FLOSS Final Report – Part 2

Free/Libre Open Source Software: Survey and Study

**Firms' Open Source Activities:
Motivations and Policy Implications**

Disclaimer:

The views expressed in this report are those of the authors and do not necessarily reflect those of the European Commission. Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use that might be made of the following information.

Nothing in this report implies or expresses a warranty of any kind. Results from this report should only be used as guidelines as part of an overall strategy. For detailed advice on corporate planning, business processes and management, technology integration and legal or tax issues, the services of a professional should be obtained.

Names and trademarks mentioned in the report are the property of their respective owners.

V 1.0 - 020630

Table of Contents

1 Introduction

The engagement in Open Source (OS) projects and the creation of Open Source software (OSS) is seen by most commentators and researchers as a private activity. Individual persons contribute to Open Source projects for private reasons, be it just for fun, in the hope of getting something in return, or because this activity yields private indirect returns on the labour market.

Creation of Open Source software mostly seen as private activity

While the assumption of altruistic behaviour dominated the Open Source discussion in earlier years, the current work emphasises more reciprocity or individual labour market considerations. E.g., Lerner and Tirole (2002) argue that a programmer can signal his coding abilities by participating in Open Source projects. This should raise his expected future wage or give him access to programming jobs, as already Raymond (2000, Chapter 5) has pointed out, although he considers the latter as rare and marginal motivation for most hackers.

Current emphasis on reciprocity and labour market motivations

While the FLOSS developer survey has shown that developing new skills and sharing their knowledge were the primary motivations for participation in Open Source projects, the expected monetary benefits are not negligible: About a third of the surveyed developers indicated that improving their job opportunities was a motivation for their Open Source activity. (Ghosh et al., 2002).

Improving job opportunities important reason for participation

Although important for explaining the Open Source phenomenon, this focus on the individual programmer neglects an important Open Source driver: firms. Part of the Open Source community consists of individuals employed explicitly for developing Open Source software. Ghosh et al. (2002) point out that about a third of the surveyed developers are being paid directly for developing Open Source software. Thus, their contribution to Open Source projects is the result of firms' deliberate decisions to finance the development of Open Source software. In addition there are several examples of companies that have made available formerly proprietary software as Open Source software.

firms' deliberate Open Source activities important

The sheer amount of resources devoted by companies to OS development can be large. IBM alone claims to have spent $1 billion on Linux alone and is also active in several other Open Source projects (Wilcox, 2000). Other companies also devote considerable resources to the development of OS software. Due to their size these initiatives are important contributions to the development of OS software in total.

Some firms' activities are large

Despite this size the companies' motivation behind their OS engagement is not as well understood as the motivation of individual developers. Although discussed in passing by some authors like Lerner and Tirole (2002) or Schmidt and Schnitzer (2002), much less attention has been devoted to firms' Open Source activity than to Open Source activity of individuals.

firms' Open Source activities less well understood than individuals'

Also in the context of public policy the firms' Open Source activities are of importance. If commercial firms produce Open Source software alongside their proprietary software, a strict distinction between the "commercial world" and the "free world", as it can sometimes be seen in the public policy debate,[1] might not be sensible. Rather

Public policy has to consider firms' Open Source activities

one would have to ask, whether commercial firms produce the socially optimal amount of Open Source software, just like economics asks whether firms engage in a socially optimal amount of basic research. Even if one comes to the conclusion that this amount is sub-optimal, one has to take firms' behaviour into account when designing policy measures intended to foster the use of Open Source software.

Organisation of this paper   In the remainder of this paper, we will proceed as follows. Section 2 discusses the engagement of the world's 25 largest software companies in OS activities. Section 3 groups these activities. This analysis forms the basis for the subsequent work. Section 4 discusses the motivation(s) behind the companies' engagement in OS development using analogies for similar activities from economic theory. This framework is also used to discuss which policy and regulation activities influence the companies' behaviour and which conclusions can be drawn for government bodies and regulatory authorities. Section 5 concludes.

---

1. Often this debate is stylised to (commercial) Microsoft versus (free, open and not commercial) Linux. The many possible shades of grey are assiduously ignored.

2   Large firms' Open Source activities

This section describes the OS activities of globally large software firms. To have a reasonable foundation for the subsequent discussion, we investigated the OS development activities of the world's 25 largest software companies.

Section describes OS activities of large software firms

We have chosen large firms mainly for two reasons: First of all, we expect the economic motives behind their OS engagements to be more pronounced in larger than in smaller companies: As these entities are large and professionally managed, preferences of individuals play a lesser role than they might in small companies. Secondly, these companies typically have sufficient funds available to pursue the strategies they want to pursue. If there are arguments in favour of an OS engagement, we would expect to see the engagement take place. Smaller companies might in principle come to similar conclusions but might refrain from pursuing these strategies due to lack of funds or due to the need to focus on more important things.

Large firms chosen for two reasons ...

The basis for this research is the Software Magazine's 2001 list of the world's largest software companies,[2] which is published each year and is by now a well-established ranking. As these 25 companies are typically large, many of them are not software pureplays. Some of them are even not in the business of standard software, but are providers of IT services like consulting and systems integration (e.g. Accenture, PWC) or outsourcing services (EDS).

... on the basis of Software Magazine's annual ranking

These companies' web sites as well as major search engines have been used to look for signs of activity in OS development. These activities had to be indicated as being the companies' activities and not that of individual programmers working there. The outcome therefore ignores "passive" Open Source support such as letting employees work on OS projects on company time[3] and thus underestimates the companies' total commitment of resources to OS projects.

Web sites and search engines investigated

There are at least two good reasons, why one should be able to find information about firms' Open Source activities this way: First of all, OS development takes place publicly and typically tries to involve as many contributors as possible. And secondly, Open Source – especially Linux – is a topic with generally positive connotation and high growth expectations for the future. Marketing departments therefore have an incentive to let the world know about a company's Open Source activities if there are any.

Incentives for firms to make their OS activities known

---

2.   http://www.softwaremag.com.

3.   Some empirical evidence for the importance of the latter is given in Berlecon Research (2002).

Compamy	Open Source activities
IBM	Major OS activities. URL: http://www-124.ibm.com/developerworks/oss/
Microsoft	No OS activities visible. Microsoft proposes model of "shared source" as alternative to open source. URL: http://www.microsoft.com/licensing/sharedsource/
PriceWaterhouse-Coopers	No OS activities visible. Topic is discussed on web-site from consulting point.
EDS	Occasional OS activities. Process data format eXT is proposed to continue as OS after initial development by EDS. Also Dynamator, a program for maintaining server pages and developed by an EDS programmer is OS.
Oracle	No OS activities visible.
Hewlett-Packard	Major OS activities. URL: http://www.hp.com/products1/linux/
Accenture	No OS activities visible. Topic is discussed on web-site from consulting point.
Cap Gemini Ernst & Young	No OS activities visible.
Compaq	Major OS activities. URL: http://opensource.compaq.com/ URL: http://www.compaq.com/products/software/linux/
Unisys	No OS activities visible.
SAP	Major OS activities. URL: http://www.sap.com/solutions/technology/linux/ URL: http://www.sabdb.org
Computer Associates	Major OS activities. Co-founder of Open Source Development Lab. URL: http://www.osdl.org
Hitachi	Major OS activities. URL: http://oss.hitachi.co.jp/index-e.html
Sun Microsystems	Major OS activities. URL: http://www.sunsource.net
NCR	No OS activities visible.
Compuware	No OS activities visible. However, development environment shipped with Compurware product OptimalJ is based on the open source Integrated Development Environment (IDE) NetBeans.
Siebel Systems	No OS activities visible.
PeopleSoft	No OS activities visible.
SunGard Data Systems	No OS activities visible.
Fiserv	No OS activities visible.
Computer Sciences Corp.	No OS activities visible.

