Open for business: emerging business models in open source software

Jon Perr

JP Consulting,
Portland, OR, USA
E-mail: jon@perrspectives.com

Melissa M. Appleyard*

School of Business Administration,
Portland State University,
P.O. Box 751, Portland,
OR 97207-0751, USA
Fax: 503-725-5850
E-mail: MelissaA@sba.pdx.edu
*Corresponding author

Patrick Sullivan

Lewis and Clark Law School,
Portland, OR, USA
E-mail: patrick@alumni.reed.edu

Abstract: Open source software (OSS) has come of age, and a number of maturing business models allow OSS companies to make a profit even when their product is distributed for free. This article considers the dynamics of value creation fuelling the proliferation of OSS and examines the business model factors that enable value capture. After interviewing leaders from over 20 OSS firms and organisations through early 2006, we found that three factors were consistently important in defining a vendor's adoption of a given business model: software licence choice, which takes into account intellectual property ownership; management of developer communities; and the unique features of the markets and product categories in which the vendor participates. Considering these factors, we characterise seven business models. One striking finding is that it is rare to find business-model purity. The majority of firms in our sample are pursuing either blended business models or multiple models simultaneously.

Keywords: open source software; OSS; business models; value creation; value capture; community management.

Biographical notes: Jon Perr is a Consultant bringing over two decades of marketing, communications and product strategy expertise to firms in the internet, mobile applications, open source, productivity software and other technology markets. Previously, he served as Vice President of Marketing at Boston-based Ximian, where he directed marketing programmes, product management, corporate communications and customer support, playing a central role in the company's later acquisition by Novell. His executive positions also have included roles as Vice President of Customer Experience and Marketing at Vendavo, Intellisync and FileMaker (previously Claris Corporation).

Melissa M. Appleyard is an Ames Professor in the Management of Innovation and Technology in the School of Business Administration at Portland State University. Her research focuses on knowledge creation and diffusion in technology-intensive industries. She has been affiliated with Oregon's Lab2Market NSF Initiative at PSU and the Sloan Competitive Semiconductor Manufacturing Center at U.C. Berkeley. Her work has been published in leading academic journals, and she has written or supervised over 15 business cases for MBA and executive courses. She received her BA in Economics/International Area Studies from UCLA and her PhD in Economics from U.C. Berkeley.

Patrick Sullivan has worked for more than ten years in software development, marketing and sales at a range of companies from funded startups to Intel's software marketing group. As founder of lov.li, he created an international e-commerce market for art and crafts, and brought new-to-the-world domain name products to market as CTO of Nameboy.com. He is a Printmaker and Photographer. He received his BA in Political Science from Reed College and his MBA from Portland State University.

1 Introduction

Myths and misconceptions abound regarding the long-term viability of open source software (OSS) organisations. Their software products are developed and licensed under terms that allow for the use, reproduction, modification and reproduction of the software code. Essential to OSS is access to source code, which enables others to create derived works from the original code. According to traditional business strategy, OSS presents a paradox: How can OSS organisations capture the economic value that they create if the building blocks of their products are not only transparent but can be used by competitors [Chesbrough, (2006), p.2; Chesbrough and Appleyard, (2007), p.59]? We sought to understand this paradox by first considering the dynamics driving the economic value created by OSS adoption, and then determining the factors that make up successful business models that guide value capture. We found that while the software source code may be open to competitors and enhanced by developer communities not employed by the OSS organisation in question (Lerner and Tirole, 2002), the open source sector of the software industry is in fact witnessing the emergence of a number of viable business models (Bonaccorsi et al., 2006; Chesbrough, 2006; Golden, 2005; Perens, 2005; Raymond, 1999; West and Gallagher, 2006).

For the purposes of this research, we define 'business model' as not merely the method(s) by which companies derive their revenue, but also as the accompanying set of business processes and organisational arrangements required to realise the revenue model. Business models allow organisations to capture the economic value associated with their product or service. A novel aspect of OSS business models is that they include a role for a community of software developers (Benkler, 2002; Dahlander and Magnusson, 2005; Hertel et al., 2003; O'Mahony, 2003; von Krogh et al., 2003). Our definition of an OSS community is as follows: the network or 'ecosystem' of participants in the creation of the technology and its accompanying intellectual property (IP). In contrast to the OSS setting, software firms that pursue business models involving proprietary software code employ most or all of the computer programmers and testers. This 'closed' development process is generally limited to a firm's own employees and those of a narrow set of partners and customers. In the OSS setting, a software vendor may be only one of many participants, with dozens, hundreds and even thousands of other developers who may be individual enthusiasts or employed by other software firms, hardware companies, universities, or even governments. In some cases, as with the OSS applications Linux, Mozilla and Apache, the community predates the vendors trying to make money from the software. In other cases, the company came first and nurtured/developed a community as an essential part of its business model.

In order to identify the primary factors of profitable OSS business models, we employed an exploratory case study analysis based on interviews of OSS leaders (Yin, 2003). In late 2005 and early 2006, we interviewed nearly two dozen OSS executives across a range of categories and markets to probe their organisations' strategies and business models. It is clear that OSS firms are transforming the software value proposition, while at the same time borrowing approaches to value capture from proprietary companies. Increasingly, the OSS companies are not only leveraging existing open source communities, but are fostering their own for new products and services.

Of the business model factors that were consistent with value capture, we identified three factors that seemed particularly important: software licence selection, which depends on IP ownership strategies; community management; and the ability to craft a business model that is appropriate for the targeted markets and product categories. Considering OSS dynamics of value creation and these business model factors, we were able to identify seven prevalent business models: professional services and consulting, support, subscription, dual licence, hybrid with proprietary extensions, device, and community source. While the business models still are evolving, many OSS companies have matured to the point of profitability.

With the emergence of profitable OSS companies, some of the leaders of their proprietary software counterparts have called into question how OSS business models operate. In 2001, Microsoft CEO Steve Ballmer proclaimed, "Linux is a cancer that attaches itself in an intellectual property sense to everything it touches" (Greene, 2001). Shai Agassi, President of the Product Technology Group at SAP warned, "Intellectual property socialism is the worst that can happen to any IP-based society" (Marson, 2005). Microsoft Chairman Bill Gates concurred, declaring, "there are some new modern-day sort of communists who want to get rid of the incentive" for software makers to innovate – and profit (Marson, 2005).

The reality of OSS (and the mushrooming number of companies banking on it) presents a far different picture. Sales of software and hardware utilising Linux, the open source operating system pioneered by Finnish student Linus Torvalds in 1991, will reach $38 billion by 2008 according to IDC (2004). By 2005, venture capital investments in the OSS arena, topped $400 million (Lacy, 2005). By December 2005, the leading open source vendor, Red Hat, saw its fiscal third quarter revenues climb by 44% to $73 million, driving 114% profit growth and a doubling of its stock price (Red Hat, Inc., 2005a). At the same time, open source technologies have captured significant market share, with Linux now running up to a quarter of server computers (IDC, 2004), the Apache web server powering two-thirds of websites (Netcraft, 2006), and by 2006, Firefox was in use by roughly 10% of web browser users worldwide (Marson, 2006).

While the proliferation of OSS applications may erode the profit positions of competing proprietary software companies, we found that the OSS leaders were not obsessed by this possibility, but rather cared about developing business models that would sustain their enterprises in the presence of open innovation (Chesbrough, 2003). To contribute to the understanding of OSS business models, this paper proceeds as follows. In the next section, we examine the dynamics of the software marketplace for value creation that underlie the OSS business models and motivated our investigation. We then present the methodology followed for this study. In the subsequent section, we identify the specific factors that distinguish the business models. This is followed by descriptions of the seven models that we have identified. We then contrast the OSS business models with characteristics of proprietary models. In the final section, we summarise our findings and suggest paths for future research.

