parties. Unless the

corporate contributions eventually dwarf the individual ones, the projects must continue to motivate such contributions to survive.

Given these factors, an open source innovation model is inherently more "open" than a typical R&D consortium, both in terms of exploiting information from outside the consortium, and sharing that information back out to non-member organizations and individuals.

4.2 Spinouts: Jikes, Eclipse, Beehive

Open innovation can release the potential of technologies within the firm that are not creating value. In some cases, the technologies are no longer strategic, as with AOL Time Warner's decision to spinoff Mozilla into a stand-alone open source project after firing its Netscape development team (Hansen, 2003).

But in addition to spinoff (and, frequently, abandonment), firms also have opportunities to release more value from their technologies by situating them outside the firm, but at the same time maintaining an ongoing corporate involvement. Here we use the term "spinout" to refer to all cases where firms transform internal development projects to externally visible open source projects.

If a firm gives away its IP, how can such spinouts create value? One way is that the donated IP generates demand for other products and services that the donor continues to sell. Two examples of this come from IBM and its efforts to promote the Java programming language developed by Sun Microsystems, that was widely embraced by firms competing with Microsoft in web-based technologies. In such a Java-centric world, IBM would still generate revenue from sales of hardware and supporting services.

IBM's first open source spinout came from a pre-production R&D project. In response to IBM's growing interest in Java, in early 1996 two IBM researchers began work on an experimental Java compiler, which they named “Jikes”. They quickly developed a prototype that was more efficient than Sun’s industry standard compiler. After customer requests for a better Java compiler, in December 1998 IBM announced the release of Jikes in open source form to allow external programmers to extend and improve the compiler. IBM continues to host the project website, but since 2000 development has been led by non-IBM engineers (Gonsalves & Coffee, 1998; Shields, 2004). Jikes has been widely adopted, and is now bundled with several distributions of open source operating systems.

A second IBM spinout came with Java development tools. In 1996, IBM purchased a Canadian software company that created such tools for its WebSphere application server product; IBM released much of this technology in open source form when it founded the Eclipse project in 2001 (Brody, 2001). IBM was joined by other software companies involved in web application development, including Borland, Red Hat, SAP and SuSE, and well as hardware makers HP, Fujitsu and Intel; in 2004, the project became an independent non-profit corporation (“Eclipse Forms Independent Organization,” 2004), although IBM engineers retained technical leadership of key projects. As an IBM executive later explained, “It is not that we are looking to make more money off the platform. It is just that we are looking to accelerate the adoption of Java and the building up of it for all of us” (Southwick, 2004).

But despite this openness, Java rivals BEA and Sun chose not to join IBM’s coalition, instead promoting the rival Java Tools Community (Taft, 2003). Meanwhile, in May 2004 BEA announced plans to create a “Beehive” open source project, in which key application libraries from its WebLogic product family could be used with Sun’s NetBeans, Eclipse, or other development systems.

The spinout thus makes sense for technologies that either are not yet commercialized (as with Jikes), or that will eventually become commoditized and thus of limited commercial value (as many predicted for Java development tools).

Both IBM and BEA donated internal innovations to create open source projects, which were intended to fuel adoption of the innovations. As with other organizations that sponsor open source projects, the benefits included:

- helping establish their technology as *de facto* standards, which, at a minimum, reduces the likelihood of having to re-implement their technology to conform to competing standards;
- attracting improvements and complements that make the technology more attractive;
- together, the innovation and complements enable the sale of related products (such as other components of WebSphere and WebLogic); and
- generating mindshare and goodwill with the same audience that includes the potential customers for these related products.

Interestingly, these motivations for open source spinouts are contrary to one of those of one of the most-often cited firm sponsor of spinouts, Xerox Parc, which spun out technologies that no longer aligned with the company's strategy (Chesbrough and Rosenbloom, 2002). Here (consistent with West, 2003), firms relinquished control of key technologies precisely because they were strategically aligned — but relinquishing control was an effective strategy for winning adoption.