Source: The Top 25 Companies are from Software Magazine's 2001 Software 500. Their open source activity has been researched by Berlecon Research in March 2002.

Compamy	Open Source activities
BMC Software	Occasional OS activities. Cooperation with The Open Group to develop an open source Management Service Broker. URL: http://www.opengroup.org/
EMC	No OS activities visible, but development of OS part of job descriptions for currently open positions.
Cadence Design Systems	Major OS activities. TestBuilder C++ testbench class library to be available through Open Source license. URL: http://www.testbuilder.net Engagement in OpenAccess coalition for standard electronic design database. URL: http://OpenEDA.org
Adobe	Occasional OS activities. Mostly Python plug-ins for Adobe products. URL: http://opensource.adobe.com/

Source: The Top 25 Companies are from Software Magazine's 2001 Software 500. Their open source activity has been researched by Berlecon Research in March 2002.

In a first step the companies were classified into one of three groups: "Major OS activities", "Occasional OS activities" and "No activities visible". Members of the first group are active in several Open Source projects. Typically companies from the first group also have a whole section of their web site devoted to their OS activities of which the URL is provided. The results are compiled in table 2-1.

Classification into three groups

Summarizing the results from table 2-1 one can see that almost one third of the 25 largest software companies (32%) do engage in major OS development activities. 12% (3 companies) have smaller projects and the majority (56%) does not have any visible open source projects. At these companies, individuals might be involved in OS activities, though. Taking into account that OS activities mean that these companies invest considerable resources to provide the public with software they cannot (directly) make money off, this number is astonishingly high.

One third engages in major Open Source activities

To be able to understand the OSS projects and their role within the selection of software and other products offered by these companies, it is useful to consider the OS activities of the 9 companies with major OS engagements in more detail. This simple description helps to derive common features of the Open Source software components furthered by these firms.

Description of projects in detail to derive common features

Within the Open Source strategies of these companies, a fewelements are very closely related and cannot always be separated in a satisfactory way. The first such pair is Open Source software and open standards, as development of standards and reference implementations of these standards often go hand-in-hand. The second pair of this kind is the support of Open Source software (e.g. Linux) in a company's applications and the development of the Open Source software itself. Especially more complicated applications require additional Open Source code to be written until an application will be compatible with the Open Source software.

Closely related are standards and software as well as support and development of OSS

IBM

IBM's bet on a standardised Linux replacing many Unixes

IBM is active in a variety of Open Source projects. Within the Linux Technology Center alone, around 70 Linux-related projects are listed toward which IBM contributes. This reflects IBM's strong commitment to Linux: In December 2000, then CEO Louis Gerstner announced that IBM was planning to spend $1 billion on Linux (Wilcox, 2000). The main motivation behind this was the belief that in the medium term proprietary Unix systems would loose their market dominance to a unified standardised Linux.[4]

IBM's Linux investment already recouped

While the real investment in Linux might have been somewhat lower than stated, according to industry observers IBM has taken more advantage of Linux and the open-source movement than any of their competitors (Shankland, 2002a). IBM claims to have almost recouped its investment in the first year through increased sales of software and systems. Meanwhile IBM has also ported Linux to their mainframe systems. And as a consequence almost all of the mainframe processing capacity sold by IBM in the fourth quarter of 2001 was for Linux (Shankland, 2002b).

Other Open Source projects concerned with Java, XML or Web Services

But IBM is also involved in a few dozen other Open source projects. Many of these are focused on new technology trends like Java, XML or Web Services. As IBM is selling a large variety of software and services based on these or adjacent emerging standards and technologies (e.g. its application servers and related products from the WebSphere family), its Open Source activity can help to increase mind share for the standards and thus ultimately for IBM's products.

Hewlett-Packard

HP active in the field of Linux and several other areas

HP's major activities in Open Source projects are also in the field of Linux. HP is a sponsor of the Linux Standards Base (LSB). The LSB's goal is to develop and promote a set of standards that will increase compatibility among Linux distributions and enable software applications to run on any compliant Linux system. HP is also a member of the Open Source Development Lab (OSDL, see Computer Associates) and a member of the GNOME foundation (see Sun). HP has in addition provided some enhancements to basic Open Source infrastructure software like Apache, Squid and Samba.

Hewlett-Packard's Open Source strategy centres on Linux

Hewlett-Packard's Open Source strategy centres on Linux (Shacklett, 2001). HP considers Linux to be the optimal choice of operating system under certain circumstances. Much of HP's sale is for solutions, i.e., the combination of hardware, operating system, possibly additional software, and services. Within these packages, the operating system is only one element. As HP is in addition providing its customers with solutions across several platforms, HP also has to provide Linux for those market segments where it is demanded.[5] This involves the development of software components that optimise the use of Linux on HP's hardware.

HP learned early to combine their products with operating systems it does not own

IDC (2001) points out that HP has an advantage compared to other hardware manufacturers for incorporating Linux into their solutions. According to them, HP has learned early to sell their hardware and solutions together with operating systems it does not own (e.g. with Microsoft Windows) – quite contrary to other companies like

---

4.   So far, Linux itself is not fully standardised. There are different distributors (RedHat, SuSE, Caldera, etc.), which combine different pieces of software into their Linux distribution. Although the Linux kernel is identical, there are many differences in the remainder. This gave rise to several Linux standardisation projects like the Linux Standards Base or Unified Linux, the latter being the latest initiative.

5.   HP: HP's Linux Strategy, http://www.hp.com/products1/linux/linux_strategy.html.

Sun, who have until recently shipped their hardware only with their own operating system.

Compaq

Compaq is hosting a variety of Open Source projects for software that runs on Compaq computers. Of the 19 projects listed, 14 are related to Linux software and often provide special drivers or utilities for Compaq hardware. Some non-Linux projects are concerned with clustering, others provide general utilities. Most projects directly profit Compaq which, according to it's own account, has a very strong position in providing hardware for Linux-based servers (Compaq, 2001).

