

von Engelhardt, Sebastian

**Working Paper**

## What economists know about open source software its basic principles and research results

Jena Economic Research Papers, No. 2011,005

**Provided in Cooperation with:**

Max Planck Institute of Economics

*Suggested Citation:* von Engelhardt, Sebastian (2011) : What economists know about open source software its basic principles and research results, Jena Economic Research Papers, No. 2011,005, Friedrich Schiller University Jena and Max Planck Institute of Economics, Jena

This Version is available at:

<https://hdl.handle.net/10419/56899>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*



# JENA ECONOMIC RESEARCH PAPERS



# 2011 – 005

## **What Economists Know about Open Source Software Its Basic Principles and Research Results**

by

**Sebastian v. Engelhardt**

[www.jenecon.de](http://www.jenecon.de)

ISSN 1864-7057

The JENA ECONOMIC RESEARCH PAPERS is a joint publication of the Friedrich Schiller University and the Max Planck Institute of Economics, Jena, Germany. For editorial correspondence please contact [markus.pasche@uni-jena.de](mailto:markus.pasche@uni-jena.de).

Impressum:

Friedrich Schiller University Jena  
Carl-Zeiss-Str. 3  
D-07743 Jena  
[www.uni-jena.de](http://www.uni-jena.de)

Max Planck Institute of Economics  
Kahlaische Str. 10  
D-07745 Jena  
[www.econ.mpg.de](http://www.econ.mpg.de)

© by the author.

# What Economists Know about Open Source Software

Its Basic Principles and Research Results

S. v. Engelhardt\*\*

January 31, 2011

## Abstract

For a decade, economists have been fascinated by the phenomenon of open source software (OSS). OSS is marked by free access to the software and its source code. It is developed in a public, collaborative manner by thousands of non-paid volunteers as well as profit seeking firms. Today, OSS is well established in the ICT sector and represents a new intellectual property paradigm.

This paper provides an introduction into the topic OSS versus closed source software (CSS, also called 'proprietary' software). After a brief history of OSS and CSS, the differences between the open and the closed source principles and the basic logic of OSS business models are explained. Next, the paper presents what economists know about the OSS phenomena, i.e. gives an overview of the motives of the (non-paid) OSS developers, the institutions of OSS, the effects of OSS on competition, the incentives and role of firms, and finally of open source principle beyond software.

**Keywords:** open source, open source software, intellectual property rights, information and communications technologies

**JEL:** L17, O34, L86,

---

\*\*Friedrich-Schiller-University Jena, Department of Economics and Business Administration, Carl-Zeiss-Str. 3, D-07743 Jena. E-mail: [Sebastian.Engelhardt@wiwi.uni-jena.de](mailto:Sebastian.Engelhardt@wiwi.uni-jena.de)

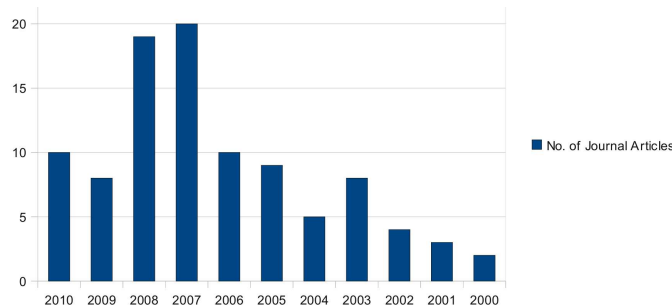
## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Open versus Closed Source Software: Two Intellectual Property Right Regimes</b>	<b>3</b>
2.1	A Brief History of Open and Closed Source Software . . . . .	3
2.2	Open versus Closed Source Principle . . . . .	5
2.3	Open Source Business Models . . . . .	7
<b>3</b>	<b>The Economics of Open Source Software</b>	<b>10</b>
3.1	The (Non-Paid) Open Source Software Developers . . . . .	10
3.1.1	Why Are They: Motives to Participate . . . . .	10
3.1.2	Who Are They: The Socio-Economic Background . . .	12
3.2	The Institutions of Open Source Software Development . . .	15
3.2.1	How Are They Organized: Governance Structures . .	16
3.2.2	Legal Protection: Software Licenses and More . . . . .	20
3.2.3	Macro-Level Institutions and Cultural Aspects . . . . .	23
3.3	Market Outcome and Competition . . . . .	24
3.3.1	Only the (Non-Commercial) Community Provides Open Source Software . . . . .	24
3.3.2	Firms Providing Open Source Software . . . . .	27
3.4	Incentives and Role of Firms . . . . .	30
3.4.1	Firms Within the Community: Role, Influence and Relationships . . . . .	30
3.4.2	Why and Which Firms Go Open Source . . . . .	31
3.5	Open Source Beyond Software . . . . .	33
<b>4</b>	<b>Summary: What Do Economists Know about Open Source Software?</b>	<b>34</b>
	<b>References</b>	<b>36</b>

## 1 Introduction

The last ten years saw a rise in economic research on so-called open source software (OSS) accompanied by an explosion of OSS-based business models and investments throughout the ICT sector. The Linux-based Kindle by Amazon or various smart phones (including the “Goggle Phone”) running Android are just the most prominent OSS-based products. Other examples indicating the importance of OSS and OSS-based business models are IBM’s OSS strategies or the market shares of MySQL and of the Apache web server software. Figure 1 indicates how economic research on OSS developed: it shows the number of published articles in economic journals, as listed in the EconLit database. Clearly, this is only a proxy for research activities, as many (more recent) papers still circulate as working papers.

Figure 1: Journal articles (EconLit) on “Open Source Software” <sup>1</sup>



The emergence of “open source” methods for producing software was intriguing for economists, because (a) the number of conceptually distinct incentives (e.g. patents, prizes, grants, contract research, etc.) that society traditionally uses to promote innovation is—despite differences in detail—remarkably small (see [Scotchmer 2004](#)), and (b) the success of OSS has challenged conventional wisdom of the role of intellectual property rights (IPRs). OSS is marked by free access to the software and its source code, and is developed in a public, collaborative manner. Thus, OSS appears to be a case of a “private provision of a public good” ([Johnson 2002](#)). At a first

<sup>1</sup>This figure is based on an EconLit database request on December 17th, 2010. Search term was “Open Source Software” and results were limited to published journal articles.

glance this seems to contradict economic rationality. Indeed, the economic justification for providing temporary monopolies on intellectual property in terms of copyrights and patents is to avoid a public good situation: (ex ante) rational agents have an incentive to invest only if (ex post) they have exclusive intellectual property, i.e. a legal mechanism which enforces excludability for a limited period of time. Nevertheless, it is a matter of facts, that thousands of software developers contribute voluntarily, i.e. without (direct) monetary reward, to OSS. Additionally, also profit seeking firms use OSS based business models and contribute to OSS projects.

OSS thus represents a “new intellectual property paradigm” (Maurer and Scotchmer 2006). In other words: the open source principle represents a new type of ownership concept for the digital economy. However, OSS has not completely replaced its counterpart closed source software (CSS), the latter also called proprietary software. As result, OSS and CSS coexist, and compete often within the same market. The difference between OSS and CSS is a difference in institutions.<sup>2</sup> OSS and CSS lead to different kinds of “institutional arrangements” (Davis and North 1971). These coexisting institutional arrangements are distinguishable by their distinct use of copyright law that is codified in the software licenses. The different types of licenses lead to different allocations of IPRs and different governance structures etc.

This paper provides an overview of research on the economics of OSS. Of course, it is neither possible nor sensible to cover all papers dealing with OSS. The paper at hand is no literature survey in terms of an exhaustive list of publications. It is rather an attempt to give an extensive overview. Naturally, the selection and classification of the research contributions are to some extent based on the subjective view of the author of this paper. Nevertheless, I think that the selection and classification is sensible, and I hope that it is fruitful for readers who want to know what economists know about OSS after one first decade of research. Furthermore, I see this paper as a complement to earlier contributions providing literature surveys and

---

<sup>2</sup>Institutions are “the rules of the game in a society, or, more formally, are the humanly devised constraints that shape human interaction” (North 1990, p 3). They “are made up of formal constraints (e.g., rules, laws, constitutions), informal constraints (e.g., norms of behavior, conventions, self-imposed codes of conduct), and their enforcement characteristics” (North 1994, p 360).

recommend the interested reader to also refer to [Maurer and Scotchmer \(2006\)](#), [Rossi \(2006\)](#), [Lerner and Tirole \(2005a\)](#) and [Schiff \(2002\)](#).

The remainder of this paper first offers a short introduction into the topic OSS versus CSS (Section 2), including a brief history of OSS and CSS, the differences between the open and the closed source principles, and the basic logic of OSS business models. Section 3 then presents what economists know about the OSS phenomena, i.e. gives an overview of the state of economic research. Broadly summarized, the existing literature on the economics of OSS deals with (a) the intrinsic and extrinsic motives of the (non-paid) OSS developers, (b) institutions like governance structures and licenses, (c) the impact of OSS on competition and market outcome, (d) the incentives and role of firms, and (e) the possibility to transfer the open source principle to other fields. We finish with a summary.

## **2 Open versus Closed Source Software: Two Intellectual Property Right Regimes**

### **2.1 A Brief History of Open and Closed Source Software**

In the early days, software was not a single product but more or less a tool to run the computers. Hence, revenue was created by selling computers, and the hardware vendors delivered software for free. Although some firms were selling the service ‘code writing’, there was no market for ready-made software products, so-called ‘software packages’.

This picture started to change in the late 1960s, when entrepreneurs realized the opportunity to sell their software to more than one customer, hence to treat it like an ordinary mass-marketable product. This new concept diffused, and finally, in the 1980s, the mass publication of packaged software by independent software vendors was established. Meanwhile the U.S. hardware producers – except IBM<sup>3</sup> – withdrew from software ([Steinmueller 1996](#), p 31 ff.). This rise of the software industry went with increasing concern about the protection of exclusive intellectual property rights. At least since the amendment of U.S. copyright law in 1980, copyright was

---

<sup>3</sup>A history of IBM’s licensing strategies, its paradigm shift to CSS and then to a balance between OSS and CSS, can be found in [Campbell-Kelly and Garcia-Swartz \(2009\)](#).

used to protect intellectual property rights with respect to computer programs. Based on this legal ground, ‘proprietary’, i.e. closed source, business models were established. As such, the early independent software vendors invented the CSS-based business models, but were also the driving force for establishing copyright protection for software. Thus, they also induced a change on the level of formal institutions.

The industry transition to CSS led to some attempts to preserve the ‘free’ programming culture based on so-called hacker ethics. The most important attempt was the foundation of a project called GNU (GNU’s Not UNIX). GNU was founded in 1984 by Richard Stallman, who worked at MIT from 1971 to 1984. Stallman was dissatisfied with the rise of the closed source principle, namely with its consequence for the use of UNIX.<sup>4</sup> Therefore he designed and introduced the GNU General Public License (GPL), nowadays the most popular type of open source license. The basic idea of the GPL was to use “copyright law, but flips it over to serve the opposite of its usual purpose: instead of a means of privatizing software, it becomes a means of keeping software free” (Stallman 1999, p 59). Thus, the GPL was created in order to preserve a certain programming culture of free software and hacker ethics.

This changed the level of institutionalization by transferring some norms of the hacker ethics—i.e. informal institutions—into a formal institution, namely the GPL. With the GPL Stallman invented a new concept of copyright-based ownership: the so-called ‘copyleft-principle’. Although Stallman was not motivated by commercial aspects and the creation of the GPL was an act of ideology, Stallman’s transformation is an economic success story. For example, all firms with business models build on Linux are based on a GPL-protected software. After the institutionalization of the OSS principle by Stallman and others, several entrepreneurs created business models based on the OSS principle.

So nowadays, firms and individual programmers have an institutional choice whether they use OSS, CSS or both. The next section describes the institutional differences between OSS and CSS. Section 2.3 then provides a brief explanation of OSS-based business models.