4.3 Selling Complements: Apache, KHTML, Darwin

Many goods in computers and electronics fall into what Katz & Shapiro (1985) term the "hardware-software paradigm". As Teece (1986) notes, the base innovation ("hardware") requires an investment into producing complementary goods ("software") specialized for that innovation, in order to make the entire system useful. In many cases, these complements are more valuable than the core innovation. For example, makers of videogame consoles deliberately lose money or break even on the hardware so that they can make money from software royalties (Gallagher & Park, 2002).

In other cases, a system architecture will consist of various components. Some mature (or highly competitive) components may be highly commoditized, while other pieces are more rapidly changing or otherwise difficult to imitate and thus offer opportunities for capturing economic value.

Two open source examples are the Apache web server and the KHTML web browser engine.

Customers access IBM's WebSphere e-commerce software using standard web browsers, so IBM originally developed a proprietary httpd (web page) server. IBM later abandoned its server for the Apache httpd server, recognizing that it would be wasting resources trying to catch up to the better quality and larger market share enjoyed by Apache (West, 2003). Today, IBM engineers are involved in the ongoing Apache innovation, both for the httpd server and also related projects hosted by the Apache Software Foundation (Apache.org website).

Similarly, in 2002 Apple Computer decided to build a new web browser called Safari, to guarantee one would be available for buyers of its computer systems. Instead of Mozilla, it incorporated the KHTML open source libraries, which in turn were developed to support the KDE desktop interface for Linux users (Searls, 2003). The move paralleled Apple's earlier use of BSD Unix as a foundation for its OS X operating system, in which it created a new open source project (Darwin) to share all modifications of the BSD code (West, 2003). For both Safari and

OS X, Apple used open source and contributed back its changes, but the company did not release the remainder of the proprietary code for its browser and OS, respectively (Brockmeier, 2003).

In the case of the Apache, KHTML and Darwin open source projects, the firms adopting open source components had four common characteristics:

- there was pre-existing open source code being developed without the intervention of the focal firms;
- the "buy vs. build" decision to use external innovation was made easier because the code was "free"[2];
- the firms were willing to contribute back to the existing projects on an ongoing basis, both to assure that the technology continued to meet their respective needs and to maintain absorptive capacity;
- the firms could continue to yield returns for internal innovation by combining the internal and external technologies to make a product offering that was not directly available through open source.

4.4 Donated Complements: Avalanche, PC Game "Mods"

In other cases, firms make their money off of the core innovation but seek donated labor for valuable complements.

For decades, I.T. companies have encouraged their users to collaborate and share user-developed software that filled in the gaps for their proprietary offerings. This has been particularly relevant for medium and large buyer organizations (companies, universities and

---

[2] Both Apache and BSD packages were open without restriction in the typology of West (2003), while KDE contained the compulsory sharing restrictions of the GPL.

government) with large internal I.T. organizations. In the 1960s, IBM sponsored its SHARE user group, while in the 1970s Digital Equipment Corporation had its DECUS.

More recently, firms have indirectly or directly supported user collaboration that is coordinated using open source techniques. One example is the Avalanche Technology Cooperative, a Minneapolis-based nonprofit founded in 2001 to pool IT customizations developed by local enterprise IT users. This would allow companies to integrate disparate packages such as PeopleSoft and SAP that do not provide their own integration modules (Lien and Black, 2004; Gomes, 2004).

Another example of donated complements is the use of "mods" for PC video games. The PC gaming industry competes with lower priced dedicated gaming platforms such as Sony's PlayStation2 or Microsoft's Xbox. The commercial publishers of PC games thus have decided to exploit the one key advantage they have versus the consoles: the ability for PC users to update and modify their games. To do this, publishers release editing tools for their games to encourage user mods that create different environments, scenarios, or even total rebuilds of the game; the users then freely distribute these mods on the Internet. A few of the mods (such as Battle Grounds) are developed as open source, but most are developed as closed source.

While mods do not directly generate publisher revenues, the novelty of the mods extends the relatively short demand period for most computer games. Meanwhile, the mods keep the name of the game in front of consumers for additional months, while the publishers need years to prepare follow-on products. This external innovation keeps the product current without tying up internal innovation resources. In rare cases, the publisher serves the need identified by the mod by creating its own game (as in Battlefield Vietnam), or even buys the mod outright (as with Counter-Strike).