Compaq's OS projects provide Linux components for its hardware

SAP

SAP has made its database, SAP DB, available under the GNU public license. SAP DB is an open, SQL-based, relational database system that provides high availability and performance scaling from small to very large implementations. This move can be seen as a strategy to increase the databases market share among SAP users (Hurd, 2001). Most SAP using companies do not use SAP DB but rather products from Oracle, IBM or Microsoft. With SAP DB being Open Source, they have additional incentives to use SAP DB, e.g. due to no upfront costs (they have to pay a support fee, though), an increased pool of trained programmers, better security due to open source code and an increasing number of third party support tools.

SAP made database available under GNU public license

SAP also hosts several programmers from different companies in its so-called SAP Linux lab with the purpose to ensure that its platform mySAP is running on Linux. The lab organises the release of mySAP on Linux and processes Linux-specific support problems. The lab works closely together with the Linux community in optimising Linux. According to SAP, its support for Linux has several reasons: Many SAP customers want to combine Intel hardware and their Unix knowledge. At the same time porting SAP to Linux was relatively easy. In addition, SAP currently has to support a variety of Unix flavours. For this purpose, SAP has to maintain, test and support each possible combination of OS, database, hardware and mySAP release. A consolidation of these variants would in their opinion be beneficial to consumers and producers alike. SAP regards Linux as having a realistic chance of becoming a unified Unix running on different platforms.[6]

Support of Linux in expectation of unified operating system

Computer Associates

Computer Associates (ca) is one of the sponsors of the Open Source Development Lab (OSDL),[7] an initiative to provide Open Source developers with computing resources to build data centre and telco class enhancements into Linux and its Open Source software stack, enabling Linux to become the leading Unix Operating System for e-business development and deployment.

ca sponsors Open Source Development Lab

Computer Associates markets several products running on Linux. Most of these are enterprise solutions, often for enterprise infrastructure management or enterprise information management. The Linux user group targeted with the OSDL initiative would also be the user group for these products. Therefore ca has an interest in wid-

Product sales would benefit from strong Linux as envisioned by OSDL

---

6. SAP: mySAP Technology on Linux, Frequently Asked Questions, http://www.sap.com/solutions/technology/linux/faq/tech_faq.asp.
7. The other sponsors are a group of Linux and hardware companies: Alcatel, Caldera, Cisco, Covalent, Dell, Fujitsu, Hitachi, Intel, Linuxcare, Miracle, Mitsubishi, Montavista, NEC, Nokia, SuSE, Toshiba, Turbolinux and VA Software.

ening the number of Linux instalments for which they can deliver additional software.

Hitachi

Hitachi supports internationalisation of Linux

Hitachi is mostly involved in Linux development. Six of the seven projects listed on the web site are concerned with Linux. Some of them focus on Linux for Hitachi hardware, others provide tools for Linux development or help in internationalising the operating system. (Hitachi is Japanese, and support for Japanese Kanji characters in software has for a long time been a difficult issue.) Hitachi is also a sponsor of the Open Source Development Lab (OSDL).

Sun Microsystems

Sun actively contributes to a variety of free and open source projects

Sun actively contributes to a variety of free and open source projects, including: OpenOffice (the Open Source version of its office suite StarOffice), GNOME, Mozilla, Apache, NetBeans, X-Windows, WBEMsource Initiative, the University of Michigan NFS version 4 Linux port, the Grid Engine Project, and Project JXTA. In addition, Sun has recently raised its bets on the Linux operating system and will ship an increasing variety of hardware with Linux (Sun, 2002).

Diverse set of motivations behind Sun's OS activities

These projects are rather diverse and so are the motivations behind Sun's engagement in Open Source projects. Buying the proprietary StarOffice and later releasing it as Open Source software OpenOffice, e.g., reportedly was mainly a marketing move to increase Sun's reputation in the Open Source community and at the same time weakening Microsoft (McMillan, 2001).

Some Open Source initiatives support parts of Sun's operating system

Other Open Source initiatives centre on products that are part of Sun's operating system. X-Windows is an example for this, where Sun is active in X.org, the consortium empowered with the stewardship and collaborative development of the X-Window system technology and standards. Sun's engagement in the development of GNOME, a desktop for Linux, which will replace Sun's Common Desktop Environment (CDE) on its Solaris-equipped computers, is another example. These graphical user interfaces are relatively complex, difficult to develop and maintain, but have a low value within the total product package "computer + operating system" – at least in the server market which Sun mainly targets. Therefore it is reasonable for Sun not to devote own resources to its further development but rather use a widespread Open Source product. If fostering its development at the same time helps in making Linux a better desktop alternative to Microsoft Windows, this weakening of Microsoft's position would only be in Sun's interest.

Sun's Java activities

A third group of Open Source projects evolve around Java, the programming language controlled by Sun. As Java is increasingly adopted by programmers and becoming the major language for writing Internet-based enterprise applications, Sun profits from own Java-based products as well as from license revenue for certain Java usage forms. Therefore it is in the interest of Sun to foster acceptance of Java and to extend its usage into new regions. Open Source projects helping to achieve this are, e.g., NetBeans, an Open Source platform and Integrated Development Environment (IDE) for Java programs or the Tomcat reference implementation for Java Server Pages, now part of the Apache project.

Cadence Design Systems

Cadence Design Systems is producing tools for the design of chips. The company sponsors two open source projects. It provides Testbuilder, a class library, which enables C/C++ to be used as an effective testbench language, through an Open Source license. A special web site (www.testbuilder.net) has been set up to host software and discussions. Cadence also sponsors OpenEDA (openeda.org), an OpenAccess coalition for a standard electronic design database. The reason for this sponsorship are hoped-for productivity increases as, according to the consortium, design productivity is a crucial, limiting factor in creating integrated circuits. The goals of the consortium are to "provide an open standard for IC design data access, along with the supporting software and to gain adoption of the standard within the EDA industry and university research community."[8] This project can be seen as a classical standardisation effort, which is accompanied by software. As major players of the industry are involved in this project, a code of conduct has been set up to comply with antirust laws.

Cadence sponsors industry-specific Open Source projects

8.   OpenEDA: OpenAccess Overview: http://www.openeda.org/openaccess_overview.html.

3   Motivations for firms' Open Source activities

Summarizing the results of section 2 leads to the observation that most OS activities by the major software makers are in the field of Linux. Even those companies that do not visibly contribute to OS development are in many cases at least passive Linux supporters by having ported some of their software to the Open Source operating system. Compared to Linux, other Open Source projects are of much lesser importance. At least this is the case for those large software companies investigated; the behaviour of smaller, more specialised software companies might differ.

Most activity is for development of Linux

As firms typically have the target to make profits and as they cannot earn income directly from selling the Open Source software they produce, the justification for the OSS engagements must come in some way from complementary goods or other indirect effects. The economics literature points out especially the strategy to sell complementary products. RedHat, SuSE and the other Linux distributors are good examples for companies providing additional products and services related to Linux. Selling additional hardware, as IBM does, is another example. Lerner and Tirole (2002) call such a strategy "reactive".