2 The dynamics of value creation underlying OSS proliferation

To understand the gamut of open source business models, it is important first to understand the rationale for its proliferation. The underlying forces can be classified as market pull and technology push,[1] and they constitute the primary ways in which OSS creates value in the market and for society as a whole. As we examine below, OSS has been pulled into the marketplace because of demand for: reduced development costs (Haefliger et al., 2008), enhanced product margins, and technical superiority (von Krogh, 2003). OSS also has been pushed into the marketplace by organisations in order to: capture customers, disrupt markets, and achieve social and economic development goals. These pull and push processes are dynamic in nature and can be reinforcing. For example, an OSS firm may liberally distribute its software code to capture customers who are seeking technically superior software products. In this example, economic value is created because a broad user-base is able to gain access to the software, and because the software is high-quality.

2.1 Market pull

Firms have, in effect, pulled OSS into existence, because it can reduce development costs and improve product margins. Successful open source projects typically have reduced time-to-market and have enhanced the breadth of innovation. One example is Mazu

Networks, a manufacturer of hardware network security devices. Mazu not only avoids licensing costs associated with the use of proprietary operating systems and tools but also is able to access the source code, which opens up further business opportunities. By using Linux (an OSS operating system) and the open source Click Router software, Mazu can substantially improve its margins and customise its line of network security devices.

'Technical superiority' is another motivator for OSS adoption. OSS projects may offer clear technical advantages for a specific market or customer need, including increased performance, reliability, scalability and flexibility. For example, Sleepycat is a vendor of embedded database products based on the open source Berkeley DB started in the early 1990s by U.C. Berkeley graduate students Margo Seltzer and Keith Bostic. Sleepycat stresses the performance, reliability and scalability of the Berkeley DB engine required by network infrastructure vendors (e.g., Cisco), phone (e.g., Nokia), and storage vendors (e.g., EMC) and web applications vendors (e.g., Google). For Mazu Networks, the flexibility provided by access to Linux's source code coupled with the founders' extensive experience both with Linux and the Click Modular Router project were central in the development of their security devices.[2]

2.2   Technology push

'Customer capture' is the most obvious motivation for OSS organisations to push the liberal distribution of their software. By providing free downloads, companies like the early Red Hat, JBoss and Canonical rapidly build large user bases, which they can 'monetise' in the future. JBoss, in the words of Vice President of Product Management Shaun Connolly, uses "an unthrottled internet distribution model" for precisely this purpose.[3]

An early example arose in 1998. Netscape famously announced it would freely distribute and provide source code access to its communicator product to counter Microsoft's bundling of Internet Explorer with the Windows operating system. What later became the Mozilla project and the Firefox browser generated a global community with thousands of developers and millions of users.

OSS can enable market disruption in established markets, changing the competitive dynamics and altering sales and marketing models. For example, open source ERP vendor Compiere enables current and potential customers to evaluate and iterate on solutions based on its OSS application. This short-circuits long RFP cycles and high-risk, multi-year commitments to established proprietary vendors such as SAP and Oracle.[4] Similarly disruptive, SugarCRM is targeting small and medium enterprises (SMEs) which are down-market for the establish players such as Oracle and Siebel. Its sales and marketing model relies on customer trials of its free, open source download version.[5]

Some OSS organisations push adoption of their software to realise social and economic development goals associated with pervasive computing. Red Hat has repeatedly stated its philosophical commitment to offering only General Public License (GPL) software. Serial Entrepreneur Mark Shuttleworth, Founder of Canonical and the Ubuntu Linux operating systems it distributes, wants to help bridge the global digital divide by offering a complete desktop Linux environment that is ''universally available, absolutely free.''[6] Canonical derives revenue from support and customisation contracts, showing that in OSS, you can do well by doing good.

On this last point, it is worth noting the role and growing influence of governments and regional institutions in encouraging the adoption of OSS to foster domestic

technology industries, as well as the deployment of OSS infrastructure for the cost-effective delivery of public services. This is reflected in the common Asianux Linux platform in China, Japan and South Korea; 'People's PCs' in India and Thailand; and open source-based government services in Spain and Brazil.

3 Methodology

The pull and push dynamics reflect value creation through OSS adoption, and this study examines the attributes of business models consistent with capturing the created value in the context of an open process of innovation [Chesbrough, (2006), p.43]. The adoption dynamics informed an interview guide that we developed to identify specific business models factors. By conducting semi-structured interviews [Ghauri and Grønhaug, (2005), p.123], we engaged in an exploratory, multiple-case approach in order to understand the pertinent business model factors [Yin, (2003), p.5].

The interviewed guide contained roughly 30 questions covering company background, product and technology strategy, and OSS business models (including revenue methods, licensing, distribution channels, ecosystem management, etc.). We conducted the interviews from December 2005 to February 2006. The individuals interviewed were all members of the executive management team (in one case, a board member). Some were founders, and most were either the CEO, chief technology office, vice-president of marketing or vice-president of engineering.

Over 20 OSS organisations were specifically chosen for the interviews, because they represent a diversity of technology markets and sectors. These included different markets (consumer, small business, enterprise) and product categories (database, middleware, network/system management, ERP, CRM, etc.). The companies selected also were generally high profile, early leaders in their market segments, and many offered an OSS alternative to existing proprietary vendors (e.g., database – MySQL, web servers/applications – Jboss, ERP – Compiere, CRM – SugarCRM). A number of the organisations in our sample had cycled through several business models, and we felt that by analysing their current model, we could gain insight into business models that were on their way to being sustainable in terms of ensuring longevity of the organisation through a positive profit position. Most of the organisations were for-profit firms, and their target customers were generally other businesses rather than end-consumers. In the sample, both horizontal (providing similar software functionality to users in different industries) and vertical (providing software products for users along a specific industry's supply chain) software companies are represented.

To determine the business model factors that appeared most responsible for success in value capture, we compiled the notes from all of the interviews and looked for patterns. Three factors rose to the top in terms of:

a how many times they were mentioned

b the emphasis place on them by the organisations that either were profitable or rapidly approaching profitability (for the not-for-profit organisations in our sample, we considered longevity of the organisation as in indicator of success).

We discuss the three factors in the next section.

4   Business model factors

The compiled interviews revealed patterns that allowed us to identify the principal factors that distinguish OSS business models in their ability to capture value. The three factors that appear to be particularly salient are strategies for dealing with IP and the associated choice of software licence; the role of the community; and the choice of market or product category. We provide details about each of the business model factors below.

4.1   IP ownership and the licence choice

While detractors to the rise of OSS use terms like 'infect', 'contaminate', 'viral' and 'cancer' to foster the misconception that OSS undermines a company's profit potential and its underlying IP, in practice, a range of open source licenses provide software developers with a host of approaches to value capture (Lerner and Tirole, 2005). Control over the code through IP rights like copyrights helps to determine which licenses can be selected to govern the distribution of the software (O'Mahony, 2003; Välimäki, 2005). The selection of the software licence in turn influences which business models can be pursued.

By way of background, a software licence is the set of legal terms governing the use, reproduction and redistribution of computer software. Generally, a proprietary software licence allows the user the right to use one copy of the software and prohibits its reproduction and redistribution. Access to the source code, which is the underlying instruction set, is typically not provided. In contrast, OSS licenses usually afford the user the 'four freedoms' to use, copy, modify and redistribute the software (whether for free or for a fee).[7] Users are given access to the source code, which they can modify. While there are numerous OSS licenses, the GNU GPL is the most common and is best known for its 'give-back' principle. That is, software licensed under the GPL confers the four freedoms in addition to the give-back principle, which requires that if the software is redistributed, the source code – along with any modifications made to it –must also be made available. These rights differentiate GPL software from traditional proprietary products from vendors such as Microsoft. They also make it virtually impossible for OSS companies to sell software licenses in the traditional sense.