As with open source, a key issue for mods is motivating the contributors. The motivations parallel those for open source: direct utility, intrinsic reward or external signaling. Individuals (or virtual teams) contribute mods because of their creative nature, love of either the computer game they modified or the milieu they recreated via their mod (Todd, 2004). Students are also frequent contributors, increasing their enjoyment of a favorite game (direct utility) as well as signaling their value to potential employers.

The computer game industry highlights three key ideas for attracting external innovation that similar to those for open source:

- *minimizing technical obstacles*. Contributors develop mods because they can build upon the publisher's proprietary innovation to make a compelling game experience. As with other software development platforms (such as operating systems or databases), third party developers are attracted by platform capabilities and the prompt availability of development tools.
- *creating an infrastructure* that encourages participation and collaboration. For open source, this is a project website and e-mail lists, but for mods this would be a distribution site that highlights the mods. Modern technologies make the cost of such infrastructure quite low.
- *recognition for contributors,* including added visibility for the most popular creators. For example, since 2002 Apple has given annual awards for the best use of open source related to its OS X operating system.

However, the mods also help address one problem that's very different from those of business-oriented source projects. As with other entertainment products, novelty-seeking consumers eventually grow bored with a PC game; by combining the core game engine with new externally generated game scenarios, the external innovation extends the life of the core (internally developed) innovation.

5. DISCUSSION

5.1 Assessing Software Industry Patterns

How did firms in open source software efforts effectively exploit open innovation? We identified four strategies of open innovation in software that addressed the unique combination and exploitation of innovation from multiple sources challenges. Table 4 cross references these four strategies with the key open innovation challenges we identified.

Despite the wide popularity of open source collaboration, software firms appear to embrace open innovation only when there is no alternative — specifically a broad dispersal of both production knowledge and market share that forces vertically integrated producers to admit that they no longer can "do it all." The use of open source by firms typically begins in ways that does not disrupt their fundamental business model (e.g. selling complements), or comes at a time when their existing business model is so threatened that they are forced them to make drastic changes.

At the same time, firms faced (and often acknowledged) the risks of collaborating, risks that have not yet been fully realized. For example, the pooled R&D supporting a common Linux platform by reduces the barriers to entry for new systems vendors (West & Dedrick, 2001). And encouraging users to develop complements could reduce the availability of vendors to achieve proprietary lock-in, which is an explicit long-term goal of the Avalanche project (Lien & Black, 2004).

We believe that open source software helps explain one finding of Almeida et al (2003), who found that larger semiconductor firms are most likely to build on the knowledge of others. Among our cases, it appeared because of their scope of products, the largest firms could not ignore any significant source of external innovation available to rivals. Meanwhile, smaller firms had a choice of market niches: some explicitly positioned their innovation strategies to align with open source external innovation, while others sought to remain in niches that would be unaffected by open source competition.

5.2 Implications for Other Industries

The four patterns of combining internal and external innovation in open source could be applied to more general forms of open innovation:

- *Pooled R&D*. This has been used in many industries, but the open source examples offer new insights for coordination and governance of collaboration to support competing interests.
- *Spinouts*. Since spinouts are valuable for technologies locked in the laboratory, they are most relevant to the largest firms, which tend to both have the largest innovation budgets and also the largest bureaucracies that can defeat commercialization. While Xerox PARC is the poster child for such obstacles, Chesbrough and Rosenbloom (2002) note that in some cases Xerox was able to spinout the technology and participate financially in its commercialization.
- *Selling complements*. Firms have long sold support services around public goods. However, open innovation requires overcoming the "not invented here" bias of in-house R&D to maximize reuse by designing external innovations into internally developed products.

- *Donated complements*. These fit the category of what von Hippel and Katz (2002) refer to as "user toolkits," where general purpose technologies are sold to users capable of generating their own modifications and improvements. Such strategies are most feasible when selling to technically proficient buyers, whether corporate engineers or hobbyist-programmers.

Open source software as part of corporate open innovation strategies is still a comparatively recent phenomenon, and there are some unresolved issues. Open source built on a confluence of ideology, professional norms and enthusiasm; some question the long-term sustainability of such motivations. Also, many projects have been created as challenges to an entrenched incumbent (e.g. Microsoft), and if such challenges are largely unsuccessful, vendor interest in sponsoring future open source efforts could wane.

