**1**

# BENEFITING INNOVATIVE CAPABILITIES OF SOFTWARE DEVELOPER/USER COMMUNITIES IN DEVELOPING COUNTRIES

Hacer Ansal [a 1], Nihan Yıldırım [b]

a: Isik University, Department of Humanities and Social Sciences, Kumbaba Mevkii Şile, 34980, Istanbul, Turkey
b: Istanbul Technical University, Faculty of Management, Macka, 34367, Istanbul, Turkey

## Abstract

Since technological innovation is generally considered to be a major force in global economic growth, the development of innovative capabilities in developing countries has been a very important policy issue. Free/Libre Open Source Software (FLOSS) has reshaped software technology through the creation of developer/user communities which enabled the collaboration of different parties resulting in the production of Linux and similar software projects. FLOSS user/developer community networks serve not only as "learning, reviewing, and testing" environments for developers, but they may also act as innovation networks that contribute to the improvement of the innovative capabilities of individual developers within the community. Therefore, understanding the characteristics, the motivating factors and the innovative-dynamics of these developer communities will provide valuable insight into how to improve the innovative capabilities of developing countries in relation to software.

The aim of this paper is to explore the characteristics of FLOSS developer communities in order to discover what benefits they may offer developing countries in generating innovative capabilities related to software. By conducting a survey in the FLOSS user/developer community in Turkey, the demographic characteristics, motivation factors and innovative characteristics of the community are explored and the question of whether these communities may act as innovation networks is examined. It is concluded that FLOSS community networks mostly serve as knowledge sharing and collaboration platforms, however, they do have the potential to evolve into innovation networks if they receive support from the local software industry and academic institutions.

***Keywords:*** *open source code; software development; innovation; developing country*.

[1] Corresponding author: Tel: + 90 216 528 71 91, E-mail: ansalh@isikun.edu.tr

## Introduction

Since the introduction of innovation into formal economic growth models by Solow [46], innovative activity has been the single, most important component of the growth of output and long-term economic growth in highly industrialised economies [44], while a lack of innovation has resulted in slow growth, as in developing countries. The most distinctive feature that dominates the search for new or improved technology in the world of the highly industrialised economies is uncertainty which makes innovative activities extremely risky and expensive. Therefore, generating technological capabilities, innovative competencies and the financial resources required for innovative activities have been a major problem, as well as a very important policy issue, in developing countries.

On the other hand, innovation is facing new challenges as economies are becoming more interdependent. As a result of the recent developments in information and communication technologies (ICT), like open source software, the dynamism that has been generated requires, in many ways, a rethinking of innovation itself. More and more it is realised that "innovation can originate from anyone. Anyone can innovate, as innovation requires a mindset that probes perceived boundaries to bring ideas to fruition" [14]. In the face of the new developments in ICT, the main objective of this paper is to explore the innovative characteristics of FLOSS user/developer communities to discover whether they can benefit developing countries by generating innovative capability in software.

## Background to the Study

### *Free / Libre / Open Source Software - FLOSS*

Free / open source software development was first applied in the computer departments of major U.S. universities [40], enabling programmers' the freedom to cooperate with each other before the 1980s. Then, software started to have owners or to be proprietary, and hence

programmers were prevented from sharing the source codes of their programmes and from developing software collectively [16]. Reacting to this change in software development, Richard Stallman, who was a resigned software developer from MIT, started the free software movement in 1983 and announced the GNU Project (acronym for "GNU's Not Unix") with some colleagues. The aim of this project was to develop a completely "free" software GNU operating system, and hence bring back the cooperation and sharing culture of software development. The term "free" meant "freedom", not being "free of price"[9]. The Free Software Foundation (FSF) set the free software concept as a way for the users' to have the freedom to run, to distribute, to study, to change and to improve the program (access to source code is a precondition for this) [9]. Linus Torvalds developed and distributed Linux, a Unix-like kernel that is the only missing part of the GNU system that was being-developed. The GNU/Linux operating system was formed and distributed in 1992, starting the economic success of Free/open software.

Although open source software and Free Software have similar meanings and goals, their basic values are different. The term Free software is similar in meaning to a social movement, an ethical imperative in software, while open source is more of a development methodology. According to the Open Source Initiative [37], open source does not just mean access to the source code, it also requires compliance with the distribution terms of software with open source initiative's criteria. The philosophy of open source considers practical issues in developing successful software that effectively meet the requirements of the users. Using the term "Open Source Software (OSS)", open source software supporters tried to provide marketing support to free software and prevent the possible misunderstandings of free software by putting an emphasis on practical issues. Therefore, a developer may not follow the open source development model, but may licence the software as OSS or free software, or vice versa.