But while the GPL accounts for roughly 70% of the licenses used by projects in the leading OSS repository SourceForge (Cohen, 2005), there are other licenses without the same source code sharing proviso, which may offer additional opportunities for value capture. The Lesser General Public License (LGPL) is designed to allow other developers or independent software vendors (ISVs) to incorporate open source libraries into their own application code licensed under other, often proprietary, terms. The Apache Public License and the Berkeley Software Distribution (BSD) licence describe when developers of derivative works may, as opposed to must, make source code available with the software they distribute. The Open Source Initiative (OSI) website recognises over 50 licenses, including the Mozilla Public License governing the Firefox browser and other Mozilla project-based technologies (Nelson, 2006).[8]

As analysed below, licence choice plays a major role in defining the options OSS companies have in creating viable business models and revenue streams. A separate but related issue is IP ownership including copyright assignment. Fundamentally, companies

that retain (or at least share) the copyright assignments for the open source code they build on are able to pursue:

a   dual licensing for commercial versions of their software

b   the distribution of product extensions that are covered by proprietary licenses.

4.2   *The role of the community*

Another misconception regarding OSS-based businesses is the nature of the communities driving the strategy, development and testing of OSS products like Linux, Apache, or Mozilla. As Matt Tucker, Co-founder and VP of Engineering at Jive Software noted, "Open source has a reputation that there is a huge collection of people, all chipping in a little bit, to get stuff done. That's almost never true."[9] Instead, a wide array of community models exists and increasingly includes vendors who provide much of the core development (if not testing and documentation) staff for open source projects.[10] This means that the companies exert control over the software code, which allows them to pursue a variety of business models.

Even in the largest global communities, many key developers are now employed by leading software, hardware, and services companies like Red Hat, Novell, IBM, HP, Sun, and Nokia. Far from being 'parasites' (Foley, 2001), these vendors often contribute resources and dollars to non-profit 'dot orgs' and foundations driving the projects like the Linux kernel, the Apache web server, the Eclipse development tools, and the GNOME desktop.[11]

Many vendors are now going beyond leveraging the work of open source enthusiasts in existing communities to drive the funding and development of projects in new vertical markets or 'mission-critical' enterprise applications. For example, SugarCRM does virtually all of the development on the core open source SugarCRM application. A healthy ecosystem of communities, however, building SugarCRM add-ons and extensions for the application can be found at SourceForge. By the end of 2005, at JBoss, an open source enterprise middleware provider, roughly 55% of code 'committers' were JBoss employees within a global community of approximately 700 contributors. They were responsible for producing roughly 85% to 90% of the code. The hiring of its most prolific community developers and a focus on software stability through central control was key to what the company calls its JBoss Professional Open Source business model.[12] SleepyCat and Compiere similarly produce virtually all of the code in their open source applications, and importantly, own or share the copyrights for all of it. All of these vendors do have policies regarding the sharing of copyrights with, or even full assignment to, external contributors.

4.3   *Market and categories*

Closely coupled with misconceptions about the nature of OSS communities are assumptions regarding targets markets, users, and customers. OSS applications have extended far beyond operating systems such as Linux into both horizontal and vertical marketplaces, and this extension has contributed to the rise of additional OSS business models. As a result, while some OSS enjoys the support of large global communities of enthusiast, end-user developers, newer vertical market offerings often require the subject matter expertise and engineering resources that only the vendor can provide.

We found that value capture was enhanced when choices regarding IP management, licenses, and community management fit with the characteristics of the market pursued. For example, a horizontal market focus would permit optimisation of specific features and the development costs could be spread over multiple industries. Horizontal OSS players would be expected to favour broad distribution of their products and thus favour the GPL and broad community support. In contrast, concentrating on a vertical strategy would allow code optimisation for seamless handoffs and integration of systems along supply chains. This capability would benefit from control over the code and hence IP ownership, which would tend to suggest a narrower community of contributors.

Even with the relatively small market share of the Linux desktop [still less than 5%, according to IDC (2004)], vendors across horizontal markets such as Mandriva, Linspire, Xandros (all Linux OS providers) and Codeweavers continue to make gains. In part due to maturation of associated applications, such as the OpenOffice productivity suite, the Evolution groupware client, and the Firefox browser, as well as its vendors, Linux has surpassed the Macintosh as the number two desktop platform worldwide in terms of new units deployed (IDC, 2004).

Another wave of open source applications has emerged with OSS startups supporting the applications for the enterprise and key vertical markets. These include areas such as customer relationship management (SugarCRM), enterprise resource planning (Compiere ERP), content management (Alfresco), and databases (MySQL).

OSS has started to populate an increasing number of product categories. For example, OSS infrastructure product offerings have multiplied, as many observers had predicted. Beyond the Linux operating system vendors (such as Red Hat, Novell) and core technologies (Covalent for Apache, Zend for PHP), the marketplace has seen as explosion of companies offering OSS 'stacks', testing and certification for application servers (JBoss), web services (SpikeSource, SourceLabs), and development and deployment tools (Innoopract, ActiveGrid).

4.4   OSS business models

By weighing the dynamics of value creation and the tradeoffs inherent in each of the three business model factors, OSS organisations have devised their business models. All of these dimensions – underlying value creation dynamics, IP ownership/licence choice, community management, target market/product categories – affect the options they have available in honing their business models. As developed below, we characterise seven business models. In practice, there was very little business model purity with the majority of the organisations in our sample pursing multiple models simultaneously or blending the models. Few vendors are 'all one thing', observed Dave Hersh, CEO of Jive Software. He noted, "We use every business model."[13]

In many cases, the vendors capture value not in the underlying software they provide, but in the technical support, professional services, automated software maintenance and updates, even the proprietary add-ons they provide. As presented in Table 1, we characterise the business models as professional services and consulting, support, subscription, dual licence, hybrid with proprietary extensions, device, and community source often involving consortia.[14] We describe each model in detail below.

Table 1 OSS business models in action

Model	Description	Licensing and IP issues	Community profile	Target markets	Examples
Professional services/consulting	Revenue derived from professional services, training, consulting or customisation of OSS.	• Typically GPL. • LGPL, or other (Apache, Mozilla) for derived works. • Lack of IP ownership, copyright assignment.	• Broad. • Not controlled by single vendor.	• Often horizontal.	• Early (pre-2002) Red Hat • IBM • Certified training (JBoss, Compiere, MySQl, etc.)
Support	Revenue derived from sale of customer support contracts.	• Typically GPL. • LGPL, or other (Apache, Mozilla) for derived works. • Lack of IP ownership, copyright assignment.	• Broad. • Not controlled by single vendor.	• Often horizontal.	• JBoss • Compiere
Subscription	Revenue derived from annual service agreements bundling OSS, customer support and certified software updates delivered via the internet.	• Typically GPL. • LGPL, or other (Apache, Mozilla) for derived works. • Lack of IP ownership, copyright assignment.	• Broad. • Not controlled by single vendor.	• Often horizontal.	• Current Red Hat (Red Hat Enterprise Linux) • SpikeSource • Innoopract • JBoss
Dual licence	Vendor licenses software under different licenses (free 'public' or 'community' licence vs. paid 'commercial' licence) based on customer intent to redistribute.	• Public/community licence designed for derived works. • Fee-based commercial licence tied to redistribution. • Vendor owns IP and has exclusive or shared copyright assignment for core product.	• Generally narrow. • Core open source project is largely controlled by single vendor.	• Some vertical, enterprise or embedded applications.	• MySQL • Sleepycat

Table 1   OSS business models in action (continued)