Justification for OSS engagement must come from complementary goods or indirect effects

As they point out, it also makes sense under certain circumstance for a company to react more "proactively", e.g. by releasing its software as Open Source. This is the case if the company can expect to boost profits in a complimentary segment by doing so. As additional condition the profit increase in the complimentary segment must be larger than the profit that could have been made in the primary segment had the software not been converted to Open Source.

Proactively releasing software as Open Source

This sounds like a fairly straightforward strategy, and as a consequence several companies started Open Source projects during the last years. However, as the focus of the large software companies' activity on Linux shows, most Open Source projects did not get the same attention as Linux did. Most notorious example is Netscape's decision of 1998 to make a portion of its browser source code freely available to catch up with Microsoft's Internet Explorer. Initially, the Mozilla project did evolve very slowly with a limited number of outside developers taking part. It took until 2002 to make Mozilla 1.0 available.[9] Therefore not all OSS projects that follow the simple "complementary goods logic" seem to work.

Not all Open Source strategies of software companies successful

Therefore one has to look deeper into the OSS projects as well as into the companies' motivations to participate, to gain a better understanding about the context in which a commercial engagement in OSS projects can be justified as well as situations where it is not the case. Doing so, we have identified four major motivations behind the companies' Open Source activities. Some of these correspond to the complimentary goods argument, others go a bit further.

Four motivations behind Open Source activities

---

9.   The history of the project can be seen at http://www.mozilla.org.

These motivations are:

- ❑ Standardisation: overcoming the ghost of Unix wars,
- ❑ Open Source software as low-cost component,
- ❑ Strategic considerations, and
- ❑ Enabling compatibility.

We go into more detail for these motivations in the following sections.

Standardisation: overcoming the ghost of Unix wars

Several different variants of Unix exist by now

One of the main reasons for Linux activities of many companies – be it actively contributing or passively adopting their software – is the hope to "overcome the ghost of Unix wars." Since Unix has forked in an early stage of its development, and since hardware manufacturers have developed Unix variants specially targeted to their hardware, several different variants of Unix exist by now. Examples are Sun's Solaris, IBM's AIX, HP's HP-UX, Silicon Graphics' Irix, etc.

Keeping basic operating system up-to-date is costly

With the key selling arguments for hardware moving away from the operating system towards higher-level software support (e.g. application server, Java) and towards more general features (e.g., reliability, security, low total cost of ownership), neither do customers want to worry about the operating system nor do hardware manufacturers want to spend considerable resources on keeping their basic operating system up-to-date, as this becomes less important for gaining a competitive advantage. Also software manufacturers have to support an increasing number of combinations between operating systems and hardware and would love to reduce the necessary effort for this task.

Common interest in favour of a single operating system

Thus, there is a common interest in favour of a single operating system providing basic functionalities within the package of hardware, software and services that these companies sell. For every party involved this would be advantageous. It would decrease the cost of operating system maintenance by sharing the cost for introducing additional functionality. It would decrease the support costs for other software manufacturers and it would decrease the risk for customers to bet on the wrong operating system, which might become unsupported at a later point in time.

Open Source foundation allows focus on other elements of product bundle

In addition, companies can focus on other elements in the product bundle they offer, i.e. focus on their core competencies. Especially for large software companies, these are typically more in creating solutions to actual business or IT problems and in marketing these than in providing basic computing functionality. By focusing on their core competencies and integrating the basic OS operating system into their product bundles, they can also speed up their rate of innovations.

GPL license ensures that foundation cannot be hijacked…

An Open Source operating system is well suited for this purpose as its license model ensures open access to the software as well as the possibility for everyone to influence and participate in its development. Due to its viral GPL license Linux is well suited for such a role as basic foundation as it cannot be "hijacked" by any party, at least not back into a proprietary software. Any derived works must be made available again under the GPL license. Problem of the GPL is, however, that it strictly limits integration into other software.

… but influence depends on resources devoted to project

Nevertheless, in practice the influence about the project's future development is bigger, the more resources can be devoted to participating in the project. Therefore it is especially interesting for large companies like IBM to become active participants in the Open Source development process to shape it in their interest. This is the same behaviour as can be observed by standard-setting organizations. OS development

processes are open for participation by everybody, but they are not democratic in a one-man-one-vote sense.

Open Source software as low-cost component

A second reason for the increasing usage of Open Source software, especially Linux, as an input into the production of product bundles and, in some companies, increasing participation in its development, is simply its lower cost. With the basic operating system itself being not any more the key property to distinguish products, the ability to produce bundles or hardware and software meeting specific requirements at low cost is a competitive advantage. The combination of relatively cheap Intel hardware and a free operating system fits these requirements better than an expensive albeit powerful proprietary hardware combined with a proprietary operating system.

Product bundles containing Open Source cheaper than fully proprietary bundles

This recent development is to a large extent customer driven. Customers have experienced that the rapidly increasing IT-requirements for processing e-business transactions can not only be met by buying highly powerful expensive solutions (vertical scaling) but also by combining a high number of cheap Intel/Linux-systems (horizontal scaling). But if they want to combine several dozen machines to a cluster, the license fee for a traditional operating system becomes a significant cost factor. This can be avoided by using Linux. The same argument applies to another range of recently introduced products, the so-called appliances. These are combinations of hardware and software for a single purpose (e.g., firewall, email-server), which are often positioned in the low-cost segment.

Development is to a large extent customer driven

Strategic considerations

While most OS activities are in the Linux field, some companies also contribute to a variety of other OS projects or have even released their software to the Open Source community, like SAP did with its SAP DB database. Typically, just like the Linux engagements, these activities are in areas which provide a rather basic functionality for solutions built on top of it than in the development of actual products. SAP, e.g., is producing and selling enterprise resource planning (ERP) software. A database is only a precondition for running such software. Likewise, Sun is selling complete server solutions. The X-Windows window system or the GNOME desktop is only one element of this solution and not even an important selling point.[10]

Release of software as Open Source in non-essential areas

But for other companies, which are in some areas competitors to SAP or Sun, these Open Source projects are direct competition. Oracle, for example, sells databases as well as ERP-systems. By releasing the SAP DB as open source – which makes it much cheaper for users – SAP can hope to snatch market share from Oracle in the database field and thereby weakening Oracle as a whole, possibly making it a less strong competitor in the ERP field, too. The same applies to Sun. By helping the improvement of GNOME, Sun raises the chances that Linux (which can also use GNOME) becomes a serious operating system for the desktop. Sun itself is not really active in the desktop computer business, but its archrival and competitor in the server business, Microsoft, is. This strategy is even more pronounced in Sun's support of the OpenOffice project, which is also a potential threat to Microsoft.

Releasing software as Open Source can weaken competitor

---

10. An exception is OpenOffice. However, as noted above, the move to make this software available as Open Source can be seen as strategy to please the Open Source community.