Because of the linguistic uncertainty of the English term "free", in 1999 the term “Libre” (meaning free in Latin) was introduced by the European Working Group on Libre Software (created by the initiative of the Information Society Directorate General of the European Commission). The terms Free software, of FSF, and Open source software of the Open Source Initiative, were used together with the term “Libre software” in the context of the EU-funded project "Free/Libre and Open source software: survey and study" leading to the acronym “FLOSS” which represents the initials of “Free/Libre Open Source Software” [6]. “FLOSS” covers all the topics of Free software, Libre Software and Open source software, and avoids taking any side in the "free software" vs. "open-source software" debate. We used the term FLOSS "Free/Libre/Open-Source Software" in our study.

*FLOSS Developer Communities*

The community is an important part of the growth and maintenance of FLOSS projects. As Krogh [22] argued, participation in these developer communities depends on a common vision and the objective of improving, developing and deploying a software application, a software type, a development method, a programming language or a platform around which people can unite. Coinciding with the diffusion of the Internet, it became possible for developers around the world to participate in the development of Linux and other FLOSS projects [51], hence it is regarded as a new phenomenon [42] which has drawn the attention of many researchers.

O’Reilly [38] sees network-enabled collaboration as one of three deep, long-term trends that are expressions of FLOSS. The other two trends are the commoditization of software and software customizability (software as a service). Since software is a product composed of written codes, it is easy to share all the information about a software product (which may be in pre-, in- or post-production phases) within a network. Typically, in FLOSS communities, members give direct, specific and immediate feedback on the software code that others write

and submit. This peer review process is not only valuable for the individual who submits the code, but it also ensures the overall quality of the software [22]. Because the code is available for all to review and contribute to, FLOSS applications can evolve rapidly and bugs can quickly be identified and resolved [53]. Moreover, the peer-review process of software development within these communities can be very beneficial as a learning tool for an individual developer. On the other hand, there are some claims that peer production is better for refining old rather than inventing new [3].

One of the most common questions posed about FLOSS communities, is why and how software developers join and participate in such efforts (often without pay) for long periods of time. Researchers who studied FLOSS developers' motivation regarding their interactions and relations, created some classifications. For example, Ghosh [15] grouped them as "social", "political", "monetary", "signalling" and "product related", while Lakhani and Wolf [25] categorized them as intrinsic and extrinsic motivations. The motivation and reasons for the participation of developers in FLOSS projects reported in these studies are as follows:

- *The opportunity to learn and share knowledge*: FLOSS developers generally find that the greatest benefit from participation is the opportunity to learn and share their knowledge about software systems' functionality, design, methods, tools and the practices associated with specific projects or community leaders [23][25][56]. Programming methods and procedures often remain faintly codified and of a wide diversity, even within the productive organizations of large software editors [55]. Hence, following and taking part in the efforts of a FLOSS community results in feedback effects for the programmer in terms of the improvement of his or her programming skills and knowledge.

- *Getting use of the projects:* Many are working on projects that they themselves find useful. According to Raymond [42], this is their only motivation. The FLOSS community

provides a wide-testing and improvement environment for developers [56], provides, at their disposal, the products they needed and ~~to~~ helps them avoid duplications in software development or improvement [57].

- *Financial and career related benefits:* External motivating factors, in the form of extrinsic benefits (e.g.: better jobs, career advancement), are the main drivers of effort [24]. Not all FLOSS developers are volunteers, a substantial fraction is paid for work [25]. FLOSS developers interact with entrepreneurs with high-income perspectives, and even developers may become entrepreneurs, suppliers of the competitive products that raise market activity [57]. However, Lakhani and Wolf [25] concluded that there was no significant difference between the reasons for the contributions to FLOSS projects of paid and volunteer participants in terms of intrinsic motivations (enjoyment, intellectual stimulation, reputation in the community, political reaction to proprietary software etc.). In fact payment affected the product/code work related to extrinsic motivations. On the other hand, participating in a community also increases the chances of developers to find investors for their projects [28].

- *Taste for creativity, enjoyment and intellectual stimulation*: Since developers determine their contribution to FLOSS projects themselves, they enjoy the work they do [10]. Intrinsic motivations like the taste for creativity, satisfying a user need, intellectual stimulation and enjoyment are strong drivers [25].