Model	Description	Licensing and IP issues	Community profile	Target markets	Examples
Hybrid/proprietary extensions	Firms broadly proliferate open source application and monetise through sale of proprietary versions or product line extensions. Variants include mixed open source/proprietary technologies or services with free trial or 'community' versions.	• Base offering (often free download) typically GPL, or other for derived works. • 'Professional', 'enterprise' or other supported version sold under proprietary licence. • Vendor owns IP and has exclusive or shared copyright assignment for core product.	• Generally narrow, but high degree of reliance on community for proliferation of base offering. • Core open source project usually controlled by single vendor.	• Some vertical or enterprise applications.	• SugarCRM • ActiveGrid • Ximian (Connector) • Codeweavers • Zend • Sendmail • Scalix • Black Duck Software
Device	Vendor sells and supports hardware device or appliance incorporating OSS.	• Typically GPL • LGPL or other (Apache, Mozilla) for derived works. • Lack of IP ownership, copyright assignment.	• Broad or narrow. • Not controlled by single vendor.	• May combine horizontal and vertical technology.	• Mazu Networks
Community source/consortia	Consortia of end-user organisations or institutions jointly develops application to be used by all.	• Typically GPL. • LGPL or other (Apache, Mozilla) for derived works. • Lack of IP ownership, copyright assignment.	• Community may be broad or narrow.	• Often targeting vertical, enterprise or 'backoffice' application.	• Kuali and Sakai projects (university collaborative education, financial management) • Vendor - rSmart

4.4.1 Professional services and consulting

Among the earliest open source business models were models focused on professional services and consulting. From training to consulting engagements around customisation and implementation, vendors sought to help customers trim the costs associated with open source deployment and maintenance. Customers were motivated to pull open source solutions into their software infrastructure because of the potential for technical superiority and decreased development costs, but some of the code was immature or not tailored for their use. Service providers smooth out the rough edges of OSS, facilitating the disruption of established software markets. These companies promote the wide distribution of OSS to increase customer capture and scale up their service offerings. In terms of market focus, the professional services and consulting business model tends to be horizontal so that the services can scale across industries.

Training programmes targeting end customers, systems integrators, and other developers are a staple of companies such as MySQL, JBoss, Red Hat, Compiere and many more. One of the service pioneers was Cygnus, acquired by Red Hat in 1999. Its mission was famously described by its founder John Gilmore, "Cygnus makes free software affordable" (Olson, 2005). By 2001, services consistently made up about half of Red Hat revenue, which had reached a plateau of roughly $20–25 million in quarterly revenue. It was only with the later adoption of its Red Hat Enterprise Linux (RHEL) model that Red Hat saw its total revenue and mix change markedly for the better (Red Hat Inc., 2005b).

Perhaps the highest profile open source services provider is IBM Global Services. Like HP, Dell and others, IBM recognises the importance of Linux in driving server sales. Just as important, IBM looks to OSS as the means to capture long-term customer relationships based on service and support, even when non-IBM hardware and software are involved. As IBM's Irving Wladawsky-Berger put it in 2000 (IBM, 2000):

> "[Linux] alters the way our industry delivers value to its customers, which is very good news for IBM… When applications are no longer lashed to a specific operating platform, control and choice shift away from the technology company, and into the hands of the customer. This makes possible an equally seismic shift in how value is delivered – through services, through middleware, through servers."

In December 2000, IBM announced a $1 billion investment in Linux development and services (Evans, 2000). By 2004, IBM sales of servers running Linux reached $2 billion, and at the time, services revenue was expected to exceed its hardware sales, which in fact happened by 2005 (Shankland, 2004; Lohr, 2005).

4.4.2 Support

The majority of OSS companies surveyed for this article derived some portion of their revenue from the provision of customer support. Whether delivered via end-user online knowledge bases, e-mail, and phone support or through enterprise help desks, most companies drove revenue through the sales of support services.

With regards to the business model factors, companies pursuing the support business model typically rely on broad community support for software development and do not focus on securing IP rights. While employing some programmers in-house, external communities are relied on to ensure the technical superiority of the software code.

Similar to the professional services model, support providers tend to promote GPL licence choices and typically have their sites on horizontal markets.

Compiere, for example, lets users download the open source ERP application for free and then sells annual support contracts to paying customers. As noted above, Compiere's business model has disrupted the business models of proprietary ERP vendors like SAP that traditionally have relied on software sales in addition to support contracts. Red Hat continues to develop a free, rapidly changing 'community' version of its Linux operating system called 'Fedora'.[15] For its business customers, however, Red Hat sells annual support agreements as part of its RHEL offerings for server and desktop systems.

Enterprise middleware provider JBoss also is illustrative of the support model. Best known for its open source JBoss application server, JBoss competes against proprietary vendors such as BEA and IBM. While hundreds of thousands of users freely download its JBoss Enterprise Management Suite (JEMS), JBoss generates revenue from support agreements. As JBoss's Connolly put it, "Our competitive differentiation is high-quality service."[16] By 2005, JBoss employed over 150 people, had received venture funding from Matrix Partners, Accel Partners, Intel Capital and Bain Capital, and was profitable on a cash flow basis.

4.4.3 Subscription

For most of these vendors, however, customer support is increasingly offered not through stand-alone contracts, but through annual subscriptions where service agreements bundle OSS, technical support, automated software updates, and in some cases, platform certification.

These subscription models leverage one of the unique attributes of Linux and open source development – a decentralised development model that rapidly offers new features, bug fixes and security enhancements across hundreds of shared software packages to improve technical superiority. This means again that drawing on a broad developer community is desirable, which is consistent with the use of the GPL and LGPL. Also similar to the cases of the services and support business models, companies pursuing the subscription business model often focus on horizontal markets.

While customers benefit from the continuous availability of new fixes and functionality, integrating, testing and deploying packages from disparate sources can quickly become an IT manager's nightmare. As a result, OSS vendors increasingly offer web-based services to automate the notification, installation and deployment of new packages directly to customers' systems.

The case of Red Hat, the leading Linux vendor in North America and perhaps the most recognisable brand in OSS, illustrates the migration to a subscription business model. Prior to 2002, Red Hat built on its 1999 acquisition of Cygnus to derive almost half of its $79 million revenue from services, including support, consulting and custom development (Red Hat Inc., 2002). In FY 2003, however, Red Hat moved to a new subscription model based on its RHEL offerings. With RHEL, customers purchase annual 'per system' service agreements that provide them with GPL Linux software, technical support, and software updates delivered via the web-based Red Hat Network. For enterprise customers considering proprietary UNIX or Windows server systems, the Red Hat price points ranging from $349 to $2,499 were quite competitive at the time of our interviews. By FY 2005, subscription sales accounted for $151 million, or 77%, of Red Hat's $196 million in revenue (Red Hat Inc., 2005b).

Other vendors have followed suit. SpikeSource, a startup led by Marimba Founder Kim Polese, provides open source IT services and software certification to enterprise customers. SpikeSource offers business ready 'LAMP' (Linux, Apache, MySQL, Perl/Python/PHP) stacks for the enterprise. The LAMP stacks are free for download and SpikeSource offers annual subscriptions providing technical support with certified software updates delivered via 'SpikeNet'. The company derives its revenue primarily from these subscriptions. As Spikesource Co-founder and CTO Murugan Pal explained, "Think of SpikeSource as the Red Hat for everything other than the OS."[17]

Innoopract's Managing Director Eric von der Heyden uses a similar analogy to describe his company, which offers visual web development tools, services and distributions based on the open source Eclipse project. "Innoopract offers distribution services for Eclipse similar to Red Hat for Linux," von der Heyden explained.[18] During the time period of our interviews, though, Innoopract's business still depended more on professional services and consulting than on annual subscription sales.

JBoss, too, monetised its huge base of free download users (8.5 million via SourceForge) through subscription sales. In 2005, subscriptions constituted 70% of its revenue, compared to 20% for certified training and 10% for short-term consulting engagements.[19] Akin to Red Hat and SpikeSource, JBoss subscriptions include the JBoss Network, which provides management tools and delivery of software patches and updates to customers.