Enabling compatibility

OS projects to make own software or hardware compatible with OS software

Some companies show only limited OS activities, mainly by making their software or hardware compatible with OS software. For example, Adobe has released some Python plug-ins for its products. Another example are several hardware makers like IBM or HP, which have released OS software that enables Linux to run on their computers and to make optimal use of special hardware features built into their products.

Viral licenses force the companies to provide Open Source software

As long as the respective OS software is licensed under a viral OS license (e.g. GPL), the company has to provide its OS-related software as OS again. But even if this is not the case, marketing considerations might be sufficiently strong reasons for releasing the source code. Obviously, this makes only sense when the OS code makes only use of publicly known APIs of the proprietary hard- or software. If the company does not want to make these known, it would not let itself being forced to do so by a viral Open Source license.

Motivations of Open Source opponents

Most of the non-contributors are probably free-riders

Only about half of the companies investigated engage in Open Source development projects. It is therefore useful to ask why others do not contribute. Most of the non-contributors are probably simply free-riders, who use the OS software – especially Linux – but do not contribute to its further development. While this may be morally objectionable it is a perfectly rational behaviour if the company can make direct use of the open source software as it is.

Objection to OS model for principal reasons

There are other companies, though, that object the OS development model for principal reasons. Microsoft is probably the most prominent one, which has proposed an alternative licensing model called "shared source" where some companies and other institutions like universities get so see selected parts of the source code. While there are several high-brow principal arguments provided by Microsoft and others against the Open Source model, there are also plain business reasons for some companies to object to the Open Source model.

Open Source components incompatible with strategy of tight integration

The most pressing business argument is that some OS software is or can become an immediate competition to products offered by these companies. Linux in combination with other OS software is a competition to Microsoft's products, as the debate about Linux in the German Bundestag has shown vividly. Also, Microsoft has decided to integrate all its software components as well as coming services like .NET very tightly. Within this strategy, there is no place for certain components – for example the server operating system – to be something other than a Microsoft product (Wilcox and Shankland, 2001).

Microsoft not the only OS opponent

But Microsoft is not the only company opposed to OS software when it comes to competing products. JBoss, for example, an OS Java application server, is a direct competition to some of BEA's products. Therefore BEA made clear that according to their opinion Open Source is not suited to mission-critical systems but rather a vehicle for low-end non-critical systems (Coleman, 2001). This is an assessment that the sponsors of the Open Source Development Lab, which explicitly has the goal to make Linux an operating system for large critical systems, certainly would not share.

Objection to Open Source for fear of hijacking

Finally, some companies object to make their software available as Open Source software, since they fear that its further development process could be hijacked by others. Sun's hesitance of giving third parties more saying into the future development of Java belongs into this category, as there exists the potential threat of Microsoft engaging too much in this activity. This example also shows that companies can take different positions for different products, depending on the actual circumstances.

Conclusions

From these different motivations for and against engagement in Open Source activities it becomes clear that companies take different positions towards Open Source. These positions depend on the kind of software in question and on its relation to the companies' major products. As soon as these are in direct competition, the companies' position towards Open Source is much less favourable than in a setting, where they are complements or where the (cost-saving) Open Source software replaces a part of the product package that is not considered core business.

Companies take different positions towards Open Source

The motivations for engaging in Open Source projects can be considered to be of two kinds: One set of motivations is rather short-run and focused on single software products. Making certain software packages available as Open Source for strategic reasons or developing Open Source programs to make one's own software compatible with Open Source software belong into this category. The second set of motivations is long-run in comparison and focused on Open Source components within larger packages of hardware and software. Helping to let Linux become a unified Unix operating system and replacing own software components with Open Source ones belong into this category.

Short-run and long-run motivations

It is the latter motivation that is often behind the analysed companies' engagement in Linux development. Such a long-term goal of using Open Source components as basic building blocks – very much in the sense of infrastructure upon which those products are build that actually compete in the market – has several analogies in other areas of business. These analogies are explored in the following section.

Open Source software as basic infrastructure

4   Policy implications of Open Source activities

The economics literature suggests two areas that resemble in some way the engagement of firms in development of infrastructure-type Open Source software. The relevant areas are the economic analysis of standardisation as well as the economic analysis of basic research by companies. The parallels between these two areas and the analysis of Open Source activity will be drawn in this section.

Standardisation and basic research activities as analogies

These two frameworks can also be used to analyse the relationship between Open Source activity, regulation and public policy. This is done on a rather principal level and focused on indirect government policies. These are activities by government bodies and regulatory authorities aimed at setting the right business or legal framework within their entities. They include competition policy and the protection of intellectual property rights. The analysis of direct policy measures to foster use and development of Open Source has already been conducted extensively by others (e.g., Schmidt and Schnitzer, 2002; Evans and Reddy, 2002).

Framework to discuss policy issues

4.1   Open Source activity as standardisation effort

As pointed out in the previous section, the main Open Source activities of firms are in the field of Linux. A major motivation for fostering its further development is the potential reduction of Unix variants towards one major Unix operating system and the associated cost savings. This is actually a motivation that is shared by many other standardisation efforts.[11] Standards provide better interoperability (a program for Linux on IBM can – more or less – run on Linux for HP) and thus a larger market for additional products. They also lead to more trained personnel being available for this operating system and thus lower costs for this personnel.

Aim of unified Unix resembles standardisation

In addition, reduction of variety as one outcome of standardisation reduces also the risk of investments. If Linux becomes the major Unix OS, developing an application for Linux is in the long-run a safer bet than developing it for, let's say, Sun's Solaris. This point is especially important for small companies, as betting on the wrong operating system can force them to close shop. The opportunities of a widely accepted operating system for small companies are shown by the considerable number of SMEs that develop applications for Microsoft Windows, a standard that is also expected to stay.

Standardisation reduces investment risk

Finally, standardisation can make the future path of product innovation more focused. While innovation in not-standardised environments often resembles more a bush – for each branch, there are several innovations leading to new branches – the sequence of innovations in standardised environments resembles more closely a well-

Standardisation can make the future path of product innovation more focused

11. For an overview of the economic aspects of standardisation see, for example, Swann (2000).

trimmed tree with a strong trunk. This requires, however, that the standard evolves over time and incorporates the best of competing innovations into the standard set.

Motivations for participating in standardisation process

As in all other forms of standardisation, it is rational for companies to participate in the standardisation process mainly for two reasons. First or all, if the proposed standard indeed becomes an accepted standard, they have a head-start before other companies that did not participate to the same extent. And secondly, they can try to influence the standardisation process in their particular interest, for example by bringing the standard as closely as possible to their existing technology. It is this standardisation process and the participation issues that raise the most questions for public policy and regulation. Two items are particularly important, competition policy and issues of intellectual property rights.

Implications for competition policy

Standardisation efforts a two-edged sword

For competition policy standardisation efforts are a two-edged sword. On the one hand, standardisation decreases the barriers for entry into a market and levels the playing field, which is favourable to competition. Such group standards might intensify the competition in other product markets that use the standard as basis. Examples like appliances or other packaged products using Linux point into this direction. On the other hand standardisation bodies and their activities do provide a potential for collusive behaviour among those participating, which can decrease competition. Therefore competition authorities watch standardisation efforts very closely.