---

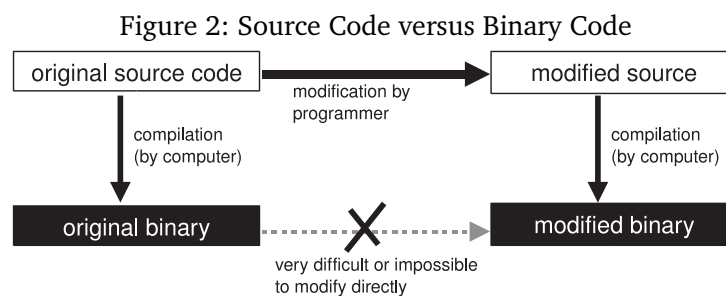
<sup>4</sup>At this time, UNIX was the most powerful operating system. In the 80s firms started selling incompatible, closed source versions of UNIX.



## 2.2 Open versus Closed Source Principle

Software is traditionally protected by copyright<sup>5</sup> (Graham and Somaya 2004), and the copyright-based license agreements define the transfer of the intellectual property rights. The crucial feature which distinguishes OSS from CSS is the scope of rights transferred by the OSS vs. CSS licenses. Technically this is mirrored by the question of whether there is general access to the source code or not.

The source code is the human-readable recipe of a software program: it is the program code written in a programming language. To run the program on a computer, the source code has to be compiled, i.e. transformed to a (only machine readable) binary code. The crucial point is that the binary code is not readable for humans. Having only the binary software, it is virtually impossible to modify the software. But with the source code, developers can understand how the software is programmed, can eventually learn from it, and—of course—are able to modify the source code, thus further develop the software. Figure 2 summarizes the implications of having access to the source code or not.



(Source: Schwarz and Takhteyev 2010)

CSS vendors like Microsoft typically transfer their software only as binary code. They sell the right to use the software, with the scope of legal

<sup>5</sup>The discussion about so-called software patents is beyond the scope of this paper. For the topic of software patents see e.g. Blind et al. (2005), Hall and MacGarvie (2006), Lerner and Zhu (2007), Bessen and Hunt (2007, 2004), Kahin (2004), Pilch (2004).

usage defined by the respective CSS license. For example, copying is not allowed, and this is enforced by law and backed by technical solutions like copy protection. Furthermore, users do not have the right to change the software, and they are also unable to do so as they have no access to the source code.

In contrast to this, an open source code enables users to copy the program code, to understand how the software works, and to change it. Thus, OSS is based on a principle of openness, which is codified in the copyright-based OSS licenses. These OSS licenses permit users to read, modify, improve, and redistribute the code under certain conditions. These conditions vary within a wide range. Liberal licenses—like the BSD license<sup>6</sup>—allow, for example, the use of the open source code to produce CSS. Such liberal licenses are also called ‘public’ licenses as they put the code in the public domain without any restrictions in use. More restricted licenses—like the GPL<sup>7</sup>—confine the scope of usage, mainly to ensure that the open source code stays open source, see also the footnote in Table 1.

OSS and CSS licenses therefore differ in the scope of transferred rights, which is summarized in Table 1. The two IPR regimes thus yield different

Table 1: Closed Source and Open Source Licenses: Transferred Rights

	Usage	Change	Redistribute
CSS	✓	✗	✗
OSS (restrictive license)	✓	✓	(✓)*
OSS (liberal license)	✓	✓	✓

*\*Restrictive OSS licenses restrict users’ right to redistribute: Any further developed software as well any derived work must be licensed as a whole under the same type of license, if this new code is further transferred, hence (re)distributed.*

allocations of IPRs. This has further implications. The kind of ownership of the source determines the governance structures: The principle of CSS is to hold exclusive rights regarding the source code. Closed source code is thus

<sup>6</sup>The BSD license is the license of the Berkeley Software Distribution.

<sup>7</sup>GPL stands for the GNU General Public License, see also p 4.

an exclusive asset, and CSS-products are typically developed within single firms. In this hierarchical structure based on exclusive ownership of assets, coordination is achieved by giving orders. Conversely, the source code of OSS is a shared asset. OSS is developed by a decentralized but nevertheless well-organized community. A complex system of rules has emerged to govern OSS development. There exist some hierarchical elements with respect to decision rights, but no one can give orders to other OSS developers. We will come back to the governance structures of OSS in Section 3.2.

The OSS community consists of thousands of volunteers who develop software, often without direct monetary reward. Additionally, more and more profit-seeking firms engage in OSS development, thus paying programmers to develop OSS code. Large companies as well as small- and medium-sized enterprises use OSS-based business models.

### 2.3 Open Source Business Models

Almost all of today's high tech products are computerized. While this is most obviously true for application software (e.g. games), the point increasingly extends to hardware like cell phones and DVD players. In these industries, a product's quality—and hence consumer appeal—often depends sensitively on the software it contains. Before the 1990s, companies usually developed this as CSS in-house. Since then, however, companies have increasingly turned to shared open source code instead.

Thus, in many markets software is sold and/or used bundled with other goods and services. Consequently, OSS business models are based on these complementary products, as the open source code itself can not be a profit center (Maurer and Scotchmer 2006, p 289, 290ff). These complements can be hardware like servers or cell phones, premium versions of the software, or different kinds of service like maintenance, and so on. The following examples provide some idea of the range of OSS business models:

- Many different products—washing machines, mobile phones, flat-screen televisions etc.—are controlled by *embedded software*. Such embedded software can be OSS. Examples of hardware running embedded Linux are Amazon's Kindle, Cisco's MDS and Nexus data switches, Linksys's WRT54G W-LAN router, different Motorola, Nokia,

and Panasonic mobile phones, Philips's LPC3180 microcontroller, the TomTom GPS navigation systems, and various LG, Panasonic, Samsung, and Sony LCD and plasma televisions. The most recent example of embedded OSS is Android. Android is a Linux-based software stack (operating system, middleware and key applications) for mobile devices. Acer, Barnes & Noble, Dell, HTC Corporation/Google, Lenovo, LG, Motorola, Samsung, and Sony Ericsson all manufacture and sell products that come Pre-installed with Android.

- Firms in the software industry typically sell a stack of software and services. So-called system integrators even sell a stack of hardware, software, and services (Riehle 2007). For example, IBM is selling several servers with pre-installed Linux like the Red Hat Enterprise Linux or SUSE Linux Enterprise Server operating system. This is often based on a collaboration between IBM and the respective Linux distributor. Red Hat, Novell's SUSE and other Linux-distributors make money with ready-to-install 'distributions' and the corresponding services like support and maintenance. Such distributions consist of a large collection of well-matched OSS applications, often bundled with further CSS for 'enterprise class' premium versions.
- Internet-based businesses like webhosting and webservice have a high share of OSS-usage. Most web servers are driven by an OSS "Lamp Stack" software suite that includes a Linux operating system, Apache web server, MySQL database, and PHP/Perl/Python programming languages. Development is supported by corporations like Novell, IBM, Oracle, and Borland who then bundle Lamp with their proprietary hardware and software. Small web developers also use Lamp in their businesses and contribute code back to the project.

Furthermore, firms with OSS-based business models have joined several projects and consortia. A prominent example is the Open Handset Alliance ([www.openhandsetalliance.com](http://www.openhandsetalliance.com)), a business alliance of 78 firms for developing open standards for mobile devices, namely the above-mentioned Android. Another important consortium is the Linux Foundation, members are for example Cisco, Fujitsu, IBM, Nokia, and Oracle (for a full list of members please visit [www.linuxfoundation.org/about/members](http://www.linuxfoundation.org/about/members)). The

Linux Foundation was founded by the merger of the Free Standards Group and the Open Source Development Labs in 2007. The Open Source Development Labs consisted of a wide range of Linux-related providers of hardware, software and services ([West and Gallagher 2006](#)). Table 2 provides an overview of the members and their motivations.

Table 2: An Example for Commercial OSS: the Members of the Open Source Development Labs

Category	Companies	Motivation
Computer systems vendor	Dell, Fujitsu, Hitachi, HP, IBM, NEC, Sun	Replacing proprietary Unix in computers with shared Linux
Telecommunications vendor	Alcatel, Cisco, Ericsson, NEC, Nokia, NTT, Toshiba	Replacing proprietary Unix with Linux in telecom equipment
Microprocessor producer	AMD, Intel, Transmeta	Enter Unix market using Linux
Linux distributor (server and desktop)	Miracle Linux, NEC Soft, Novell, Red Hat, SuSE, Turbolinux	Sell Linux distributions and services
Embedded Linux distributor	LynuxWorks, MontaVista, TimeSys, Wind River	Design Linux into custom products for customers
Linux support company	VA Software, Linuxcare, LynuxWorks	Sell Linux services
Software developers	Computer Associates, Trolltech	Adapt proprietary applications to Linux

(Source: [West and Gallagher 2006](#))

Several authors have set up taxonomies of OSS business models. See for example [Ghosh et al. \(2002a\)](#), [Fink \(2002, Chapter 11\)](#), or [Daffara \(2007\)](#). These typologies are of interest for practitioners like managers or for researcher who do detailed empirical research on firms with different OSS business models. However, the different typologies are not of interest for

the paper at hand. We will not discuss these taxonomies further but keep to the following definition that subsumes all the different OSS business models: *Firms with OSS business models generate revenue by selling products (goods or services) that are complements to the freely-accessible OSS.*

### **3 The Economics of Open Source Software**

During the first decade of the 21st century, the new IPR paradigm of OSS was attracting more and more interest by economists. In 2000 and 2001 the working-paper versions of three contributions were published, each of them representing one branch of research on OSS. First, the motives of volunteers participating in OSS (Lerner and Tirole 2000). Second, the coordination of these contributions, hence the governance structures of OSS (Weber 2000). And third, the impact of OSS on market outcomes and competition (Mustonen 2001). These three branches were later supplemented by two more recent research aspects: OSS and firms, and open source beyond software. The following provides an overview of each of these branches.

#### **3.1 The (Non-Paid) Open Source Software Developers**

The first branch of research on OSS deals with the volunteers participating in OSS. Often the research questions concentrate on why they participate. Some contributions have a wider focus and are also interested in the socio-economic characteristics of the OSS developers including their country.

##### **3.1.1 Why Are They: Motives to Participate**

Probably the most famous research question regarding the economics of OSS was asked by Lerner and Tirole (2000, 2002): “Why should thousands of top-notch programmers contribute freely to the provision of a public good?” Lerner and Tirole emphasize the role of extrinsic motivation, namely the acquisition of a reputation-signal. They separate this aspect into two different incentives: career concern incentives, referring to future job offers or access to venture capital, and the ego gratification incentives,

which stems from a desire for peer recognition. This seminal article inspired further research on the motives of OSS contributors, analyzing extrinsic as well as intrinsic motives. An overview of this research can be found in [Rossi \(2006\)](#). Today, the consensus is that a mix of extrinsic and intrinsic motives explain the behavior of unpaid<sup>8</sup> OSS developers.

Most of the research on OSS developer motivation consists of empirical studies. These surveys indeed report extrinsic motives like peer recognition and reputation within the community, self-marketing, and career-related motives like the improvement of programming skills and reputation signals ([Lakhani and Wolf 2005](#), [Hertel et al. 2003](#), [Ghosh et al. 2002b](#), [Hars and Ou 2002](#), [Lakhani et al. 2002](#)). However, most of these surveys find intrinsic motives ranking higher than extrinsic ones. [Lakhani and Wolf \(2005\)](#) find that enjoyment-related intrinsic motivations in the form of a sense of creativity are more important than extrinsic motivations. According to [Lakhani et al. \(2002\)](#) the two top-ranked motives are that developing OSS is “intellectually stimulating” and “improves skill”. [Ghosh et al. \(2002b\)](#) find that the most important reasons why developers have joined and stay in the community are that they want to learn and develop new skills, and that they want to share their knowledge and skills with other developers.