4.4.4 *Dual licensing models*

In contrast to the models above, IP ownership through shared or exclusive copyright assignment is pivotal in another, less common open source business model: dual licensing. Under a dual licensing model, a vendor may offer versions of software using different licenses depending on the class of user or his/her intent to redistribute the software or create derivative works. Typically, a dual licence business model involves a free 'community edition' licensed under the GPL coupled with the offering of other commercial editions under a proprietary or commercial licence.

One such example is MySQL, which builds on the open source MySQL project and has emerged as an alternative to Oracle and other proprietary vendors for mission-critical enterprise database applications and web services. A GPL community edition is available for free download, including the latest (sometimes-bleeding edge) features under development. The commercial MySQL Pro Server, however, is licensed to those who need to redistribute applications without the give-back implications of the GPL. It also includes the certified, supported MySQL version, data access drivers and tools for database administrators (DBAs), with updates provided by the MySQL Network. Interestingly, MySQL also couples a subscription model with its dual licensing approach, with prices ranging from $595 to $4,995 per server per year at the time of our interviews, depending on support levels and other services.

One of the innovators of the dual licensing model was Sleepycat, the embedded database provider. Its transactional storage system is designed for developers, ISVs and service providers building applications where performance, scalability and 'five 9's' reliability are essential. The Sleepycat dual licence model hinges on redistribution. For OSS projects or applications not distributed to third parties, the company offers the OSI-certified Sleepycat Public License (SPL). The Sleepycat Commercial License (SCL), however, permits proprietary application redistribution. The SCL is sold as a traditional

licence with annual maintenance fees similar to the approach used by many proprietary firms, and it constituted roughly 70% of Sleepycat's revenue at the end of 2005.[20]

The key in dual licensing models is, as Augustin notes, "Who owns the code?"[21] For vendors looking to dual licensing business models, Sleepycat's Olson cautions that "the stars have to line up. You need IP ownership, technical leverage and a clear choice between varieties of pain" (Olson, 2005). Customers need to choose their type of 'pain'. That is, if they want support and/or the ability to redistribute their application, they will need to pay for the commercial version. But some customers may be content to use the free or 'community' licence, recognising that they have to provide support themselves.

4.4.5 The hybrids: proprietary extensions to open source

Where the dual licensing vendors are focused on the user's intent to redistribute any subsequent applications that are developed, the hybrid vendors sell either proprietary extensions to an open source application or proprietary versions of the application with additional functionality. In the hybrid business model, OSS is the essential ingredient to what Navica CEO Bernard Golden calls a 'razor and razor blades' model.[22] Here, vendors seek to capture large user bases by broadly proliferating free OSS applications (the razors). They then sell proprietary add-on products or optimised versions (the blades). In these hybrid models, vendors derive their revenue from the sale of proprietary software. Companies pursuing hybrid models are highly dependent on both the level of control over the associated IP and the external developer community to improve the focal software. IP ownership over the add-ons generally plays a central role in these revenue streams. By specialising in certain industries, the companies adopting hybrid business models can optimise their software along the supply chain, although a horizontal market focus also can be pursued.

An increasingly common opportunity for monetisation is at the nexus where open source and proprietary software meet. For example, desktop Linux provider Ximian offered its standards-based Evolution e-mail and groupware client (the analogue of Microsoft Outlook for Linux systems) for free under the GPL. For enterprise customers integrating Linux desktops into Microsoft Exchange collaboration environments, Ximian sold a separate 'Exchange Connector' under a proprietary licence. The company largely avoided a backlash from its GNOME developer community, because community members generally did not work in Exchange environments.[23] Ximian consciously focused on proprietary add-ons that would not be a priority of open source developers in order to ensure a healthy coexistence with the community.

SugarCRM may be the highest profile hybrid vendor at this time. Users can freely download the unsupported SugarCRM application. The company then sells two separate, supported products for individuals ('Professional Edition') and enterprise customers ('Enterprise Edition'). Each edition contains substantial functionality not found in the OSS application, a strategy that SugarCRM Board Member Larry Augustin acknowledged requires 'a balancing act' for customer acceptance.[24]

Central to both the Ximian and SugarCRM strategies, however, is the company's IP ownership and control of the source code. ActiveGrid, which provides tools and a platform to deliver service-oriented web applications, similarly offers a freely downloadable open source application builder, while selling a commercial server for large-scale deployment.[25]

Codeweavers provides another example of a company selling a proprietary product on top of an OSS base. Its CrossOver Office application lets users run off-the-shelf Windows applications such as Microsoft Office, Quicken and Lotus Notes on Linux PCs or 'thin-client' environments. The company builds on the open source Wine project, which is licensed under the LGPL. Codeweavers derives its revenue from proprietary end-user licenses, service contracts to improve Wine, OEM agreements and non-recurring engineering (NRE) work for other ISVs.[26]

Some of the vendors behind some of the earliest and most widely adopted open source technologies have turned to hybrid open-closed business models. The PHP company Zend is one such example. Another is Sendmail, developer of the e-mail management systems of the same name. The company offers an open source version of its Sendmail MTA e-mail and messaging server software, and then sells proprietary extensions addressing policies, spam, anti-virus and other essential e-mail functionality.

A company leveraging open source Sendmail technology as part of a hybrid model is enterprise e-mail and collaboration vendor Scalix. Using a slightly different approach, Scalix offers both a free download version ('Community Edition') and sells a more feature rich, scalable version ('Enterprise Edition'). Each version is licensed under proprietary terms, but both leverage an open source stack including Linux, Apache, PHP, Sendmail and other core technologies.[27] It is worth noting the expansion of open source offerings by Scalix, such as the ScalixConnect for Novell Evolution, as well as the company's ScalixReady for Open Source programme, which certifies interoperability with a wide range of e-mail-related open source components.

A company targeting open source development, Black Duck Software, also uses a hybrid approach using OSS in their products without adopting a free software business model. Black Duck conceived of its protexIP product line to help enterprise IT developers and ISVs ensure open source IP compliance. This product line assists with software licence management by drawing on a leading source code repository and providing defect-tracking software. In practice, the closed source protexIP platform has been used by proprietary software developers as well because of the concern over 'code contamination' and software asset management.[28]

4.4.6 Device

Companies also are leveraging open source in other business models. The device business model is represented by embedded devices, handhelds and appliances that feature Linux and other OSS. Device vendors can improve margins and extensibility through their utilisation of OSS. In addition to Mazu Networks network security devices mentioned above, Sharp, for example, uses Linux in handheld products. High-profile consumer devices such as Tivo and PlayStation also have adopted open source operating systems to improve product margins.

4.4.7 The rise of community source and end-user consortia

A relatively new phenomenon is the emergence of consortia of enterprises or institutional end-users to deliver shared OSS solutions. Dubbed 'community source', this model allows organisations to dramatically reduce software licence costs by contributing to the joint development of major applications. Community source projects may feature a vendor driving the consortium, positioning itself for market leadership in support,

services, and training as the joint open source application reaches completion. For the consortium vendors, the resource commitments involved in a shared community source project are a fraction of the budgets otherwise allocated to the purchase and maintenance of proprietary systems. This pooled expertise provides what Jason McKerr of the Oregon State University's Open Source Lab has deemed the 'collaborative return on investment'.[29] In many instances, public organisations have been involved, making these efforts consistent with the social goal of deploying a high-quality computing infrastructure at reasonable costs.

One of the most visible community source efforts involves several leading universities, including Michigan, Indiana, MIT, Cornell, Arizona, and Stanford, jointly developing financial management and collaborative learning systems for their campuses. The Kuali and Sakai projects feature multi-million dollar investments from the schools and are being shepherded by Phoenix-based rSmart and others, with growing support from vendors such as IBM, Sun, Apple, and Unisys.