Openness for participation typically the case in Open Source projects

There are two major points of importance here. The first is openness for participation. If everybody can join the club on non-discriminatory terms, the probability that the standardisation process is used to keep others out of the market is lower. On the other hand, if access to the standards body is restricted, the activities might well be worth further investigation. Open Source activities do not pose much of a threat regarding this issue. As everybody can join Open Source activities and as also the outcome of the activities is available to everyone, it is very unlikely that they pose a threat to competition.

Abuse of OS projects to exchange sensitive information also unlikely

The second issue is the potential abuse of a standardisation body as a platform for exchanging sensitive information with the aim to reduce competition. Pricing agreements are the most obvious kind of forbidden behaviour. Exchanging sensitive information regarding targeted markets, planned strategies, etc. is similarly problematic. Again Open Source projects pose less of a threat to competition than other standardisation efforts. Open Source activities typically take place in an open space, i.e., on the Internet, and not in smoke-filled backrooms.

As they are open to everyone and as most of the information flow in form of e-mails and group discussions takes place in written form, is archived and visible to most of the world, the incentives to abuse the Open Source projects for such anticompetitive behaviour are rather low. By the same token, absence of such an open communication could be a signal for authorities to watch the activities more closely.

Code of conduct within Open Source projects

To clarify this issue and define appropriate behaviour some initiatives, e.g. OpenE-DA, have given themselves a code of conduct explicitly forbidding the misuse of the initiative for anti-competitive activities.

Competition issues of restricted shared source initiatives to be investigated

Somewhat more problematic from the point-of-view of competition policy might be shared source initiatives like Microsoft's, where access to the source code is limited to certain kinds of companies or institutions. This becomes especially relevant if the code is considered to be of essential importance, e.g. because it defines an operating system with large market share. If not everyone has access to the provided informa-

tion or only under rather strong restrictions, some might judge such activities as being detrimental to competition.[12] However, up to now there has not been much discussion of these issues. Therefore it is too early for a final assessment.

Issues related to intellectual property rights

A further issue arising in classical standardisation activities is the possibility that a participant tries to hijack a standardisation group. Typically this works in the way that this party participates in a standardisation body, agrees to a standard, helps in promoting it and eventually, when the standard becomes widely accepted, pulls a patent covering part of the standard out of his pocket. There do exist a few examples for such behaviour, like Unisys' discovery of owning the rights for the LZW compression algorithm employed in the GIF graphic format or Dell's patent covering part of the Vesa Local Bus standard (Lemley, 1996). It is not completely clear, whether the behaviour of these companies was intentional from the beginning or whether the companies only by accident noticed that they are sitting on a potential IPR goldmine. While these examples concern hardware and algorithms, the increasing importance of software patents also makes this problem more acute for Open Source projects like Linux.

Hijacking standardisation group problematic issue

To avoid such potential problems, there is the possibility of imposing that participating in the standardisation process requires an assurance of every participant that he does not own any intellectual property right related to this standard. The Internet Engineering Task Force IETF has chosen this approach. Others like ANSI require only that access to the intellectual property is granted on fair conditions (Lemley, 1996). Recently this issue has led to a vivid discussion within the World Wide Web Consortium W3C.[13]

Assurance of participants that no intellectual property claims will be made

Such a rule can be followed comparatively easily in small groups of participants. In Open Source projects, where often many participants are involved, who in addition might move into and out of the projects running for several years and in the process might even change their employers, it becomes rather difficult to handle. In addition the participants in Open Source projects are typically people, even if they are paid by their employers to contribute, while the owners of patents are mostly companies. It is unclear, under what circumstances the participating individuals can provide legally binding statements in the form of those given above.

Open Source projects pose several problems to intellectual property clearance

With the increasing acceptance of software patents also in the European Union, this potential problem is likely to get more attention – also for Open Source projects. The issue has been addressed to some extent in OS licenses, for example in the GPL. Section 7 states that if due to patent infringement claims conditions are imposed on distributing a software that contradict the terms of the GPL (e.g., royalty payments), the software must not be distributed.[14] As the GPL is viral, this obviously leads to problems if patent claims are made for very basic pieces of software, which are included in several other derived works. The consequences of this problem – in theory as well as in reality – are still to be investigated.

Problem is likely to get more attention in the future

---

12. For example, Microsoft's OEM (Original Equipment Manufacturers) Source Licensing Program is only open to such OEMs that "[m]eet the OEM group's definition of 'Multinational' or 'Datacenter OEM,' or be a Windows CPU vendor."
Cf. http://www.microsoft.com/licensing/sharedsource/oem.asp.
13. http://www.w3c.org/2001/ppwg/.
14. http://www.gnu.org/copyleft/gpl.html.

The role of government for Open Source as standardisation

Do typical recommendations about standardisation apply to Open Source?

As standardisation is an activity that has taken place for quite some time in several different areas, there are several opinions about what the government should or should not do with respect to such standardisation activities. It is worth asking, whether the typical recommendations also apply to Open Source activities when interpreted as standardisation.

Is level of standardisation optimal?

The first, and probably most debated issue is, whether the amount of standardisation activity taking place is optimal, given that standardisation has positive implications for competition and thus for consumer welfare. Economic theory that could provide some insight, however, has been more concerned with standards races, where several (at least partially) incompatible standards compete for market dominance and has tried to answer questions about fair and unfair practices in these races and about the determinants of the optimal outcome.

Government should be careful when pushing companies into standardisation-focused Open Source activities

An assessment about the optimal degree of standardisation is rather difficult, as consumers' utility from variety, the importance of variety as driver of technical progress and the static efficiency gains from the use of standards have to be balanced somehow. The conclusion for government policy would be that it is hard to assess the optimal degree of standardisation. There are no convincing arguments, why government should be able to determine this optimal level better than the market does. Therefore it should be very careful when pushing companies into standardisation-focused Open Source activities – as tempting as the vision of a fully standardised Open Source technology core and the associated static gains may be.

Government should not pick standards

What frequently happens in traditional standardisation processes is that the process stalls and participants cannot agree to a standard. In such situations it is sometimes said that government should choose a standard, as any standard is said to be better than no standard. Mobile communication in Europe, which is based on the single GSM standard, is often cited as an example for the benefits of standardisation. However, a closer look at this example shows that it also had its drawbacks: Japan, for example, has a much more advanced mobile phone system and a much larger variety of mobile data services than Europe, despite several incompatible standards competing with each other. Therefore public policy would have to be very careful in taking sides and trying to push development in one or another direction.

Balance participation and represent excluded interests

A further claim about government's role in the standardisation process is that it should try to balance participation and represent excluded interests. This claim is also very much influenced by traditional standardisation procedures where industry-heavyweights agree about the future course of technology. With respect to Open Source projects like Linux, though, participation is in principle open to everybody. Thus, the role of government is much smaller here than in traditional standardisation efforts.