It is important to note that most of the empirical studies on OSS motives are surveys. Such surveys reflect what the OSS developers report as being their most important motives but do not take into account the importance of the developers, their effort levels, etc. An exception is the article by [Hars and Ou \(2002\)](#). They connect reported motives with the individual effort and find that, although intrinsic motivations play a role, external motives have greater weight, see Table 3. [Hars and Ou \(2002\)](#) also point out, that different types of OSS programmers exist. For example students and hobby programmers are more internally motivated than professionals.

As noted before, the most prominent external motive is job-market signaling: programmers engage in OSS development in order to disclose unobservable skills. Such a classical signaling of skills, as described by [Spence \(1973\)](#), then yields higher wages for the employees. In a recent working paper, [Bitzer et al. \(2010\)](#) test for a wage premium associated with OSS-

---

<sup>8</sup>Some developers are paid by firms for developing OSS. In such a case their motivation is trivial. For why firms engage in OSS (thus pay OSS developers) see Section 3.4.

Table 3: Motivations and Effort of OSS Programmers

	<b>All</b>		<b>Students and hobby programmers</b>		<b>Salaried and contract programmers</b>		<b>Programmers paid for open-source development</b>	
<b>Motivation</b>	<b>Percent</b>	<b>Corr. with effort</b>	<b>Percent</b>	<b>Corr. with effort</b>	<b>Percent</b>	<b>Corr. with effort</b>	<b>Percent</b>	<b>Corr. with effort</b>
<b>1. Internal</b>								
Self-determination	79.7%	0.072	81.8%	-0.015	92.6%	-0.303	61.5%	0.221
Altruism	16.5%	0.192	24.2%	0.356	11.1%	0.061	7.7%	-0.163
Community identification	27.8%	0.116	36.4%	0.361	18.5%	-0.130	30.8%	-0.307
<b>2. External</b>								
<i>2.1 Future rewards</i>								
Selling products	13.9%	0.363	6.1%	0.011	3.7%	0.488	53.8%	0.304
Human capital	88.3%	0.139	96.9%	0.080	88.5%	0.073	84.6%	0.065
Self-marketing	36.7%	0.317	33.3%	0.206	29.6%	0.208	69.2%	0.424
Peer recognition	43.0%	-0.021	42.4%	-0.023	48.1%	-0.145	46.2%	-0.178
<i>2.2 Personal need</i>								
	38.5%	0.304	36.4%	0.301	38.5%	0.186	38.5%	0.328

(Source: Hars and Ou 2002)

based signaling exploiting a unique dataset about 7,000 German IT employees. Remarkably enough, their empirical analysis does not show any evidence that the signaling actually works. This result is even more striking as surveys consistently report that OSS developers belief in the signal.

### 3.1.2 Who Are They: The Socio-Economic Background

Research on motives of the OSS developers is often connected with research on their socio-economic background (Lakhani and Wolf 2005, Lakhani and Hippel 2003, Hertel et al. 2003, David et al. 2003, Hars and Ou 2002, Ghosh et al. 2002b, Robles et al. 2001). So beside the motives, questions of interest are such different issues like the share of female, the education level, job-position, and so on. In a recent paper David and Shapiro (2008)



provide a table listing the results of several surveys, see table 4. As the table shows, virtually all OSS developers are male (98.9% to 95%). The vast majority of developers are either students or employed, with the students being the smaller fraction of both. Furthermore, the OSS developers are well educated.

The aim of [David and Shapiro \(2008\)](#) is to bring together the motivations, personal attributes and behavioral patterns among OSS developers. They use hierarchical cluster analysis to extract a set of distinctive “motivational profiles”. David and Shapiro find that OSS developers differ depending on whether they engage in community-based projects, very small projects or work independently. These differences in attributes refers to their motivational profiles as well as to aspects like demographic characteristics, education and experience, the likelihood of receiving direct monetary, and so on.

However, one of the socio-economic variable is the geographic origin of the developers. As the OSS community is often described as global, OSS seems to be a digital public good with a truly globalized private provision. But, apart from anecdotal evidence for the internationality of certain OSS project teams, the question remains how global the OSS community actually is and how the supply side of OSS differs among countries. This has motivated researchers to study the geographical allocation of OSS developers. It turns out that the most OSS developers come from North America and Europe. This result is quite consistent regardless of the method used.

The methods to gather information about the geographic origin of OSS developers can be broadly distinguished into two approaches. Some studies are based on survey-data, while other work is based on specific data drawn from code of certain OSS projects such as credit files, mailing lists or data from platforms like SourceForge. [Robles et al. \(2001\)](#) provide a combination of both types of data collection. In [Ghosh \(2006\)](#), [David et al. \(2003\)](#) and [Ghosh et al. \(2002b\)](#) one can find survey-based information about the origin of OSS developers. [Lancashire \(2001\)](#) provides information about the world-wide distribution of Linux and Gnome developers, based on data collected from the Linux Credit file and in case of Gnome developer-contact information from the project’s web-site. The most recent research dealing with the geographic origin of OSS developers is [Gonzalez-Barahona et al. \(2008\)](#). The authors provide a worldwide picture of OSS develop-

Table 4: Socio-Economic Characteristics of OSS Developers

FLOSS-US compared with other pre-2005 surveys of FLOSS developers.

Type	Online survey, website posts, emailed lists				Emailed developers			Emailed developers from a single project	
Survey title	TU Berlin	FLOSS-EU	FLOSS-US	FLOSS-ASIA	BCG			Apache	Linux kernel
Reference if different from title	Robles et al. (WIDI 2001)	Ghosh et al. (2002)	David et al. (2003)	Mitsubishi (2004)	Wolf et al. (2002) and Lakhani and Wolf (2005)	Hars and Ou (2002)	Haruvy et al. (2003)	Lakhani and von Hippel (2002)	Hertel et al. (2003)
Data collection	June–August 2001	February–April 2002	January–June 2003	December 2003–January 2004	October 2001, April 2002	–	–	October 1999–February 2000	February–April 2000
Method	Online survey, website posts, emailed lists	Online survey, website posts, emailed lists	Online survey, website posts, emailed lists	Online survey, website posts, emailed lists	Emailed SourceForge project contributors	Emailed developers	Emailed developers	Emailed developers	Linux mailing list announcements
Usable responses (rate)	5478	2784	1588	138	684 of 1994 (34%)	79 of 389 (20%)	160 of 2000 (8%)	336 of 1709 (19.6%)	141 (half developers)
Motivation questions	None	Reasons for joining FLOSS community; reasons for staying in FLOSS	Reason for developing FLOSS, reasons for staying in FLOSS	Reason for developing FLOSS	Reasons for working on specific FLOSS project	Reasons for developing FLOSS	Reasons for developing FLOSS or working on specific project	None	Reasons for working in Linux community
Number of projects	–	–	1811 <sup>a</sup>	–	287	41 (with 25% from Linux)	More than 90	1	1
Respondents per project	–	–	0.88 <sup>a</sup>	–	2.38	1.93	1.78	–	–
Source	TU Berlin	FLOSS-EU	FLOSS-US	FLOSS-ASIA	BCG	Hars and Ou (2002)	Haruvy et al. (2003)	Lakhani and von Hippel (2002)	Hertel et al. (2003)
Location									
North America (%)	35%	13%	27%	1%	45%	–	–	–	48%
Western Europe (%)	47%	71% (EU)	53%	2%	38%	–	–	–	37% (all Europe)
Number continents	–	–	6	3	6	–	5	–	–
Number countries	94	–	65	16	52	–	“At least 18”	–	28
Age (mean)	27	27	29.0 <sup>a</sup>	27	29.8	–	–	–	–
Male	1	98.9%	98.4%	98.5%	97.5%	95%	–	–	–
Employment status									
Not employed	–	4%	4%	–	–	–	11%	–	5%
Employed	–	79%	68%	–	–	–	89%	–	72%
Student	0	17%	29%	16%	–	14%	32%	–	23%
Highest education									
High school	–	–	19%	–	–	–	–	–	–
Bachelor's degree	41%	33%	36%	–	–	48%	31%	–	–
Master's degree	11%	28%	43%	–	–	21%	16%	–	–
Ph.D.	4%	9%	–	–	–	3%	21%	–	–
FLOSS experience	–	4.1	5.1	4	5.3	–	4.7	–	–
Mean effort (h/week)	–	–	10.0	–	14.1	–	–	–	18.4

Notes: response rate shows number of usable observations out of total number solicited. FLOSS experience in years, mean effort in hours per week. “Employed” includes self-employment or work at a firm. BCG randomly chose 10% of SF projects with both more than one developer and reported maturity stage of Alpha, Beta, or Production/Stable, and they emailed developers listed in these projects to obtain 526 responses. BCG also emailed all participants in SF projects with reported maturity stage of Mature who had multi-person teams to obtain 158 additional responses. Hars and Ou (2002) obtained email addresses from “discussion lists and news groups ... both general open-source communities and specific open-source programmers’ forums” (p. 5). Haruvy et al. (2003) emailed “2000 programmers listed as contributors on open-source web pages and on selected open-source developer lists” (p. 16). They provide no further details or exact websites. Haruvy et al. (2003) do not count modules as separate projects. Hertel, Nieder, and Herrman: half of respondents were active developers, others merely read the Linux kernel mailing list. They have no defined response rate since they emailed lists recent/current projects (and their paper does not declare the lists’ membership size) rather than directly contacting a set number of individual developers. FLOSS-EU, FLOSS-US, BCG: FLOSS experience is years since first FLOSS contribution. FLOSS-US: mean commitment averages current/most recent and first projects.

<sup>a</sup> Implies new estimates from underlying FLOSS-US data rather than stated in David et al. (2003). Number of projects: FLOSS-US counts first and current.

(Source: David and Shapiro 2008)

ers, weighted by population, internet users and GDP. [Gonzalez-Barahona et al. \(2008\)](#) build on [Robles and Gonzalez-Barahona \(2006\)](#). [Robles and Gonzalez-Barahona \(2006\)](#) use information about the email addresses of registered users and the indicated time-zone to assign developers at SourceForge<sup>9</sup> in 2005 to their countries. However, they were unable to assign 25% to countries and therefore had to develop methods to estimate the geographic allocation for this part. [Engelhardt et al. \(2010\)](#) similarly make use of data of developers at SourceForge but are able to geographically identify 94% of all registered OSS developers in 2006 by analyzing IP address, email and time-zone. They also use data about the number of posted messages as proxy for the activity of each registered OSS developer. This is important, since members of the OSS community differ in their effort levels, numbers of contributions etc. (see e.g. [David and Rullani 2008](#)).<sup>10</sup> Hence, [Engelhardt et al. \(2010\)](#) find that while most registered developers were not active in 2006, the average activity level by active developers is remarkably similar. Furthermore, they find that both, active developers per capita and activity levels differ over the world, and that GDP per capita and internet usage alone cannot explain these differences.

### 3.2 The Institutions of Open Source Software Development

Aside from the question of motives, [Weber \(2000\)](#) asks how the OSS developers “coordinate their contributions on a single ‘focal point’?” ([Weber 2000](#), p 5). Research on OSS has thus to understand how the implications of the complexity of large OSS projects like Linux are managed. Consequently, in [Weber \(2004\)](#) he describes collaborative methods in the context of developing OSS. This points to the institutions of OSS, including organizational issues and governance structures, the role of hacker-ethics and the role, choice and rationale of OSS licenses.<sup>11</sup>

---

<sup>9</sup>SourceForge is the world’s largest site hosting OSS projects, an internet platform where OSS developers discuss, coordinate their tasks, upload new developed codes etc.

<sup>10</sup>For further research on the division of labor within open source projects see among others [Besten et al. \(2008\)](#), [Giuri et al. \(2008\)](#); or [Krogh et al. \(2003\)](#).

<sup>11</sup>[Gehring \(2006\)](#) and [Lessig \(2006, 1999\)](#) interpret the code itself to be an institution. However we do not follow this view here.