5   Open source vs. closed source business models

One of the most striking aspects of the evolution of OSS business models is how closely it mirrors trends in proprietary software. Over the last several years, proprietary vendors including Microsoft, Lotus, Adobe, have seen a dramatic shift in their revenue composition, as sales of new software licenses decline and must be replaced by installed-base upgrade and maintenance business. Forrester Research estimates that in mature applications markets, maintenance now accounts for up to 50% of overall revenues for the major vendors, a share that is growing (Hamerman and Harrington, 2005).

As IBM's Wladawsky-Berger noted above, Linux and other OSS applications are accelerating the redefinition of value for customers. As Linux consumes the proprietary UNIX market and OSS competes across the gamut of computing, we expect the similarities across proprietary and OSS business models – in particular, services, support, and subscriptions – to increase. For Miko Matsumura of SOA vendor Infravio, OSS enables a customer acquisition strategy based on 'qualification, commoditisation and standardisation'.[30] Tim O'Reilly, founder and CEO of O'Reilly Media describes the shift in different terms:

> "I believe we're actually moving away from software as a locus of value… My premise is that value is now being pushed 'up the stack' to information services provided over the internet (O'Reilly, 2004)."

To make their revenue streams (and share prices) more predictable, proprietary vendors have been moving beyond maintenance models to annual subscriptions or 'software as a service' business model. In 2002, for example, Microsoft introduced its Licensing 6.0 and Software Assurance programme, which required enterprise buyers to make up-front, multi-year commitments in order to lock in favourable upgrade pricing (Galli, 2002). Despite pushback from customers, these sorts of programmes from Microsoft and other proprietary vendors have helped set enterprise-customer expectations regarding the move away from one-time software licenses and towards subscription-based purchasing and internet-based services. This change in customer expectations has helped OSS companies execute their non-traditional business models.

While business models may continue to converge across open source and closed source software companies, the underlying differences in how they execute similar business models may require different managerial skills. As examined above, IP ownership, licence choice and nature of the developer community all influence the options available to OSS companies when crafting their business models. OSS companies may not own all of the IP associated with their products, which in turn determines the range of possible licenses, and they may enlist a broad community of developers although many do now have a critical mass of internal developers. In contrast, by definition, proprietary companies own the IP associated with their products, and typically rely exclusively on company employees for software development.

Table 2 Additional attributes of proprietary vs. OSS business models

Attribute	*Proprietary software*	*OSS*
Software development model	• Closed – within boundaries of vendor and designated partners. • Customers/users may use/see only under non-disclosure.	• Open to external input. • Project definition and pre-releases made available. • Global development via internet-connected community.
Typical software development process	• Product/project idea. • Research/validate idea. • Build prototype, pre-releases. • Announce to world. • Ship final product.	• Product/project idea. • Announce to world to build developer community and customer interest. • Iterative releases of early versions and source code. • Ship and refine.
Sales and marketing process	• Traditional demand generation cycle. • Awareness/lead generation. • Up front investment in sales and marketing programs and staff. • Vendor-guided pilot or trial programmes.	• Customer-self-selection (download free open source application). • Monetise large installed base of free users. • Sales and marketing costs/headcount limited.
Examples	• Microsoft, Adobe, Lotus. Novell	• Red Hat, Novell (SuSE), Jboss, Mozilla, OpenOffice

As shown in Table 2, additional attributes distinguish OSS and proprietary business models including the model for software development, the typical software development process, and the sales and marketing process. They follow different software development processes where user input is typically limited and occurs later in the development cycle for proprietary products. In contrast, the OSS setting is characterised by iterations based on customer and community feedback (Franke and von Hippel, 2003; von Hippel, 2005; von Hippel and von Krogh, 2003), and this input may be sought regularly even after the product is shipped to customers.[31] Proprietary companies typically face much higher sales and marketing costs, needing to invest in traditional demand generation programmes to raise awareness and create leads for what is often a large direct sales force. In contrast, OSS companies typically do not have large sales and marketing staffs or budgets, instead benefiting from customers' products trials and self-selection made possible by freely downloadable OSS. The challenge over time is to

convert these 'self-selected' customers of the free downloads into customers willing to pay for consulting services, support, an enterprise version, a complementary device, etc.

The differences between OSS and proprietary business models may require different managerial skills. For example, the legal staff employed by an OSS firm may need to manage the expectations of external copyright holders and be skilled in negotiating multiparty IP ownership contracts to divide up any proceeds. Another example is during product development. The leader of a product development team in a proprietary company may not need to address 'not-invented-here' biases very often but may need to inspire creativity to ensure the internally developed product is more attractive to the marketplace than an OSS alternative.

6   Discussion, conclusions and extensions

In a common myth of open sources software, a rag-tag army of free software 'hackers', connected around the world via the internet and coding around the clock, strives to make software and its source code available to all. While this may describe some (especially the early) OSS developers, it does not do justice to the firms that are constructing their business models around OSS. For a generation, software users, creators and analysts alike have been well acquainted with proprietary business models. In the past, companies like Microsoft, Lotus, Oracle and Adobe produced revenue through the sale of software licenses and maintenance/upgrade agreements for products whose IP they alone owned and whose development was done almost exclusively by their own employees. But end users, enterprises, governments and large institutions have increasingly come to rely on OSS. OSS vendors are not radicals storming the barricades. On the contrary, the open source ecosystem is creating value through what O'Reilly calls 'the architecture of participation', where 'you actually build value by the amount of cooperation you enable' [O'Reilly, (2004), p.5]. External developer communities have provided the backbone to OSS products like Linux, as well as serving a role – albeit more minor – in the development of commercial versions of software like products sold by MySQL or Sleepycat.

The attractiveness of OSS, which is based on this open process of innovation [Chesbrough, (2006), p.43], is a result of value creation through dynamic processes including reduced development costs, increased product margins, technical superiority, customer capture, market disruption, and the ability to attain social and economic development goals based on an affordable computer infrastructure. By interviewing OSS leaders, we wished to uncover the primary business model factors that are consistent with the capture of this created value.

The three business model factors that emerged as primary considerations for OSS organisations that have enjoyed longevity in the marketplace and in a number of cases, profitability, were: IP ownership and licensing strategies; management of the community; and selection of market segments and product categories that were consistent with the former two factors. By making different choices regarding these factors, OSS organisations have formulated seven different business models: professional services and consulting, support, subscription, dual licence, hybrid with proprietary extensions, device, and community source. Among the over 20 OSS firms and organisations in our sample, we found evidence that each of these business models could be viable. These models are leading to sustainable footholds where the label 'open source' is not

inconsistent with being profitable. As with the rest of the software industry, OSS companies are turning to tried – and increasingly true – business models to maximise profit and shareholder value.

This exploratory research opens up a number of paths for future research. For starters, it would be of interest to understand the magnitude of potential profitability when comparing OSS versus proprietary software business models. It could be productive to develop a theoretical model that links the dynamics of value creation to the ability to capture value through business model factors. The model could include interaction effects between the dynamics of value creation and the business model factors. For example, OSS 'sweet spots' are present for software products that face high development costs, but will garner high profit margins, and are in a market in which a dominant technology will likely emerge. These dynamics will benefit from an OSS licence choice that will lead to customer capture (interaction between the development costs, profit margin, and customer capture dynamics and IP ownership/licence factor). However, the OSS firm would not want to alienate the developer community because it would likely want to achieve technical superiority in order to guarantee that its product will become the *de facto* industry standard (interaction between the technical superiority dynamic and community management factor). If instead the OSS firm is choosing to develop a product for a market crowded with established proprietary software firms with deep pockets, it may face years in court instead of unfettered software development (interaction between the market disruption dynamic and market/product category factor). When an OSS company gives back to the worldwide community of software developers, e.g., through donations to the dot orgs mentioned above that drive applications with broad applicability like the Linux kernel, it can build goodwill that leads to a heightened adoption rate of its products (interaction between the social goals and customer capture dynamics and community management factor). A theoretical model could frame which interactions would be expected to be most potent for sustainable profits in an OSS versus proprietary setting, and then a longitudinal empirical analysis could test the predictions of the model.