Also, open source has yet to fully resolve the IP issues of accepting donations from a wide community of unknown contributors. Contributors to Linux have been accused by SCO of stealing copyrighted or trade secret-protected source code from SCO's proprietary Unix implementation. While such potential infringement has been attributed to ignorance, others have suggested that infringing "stealth" IP could be deliberately donated to projects to sabotage their success (Cargill & Bolin, 2004).

5.3 Further Research

In considering what "open innovation" means, there is a huge gap between the free-riding on basic research (e.g. Chesbrough, 2003b) and this study of the partitioning of software development between firms and open source projects.

From our own study, we suggest that future research should consider whether these characteristics of open source are prerequisites for other forms of open innovation:

- *feasibility of virtual teams* as a way to organize innovation, enabling pooled R&D and other collaboration between organizations;
- *a culture of open innovation* throughout such teams that spans organizations, vanquishing both a "not invented here" attitude towards external innovation and a "crown jewels" attitude of controlling internal innovations;
- *modularization* of technologies and products, to allow the external production of components or complements; and
- *formal IP mechanisms* that encourage collaboration.

Our attempts to define open innovation uncovered questions beyond those specific to software. Two relate to the availability of external spillovers:

- *commercialization of public research*. Universities have increasingly sought to profit from their research spillovers, a trend encouraged in the U.S. by the Bayh-Dole Act (Colyvas et al, 2002). Will this restrict the flow of external innovations or provide an ongoing incentive for greater supply?
- *increasing conflict over patents*. The increasing scope and commercial value of patents has spawned various concerns that patents will inhibit traditional closed innovation (e.g. Jaffe & Lerner, 2004); the threat to external spillovers is likely greater.

Other questions relate to potential patterns for leveraging external knowledge:

- *boundaries of the firm*. If R&D is performed in virtual teams, collaborative R&D consortia or other shared fora, is this evidence that R&D is no longer necessary to internalize in firms, reflects examples of specific innovations that cannot be appropriated by firms, or symptomatic of industry segments that have become commoditized and thus R&D produces little competitive advantage?

- *role of process innovations*. Companies like Dell combine external product innovations with internal process innovations. Research on open innovation has focused on innovation to produce products, so would the process of open innovation be fundamentally different when it incorporates process innovations?
- *low R&D intensity firms*. Many firms have low R&D intensity, either due to size (e.g. small businesses) or industry characteristics (low tech). Are they pursuing "external innovation," "open innovation," or (as commonly assumed) "no innovation" strategies?

These are but some of the questions about how open innovation can be applied to a wide range of industries and contexts.

6. References

Afuah, A. (2003) *Innovation Management.* 2nd ed. New York: Oxford University Press.

Almeida, P., Dokko, G. and Rosenkopf, L. (2003) Startup size and the mechanisms of external learning: increasing opportunity and decreasing ability? *Research Policy,* **32**, 2, 301-315.

Baptista, R. and Swann, P. (1998) Do firms in clusters innovate more? *Research Policy*, **27**, 5, 525-540.

Bekkers, R., Duysters, G. and Verspagen, B. (2002) Intellectual property rights, strategic technology agreements and market structure: the case of GSM. *Research Policy,* **31**, 7, 1141-1161.

Brandenburger, A. M. and Nalebuff, B.J. (1996) *Co-opetition,* New York: Doubleday.

Bresnahan, T. and Yin, P-L. (2004) Standard setting in browsers: technology and (real world) incomplete information. Standards and Public Policy conference, Federal Reserve Bank of Chicago, May 13-14.

Brockmeier, J. (2003) "Is Open Source Apple's Salvation?," NewsFactor Network, April 21, 2003, URL: http://www.newsfactor.com/perl/story/21318.html

Brody, Steve, "Interview: The Eclipse code donation," IBM developerWorks, 1 November 2001 URL: http://www.ibm.com/developerworks/linux/library/l-erick.html

Cargill, C. & Bolin, S. (2004) Standardization: a failing paradigm. *Standards and Public Policy Conference*, Chicago Federal Reserve, Chicago, USA. May 12 - 14.

Chandler, A.D. (1990) *Scale and Scope*. Cambridge, MA: Belknap.