Are European interests underrepresented in Open Source projects?

One potentially underrepresented group might be Europeans. It has been argued above that especially large software companies have an incentive as well as sufficient resources for participating in or even for driving Open Source projects. As these companies are mainly located in the United States and as the US market is larger than that of any single European country, it might be the case that US conventions – language, character sets, business rules – are better represented in the resulting software than European ones. This problem is unlikely to be very large, though. As the FLOSS developer survey showed, European programmers make up the large majority of those developers surveyed (Ghosh et al., 2002). Even taking into account that the survey is probably biased towards European developers, there remains a substantial number of European Open Source developers.

Taking all these points together from the point of view of Open Source projects as standardisation activities, the most important role of government is probably to educate companies about this role OS projects can have and to make clear what the chances of participation and the potential threats of ignoring these activities are. It might also be useful to investigate whether other Open Source activities besides Linux are standards-like and point out these opportunities to companies without forcing them to participate.

Education role remains

4.2   Open Source activity as basic research

Besides the interpretation of OS activity as standardisation activity, there is a second reasonable analogy that can be drawn. The engagement of software manufacturers in Open Source projects is very similar to firms' engagement in basic research. In both cases the immediate gain from this activity seems to be much smaller than from devoting the resources to the development of other products that can be sold directly.

Open Source engagement similar to basic research

The economics literature has investigated thoroughly why firms nevertheless invest in basic research (e.g., Rosenberg, 1990). This question is of huge importance for policy makers, since the social returns from basic research, for example in form of growth effects, are considered to be extraordinarily large (Griliches, 1996). The same is often said about Open Source software. As everybody can learn from reading this source code and using it as a basis for own software, positive externalities of these Open Source activities exist.

Basic research by firms with positive spillover effects

The major arguments from the economics literature explaining basic research activities have already been met during the analysis of the OS projects from the largest software companies. For example, it is said that an engagement in basic research is rational, when firms can capture at least some of the gains from this activity. They do not need to capture the entire returns from basic research. Indeed, this would by socially inferior, as it is especially the externalities that make basic research desirable. There are several ways in which a company could capture some of the benefits: For example, it can use it in combination with other inputs for new products (just like Linux on IBM mainframes) or it can try to obtain first mover advantages. These arguments certainly apply for Open Source projects.

Engagement in basic research when some returns can be captured

A further reason for basic research activities is that knowledge from basic research can only be understood and thus be incorporated in innovations, when companies are familiar with the research process, the jargon used, etc. This knowledge is best gained by employing at least a few people with strong ties to the scientific community. Knowledge is not, as it is sometimes argues, "lying on the shelf". The same might be true to some extent for Open Source projects. While much Open Source software can easily be used as is without being involved in its production process, incorporating Open Source software in one's products requires a deeper understanding of its capabilities and problems, which is best gained by participating in its development process.

Knowledge from basic research can best be understood when companies are involved in research process

Issues related to intellectual property rights

Economists have long since argued that society would invest insufficient resources in basic research in all market regimes. While monopolists do have a lower incentive to innovate than firms in perfect competition, the latter have the problem that a competitor can quickly exploit the useful new knowledge when it is unprotected. The obvious solution to this problem is IPR protection, e.g. in the form of patents. However,

IPR protection raises incentives for basic research

these introduce another inefficiency. Since the knowledge has been produced and can be distributed almost without cost, it is inefficient not to do so. This is a dilemma without a simple solution.

Observed outcome of Open Source activity optimal for firms

As firms are voluntarily choosing to participate in Open Source projects, the observed volume of Open Source projects by firms can be interpreted as the (local equilibrium) outcome of their research investment. As firms can protect most of their intellectual property in the domain of software development, it can be assumed that they only give as much intellectual property away in the form of Open Source software as is optimal for them.

Are Open Source peculiarities keeping companies from releasing code as Open Source?

The question to address is thus, whether this level of Open Source activity could be increased without weakening the rate of innovation within these companies. Are there certain peculiarities of the Open Source process that keep such firms from participating that under different regimes would be willing to make their knowledge available to others?

Viral nature of GPL

One issue pointed out, for example by Microsoft,[15] is the viral nature of the GPL (which governs Linux) and especially ambiguities in its virality, which supposedly makes it difficult to build commercial software on top of Open Source software. While a discussion of this legal issue is beyond the scope of this paper, it has to be taken in account that unclear legal implications might indeed be issues keeping companies from taking part in those Open Source projects governed by such licenses or from including such software as infrastructure components into their products.

However, these ambiguities do not concern the release of formerly proprietary software as Open Source, as companies are free to choose the license they want when doing so. The strictness of the GPL is one reason why many software companies use BSD Unix, governed by the more liberal BSD license, as foundation for their commercial software.[16]

The role of government for Open Source as basic research

Basic research often limited to certain sectors and large firms

Empirical research about basic research has frequently shown that basic research is typically limited to very few sectors. Within these sectors there exists a handful of firms, typically large firms, that dominate the basic research picture (Rosenberg, 1990).

Issue seems less pronounced in Open Source

However, this issue seems to be less pronounced in the Open Source area. For example, the hurdles for participation (barriers to entry) in Open Source projects are lower than for conducting classical basic research, as only smart people are needed and no costly research infrastructure. Indeed, the FLOSS user survey does not show significant differences between large and small companies in letting their developers pursue Open Source projects on company time (Berlecon Research, 2002). Nevertheless, there might be differences between large and small companies in deliberate decisions to participate actively in Open Source projects for reasons like those set out above.

Often large companies can capture indirect returns better than smaller ones

A major hypothesis trying to explain the dominance of large companies in basic research is that their product diversity as well as their sales and marketing power raises the potential of being able to use the research outcome. Small companies in comparison need to target their research investment much more to their immediate needs. This argument also applies to participation in Open Source activities. As companies

---

15. See the discussion with Microsoft's Craig Mundie at the O'Reilly Open Source conference in July 2001.
http://linux.oreillynet.com/pub/a/linux/2001/08/09/oscon_panel.html
16. Also the Apache project uses a BSD license explicitly for these reasons.

cannot earn directly from these activities, they must find a way to capture some of the gains in an indirect way. Large companies can do this often better than small ones, although there do exist counterexamples like Sendmail, a company selling messaging solutions on the basis of software components it released as Open Source.

Thus, the extent to which the Open Source activities of small and large companies differ, has to be investigated in more detail. If SMEs participate less than optimal in these activities, then for government policy the same arguments apply for supporting Open Source engagement than for supporting (other) basic research. Policy should create incentives to increase this participation. Not only would this give smaller companies similar benefits as larger companies, but it would also increase the amount of Open Source software available and would thus most likely raise the rate of technical progress in software manufacturing.