In addition to modelling the importance of interaction effects between the value creation dynamics and business model factors, the theoretical framework could provide guidance when assessing the profit potential facing proprietary and OSS firms in the same market. Depending on the market's willingness to pay, proprietary software sales historically may have provided more attractive product margins and hence a larger pool of development funds because the marginal costs associated with replication of a software programme is so low. The marginal costs associated with other revenue streams like support, would likely be much higher, thus possibly hurting potential profitability unless revenues were to rise more than proportionately. Again, the implications of the model could be tested empirically, and it would be of interest to see whether there has been a crossover point where the profit potential is now higher for OSS firms than for proprietary firms as their business models converge.

Another finding that merits systematic empirical analysis is the degree to which 'fit' matters for targeted markets and product categories and the choice of IP ownership, licensing strategy, and approach to community management. While we found evidence that such fit matters for profitability, we could not definitively determine the optimal pairing of factor decisions. An analysis of the fit across business model factors could be complemented by a deeper examination of the accompanying set of business processes and organisational arrangements in an OSS firm's ecosystem. It may be that while the choices regarding business model factors appear similar, the underlying processes like the

product development process and the sales and marketing process may be quite different. For example, OSS firms pursuing a hybrid business model would likely need a skilled direct sales force, but our interview guide did not probe how the sales function worked or permit an assessment of its quality across vendors.

Another area ripe for further research is the expansion of so-called 'software as a service' (SaaS) and Web 2.0 companies building upon OSS. Many of the leading web services such as Google and Amazon.com provide free offerings to customers by building on open source technologies and infrastructure, which can dramatically cut development costs and time-to-market. Increasingly, some of the most valued technology companies offer services, which depend on OSS 'under the hood' that their users never see.

It is important to note that since the time of the interviews for this research in 2005 and 2006, a number of the companies mentioned have been acquired or have altered their business model. The ability of OSS-based companies to make inroads into a variety of horizontal and vertical markets is reflected by acquisitions, such as Xandros's acquisition of Scalix, Sun Microsystems's acquisition of MySQL, Oracle's acquisition of Sleepycat, Red Hat's \$350 million acquisition of JBoss (LaMonica, 2006a, 2006b). These acquisitions have helped to legitimise the revenue potential of OSS companies and their role in the broader software ecosystem. Other companies mentioned in this research, such as SpikeSource, Jive and Compiere, have changed their business models.

Overall, the myth of the OSS sector as an 'anarchistic, caffeinated, hirsute world of hackers' is just that (Evans and Wolf, 2005). The role of open source technologies and the vendors backing them are increasingly central to corporate computing. A February 2005 Gartner report identified open source as one the top five issues for information technology (Cearley et al., 2005). The Gartner study also found that 95% of global 2000 organisations were planning on addressing OSS acquisition and management strategies across virtually every software category. Finally, in a report issued in the summer of 2006, IDC found that 71% of software developers around the world were using OSS (CDN Staff, 2006).

Acknowledgements

References

Benkler, Y. (2002) 'Coase's Penguin, or, Linux and the nature of the firm', *Yale Law Journal*, December, Vol. 112, No. 3, pp.369–446.

Bonaccorsi, A., Giannangeli, S. and Rossi, C. (2006) 'Entry strategies under competing standards: hybrid business models in the open source software industry', *Management Science*, July, Vol. 52, No. 7, pp.1085–1098.

CDN Staff (2006) 'The real impact of open source: IDC study concludes it will play a role in the life-cycle of every major software category', *ITBusiness.ca*, 16 August, available at http://www.itbusiness.ca/it/client/en/EDGE/News.asp?id=40328.

Cearley, D.W., Fenn, J. and Plummer, D.C. (2005) 'Gartner's positions on the five hottest IT topics and trends in 2005', *Gartner Research*, Gartner, Stamford, CT, 12 May, available at http://www.gartner.com/resources/125800/125868/gartners_positi.pdf.

Chesbrough, H. (2003) *Open Innovation*, Harvard Business School Press, Boston, MA.

Chesbrough, H. (2006) *Open Business Models*, Harvard Business School Press, Boston, MA.

Chesbrough, H.W. and Appleyard, M.M. (2007) 'Open innovation and strategy', *California Management Review*, Fall, Vol. 50, No. 1, pp.57–76.

Cohen, S. (2005) 'Foolproofing open source', *BusinessWeek Online*, 9 August, available at http://www.businessweek.com/technology/content/aug2005/tc2005089_9209.htm?campaign_id=search.

Dahlander, L. and Magnusson, M.G. (2005) 'Relationships between open source software companies and communities: observations from Nordic firms', *Research Policy*, Vol. 34, pp.481–493.

de Laat, P.B. (2005) 'Copyright or copyleft?: an analysis of property regimes for software development', *Research Policy*, December, Vol. 34, No. 10, pp.1511–1532.

Evans, J. (2000) 'IBM to invest almost $1 billion in Linux development', *Infoworld*, available at http://www.infoworld.com/articles/hn/xml/00/12/12/001212hnibmlin.html (accessed on 12 December).

Evans, P. and Wolf, B. (2005) 'Collaboration rules', *Harvard Business Review*, July–August, pp.1–8.

Foley, M.J. (2001) 'Stallman strikes back at Microsoft', *ZD Net Asia*, 30 May, available at http://www.zdnetasia.com/news/hardware/0,39042972,21207321,00.htm.

Franke, N. and von Hippel, E. (2003) 'Satisfying heterogeneous user needs via innovation toolkits: the case of Apache security software', *Research Policy*, July, Vol. 32, No. 7, pp.1199–1215.

Galli, P. (2002) 'Microsoft tight-lipped on licensing numbers', *eWeek*, 2 August, available at http://www.eweek.com/article2/0%2C1895%2C431555%2C00.asp.

Ghauri, P. and Grønhaug, K. (2005) *Research Methods in Business Studies*, Pearson Education Limited, Harlow, England.

Golden, B. (2005) *Succeeding with Open Source*, Addison Wesley, Boston, MA.

Greene, T.C. (2001) 'Ballmer: 'Linux is a cancer'', *The Register*, 2 June, available at http://www.theregister.co.uk/2001/06/02/ballmer_linux_is_a_cancer/.

Griliches, Z. (1957) 'Hybrid corn: an exploration in the economics of technical change', *Econometrica*, October, Vol. 25, No. 4, pp.501–522.

Griliches, Z. and Schmookler, J. (1963) 'Inventing and maximizing', *American Economic Review*, September, Vol. 53, pp.725–729.

Haefliger, S., von Krogh, G. and Spaeth, S. (2008) 'Code reuse in open source software', *Management Science*, January, Vol. 54, No. 1, pp.180–193.

Hamerman, P. and Harrington, J. (2005) 'Third-party application support promises lower costs, with tradeoffs', *Quick Take*, Forrester Research, Cambridge, MA, 11 March, available at http://www.oracle.com/support/collateral/3rd-party-app-support_forrester.pdf.

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*, July, Vol. 32, No. 7, pp.1159–1177.

IBM (2000) *Annual Report*.

IDC (2004) 'The Linux marketplace – moving from niche to mainstream', 14 December, available at http://www.osdl.org/docs/linux_market_overview.pdf.

Lacy, S. (2005) 'A watershed for open source', *BusinessWeek Online*, 28 December, available at http://www.businessweek.com/technology/content/dec2005/tc20051228_262746.htm.

LaMonica, M. (2006a) 'Red Hat scoops up Jboss', *CNET News.com*, 10 April, available at http://news.com.com/Red+Hat+scoops+up+JBoss/2100-7344_3-6059293.html.

LaMonica, M. (2006b) 'Oracle snags open-source database company', *CNET News.com*, 14 February, available at http://news.com.com/Oracle+snags+open-source+database+company/2100-7344_3-6039070.html?tag=nl.