### 3.2.1 How Are They Organized: Governance Structures

What kind of organization OSS projects represent and how they are governed is a question that is widely discussed nowadays [Markus \(2007\)](#) offers a sound survey and a synthesis of this literature. Markus defines OSS governance as “the means of achieving the direction, control, and coordination of wholly or partially autonomous individuals and organizations on behalf of an OSS development project to which they jointly contribute” ([Markus 2007](#), p 152). [Markus \(2007\)](#) rightly points out that two branches of research on OSS governance can be distinguished. Some scholars analyze OSS as a new, distinct but unitary organizational form which can be differentiated from CSS development ([Raymond 1998](#)), characterized as a private-collective model ([Hippel and Krogh 2003](#), [Osterloh and Rota 2007](#)), and so on. Others emphasize the different types of OSS governance mechanism, or focus on one of these mechanisms specifically. An overview of the governance mechanisms is provided by [Laat \(2007\)](#). He groups the main tools of OSS governance into six categories: modularization, division of roles, delegation of decision-making, training and indoctrination, formalization, and authority versus democracy. What follows contains examples from both types of literature.

Let us start with OSS as a unitary organizational form. Very common in organizational theory is the tripartite division of types of organization into ‘market’, ‘firm’ and ‘network’ (see [Table 5](#)). In this context, OSS projects

Table 5: The three Types of Organization

Type of Organization	Type of Coordination	Coordination Instrument
Market	Competition	Price
Firm	Hierarchy	Order
Network	Cooperation	Consensus

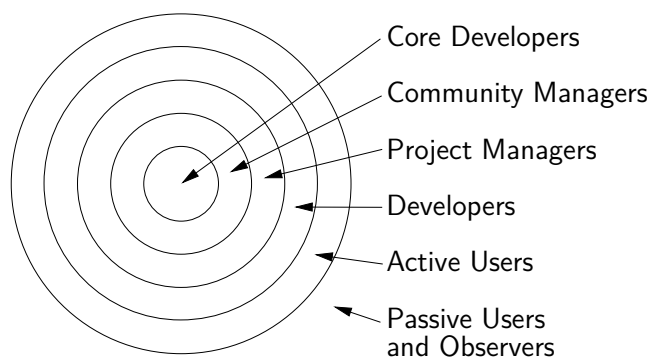
(Source: [Brand and Schmid 2005](#))

are mostly characterized as networks. An exception is the interpretation of [Demil and Lecocq \(2006\)](#). They argue that OSS projects differ from net-

works in that they do not require long-term relations, have no mechanism to restrict access, etc. Also [Garzarelli \(2003\)](#) points out the organizational uniqueness of OSS, arguing that its organizational characteristics can be explained by a combination of the organizational theory on clubs with the theory of professions. However, many authors (e.g. [Brand and Schmid 2006, 2005](#), [Hippel 2005](#), [Iannacci and Mitleton-Kelly 2005](#), [Benkler 2002](#)) interpret OSS projects as (special) networks. Based on a case study on the KDE project<sup>12</sup> [Brand and Schmid \(2005, 2006\)](#) find that OSS combines the coordination mechanism of networks with elements of hierarchy, the latter typically associated with firms.

The hierarchical elements of OSS organizations are mainly based on decision rights and the tasks of certain developers. The basic structure of these hierarchies is often labeled the ‘onion layer’ model, see [Figure 3](#). (See also

Figure 3: The ‘Onion Layer’ of OSS Projects



(Source: [Jensen and Scacchi 2007](#))

[Jensen and Scacchi 2007](#), [Crowston et al. 2006](#), and [Wendel de Joode et al. 2003](#) pp 18,19.) The career of developers within projects (a participant becomes project leader or specialist etc.), i.e. their movement into the core of

<sup>12</sup>KDE (K Desktop Environment) is an open source graphical user interface (GUI). Together with the GNOME desktop it is likely the most-known desktop environment (and development platform) for Linux and Unix workstations (Webpage: [www.kde.org](http://www.kde.org)).

the ‘onion’ is studied by [Giuri et al. \(2008\)](#). The acceptance of the ‘onion’-hierarchies and the authority of e.g. project leaders are based on the meritocratic norms of the OSS community, rooted in hacker-ethics ([O’Mahony and Ferraro 2007](#)). [Johnson \(2006\)](#) and [Lee and Cole \(2003\)](#) emphasize the importance of peer review processes in the control structures of OSS. In this context [Krogh et al. \(2003\)](#) point to the role of extrinsic motives and incentives like reputation and signaling.

Several authors underline that the evolution of OSS projects over time has implications for their organizational forms ([Sadowski et al. 2008](#), [Lattemann and Stieglitz 2005](#), [Schweik and Semenov 2003](#), [Wynn 2003](#)). Typically, OSS projects start with one or only a few developers who coordinate via direct communication based on trust. With the growth of the project size, more and more official coordination structures are implemented. Finally the projects characterized by well-defined roles (code-tester, release-manager, core-developer etc.) combined with a decentralized and modularized organization structure. For example, [Crowston and Howison \(2005\)](#) have analyzed 120 project teams from SourceForge and find that projects which grow become more modular, with different people responsible for different modules. [Langlois and Garzarelli \(2008\)](#) explicitly focus on modularity in open source collaborations, including OSS. They argue that in such collaborations the division of labor is coordinated through voluntary exchanges of effort rather than of products.

In his synthesis of research on OSS governance, [Markus \(2007\)](#) concludes that three OSS governance purposes are linked to six OSS governance categories, see Table 6. According to [Markus \(2007\)](#), OSS governance has to solve collective action dilemmas and coordination problems, and has to create a climate such that developers contribute to the particular project (rather than to others). To achieve these goals tools from the six governance categories are used. For example, rules about the software development process and rules about how information will be communicated and managed using certain tools (repositories), both support coordination. Conflict rules as well as meta-rules solve coordination problems but also create a good climate for contributors. The community rules – which determine who can become a member, what roles members can play etc. – clearly support all three purposes. Finally, tools that solve collective action dilemmas and create a contributor-friendly climate belong to the categories

Table 6: Relationship between OSS Governance Purposes and OSS Governance Categories

OSS governance purposes vs. OSS governance category	Solve collective action dilemmas	Solve coordina- tion problems	Create a climate for contributors
Ownership rules	x		x
Chartering rules	x		x
Community rules	x	x	x
Software development process rules		x	
Conflict rules and rules about rules		x	x
Information and tools rules		x	

(Source: Markus 2007)

of chartering rules and ownership rules. The first refers to statements about the goals of the project, what the software should look like, and so on. The second refers to the use of (intellectual) property law: the formal legal organizational structure (e.g. a foundation) and the type of license etc.

### 3.2.2 Legal Protection: Software Licenses and More

O'Mahony (2003) shows in detail how OSS projects use intellectual property law to protect their work. OSS projects make use of restrictive license terms or trademark registration etc. Additionally, often the copyright, trademark etc. is transferred to a foundation. Such foundations are better suited to enforcing e.g. the license restrictions or protecting the brand of the project.

The type of OSS license is an important institution, as it defines how the code can be used. Some authors analyze the importance of the respective license for the governance of an OSS project. For example, Franck and Jungwirth (2003) argue that the GPL is constructed such that egoistic motives ('rent seeking') do not crowd out altruistic motives ('donation'). Moreover, according to Franck and Jungwirth (2003), the GPL creates incentives for participation for both rent seekers and donators.

Sen et al. (2008) examine how the OSS license type (ranging from very restrictive to very liberal) can be explained by the motivations and attitudes of the OSS developers. They find that intrinsic motivation of challenge (problem solving) is connected with a preference for moderate restrictions, while extrinsic motivation of status (peer recognition) is linked to licenses with least restrictions. Another study on the determinants of OSS license choice is offered by Lerner and Tirole (2005b). They first develop a theoretical model and then test the model predictions empirically. According to Lerner and Tirole, OSS projects are more likely to have restricted licenses if they are consumer-oriented (e.g. desktop tools or games) and if they are developed in a corporate setting. Projects oriented toward developers and/or designed to run on commercial operating systems have less restrictive licenses. Finally, less restricted projects tend to attract more developers. A discussion about the relationship between OSS business models and the type of OSS licenses is provided by Laat (2005). This includes a description of Netscape's experience with different licenses, when the company turned



its browser Netscape Navigator into an OSS project—today this OSS project is named Mozilla with its products Firefox and Thunderbird.

D'Antoni and Rossi (2007) present a model with incomplete contracting to analyze the rationale for liberal versus restricted licenses, hence the BSD license versus the GPL. D'Antoni and Rossi (2007) find that the GPL is superior to coordinate and encourage joint effort by many (possibly small) developers; while the BSD is better suited to generate positive spillovers to other developers when no feedback is required. Gaudeul (2005) also compares the GPL and the BSD license, but her model also allows for a CSS license. Each software project consists of a sequence of cumulated innovations. The software has the same value for the project leader, the developers, and consumers. If the software has a high net market value and thus high profits, the project leader develops the software on its own and sell it under a CSS license. Otherwise it is rationale to choose a OSS license. The project owner forsake the profits of the CSS case but this is overcompensated by the cost effect: in the OSS case the project leaders (who draws utility from consuming software) gets the software developed by others 'for free'. The model predicts that the GPL will be chosen when the opportunity costs of developing the software are low relative to the value of the software (small efforts bring big rewards). Otherwise the project leader chooses the BSD. In this case developers simultaneously develop the code as they hope to be the first one. If a developer would win this race he or she could then claim her rights exclusively: the BSD allows to turn OSS code into CSS, i.e. sell it for a positive price. Furthermore, Gaudeul (2005) finds that competition makes OSS licenses more attractive because competition reduces the prices and hence profits of an CSS project leader: In a 'first mover, second mover' set up, the first mover must lower its prices in order to deter the development of an OSS alternative by the second mover.

Typically the literature on OSS versus CSS licenses does not differentiate between the divergent OSS license types. Using a centipede-type game, Polanski (2007) analyzes CSS versus OSS licensing as a mechanism design issue. He models cumulative production, thus sequential production where the outputs of the stages  $1 \dots k - 1$  are inputs for stage  $k$ . In such an environment, according to Polanski, a public (open source) license is better suited if the project is highly modular and there are significant returns to scale. Scotchmer (2010) considers a model with two sequential

innovations (complements) and two firms that are the first and the second innovator. Firms can choose between a closed source license (proprietary license) and an open source license (namely the GPL). The firm that turns out to be the first innovator will then prefer closed source licensing over open source licensing (GPL) in the intermediate stage. The reason is that for the first innovator the closed source license is more profitable. [Scotchmer \(2010\)](#) also finds that closed source licensing by the first innovator in the intermediate stage is efficient in the sense that it creates the same outcome as ex post bargaining. Nevertheless, industry profits would be higher if the industry uses an open source license. Furthermore, it is not known ex ante who will be the first innovator. So in the beginning, the two firms are behind a “veil of ignorance” where they then should favor to commit to open source licensing. [Bessen \(2006\)](#) uses a model based on incomplete contracting and the hold-up problem to shed light on the rationale of OSS licensing. The result of his model is that OSS licensing can be more efficient than CSS licensing in the case of complex products like software. According to Bessen, OSS will be mainly used by firms that have complex specialized needs and their own development capabilities.

Inspired by the case of OSS, [Benkler \(2002\)](#) discuss the rationale for what he calls “commons based peer production”. He refers to Demsetz’s explanation of the emergence of property rights and Coase’s theory of the firm ([Demsetz 1967](#), [Coase 1937](#)). Benkler focuses on the information problem of who is the best person for a given task. His main argument is that under certain circumstances peer production is better suited than markets or hierarchies (firms) “in matching the best available human capital to the best available information inputs in order to create information products” ([Benkler 2002](#), p 444). Another contribution that links the phenomenon of OSS to the concept of transaction costs is [Engelhardt \(2008\)](#). Based on a property rights approach the rationale for both IPR paradigms (open source versus closed source) is discussed. He argues that open source and closed regimes coexist because both are second-best solutions. Ex-post transaction costs make the first best set of contracts regarding a source code impossible as some IPRs of the code are not exclusively separable. Rights that are not exclusively separable are bundled. It can be rationale to exclusively claim these bundled rights (closed source). But it can also be rationale not to claim these rights. First, it does not cause costs to forgo

rights if these rights would not have been traded anyway because of transaction costs. Second, limits in separating and trading rights inhibit some feedback-effects. Enabling such effects can overcompensate the costs (e.g. lost revenue) to forgo the rights.