Chesbrough, H. and Rosenbloom, R.S. (2002) The role of the business model in capturing value form innovation: evidence from Xerox corporation's technology spin-off companies. *Industrial and Corporate Change,* **11**, 3 (June), 529-555.

Chesbrough, H.W. (2003a) *Open Innovation,* Boston, MA: Harvard University Press.

Chesbrough, H.W. (2003b) The Era of Open Innovation. *Sloan Management Review*, **44**, 3, 35-41.

Cohen, W. M. & Levinthal, D. A. (1990) Absorptive capacity: a new perspective on learning and innovation. *Administrative Science Quarterly*, **35**, 128-152.

Colyvas, J., Crow, M., Gelijns, A., Mazzoleni, R., Nelson, R.R., Rosenberg, N. and Sampat, B.N. (2002) How do university inventions get into practice? *Management Science,* **48**, 1, 61–72.

Conant, J. (2002) *Tuxedo Park*. New York: Simon & Schuster.

"Corporate Overview," Open Source Development Labs, April 1, 2004, URL: http://www.osdl.org/about_osdl/OSDL_overview_website.pdf

David, P.A., Hall, B.H., and Toole, A.A. (2000) Is public R&D a complement or substitute for private R&D? A review of the econometric evidence. *Research Policy*, **29**, 497-529.

Dotzler, A. (2004), interview, Mozilla.org, March 8.

"Eclipse Forms Independent Organization," press release, February 2, 2004, URL: http://www.eclipse.org/org/index.html

Eisenhardt, Kathleen M., "Building Theories from Case Study Research," *Academy of Management Review* 14, 4 (Oct. 1989): 532-550.

Glaser, B. and A. Strauss, 1967, *The Discovery of Grounded Theory: Strategies of Qualitative Research*. London: Wiedenfeld and Nicholson.

Gomes, L. (2004) Avalanche Project is Clearing the Path for Tech Cooperation. *Wall Street Journal*, April 12, B1.

Gonsalves, A. and Coffee, P. (1998) Jikes! More Open Source Code, *PC Week,* **15**, 49, December 7, 6.

Gulati, R. (1998) Alliances and networks. *Strategic Management Journal,* **19**, 4, 293-317.

Hansen, Evan, "AOL lays off Netscape developers," CNET News.com, July 15, 2003, URL: http://news.com.com/2100-1032_3-1026078.html

Harris, Stanley G., Sutton, Robert I. "Functions of Parting Ceremonies in Dying Organizations," *Academy of Management Journal*. (Mar 1986). Vol. 29, Iss. 1; p. 5 (26 pages)

Hars, A. and Ou, S. (2002) Working for free? Motivations for participating in open-source projects. *International Journal of Electronic Commerce,* **6**, 3, 25-39.

Hertel, G., Niedner, S. and Herrmann, S. (2003) Motivation of software developers in open source projects: an Internet-based survey of contributors to the Linux kernel. *Research Policy,* **32**, 7, 1159-1177.

Jaffe, A. B. and Lerner, J. (2004) *Innovation and Its Discontents: How Our Broken Patent System is Endangering Innovation and Progress, and What to Do About It.* Princeton, NJ: Princeton University Press.

Kaplan, J. (1996) *Startup: A Silicon Valley Adventure,* Paperback ed, New York: Penguin.

Katz, M. L. and Shapiro, C. (1985) Network externalities, competition, and compatibility. *American Economic Review,* **75**, 3, 424-440.

Kenney, M., editor, (2000) *Understanding Silicon Valley: Anatomy of an Entrepreneurial Region,* Stanford: Stanford University Press.

Kessler, E.H., Bierly, P.E., Gopalakrishnan, S. (2000) Internal vs. external learning in new product development: effects of speed, costs and competitive advantage. *R&D Management*, **30**, 3, 213-224.

Lakhani, K. R. and von Hippel, E. (2003) How open source software works: "free" user-to-user assistance. *Research Policy,* **32**, 6, 923-943.

Lawler, E.E. (1971) *Pay and Organizational Effectiveness: A Psychological View*, New York: McGraw-Hill.

Leifer, R., McDermott, C.M., O'Connor, G. C., Peters, L.S., Rice, M.P., Veryzer, R.W., and Rice, M. (2000) *Radical innovation: how mature companies can outsmart upstarts,* Boston: Harvard Business School Press.