Lerner, J. and Tirole, J. (2002) 'Some simple economics of open source', *The Journal of Industrial Economics*, June, Vol. 50, No. 2, pp.197–234.

Lerner, J. and Tirole, J. (2005) 'The scope of open source licensing', *The Journal of Law, Economics, & Organization*, April, Vol. 21, No. 1, pp.20–56.

Lohr, S. (2005) 'I.B.M. surpasses estimates as tech spending improves', *NYTimes.com*, 18 October, available at http://www.nytimes.com/2005/10/18/technology/18blue.html.

Markus, M.L., Manville, B. and Agres, C.E. (2000) 'What makes a virtual organization work?', *Sloan Management Review*, Fall, pp.13–26.

Marson, I. (2005) 'SAP slams open source socialism', *ZDNet UK*, 11 November, available at http://news.zdnet.co.uk/software/linuxunix/0,39020390,39236750,00.htm.

Marson, I. (2006) 'Firefox gets a fifth of European market', *CNET News.com*, 17 January, available at http://news.com.com/Firefox+passes+20+percent+market+share+in+Europe/2100-7344_3-6027517.html?part=rss&tag=6027517&subj=news.

Nelson (2006) 'Open source licenses', *Open Source Initiative*, 19 September, available at http://www.opensource.org/licenses/.

Netcraft (2006) 'January 2006 web server survey', January, available at http://news.netcraft.com/archives/2006/01/05/january_2006_web_server_survey.html.

O'Mahony, S. (2003) '0Guarding the commons: how community managed software projects protect their work', *Research Policy*, July, Vol. 32, No. 7, pp.1179–1198.

O'Reilly, T. (2004) 'Sharing to advantage: a new paradigm for business', *Strategy & Innovation*, July–August, pp.3–5.

Olson, M. (2005) 'Open source business models', SCCT presentation at the University of California, September 22.

Perens, B. (2005) 'The emerging economic paradigm of open source', available at http://perens.com/Articles/Economic.html (accessed on 16 February).

Raymond, E. (1999) *The Cathedral and the Bazaar*, O'Reilly, Beijing.

Red Hat, Inc. (2002) 'Red Hat reports fiscal fourth quarter and year end results', 19 March, available at http://investors.redhat.com/releasedetail.cfm?ReleaseID=356011.

Red Hat, Inc. (2005a) 'Red Hat reports fiscal third quarter results', 21 December, available at http://investors.redhat.com/releasedetail.cfm?ReleaseID=355589.

Red Hat, Inc. (2005b) 'Red Hat reports fiscal fourth quarter and year end results', 21 March, available at http://investors.redhat.com/releasedetail.cfm?ReleaseID=355739.

Scherer, F.M. (1982) 'Demand-pull and technological invention: Schmookler revisited', *The Journal of Industrial Economics*, Vol. 30, No. 3, pp.225–237.

Shankland, S. (2004) 'IBM's Linux revenue: services to overtake servers', *ZD Net News*, 4 August, available at http://news.zdnet.com/2100-3513_22-5297392.html.

Välimäki, M. (2005) *The Rise of Open Source Licensing*, Turre Publishing, Helsinki, Finland.

von Hippel, E. (1998) 'Economics of product development by users: the impact of 'sticky' local information', *Management Science*, May, Vol. 44, No. 5, pp.629–644.

von Hippel, E. (2005) *Democratizing Innovation*, MIT Press, Cambridge, MA.

von Hippel, E. and von Krogh, G. (2003) 'Open source software and the 'private-collective' innovation model: issues for organization science', *Organization Science*, March–April, Vol. 14, No. 2, pp.209–223.

von Krogh, G. (2003) 'Open-source software development', *MIT Sloan Management Review*, Spring, Vol. 44, No. 3, pp.14–18.

von Krogh, G., Spaeth, S. and Lakhani, K.R. (2003) 'Community, joining, and specialization in open source software innovation: a case study', *Research Policy*, July, Vol. 32, 7, pp.1217–1241.

West, J. and Gallagher, S. (2006) 'Challenges of open innovation: the paradox of firm investment in open-source software', *R&D Management*, Vol. 36, No. 3, pp.319–331.

Yin, R.K. (2003) *Application of Case Study Research*, SAGE Publications, Inc, Thousand Oaks, CA.

Notes

1 An introduction to the concepts of demand pull and technology push can be found in Griliches (1957), Griliches and Schmookler (1963) and Scherer (1982).

2 Glenn Brewer of Mazu Networks, interview with authors, 5 December 2005.

3 Shaun Connolly of JBoss, interview with authors, 16 December 2005.

4 Jorg Janke of Compiere, interview with authors, 16 December 2005.

5 Larry Augustin, SugarCRM Board Member, interview with authors, 8 December 2005.

6 Mark Shuttleworth of Canonical, interview with authors, 9 December 2005.

7 It is important to note that the terms 'free software' and 'open source software' are not synonymous. Free software licensed under the terms of the GPL is OSS; however, not all OSS is free software. The term 'free' refers not to the price of the software, but to the four freedoms conferred by the GPL.

8 For more analysis of the implications of potential open source license fragmentation and incompatibility, see de Laat (2005).

9 Matt Tucker of Jive Software, interview with authors, 9 December 2005.

10 von Hippel (2005) documents the pervasiveness of 'innovation communities' across a host of industries ranging from OSS to kitesurfing. Markus et al. (2000) note that established open source communities frequently are governed through formal mechanisms.

11 While this article focuses on business models pursued by firms based on open source foundations, other studies have examined how 'open innovation' through liberal IP sharing by 'proprietary-oriented' firms can be justified. West and Gallagher (2006), for example, find that firms like IBM can benefit from pooling software IP, spinning out entities into the public domain based on software IP, or developing complements that are either sold off or 'donated' that in turn enhance other lines of the originator's business.

   Consistent with the findings of West and Gallagher (2006), Chesbrough (2006) demonstrates how value creation can be strengthened through openness by improving complementary technologies, by integrating open source-based code into 'system architectures', or by contributing IP to intellectual 'commons' to deter litigation. On this last point, he notes how crucial it is to trace IP ownership of the open source 'building blocks' to avoid hold-up when IP rights are asserted, e.g., by the Santa Cruz Operation [Chesbrough, (2006), p.44].

12 Shaun Connolly, op. cit.

13 Dave Hersh of Jive Software, interview with authors, 9 December 2005.

14 In addition to variants on these models, Chesbrough (2006) also includes a business model whereby a vendor incorporates OSS into a customer's existing IT infrastructure [Chesbrough, (2006), p.45].

15 Novell similarly offers 'OpenSuSE'.

16 Shaun Connolly, op. cit.

17 Murugan Pal of SpikeSource, interview with authors, 13 December 2005.

18   Eric von der Heyden of Innoopract, interview with authors, 16 December 2005.

19   Shaun Connolly, op. cit.

20   Michael Olson of Sleepycat, interview with authors, 7 December 2005.

21   Larry Augustin, op. cit.

22   Bernard Golden of Navica, interview with authors, 8 December 2005.

23   Novell, which acquired Ximian in late 2003, later open-sourced the Connector technology to help spur adoption of the emerging iCalendar standard.

24   Larry Augustin, op. cit.

25   Rick Fleischman of ActiveGrid, interview with authors, 12 December 2005.

26   Jeremy White of Codeweavers, interview with authors, 14 December 2005.

27   Julie Farris of Scalix, interview with authors, 18 January 2006.

28   Doug Levin of Black Duck Software, interview with authors, 6 December 2005.

29   Jason McKerr of the Oregon State University Open Source Lab, interview with authors, 18 January 2006.

30   Mike Matsumura of Infravio, interview with authors, 17 January 2006.

31   For an examination of the motivation behind user-led innovation in other technology-intensive industries see von Hippel (1998).