### 3.2.3 Macro-Level Institutions and Cultural Aspects

All the research contributions mentioned so far have in common that they deal with micro-level institutions. (Also in general, OSS is mainly a topic of microeconomic research.) To the best of the author's knowledge, there are only two papers that focus on macro-level institutions. More precisely, here the research question is how a country's institutional and cultural framework influence on the supply side of OSS.

[Ramanujam \(2007\)](#) links cultural factors with the geographics of OSS developers. Ramanujam uses data from [Ghosh \(2006\)](#) and Hofstede's cultural indicators to analyze how differences in national culture affect or influence the participation in OSS. He links the geographical distribution of developers with the four dimensions of national cultures considered by [Hofstede \(1991\)](#). Ramanujam states a positive correlation between the share of OSS developers and 'Individualism', whereas 'Power Distance' and 'Uncertainty Avoidance' are negatively correlated each. However, the results might be interpreted with care, as Ramanujam does not control for aspects like number of inhabitants, GDP, or internet access, and he distinguishes four regions only. With their cross-country study, [Engelhardt and Freytag \(2010\)](#) analyzes how several cultural and institutional factors including norms and attitudes can explain the global differences in OSS activities. Based on [Engelhardt et al. \(2010\)](#), they run regressions with data from about 70 countries. Their findings are that social capital (interpersonal trust), individualism/self-determination, and an optimistic view of scientific progress support OSS activities. OSS also benefits from a low degree of regulation, and good protection of IPRs. According to [Engelhardt and Freytag \(2010\)](#) these results support a view of OSS as being an entrepreneurial activity that relies on trust and IPR protection. It has a strong individualistic/self-deterministic aspect, combined with a spirit of individual initiatives. The results for market regulation and IPR protection show the importance of OSS activities connected to business models. Finally,

Engelhardt and Freytag point out that OSS benefits from good de facto protection of IPRs because OSS is not anti-IPR but a new IPR paradigm.

### 3.3 Market Outcome and Competition

Mustonen (2001, 2003) was the first to analyze the impact of OSS on competition and market outcome. This was followed by several contributions on this topic. In this context it is useful to distinguish models with non-commercial OSS from those with commercial OSS. In case of the first, all the OSS is provided by the non-commercial community. There can be firms who use this OSS as input, but there is no OSS developed by firms. The second branch focuses on markets where firms contribute to OSS.<sup>13</sup>

#### 3.3.1 Only the (Non-Commercial) Community Provides Open Source Software

Mustonen (2001, 2003) belongs to the branch focusing on non-commercial OSS. He models the interaction between the OSS community and a CSS monopolist. The monopolist is affected by OSS in two markets: the product market and the labor market. Consumers can either buy CSS or use OSS for free. However, both types of software cause implementation costs. Programmers choose to work for the monopolist at a wage that the monopolist sets, or develop OSS and thus build reputations that results future income. In Mustonen's setting, highly talented programmers have incentives to join the OSS community. The basic result is that if the software implementation costs are low, OSS and CSS coexist. The presence of OSS lowers the CSS vendor's monopoly power in both markets. The impact of non-commercial OSS on the outcome of software markets is also analyzed by Casadesus-Masanell and Ghemawat (2006). Their model is inspired by the competition between the operating systems Linux vs. Microsoft's Windows and is a dynamic mixed duopoly of CSS and non-commercial OSS.

---

<sup>13</sup>Models with commercial OSS also contribute to the topic 'OSS business models'. The distinction is made based on the focus of the paper. If the main purpose is to analyze the market outcome, then we consider it in this section. If the main purpose is to explain the rationales for OSS business models, then it is mentioned it in the next section.

Casadesus-Masanell and Ghemawat (2006) take into account dynamic effects which yield network externalities: the cumulative output of each operating system (installed base) affects their relative position over time. They find that Windows can survive in the market, hence coexist with Linux, if the installed base effect is strong enough.<sup>14</sup> Additionally, they show that welfare in the mixed case can be smaller than under a Windows monopoly. Economides and Katsamakas (2006) focus on the fact that operating systems like Linux and Windows are platforms. Making use of the theory of two-sided markets, Economides and Katsamakas (2006) compare industry structures based on an OSS platform with those based on a CSS platform. They compare a vertically integrated CSS-industry, a vertically disintegrated CSS-industry, and an industry with an OSS platform and CSS applications. Economides and Katsamakas (2006) provide conditions for each of these industries to have the highest industry profits. They also find that welfare is maximized if the industry is characterized by an OSS platform with different CSS applications.

Gaudeul (2008) presents a circular city model (Vickrey-Salop model) that focuses on markets where users are ICT specialists and thus all of them are (potentially) user-developers (like in web server software markets). In her model users decide whether to buy a CSS project or contribute to an OSS project. Gaudeul (2008) compares a pure OSS industry, a pure CSS industry, and a mixed industry where CSS and OSS projects alternate in location of the circle. She finds that welfare in such a mixed industry may be higher than in a pure OSS industry and is higher than in the pure CSS case. Also in Bitzer (2004) OSS is developed solely by the community. He analyzes a case where a CSS firm faces the emergence of OSS. However, Bitzer (2004) takes into account firms with OSS-based business models. Firms with OSS business models can use the OSS code for free and bear only the costs for producing the complementary products. The CSS firms on the other hand have to bear the costs for both software development and production of complements. Bitzer (2004) uses a Launhardt-Hotelling model set-up and derives the result that product heterogeneity is the crucial factor in this setting. If the heterogeneity between the OSS and CSS based

---

<sup>14</sup>They assume that in  $t = 0$  Windows is perceived more valuable than Linux. Hence Windows has an advantage in the beginning.

products is sufficiently high, the CSS firm will stay in the market. A model with competition in technological levels rather than in prices or quantities is proposed by [Bitzer and Schröder \(2007\)](#). They find that the chosen technological level is higher in markets with OSS and CSS than in pure CSS markets. The highest technological level is achieved in pure OSS markets. The growth-effect of innovations that come from the non-commercial OSS community is the topic of [Saint-Paul \(2003\)](#). In his model, a profit motivated R & D sector coexists with the introduction of free blueprints invited by philanthropists. In other words: software-innovations come from for-profit firms (CSS) and from a non-commercial OSS community. His main finding is that the growth effect of philanthropy (the OSS community) can be negative: while non-proprietary innovations boosts growth in the short run, it can reduce growth in the long run. The reason is that the existence of OSS reduces profits of the CSS firms and hence reduces their incentives to innovate. This effects can be so strong that long-run growth also falls.

[Sen \(2007\)](#) models competition in software markets where CSS vendors compete against firms who sell *improved* versions of OSS. The latter represents the business model of OSS ‘distributors’ as explained in Section 2.3, p 8. Firms take a given OSS and improve its usability with support and service (SS). Therefore, Sen calls this type of software OSS-SS. Consumers can thus choose to use either OSS for free, or purchase either OSS-SS or CSS. The CSS and OSS-SS firms decide on the usability of their software, while OSS has a fixed, low usability. Furthermore, users differ in their valuation of software usability (a Hotelling’s model approach). [Sen \(2007\)](#) takes into account network effects in terms of installed base. Here OSS and OSS-SS users belong to the same installed base, as both use the same software in technical terms. [Sen \(2007\)](#) finds the following results, from which he then also draws management implications. With weak network effects, CSS always have a market share of more than 50%, with its usability and prices being higher than those of OSS-SS. If network effects are high, profits for the OSS-SS vendors are maximized if they offer the same usability as their CSS rivals. In such a case, CSS is driven out of the market.

### 3.3.2 Firms Providing Open Source Software

Firms that do not only use but also *develop* OSS (commercial OSS) are the topic of Verani (2006) and Schmidtke (2006). In both cases OSS firms develop code and produce complementary products. Verani (2006) uses a duopoly model to analyze under which conditions firms produce more code, under an OSS rather than under a CSS regime. Verani finds that firms develop more code when their products are substitutes and OSS is chosen. For products being strong complements a CSS regime yields higher investments. Schmidtke (2006) uses a non-differentiated Cournot oligopoly model to analyze an industry with OSS firms only. Firms invest in a homogeneous private good (e.g. a computer server) and a homogeneous public good (i.e. OSS). Schmidtke describes the conditions under which government provision of OSS yields an increase of private provision of OSS (crowding in) rather than an decrease (crowding out). Furthermore, he analyzes the impact of market entry Schmidtke finds that increasing the number of firms in the market promotes welfare, while the effects on investments, prices and profits are ambiguous and depend on the parameter settings. Lambardi (2009) analyzes innovation investment in a Stackelberg duopoly with different initial technological levels. He compares a pure CSS duopoly (both firms are CSS firms) with a mixed duopoly where the leader is a CSS firm but the follower is an OSS firm. Both firms face consumers divided in two types: the first group just needs the basic software and does not require the complementary good, while the second group of consumers consume both. As the OSS firm cannot sell the software solely it can generate revenues only by selling the OSS-based complementaries to the second type of consumers. On the other hand the OSS firms receives development help from the OSS community which reduces its costs. Furthermore Lambardi (2009) also endogenizes the decision of the follower to become an OSS firm. The leader takes this option into account when choosing its optimal investment. His findings are that this OSS “threat” can yield to an decrease in overall innovation: the CSS leader invests less in order to restrain the follower from becoming an OSS firm. In this context Lambardi also points out that a government subsidy to OSS firms can be potentially harmful for innovation.

To the best of the author’s knowledge, the only contributions that model

commercial OSS and analyzes a mixed industry—that is  $n > 2$  firms (i.e. oligopoly) with OSS and CSS firms—are [Engelhardt \(2010\)](#), [Engelhardt and Maurer \(2010\)](#), and [Llanes and de Elejalde \(2009\)](#):

Applying a general two-stage model with horizontal product differentiation, [Engelhardt \(2010\)](#) analyzes the strategic nature of open source versus closed source business models and the role of OSS license-type. In stage one firms develop software, as OSS or CSS. If the license is of liberal type (e.g. the BSD), firms can use an OSS-CSS-mix, otherwise not. In stage two, firms bundle this software with complementary products and compete à la Cournot. The software determines the quality of the products. Based on the model findings [Engelhardt](#) points out that OSS lets firms avoid quality competition as they can cooperate on quality without an explicit contract. Regarding the strategic interactions [Engelhardt](#) finds that while CSS-decisions are always strategic substitutes, OSS-decisions can be strategic complements. Furthermore, CSS is a strategic substitute for OSS and vice versa. An important result of [Engelhardt's \(2010\)](#) model is that the type of OSS license plays a crucial role: Nash-equilibria with firms producing OSS for all parameters exist only for restricted licenses (e.g. the GPL). Next, he analyzes the equilibrium ratios of OSS/CSS firms in a mixed industry. The findings are that OSS-firms offer lower quality than their CSS-rivals, and where horizontal product differentiation is low CSS-based products have the largest market share. Using the model setup of [Engelhardt \(2010\)](#), [Engelhardt and Maurer \(2010\)](#) concentrate on welfare aspects. They mainly focus on cases without non-paid OSS volunteers (all OSS is commercial OSS). They point out that from a social point of view, the cost-saving benefits from OSS code-sharing are contrasted by the OSS cartel effect: code-sharing guarantees that no OSS firm can offer better software than any other OSS firm. This suppresses quality competition between OSS firms and restricts their code output. Competition from CSS firms weakens this quality-cartel effect. As result, the equilibria of mixed industries offer higher welfare than pure-OSS or pure-CSS. Furthermore, pure-OSS (pure-CSS) industries are sometimes stable against CSS (OSS) entry so that the mixed OSS/CSS state never occurs. Even where mixed OSS/CSS industries do exist, the proportion of OSS firms needed to stabilize the market against entry is always larger than the target ratio required to optimize welfare. Next, [Engelhardt and Maurer \(2010\)](#) discuss vari-



ous government interventions for addressing this imbalance with tax policy, funding of OSS development, and procurement preferences. According to them, the first-best solution in the model is to tax OSS firms and grant tax breaks to CSS firms. Conversely, government interventions that fund OSS development or establish procurement preferences for OSS software increase the gap between desired and actual OSS/CSS ratios still further. Despite this, funding OSS development can still improve welfare by boosting total (private plus government) OSS investment above the levels that a private OSS cartel would deliver.