Lieberman, M.B. and Montgomery, D.B. (1998) First mover (dis)advantages: retrospective and link with the resource-based view. *Strategic Management Journal*, **19**, 1111-1125.

Lien, S. and Black, A. (2004) Personal Interview, Avalanche Technology Cooperative, March 16.

Miotti, Luis and Frédérique Sachwald (2003) Co-operative R&D: why and with whom?: An integrated framework of analysis *Research Policy,* 32, 8, 1481-1499.

Mowery, D. C., Oxley, J. E., Silverman, B. S. (1996) Strategic alliances and interfirm knowledge transfer. *Strategic Management Journal*, **17**,(Winter Special Issue), 77-91.

Nelson, R. R. and Winter, S.G. (1982) *An Evolutionary Model of Economic Change.* Cambridge, Mass.: Harvard University Press.

"Netscape Navigator", Wikipedia, April 16, 2004, URL: http://en.wikipedia.org/wiki/Netscape_Navigator, accessed May 22, 2004.
O'Mahony, S. (2003) Guarding the commons: how community managed software projects protect their work. *Research Policy*, **32**, 7, 1179-1198.
Ouchi , William G. and Michele Kremen Bolton (1988) The Logic of Joint Research and Development *California Management Review* 30, 3, 9-34.
"OSDL Members," Open Source Development Labs, copyright 2003, URL: http://www.osdl.org/about_osdl/members/
Perens, B. (1999) "The Open Source Definition" in Chris DiBona, Sam Ockman and Mark Stone, eds., *Open Sources: Voices from the Open Source Revolution,* Sebastopol, Calif.: O'Reilly, pp. 171-188.
Porter, M.E. (1990) *The Competitive Advantage of Nations*. New York: The Free Press.
Powell, W.W., Koput, K.W. and Smith-Doerr, L. (1996) Interorganizational collaboration and the locus of innovation: networks of learning in biotechnology. *Administrative Science Quarterly,* **41**,116-145.
Raymond, E. "The Open Source Definition," Version 1.9, 2004, URL: http://www.opensource.org/docs/definition.php
Rosenberg, N. and Steinmueller, W. E. (1988) Why are Americans such poor imitators? *American Economic Review*, **78**, 2, p229 - 235.
Sakakibara, Mariko (2001) Cooperative research and development: who participates and in which industries do projects take place? *Research Policy*, 30, 7, 993-1018.
Schumpeter, J.A. (1934), *TheTheory of Economic Development,* Cambridge, Mass., Harvard University Press.
Schumpeter, J.A. (1942) *Capitalism, Socialism, and Democracy.* New York: Harper and Row.
Searls, Doc "Surprise: Apple's New Browser Is a Sister to Konqueror," LinuxJournal.com, January 11, 2003, URL: http://www.linuxjournal.com/article.php?sid=6565
Shields, D. (2004) Personal Interview, IBM Corporation, May 24.
Smith, D. K. and Alexander, R.C. (1988) *Fumbling the Future*. New York: William Morrow.
Southwick, K. "Big Blue's Mr. Web services," March 17, 2004, CNET News.com, URL: http://news.com.com/2008-7345_3-5173667.html
Taft, Darryl K., "Sun Mulls Joining Java Eclipse Effort," *eWeek,* September 1, 2003, URL: http://www.eweek.com/article2/0,1759,1236123,00.asp
Teece, D. (1986) Profiting from technological innovation: implications for integration, collaboration, licensing and public policy. *Research Policy,* **15**, 6, 285-305.
Tennenhouse, D. (2003) Innovation breeds success at Intel. *IEE Engineering Management,* **13**, 6 44-47.

Todd, B. (2004) The whys of modding. *Computer Games*, No. 163, June, 38-40.

von Hippel, E. and Katz, R. (2002) Shifting innovation to users via toolkits. *Management Science,* **48**, 7, 821- 834.

von Hippel, Eric (1988) *The Sources of Innovation*. New York: Oxford University Press.

von Hippel, Eric, "Horizontal innovation networks - by and for users," MIT Sloan School of Management Working Paper No. 4366-02, June 2002, URL: http://web.mit.edu/evhippel/www/UserInnovNetworksMgtSci.pdf

West, J. (2003) How open is open enough? Melding proprietary and open source platform strategies. *Research Policy,* **32**, 7, 1259-1285.