If SMEs participate less than government should create participation incentives for them

With respect to basic research it is sometimes suggested that the government itself should become active and conduct basic research, as the incentives for private entities are insufficient. Likewise one could argue that the government should engage in Open Source activities as the existence of Open Source software is socially beneficial but the incentives are not sufficient to develop (enough) OS software. However, the government does not have a good record of choosing the right projects, and therefore it is likely that it will develop Open Source software that does not meet the requirements of the market but rather those of the people deciding about the project.[17]

Should government conduct basic research?

A different issue would be the support of Open Source development. To the extent that private entities do not engage sufficiently in Open Source activities, support of the latter is justified provided that three issues are taken into account. First of all, the support should be structured in a way that lets the market decide about which projects are useful. This is, e.g., the case when supporting infrastructure for Open Source development. Secondly, support should go to those kinds of Open Source projects that provide software closest to basic research, i.e. infrastructure-like software that can be used as component in many other kinds of software. And thirdly, the license regime of the supported Open Source software projects should be such that the results can be used in as many ways as possible. This would exclude strong viral license regimes such as the GPL.

Support of Open Source development justified with caveats

---

17. The debate about the use of Linux in public institutions illustrates the problem: Linux is not the only alternative to Microsoft products (BSD-based products and Macintosh are other alternatives) and it is not clear which will prevail in the long-run. Initiatives forcing public institutions to use Linux might put them on a sub-optimal technological trajectory in the long-run.

5   Conclusions

A major observation in this paper is that profit-maximising large software companies do actively contribute to Open Source projects. Not all of them do so, not all do it to the same extent, and the motivation for their activity is obviously purely selfish. Nevertheless, by doing so they contribute to the amount of Open Source software available to everybody and thereby contribute to economic growth and social welfare.

Profit-maximising large software companies actively contribute to Open Source projects

The reasons for doing so bear some resemblance to other activities of companies like the engagement in standardisation efforts or the conduct of basic research. Thus, similar issues related to competition policy or intellectual property rights can be shown to play a role in Open Source projects, although mostly to a smaller extent than in other surroundings.

Motivations bear resemblance to standardisation and basic research

There are some arguments implying that the amount of activity in Open Source projects by companies is sub-optimal, as the companies do not fully take into account the positive externalities from the availability of the Open Source code. This provides a justification for certain ways of government support to Open Source projects.

Possibly sub-optimal level of Open Source activity

Such support should not be provided "against the market" but rather "with the market". Going with the market means trying to understand why firms do contribute to Open Source software, setting the legal and regulatory framework in a way that allows them to do so and educate those that are not yet doing so about the benefits they will have from contributing to Open Source. It would, however, also mean accepting that firms are often more willing to support development of some kinds of Open Source software (e.g. for infrastructure- or standards-like software) than for others (e.g. specific applications).

Support "with the market": Removing barriers to Open Source activities by commercial entities

Going against the market means supporting Open Source projects that provide an (free) alternative to commercial programs, where only one or two manufacturers remain. This is not necessarily beneficial. As software tends towards a natural monopoly, creating an artificial duopoly might be sub-optimal. With the same resources other projects could be supported that increase the amount of Open Source software in other layers of software and thereby benefiting a larger group of users in addition to speeding up the rate of technical progress.

Support "against the market": Open Source software as free alternative

Public support of Open Source activities requires the following three issues to be taken into account. First of all, the support should be structured in a way that lets the market decide about which projects are useful. This is, e.g., the case when supporting infrastructure for Open Source development. Secondly, support should go to those kinds of Open Source projects that provide software closest to basic research, i.e. infrastructure-like software that can be used as component in many other kinds of software. And thirdly, the license regime of the supported Open Source software projects should be such that the results can be used in as many ways as possible. This would exclude strong viral license regimes such as the GPL.

Issues for public support of Open Source activities

6   References

Berlecon Research (2002): FLOSS Final report, Part 2: Use of Open Source Software in Firms and Public Institutions: Evidence from Germany, Sweden and UK.

Coleman, Murray (2001): BEA Focuses on Businesses Stuck in the Middle, October 2001, http://www.bea.com/press/ibd_qa.shtml.

Compaq (2001): Compaq maintains #1 position in Linux server market, 20.12.2001, http://www.compaq.ch/de/press/release.htm?NoteID=651.

Evans, David S.; Bernard Reddy (2002): Government Preferences for Promoting Open-Source Software: A Solution in Search of a Problem, 21.7.2002, mimeo.

Ghosh, Rishab Aiyer; Ruediger Glott; Bernhard Krieger; Gregorio Robles (2002): FLOSS Developer Level Analysis, working paper, mimeo.

Griliches, Zvi (1996): Productivity, R&D, and basic research at the firm level in the 1970's, The American Economic Review, 76 (1996) 1, 141-154.

Hurd, Jim (2001): Open Source for the Enterprise, 26.7.2001, http://www.extreme-tech.com.

IDC (2001): Expanding the Frontiers of Enterprise Computing with Hewlett-Packard Linux Solutions – An IDC White Paper, August 2001.

Lemley, Mark A. (1996): Antitrust and the Internet Standardization problem, Connecticut Law Review 28 (1996), 1041.

Lerner, Josh; Jean Tirole (2002): Some Simple Economic of Open Source, Journal of Industrial Economics, 50, 197-234.

McMillan, Robert (2001): Sharing the Dot in Dot-com, Linux Magazine, May 2001.

Raymond, Eric (2000): Homesteading the Noosphere: An Introductory Contradiction, http://tuxedo.org/~esr/writings/homesteading/homesteading/.

Rosenberg, Nathan (1990): Why do firms do basic research with their own money? Research Policy, 19 (1990), 165-174.

Schmitz, Klaus; Monika Schnitzer (2002): Public Subsidies for Open Source? Some Economic Policy Issues of the Software Market, 14.6.2002, mimeo.

Shacklett, Mary (2001): HP open source strategy centers on Linux, HP Chronicle, January/February 2001, http://www.serverworldmagazine.com/hpchronicle/2001/0102/opensource.shtml.

Shankland, Stephen (2002a): IBM: Linux investment nearly recouped, CNET news.com, 29.1.2002, http://news.com.com/2100-1001-825723.html.

Shankland, Stephen (2002b): IBM puts Linux converts on display, CNET news.com, 31.1.2002, http://news.com.com/2100-1001-826926.html.

Sun (2002): A Closer Look at Linux, 19.3.2002, http://www.sun.com/2002-0319/feature/.

Swann, G.M. Peter (2000): The Economics of Standardisation – Final Report for Standards and Technical Regulations Directorate, Department of Trade and Industry, December 2000, mimeo.

Wilcox, Joe (2000): IBM to spend $1 billion on Linux in 2001, CNET news.com, 12.12.2000, http://news.com.com/2100-1001-249750.html.

Wilcox, Joe; Stephen Shankland (2001): Why Microsoft is wary of open source. CNET News.com, 18.6.2001, http://news.com.com/2100-1001-268520.html.