Llanes and de Elejalde (2009) consider a model in which each firm sells packages consisting of a primary good (which can be OSS or CSS) and a complementary private good. Consumers have a binary demand (each consumer buys one package, or nothing) and idiosyncratic preferences so that they usually favor one firm's private good over others. However, rival firms can overcome this preference by investing in a technology that simultaneously increases the quality of both the primary good and also the complement. Llanes and DeElejalde present a two stage model in which a predetermined number of firms (a) decide whether to produce OSS or CSS in the primary good, and then (b) simultaneously decide the quality/price of the bundle that they will offer to consumers. They find that when most of the bundle's value comes from the primary good OSS firms find it hard to appropriate profits from their investment in an open complement. This leads to outcomes in which a small number of firms choose CSS and capture most of the market by delivering high quality code; the other firms become OSS and deliver comparatively low quality code at a low price. However, this situation changes where consumers value the complement roughly as much as the primary. In this case, the cost advantage of code-sharing dominates so that all firms choose to become OSS even though a hypothetical CSS firm would produce higher quality software. This (theoretical) CSS quality advantage reflects OSS firms' limited ability to recover quality gains from consumers. The advantage disappears in cases where most of the bundle's value comes from the complementary good.

### 3.4 Incentives and Role of Firms

Beside the motives of individual contributors, the engagement of firms is of interest. Here research analyzes the incentives for firms to contribute and the roles firms play within the OSS community.

#### 3.4.1 Firms Within the Community: Role, Influence and Relationships

Dahlander (2007) analyzes the role firms play in OSS projects, distinguishing between projects initiated by firms versus community-initiated ones and high versus low degree of firm participation. He focuses on de novo entrants (new organizational entities are formed) and draws conclusions for the management of OSS-based business models. The fact that an OSS project was founded by a firm rather than by the community has influence on its governance structure (West and O'Mahony 2008). Governance of community projects is largely pluralistic, while in firm-initiated projects the ultimate decisions are controlled by the company.<sup>15</sup> Furthermore, firm-initiated projects tend to have less restrictive licenses rather than the GPL. Here firms are an origin of more flexibly licensed OSS (Koski 2005), including the strategy of dual-licensing (Välimäki 2003) like e.g. an open source basic version and a closed source premium version.

Dahlander and Magnusson (2005) examines how the relationships that firms have to the OSS communities are connected with their way of doing business. He distinguishes three types of strategies. With the “symbiotic approach” the firm and the community gain, as the firm strongly contributes back. If the firm uses a “commensalistic approach” (firm uses input from the community), the firm gains while the community is indifferent. Finally, in a “parasitic” firm-community relationship the firm exploits, i.e. uses input without obeying norms, values and rules of the community. Not surprisingly, Dahlander and Magnusson (2005) report that firms who use a more symbiotic approach have more possibilities to influence the community. But such firms have to manage their dual roles of being a profit-seeking firm and part of the community. The competitive advantage of an OSS firm can hence be influenced by the relations it may have with OSS communities.

---

<sup>15</sup>This can also be a group of firms founding an alliance.

[Dahlander and Magnusson \(2006\)](#) emphasize in this context that in order to successfully cooperate with and gain from the community, firms have to have capabilities and in-house expertise.

[Henkel \(2009\)](#) focuses on the individual developers who establish the link between OSS firms and the OSS community. Here a principal-agent problem might exist, caused by the developer's double allegiance to firm and community. Thus some firms fear the risk of losing intellectual property, etc. [Henkel \(2009\)](#) uses data derived from interviews and a large-scale survey. He finds no evidence of commercially harmful behavior induced by OSS ideology ("Software has to be free" etc.). Also [Dahlander and Wallin \(2006\)](#) emphasize the role individuals play in the attempts of firms to unlock communities as complementary assets. Based on network analysis they show that firms sponsor individuals to act strategically within the OSS community.

Based on data from SourceForge (the leading online depository for OSS projects), [Lerner et al. \(2006\)](#) analyze the kind of projects to which firms contribute. They find that firms tend to contribute more to larger projects that grow faster (in terms of code lines). In their dataset, [Lerner et al. \(2006\)](#) can not find any consistent relationship between the type of OSS license and corporate contributions.

### 3.4.2 Why and Which Firms Go Open Source

[Henkel \(2006b\)](#) is focusing on the incentive for firms to contribute code back to the community even if they are not obliged to do so. The explanation he can draw from his empirical study is that the firms can expect a kind of reciprocal behavior: they receive informal development support from the community which even includes other firms. At the same time firms protect their intellectual property by revealing only parts of their code, only several modules respectively. [Henkel \(2006a\)](#) provides a duopoly model of why firms use and contribute to embedded Linux. The two firms require two technologies (or software modules) but value these technologies differently. As a result, each firm concentrates on producing the software it values most, publishes this as OSS and receives the OSS developed by the other firm. The economic logic of Henkel's model is basically a dual version of the 'exploitation of the great by the small' analyzed by [Olson and Zeckhauser](#)

(1966), [Olson \(1971\)](#). According to Henkel's model, firms are most likely to choose OSS business models when competition is low, and each firm's technology needs are different. In a duopoly model, [Baake and Wichmann \(2004\)](#) analyze the rationale of firms to publish parts of their software as OSS. Because of spillovers, publishing code as OSS reduces the firms' coding costs. But OSS encourages entry and thus increases the expenditures required to deter entry. In equilibrium, both firms publish open source code, which yields higher quality either because of reduced costs or because of the threat of entry. The latter effect is even stronger and dominates when the software-products are strong substitutes. In [Casadesus-Masanell and Llanes \(2009\)](#) consumers consume software and a complementary service. The software is further segmented into a core program which consumers can use as a free-standing unit, and extensions which are valueless without the core unit. Their model features a continuum of consumers with heterogeneous tastes, where each consumer buys just one package (bundle) or nothing (binary demand). Casadesus-Masanell and Llanes then examine how firms decide whether to develop one or both software components as OSS or CSS. Three cases are considered here: a monopoly, a firm vs. non-profit OSS project, and duopoly. They find, inter alia, that firms are more willing to open modules when (a) consumer demand for the complementary good is strong, and (b) the quality of OSS is boosted by exogenous user innovation at no cost.

The relationship between OSS and *entrepreneurship* is analyzed by [Gruber and Henkel \(2006\)](#), who focus on new ventures that apply embedded Linux. Based on data from personal interviews they conclude that the key challenges for new ventures discussed in the entrepreneurship literature are of less relevance for such OSS-firms. The empirical study of [Fritsch and Engelhardt \(2010\)](#) analyzes the characteristics of new businesses in the German ICT industry distinguished by how OSS-intensive their business model was. The analysis uses data about 680 start-ups and is based on a survey among founders of ICT firms in Germany conducted in fall 2009. This is the first study that directly compares OSS- and CSS-based start-ups and analyzes which aspects shape their OSS vs. CSS choices. The findings are that firms with OSS-based business models tend to be smaller in terms of staff and capital. OSS-firms also experience less shortages of capital. Furthermore, the data show that OSS business models seem to

be established nowadays, as only OSS-intensive start-ups in older cohorts have larger problems than their CSS counterparts to convince potential financiers to invest. According to [Fritsch and Engelhardt \(2010\)](#) their data do not indicate that the lower entry barriers for OSS firms are particularly attractive for start-ups with low human capital endowment or to necessity-motivated entrepreneurs.

There are other contributions providing comparative studies of OSS and CSS firms: [Lamastra Rossi \(2009\)](#) and [Harison and Koski \(2010\)](#). Based on a sample of 134 software solutions developed by Italian small and medium sized enterprises, [Lamastra Rossi \(2009\)](#) concludes that OSS solutions seems to be more innovative. [Harison and Koski \(2010\)](#) use survey data from 170 Finish software companies and analyze how the firms' properties shape their OSS-vs-CSS decision. They distinguish between firms with no OSS (firms that provide only CSS) and firms with OSS (either purely or as hybrid strategy, i.e. an OSS-CSS-mix). [Harison and Koski \(2010\)](#) find that the decision to use some vs. no OSS can be explained by several characteristics of the software firms. In particular, human capital (education) has a positive impact on OSS strategies, and firms that are younger and smaller more often apply OSS supply strategies. Software firms owned by one or two individuals or a family tend to be CSS-only. Finally, the magnitude of the service variety provided by the firms has a positive impact on the propensity to adopt OSS strategies.

### 3.5 Open Source Beyond Software

Some authors discuss the possibility to implement the open source paradigm in areas other than software. This must not be confused with research on online communities beyond OSS, like research on Wikipedia (e.g. [Gaio et al. 2009](#), [Ciffolilli 2003](#)). Here open source beyond software means applying similar open source mechanisms to other industries based on digital goods, i.e. "payoff-relevant bitstring[s]" ([Quah 2003](#)). For example, an important project of the open source movement in genomics-based research is the International Human Genome Project. Laboratories from all over the world jointly collaborate to map and sequence the human genome, with the resulting data deposited into the public domain.

Maurer (2008) discuss whether and how open source principles and incentives are suitable for the several stages of the drug discovery pipeline. Allarakhia et al. (2010) examine the mechanisms of cooperative knowledge production and dissemination in open source biopharmaceutical innovation. They analyze about 50 open source initiatives that focus on genomic, proteomic, and systems-based research. Based on this, Allarakhia et al. (2010) develop a two-player game model in order to further analyze the incentives to participate in open source biopharmaceutical initiatives. In her book “Biobazaar”, Janet Hope discuss the challenges and implications of applying open source principles to biotechnology. She argues that open source biotechnology would foster competition in the industry that today tends to be dominated by a few powerful players Hope (2008). Henkel and Maurer (2007) discuss the economics of open source synthetic biology, including the consequence of different access and usage rules regarding the community’s Registry of Standard Biological Parts. Finally, Roosendaal (2007) discusses the (legal) problems that occur when commercial companies are invited to join open source projects in biomedics. Such problems are a result of the tension between traditional proprietary regimes and open source approaches in this field.

#### **4 Summary: What Do Economists Know about Open Source Software?**

After a first decade of economic research on OSS, one can say that we know a lot about this issue. Economists understand quite well why ‘thousands of top-notch programmers contribute freely to the provision of’ OSS, and how the commercial and non-commercial agents manage to coordinate their provisions. The developers’ motives are basically a mix of extrinsic and intrinsic ones. For example, a programmer contributes code because he or she likes programming and at the same time he or she wants to generate a signal for the job market. OSS projects are governed by a set of sophisticated rules that enable them to stay open and permeable but at the same time delegates tasks to specific people with specific role. With their hierarchies in (passive) decision rights, OSS projects remind in some sense of firms, although firm-hierarchies are based on order.

The rationale for OSS licensing (instead of CSS licensing) as well as for the different types of OSS licenses is also understood—at least partly. In a nutshell, exclusive intellectual property can have its limits either because innovations depend on each other and are cumulative, or because of ex post transaction costs or hold-up problems. Here OSS can be an alternative, and sometimes even performs better—in terms of individual or social payoff, or both. However, all types of licenses are based on copyright law, and this explains why OSS activities are—among other country-specific institutional and cultural factors—positively affected by a good de facto protection of IPRs.

Besides differences in detail, the bottom line of the research on the impact of OSS on competition and market outcome is that the emergence of OSS increased market outcomes and welfare. However, recent research points out that an OSS-only world is also not favorable. It seems to be true that the coexistence of OSS and CSS is desirable as both principles have their pros and cons. This is also mirrored in the micro-level: Most firms in the ICT sector use both, OSS and CSS, in their business models. Furthermore, they use a wide range of license schemes and have different and sometimes quite sophisticated strategies to incorporate OSS and the OSS community into their everyday businesses. Nevertheless, the basic principle of OSS business models is the combination of the public OSS with complementary private goods or services.