West, Joel, Dedrick, Jason, 2001, Open Source Standardization: The Rise of Linux in the Network Era, *Knowledge, Technology and Policy* 14 (2), 88-112.

Williamson, O.E. (1991) Comparative economic organization: the analysis of discrete structural alternatives. *Administrative Science Quarterly*, **36**, 2, 269-296

7. Figures and Tables

Innovation Model	Management Challenges	Resulting Management Techniques
Proprietary (or internal or "closed")	1. Attracting "best & brightest" 2. Moving research results to development	1. Provide excellent compensation, resources, and freedom. 2. Provide dedicated development functions to exploit research and link it to market knowledge.
External	1. Exploring a wide range of sources for innovation. 2. Integrate external knowledge with firm resources & capabilities	1. Careful environmental scanning 2. Developing absorptive capacity, and/or using alliances, networks, and related consortia.
Open	1. Motivating the generation & contribution of external knowledge 2. Integrating those sources with firm resources & capabilities 3. Diversifying the exploitation of IP resources	1. Provide intrinsic rewards (e.g. recognition) and structure (instrumentality) for contributions. 2. As above. 3. Share or give away IP to maximize returns from entire innovation portfolio.

Table 1: Models of Innovation and Resulting Managerial Issues

Project	Product Category	Approach
Apache	web server	shared R&D
Darwin	operating system	selling complements
Eclipse	programming environment	spinout
Jikes	Java compiler	spinout
Linux	operating system	shared R&D
Mozilla	web browser	shared R&D, spinout
MySQL	database	selling complements
OpenOffice	business productivity	selling complements
Sendmail	mail router	selling complements

Table 2: Open source projects as examples of open innovation

Category	Companies	Motivation
Computer systems vendor	Dell Fujitsu **Hitachi** **HP** **IBM** **NEC** Sun	These vendors spent the late 1980s and 1990s fighting the "Unix wars" with mutually incompatible Unix implementations for their workstations and servers. In the late 1990s, they began shifting resources from their proprietary Unix implementations towards adapting and extended a shared implementation of Linux.
Telecommunications vendor	Alcatel Cisco Ericsson NEC Nokia NTT Toshiba	These vendors used Unix to run their switching systems but began shifting to Linux. As with systems vendors, interested in assuring that Linux evolved to work with their respective hardware and customers.
Microprocessor producer	AMD **Intel** Transmeta	Makers of Intel-compatible processors wanted to speed the shift of enterprise applications from proprietary RISC processors to their commodity processors.
Linux distributor (server and desktop)	Miracle Linux NEC Soft Novell Red Hat SuSE Turbolinux	Distributors have a clear interest both in free riding off the work of others in developing Linux, and making sure that the software met the specific needs of their customers.
Embedded Linux distributor	LynuxWorks MontaVista TimeSys Wind River	Similar to motivations of desktop and server Linux distributors, but need to support more heterogeneous customer needs for use with custom system configurations.
Linux support company	VA Software Linuxcare LynuxWorks	Without development capabilities, the firms both want to leverage the work of others and understand how it met customer needs.
Software developers	Computer Associates Trolltech	Want to make the operating system more reliable for running their specific applications and libraries.

Founding member in **bold**
Source: "OSDL Members," OSDL and company websites (as of May 2004)

Table 3: Members of the Open Source Development Labs

Open Source Strategy	**Maximizing Returns of Internal Innovation**	**Role of External Innovation**	**Motivating External Innovation**
Pooled R&D	Participants jointly contribute to shared effort	Pooled contributions available to all	Ongoing institutions establish legitimacy and continuity
Spinouts	Seed non-commercial technology to support other goals	Supplants internal innovation as basis of ongoing innovation	Free access to valuable technology
Selling Complements	Target highest value part of whole product solution	External components provide basis for internal development	Firms coordinate ongoing supply of components
Donated Complements	Provide an extensible platform for external contributors	Adding variety and novelty to established products	Recognition and other non-monetary rewards

Table 4: Open source strategies as solutions to open innovation challenges