The fact that OSS is successful and has its advantages has inspired various colleagues to ask whether the open source paradigm can be used in areas other than software. Namely open source biology/biotechnology is a recently discussed topic here. There are some promising aspects of this new research topic ‘open source beyond software’.

Without any doubt, the research on open source beyond software benefits from what we have learned from the software example. However, some of the results of the OSS case are too specific to draw general conclusion. Future research on open source therefore might try to focus more on general aspects that can be found not only in software. Knowing what is similar (or general) and what is different can then help to learn from OSS as one well-studied example of the open source principle.

## References

- Allarakhia, M., Kilgour, D. M. and Fuller, J. D.: 2010, Modelling the incentive to participate in open source biopharmaceutical innovation., *R&D Management* **40**(1), 50 – 66.
- Baake, P. and Wichmann, T.: 2004, Open source software, competition and potential entry, *Berlecon Research Papers* 5, Berlecon Research, Berlin.
- Benkler, Y.: 2002, Coase's penguin, or, Linux and the nature of the firm, *Yale Law Journal* **112**(3), 369–437.
- Bessen, J.: 2006, Open source software: Free provision of complex public goods, in J. Bitzer and P. Schröder (eds), *The Economics Of Open Source Software Development*, Elsevier, p. 57–81.
- Bessen, J. E. and Hunt, R. M.: 2004, The software patent experiment, in OECD (ed.), *Patents, Innovation and Economic Performance*, OECD, Paris.
- Bessen, J. and Hunt, R. M.: 2007, An empirical look at software patents, *Journal of Economics & Management Strategy* **16**(1), 157–189.
- Besten, M. d., Dalle, J.-M. and Galia, F.: 2008, The allocation of collaborative efforts in open-source software, *Information Economics and Policy* **20**(4), 316 – 322.
- Bitzer, J.: 2004, Commercial versus open source software: The role of product heterogeneity in competition, *Economic Systems* **28**(4), 369 – 381.
- Bitzer, J., Geishecker, I. and Schroeder, P.: 2010, Returns to open source software engagement: An empirical test of the signaling hypothesis, *Working Papers* 321, University of Oldenburg, Department of Economics.
- Bitzer, J. and Schröder, P. J. H.: 2007, Open source software, competition and innovation, *Industry and Innovation* **14**(5), 461 – 476.
- Blind, K., Edler, J. and Friedewald, M.: 2005, *Software Patents – Economic Impacts and Policy Implications*, Elgar, Cheltenham [u.a.].



- Brand, A. and Schmid, A.: 2005, Koordination in einem Open Source-Projekt, *Arbeitspapier aus dem dfg-projekt struktur und funktionsweise elektronischer arbeitsmärkte*.
- Brand, A. and Schmid, A.: 2006, Ist ein Open-Source-Projekt nur kooperativ? – Die Koordination der Zusammenarbeit im KDE-Projekt, in B. Lutterbeck, M. Bärwolff and R. A. Gehring (eds), *Open Source Jahrbuch 2006: Zwischen Softwareentwicklung und Gesellschaftsmodell*, Lehmanns Media, Berlin, pp. 123 – 138.
- Campbell-Kelly, M. and Garcia-Swartz, D. D.: 2009, Pragmatism, not ideology: Historical perspectives on IBM's adoption of open-source software, *Information Economics and Policy* **21**(3), 229 – 244.
- Casadesus-Masanell, R. and Ghemawat, P.: 2006, Dynamic Mixed Duopoly: A Model Motivated by Linux vs. Windows, *Management Science* **52**(7), 1072–1084.
- Casadesus-Masanell, R. and Llanes, G.: 2009, Mixed source, *Working Papers 09-06*, NET Institute.
- Ciffolilli, A.: 2003, Phantom authority, self-selective recruitment and retention of members in virtual communities: The case of Wikipedia, *First Monday* **8**(12). <http://firstmonday.org/>.
- Coase, R. H.: 1937, The nature of the firm, *Economica* **4**(16), 386–405.
- Crowston, K. and Howison, J.: 2005, The social structure of free and open source software development, *First Monday* **10**(2). <http://firstmonday.org/>.
- Crowston, K., Wei, K., Li, Q. and Howison, J.: 2006, Core and periphery in free/libre and open source software team communications, *System Sciences: HICCS (Hawaii International Conference) Proceedings*, System Sciences: HICCS (Hawaii International Conference) Proceedings, pp. 118a–118a.
- Daffara, C.: 2007, Business models in FLOSS-based software companies, *Workshop presentation at the 3rd conference on open source systems*. <http://opensource.mit.edu/papers/OSSEMP07-daffara.pdf>.

- Dahlander, L.: 2007, Penguin in a new suit: A tale of how de novo entrants emerged to harness free and open source software communities, *Industrial and Corporate Change* **16**(5), 913 – 943.
- Dahlander, L. and Magnusson, M. G.: 2005, Relationships between open source software companies and communities: Observations from nordic firms, *Research Policy* **34**(4), 481 – 493.
- Dahlander, L. and Magnusson, M. G.: 2006, Business models and community relationships of open source software firms, in J. Bitzer and P. J. Schröder (eds), *The Economics of Open Source Software Development*, Elsevier, Amsterdam, pp. 111 – 130.
- Dahlander, L. and Wallin, M. W.: 2006, A man on the inside: Unlocking communities as complementary assets, *Research Policy* **35**(8), 1243 – 1259.
- D’Antoni, M. and Rossi, M. A.: 2007, Copyright vs. copyleft licencing and software development, *Department of Economics University of Siena Working Paper 510*, Department of Economics, University of Siena.
- David, P. A. and Rullani, F.: 2008, Dynamics of innovation in an ‘open source’ collaboration environment: Lurking, laboring, and launching floss projects on sourceforge, *Industrial and Corporate Change* **17**(4), 647 – 710.
- David, P. A. and Shapiro, J. S.: 2008, Community-based production of open-source software: What do we know about the developers who participate?, *Information Economics and Policy* **20**(4), 364 – 398.
- David, P. A., Waterman, A. and Arora, S.: 2003, FLOSS-US. the free/libre/open source software survey for 2003, *Technical report*, Stanford Institute for Economic Policy Research.
- Davis, L. E. and North, D. C.: 1971, *Institutional Change and American Economic Growth*, Cambridge University Press.
- Demil, B. and Lecocq, X.: 2006, Neither market nor hierarchy nor network: The emergence of bazaar governance, *Organization Studies* **27**(10), 1447 – 1466.

- Demsetz, H.: 1967, Towards a theory of property rights, *American Economic Review* (2), 347–359.
- Economides, N. and Katsamakas, E.: 2006, Two-sided competition of proprietary vs. open source technology platforms and the implications for the software industry, *Management Science* 52(7), 1057–1071.
- Engelhardt, S. v.: 2008, Intellectual property rights and ex-post transaction costs: the case of open and closed source software, *Jena Economic Research Papers 2008-047*, Friedrich-Schiller-University Jena, Max-Planck-Institute of Economics.
- Engelhardt, S. v.: 2010, Quality competition or quality cooperation? License-type and the strategic nature of open source vs. closed source business models, *Jena Economic Research Papers 2010-034*, Friedrich-Schiller-University Jena and Max-Planck-Institute of Economics.
- Engelhardt, S. v. and Freytag, A.: 2010, Institutions, culture, and open source, *Jena Economic Research Papers 2010-010*, Friedrich-Schiller-University Jena and Max-Planck-Institute of Economics.
- Engelhardt, S. v., Freytag, A. and Schulz, C.: 2010, On the geographic allocation of open source software activities, *Jena Economic Research Papers 2010-009*, Friedrich-Schiller-University Jena and Max-Planck-Institute of Economics.
- Engelhardt, S. v. and Maurer, S. M.: 2010, The new (commercial) open source: Does it really improve social welfare?, *Berkeley Goldman School of Public Policy Working Paper 10-001*.
- Fink, M.: 2002, *Business and Economics of Linux and Open Source*, Prentice Hall Professional Technical Reference.
- Franck, E. and Jungwirth, C.: 2003, Reconciling rent-seekers and donators—the governance structure of open source, *Journal of Management and Governance* 7(4), 401 – 421.
- Fritsch, M. and Engelhardt, S. v.: 2010, Who starts with open source? Institutional choice of German ICT start-ups. mimeo.

- Gaio, L., Rossi, A., Besten, M. d. and Dalle, J.-M.: 2009, Coordination, division of labor, and open content communities: Template messages in wiki-based collections, *DISA Working Papers 0903*, Department of Computer and Management Sciences, University of Trento, Italy.
- Garzarelli, G.: 2003, Open source software and the economics of organization, *Industrial Organization 0304003*, EconWPA.
- Gaudeul, A.: 2005, Public provision of a private good: What is the point of the BSD license?, *Industrial Organization 0511002*, EconWPA.
- Gaudeul, A.: 2008, Consumer welfare and market structure in a model of competition between open source and proprietary software, *MPRA Paper 19555*, University Library of Munich, Germany.
- Gehring, R. A.: 2006, The institutionalization of open source, *Poiesis & Praxis: International Journal Of Technology Assessment and Ethics Of Science* 4(1), 54–73.
- Ghosh, R. A.: 2006, Economic impact of open source software on innovation and the competitiveness of the information and communication technologies (ICT) sector in the EU, *Study*, European Commission.
- Ghosh, R. A., Glott, R., Krieger, B. and Robles, G.: 2002a, FLOSS final report, part 3: Basics of open source software markets and business models, *Technical report*, International Institute of Infonomics, University of Maastricht.
- Ghosh, R. A., Glott, R., Krieger, B. and Robles, G.: 2002b, FLOSS final report, part 4: Survey of developers, *Technical report*, International Institute of Infonomics, University of Maastricht.
- Giuri, P., Rullani, F. and Torrì, S.: 2008, Explaining leadership in virtual teams: The case of open source software, *Information Economics and Policy* 20(4), 305 – 315.
- Gonzalez-Barahona, J. M., Robles, G., Andradas-Izquierdo, R. and Ghosh, R. A.: 2008, Geographic origin of libre software developers., *Information Economics and Policy* 20(4), 356 – 363.

- Graham, S. and Somaya, D.: 2004, The use of patents, copyrights and trademarks in software: Evidence from litigation, in OECD (ed.), *Patents, Innovation and Economic Performance*, OECD, Paris.
- Gruber, M. and Henkel, J.: 2006, New ventures based on open innovation – an empirical analysis of start-up firms in embedded Linux, *International Journal of Technology Management* **33**(4), 356–372.
- Hall, B. H. and MacGarvie, M.: 2006, The private value of software patents, *NBER Working Papers 12195*, National Bureau of Economic Research, Inc.
- Harison, E. and Koski, H.: 2010, Applying open innovation in business strategies: Evidence from finnish software firms, *Research Policy* **39**(3), 351–359.
- Hars, A. and Ou, S.: 2002, Working for free? - Motivations of participating in open source projects, *International Journal of Electronic Commerce* **6**(3), 25 – 39.
- Henkel, J.: 2006a, The jukebox mode of innovation - a model of commercial open source development, *DRUID Working Papers 06-25*, DRUID, Copenhagen Business School, Department of Industrial Economics and Strategy/Aalborg University, Department of Business Studies.
- Henkel, J.: 2006b, Selective revealing in open innovation processes: The case of embedded linux, *Research Policy* **35**(7), 953 – 969.
- Henkel, J.: 2009, Champions of revealing–the role of open source developers in commercial firms, *Industrial and Corporate Change* **18**(3), 435 – 471.
- Henkel, J. and Maurer, S. M.: 2007, The economics of synthetic biology, *Molecular Systems Biology* **3**(Article number: 117).
- 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.

- Hippel, E. v.: 2005, Open source software projects as user innovation networks, in E. v. Hippel, J. Feller, B. Fitzgerald, S. A. Hissam and K. R. Lakhani (eds), *Perspectives on Free and Open Source Software*, MIT Press, Cambridge and London, pp. 267 – 278.
- Hippel, E. v. and Krogh, G. v.: 2003, Open source software and the 'private-collective' innovation model: Issues for organization science, *Organization Science* **14**(2), 209 – 223.
- Hofstede, G. H.: 1991, *Cultures and organizations: Software of the mind*, McGraw-Hill, London and New York.
- Hope, J.: 2008, *Biobazaar: The Open Source Revolution and Biotechnology*, Harvard University Press.
- Iannacci, F. and Mitleton-Kelly, E.: 2005, Beyond markets and firms: The emergence of Open Source networks, *First Monday* **10**(5). <http://firstmonday.org/>.
- Jensen, C. and Scacchi, W.: 2007, Role migration and advancement processes in OSSD projects, *International Conference on Software Engineering*, number 29 in *International Conference on Software Engineering*, Minneapolis, MN, USA.
- Johnson, J. P.: 2002, Open source software: Private provision of a public good, *Journal of Economics & Management Strategy* **11**(4), 637–662.
- Johnson, J. P.: 2006, Collaboration, peer review and open source software, *Information Economics and Policy* **18**(4), 477 – 497.
- Kahin, B.: 2004, Through the lens of intangibles: What patents on software and services reveal about the system, in OECD (ed.), *Patents, Innovation and Economic Performance*, OECD, Paris.
- Koski, H.: 2005, OSS production and licensing strategies of software firms, *Review of Economic Research on Copyright Issues* **2**(2), 111 – 125.
- Krogh, G. v., Spaeth, S. and Lakhani, K. R.: 2003, Community, joining, and specialization in open source software innovation: A case study, *Research Policy* **32**(7), 1217 – 1241.

- Laat, P. B. d.: 2005, Copyright or copyleft?: An analysis of property regimes for software development, *Research Policy* **34**(10), 1511 – 1532.
- Laat, P. B. d.: 2007, Governance of open source software: State of the art, *Journal of Management and Governance* **11**(2), 165 – 177.
- Lakhani, K. R. and Hippel, E. v.: 2003, How open source software works: 'free' user-to-user assistance, *Research Policy* **32**(6), 923 – 943.
- Lakhani, K. R., Wolf, B., Bates, J. and DiBona, C.: 2002, The boston consulting group/OSDN hacker survey, <http://www.osdn.com/bcg/>.
- Lakhani, K. R. and Wolf, R. G.: 2005, Why hackers do what they do: Understanding motivation and effort in free/open source software projects, in J. Feller, B. Fitzgerald, S. A. Hissam and K. R. Lakhani (eds), *Perspectives on Free and Open Source Software*, MIT Press, pp. 3 –22.
- Lamastra Rossi, C.: 2009, Software innovativeness. A comparison between proprietary and free/open source solutions offered by italian SMEs., *R&D Management* **39**(2), 153 – 169.
- Lambardi, G.: 2009, Software innovation and the open source threat, *NET Institute Working Paper 09-15*, The Networks, Electronic Commerce, and Telecommunications (NET) Institute.
- Lancashire, D.: 2001, Code, culture and cash: The fading altruism of open source development, *First Monday* **6**(12). <http://firstmonday.org>.
- Langlois, R. N. and Garzarelli, G.: 2008, Of hackers and hairdressers: Modularity and the organizational economics of open-source collaboration, *Industry and Innovation* **15**(2), 125 – 143.
- Lattermann, C. and Stieglitz, S.: 2005, Framework for governance in open source communities, *Proceedings of the 38th Annual Hawaii International Conference on System Sciences 2005*, pp. 192–202.
- Lee, G. K. and Cole, R. E.: 2003, From a Firm-Based to a Community-Based Model of Knowledge Creation: The Case of the Linux Kernel Development, *Organization Science* **14**(6), 633–649.

- Lerner, J., Pathak, P. A. and Tirole, J.: 2006, The dynamics of open-source contributors, *American Economic Review* **96**(2), 114.
- Lerner, J. and Tirole, J.: 2000, The simple economics of open source, *NBER Working Paper Series* 7600.
- Lerner, J. and Tirole, J.: 2002, Some simple economics of open source, *Journal of Industrial Economics* **50**(2), 197–234.
- Lerner, J. and Tirole, J.: 2005a, The economics of technology sharing: Open source and beyond, *Journal of Economic Perspectives* **19**(2), 99 – 120.
- Lerner, J. and Tirole, J.: 2005b, The scope of open source licensing, *Journal of Law, Economics and Organization* **21**(1), 20–56.
- Lerner, J. and Zhu, F.: 2007, What is the impact of software patent shifts? Evidence from lotus v. borland, *International Journal of Industrial Organization* **25**(3), 511–529.
- Lessig, L.: 1999, *Code and Other Laws Of Cyberspace*, Basic Books, New York, NY.
- Lessig, L.: 2006, *Code: Version 2.0*, Basic Books.
- Llanes, G. and de Elejalde, R.: 2009, Industry equilibrium with open source and proprietary firms, *Harvard Business School Working Papers* 09-149, Harvard Business School.
- Markus, M. L.: 2007, The governance of free/open source software projects: Monolithic, multidimensional, or configurational?, *Journal of Management and Governance* **11**(2), 151 – 163.
- Maurer, S. M.: 2008, Open source biology: Finding a niche (or maybe several), *UMKC Law Review* **76**(2).
- Maurer, S. M. and Scotchmer, S.: 2006, Open source software: the new intellectual property paradigm, in T. Hendershott (ed.), *Handbook of Economics and Information Systems*, Elsevier, pp. 285–319.



- Mustonen, M.: 2001, Copyleft - the economics of linux and other open source software, *Discussion Papers of the Department of Economics, University of Helsinki* 493, University of Helsinki.
- Mustonen, M.: 2003, Copyleft—the economics of linux and other open source software., *Information Economics and Policy* **15**(1), 99 – 121.
- North, D. C.: 1990, *Institutions, Institutional Change and Economic Performance*, Cambridge Univ. Press, Cambridge.
- North, D. C.: 1994, Economic performance through time, *American Economic Review* **84**(3), 359–368.
- Olson, M.: 1971, *The Logic Of Collective Action*, Harvard Univ. Press, Cambridge, Mass.
- Olson, Mancur, J. and Zeckhauser, R.: 1966, An economic theory of alliances, *The Review of Economics and Statistics* **48**(3), 266–279.
- O'Mahony, S.: 2003, Guarding the commons: How community managed software projects protect their work, *Research Policy* **32**(7), 1179 – 1198.
- O'Mahony, S. and Ferraro, F.: 2007, The emergence of governance in an open source community., *Academy of Management Journal* **50**(5), 1079 – 1106.
- Osterloh, M. and Rota, S.: 2007, Open source software development—just another case of collective invention?, *Research Policy* **36**(2), 157 – 171.
- Pilch, H.: 2004, Why are software patents so trivial?, in OECD (ed.), *Patents, Innovation and Economic Performance*, OECD, Paris.
- Polanski, A.: 2007, Is the General Public Licence a rational choice?, *Journal of Industrial Economics* **55**(4), 691–714.
- Quah, D.: 2003, Digital goods and the new economy, *CEP Discussion Papers* 563, London School of Economics, London.
- Ramanujam, P.: 2007, Who produces open source software, *MPRA Paper* 4253, University Library of Munich, Germany.

- Raymond, E. S.: 1998, The cathedral and the bazaar, *First Monday* 3(3).  
<http://firstmonday.org>.
- Riehle, D.: 2007, The economic motivation of open source software: Stakeholder perspectives, *IEEE Computer* 40(4), 25 – 32.
- Robles, G. and Gonzalez-Barahona, J. M.: 2006, Geographic location of developers at SourceForge, *MSR '06: Proceedings of the 2006 international workshop on Mining software repositories*, ACM, New York, NY, USA, pp. 144–150.
- Robles, G., Scheider, H., Tretkowski, I. and Weber, N.: 2001, Who is doing it? A research on libre software developers, *Technical report*, Technische Universität Berlin.
- Roosendaal, A.: 2007, Biomedics and creative commons: A perfect match?, *International Journal of Intellectual Property Management* 1(4), 316 – 324.
- Rossi, M. A.: 2006, Decoding the free/open source software puzzle: A survey of theoretical and empirical contributions, in J. Bitzer and P. Schröder (eds), *The Economics Of Open Source Software Development*, Elsevier, pp. 15–55.
- Sadowski, B. M., Sadowski-Rasters, G. and Duysters, G.: 2008, Transition of governance in a mature open software source community: Evidence from the debian case, *Information Economics and Policy* 20(4), 323 – 332.
- Saint-Paul, G.: 2003, Growth effects of nonproprietary innovation, *Journal of the European Economic Association* 1(2-3), 429–439.
- Schiff, A.: 2002, The economics of open source software. a survey of the early literature, *The Review Of Network Economics* 1(1), 66 – 74.
- Schmidtke, R.: 2006, Private provision of a complementary public good, *CESifo Working Paper Series CESifo Working Paper No. 1756*, CESifo Group Munich.

- Schwarz, M. and Takhteyev, Y.: 2010, Half a century of public software institutions: Open source as a solution to hold-up problem, *Journal of Public Economic Theory* **12**(4), 609–639.
- Schweik, C. M. and Semenov, A.: 2003, The institutional design of open source programming: Implications for addressing complex public policy and management problems, *First Monday* **8**(1). <http://firstmonday.org/>.
- Scotchmer, S.: 2004, *Innovation and Incentives*, The MIT Press.
- Scotchmer, S.: 2010, Openness, open source, and the veil of ignorance, *American Economic Review* **100**(2), 165–71.
- Sen, R.: 2007, A strategic analysis of competition between open source and proprietary software, *Journal of Management Information Systems* **24**(1), 233 – 257.
- Sen, R., Subramaniam, C. and Nelson, M. L.: 2008, Determinants of the Choice of Open Source Software License, *Journal of Management Information Systems* **25**(3), 207 – 239.
- Spence, M.: 1973, Job market signaling, *The Quarterly Journal of Economics* **87**(3), pp. 355–374.
- Stallman, R. M.: 1999, The GNU operating system and the free software movement, in C. DiBona, S. Ockman and M. Stone (eds), *Open Sources: Voices From The Open Source Revolution*, O'Reilly Media, Inc., pp. 53–70.
- Steinmueller, W. E.: 1996, The U.S. software industry: An analysis and interpretive history, in D. C. Mowery (ed.), *The International Computer Software Industry: A Comparative Study of Industry Evolution and Structure*, Oxford University Press, Oxford, New York, pp. 15–51.
- Välimäki, M.: 2003, Dual licensing in open source software industry, *Systèmes d'Information et Management* **8**(1), 63 – 75.
- Verani, S.: 2006, Open source development in a differentiated duopoly, *Economics Discussion / Working Papers 06-05*, The University of Western Australia, Department of Economics.

- Weber, S.: 2000, The political economy of open source software, *UCAIS Berkeley Roundtable on the International Economy, Working Paper Series 1011*, UCAIS Berkeley Roundtable on the International Economy, UC Berkeley.
- Weber, S.: 2004, *The Success Of Open Source*, Harvard University Press.
- Wendel de Joode, R. V., Bruijn, J. A. d. and Eeten, M. J. G. V.: 2003, *Protecting the Virtual Commons – Self-Organizing Open Source and Free Software Communities and Innovative Intellectual Property Regimes*, T.M.C. Asser Press, The Hague.
- West, J. and Gallagher, S.: 2006, Challenges of open innovation: the paradox of firm investment in open-source software., *R&D Management* **36**(3), 319 – 331.
- West, J. and O'Mahony, S.: 2008, The role of participation architecture in growing sponsored open source communities, *Industry and Innovation* **15**(2), 145 – 168.
- Wynn, D. E.: 2003, Organizational structure of open source projects: A life cycle approach, *Proceedings of the 7th Annual Conference of the Southern Association for Information Systems*, pp. 285–290.