- *Recognition:* Developers like being admired and recognized by the rest of the community as knowledgeable and trustworthy contributors [10].

- *Political idealism for the freedom of software:* A significant portion of the FLOSS community participants feel a political idealism for the freedom of software and by defending the idea that "software should not be a proprietary good", they want to limit the power of large software companies [11]. As they want to increase the competitiveness

of FLOSS in software market, developers are satisfied with the successful achievement of the objectives of the FLOSS community [56].

- *Working with like-minded colleagues:* FLOSS development projects involve like-minded individuals who share many common technical competencies, values, and beliefs [11], and who choose to participate and contribute to a specific project themselves.

However, the above-mentioned motivations are not applicable to every community member in the same way or at the same level because the motivation may be different for each developer in the community. Typically a less experienced member of the community can be more attracted by the chance to improve his/her own skills, while an accomplished or experienced member may want to link his/her contribution to future earnings related to the commercial services of FLOSS products' diffusion [57].

Another important characteristic of FLOSS communities is that, a small group of core developers not only controls the architecture and process of development, but also develops most of the FLOSS software projects. Most participants typically contribute to a single module or develop some patches or modifications [11], as reported in a survey conducted by Hars and Ou [18], 5 % of the developers reported that they had participated in 10 or more FLOSS development projects.

*FLOSS User and Developer Communities as Innovative Networks:*

The Free Software movement's unique development practices are challenging the traditional view of how innovation should work [22]. Developer communities take their place as an "innovator party" in software technology and offer a valuable platform for the collaboration of other high-tech professionals and researchers in similar fields of technology.

When exploring the innovative attitudes and innovative networking characteristics of FLOSS communities two dimensions have to be analyzed. The first dimension is that they

act as innovative producer/developer networks, and the second dimension is that they function as innovative user networks.

Carr [3] argues that companies no longer have to pursue innovation in isolation since they can connect to the global masses through the Internet. Developer communities that offer models in which resources of innovation are widely distributed throughout the world, have members who, by using the Internet, can take part in the innovation process without being limited by the national, regional and cultural borders, as well as provide important management lessons related to finding effective ways to structure and implement innovation and knowledge sharing [4]. Carr [3] points out that they have the option of replacing the traditional closed cathedral model with the new open bazaar model, referring to the metaphors that Raymond [42] created for proprietary software development with hidden codes and FLOSS development.

In the early phases of technological innovations, the developers define and solve not only technical problems but also organizational, economic and political questions that are indistinguishable from technical problem solving [27]. During new product development, technical problem solving is a major learning tool and a source of new knowledge [5] (or knowledge is a by-product of technical problem solving [49]), hence, innovation occurs as a stream of random or planned problem solving. The diversity of a developer community, which may matter more than individual ability when solving technical problems [39], is critically important as it enables valuable contributions to innovation. From this point of view, technical problem solving activities, like defining and correcting errors or defining and developing the improvement needs of software programs in developer communities, will lead to knowledge creation and the improvement of the innovative capacity of the community members [1], and hence of the companies and industries that they work in/for. FLOSS projects are, therefore, critical resources of innovation. With the maturation and the increased number of FLOSS-based products and improvements, FLOSS development denotes a hybrid

innovation model, which takes advantage of acquiring resources from both the community and the firms [29].

In addition to developer communities, there are communities of user and developer groups who operate through mailing lists that act as information providers to users and developers. FLOSS projects, among others, have led to innovation, development and consumption communities run completely by and for users. End-users often become contributors or developers, while developers may act as end-users in FLOSS projects [19][35][44]. Similarly, many end-users often participate in and contribute to FLOSS development efforts by providing feedback, bug reports, and usability concerns. However, the vast majority of participants probably simply prefer to be users of FLOSS systems, unless, or until, their usage motivates them to act through some sort of contribution. Avid users with sufficient technical skills may actually work their way through each of the roles and eventually become a core developer [45].

Therefore, user communities can have a pattern of user innovation and trial if some users have sufficient incentive to innovate, while some others have the incentive and means to reveal their innovations voluntarily [19]. On the other hand, as Von Hippel [19] points out, users often have difficulty in expressing their needs while developers of standard, off-the-shelf, software often find it difficult to judge whether a product feature will have a major impact on satisfying user needs. Such information is costly for manufacturers to retrieve because understanding the users' problems requires that the manufacturer work with the users for a prolonged period to enable a better understanding of their problems; hence development and marketing costs increase. When compared with the measures of traditional economics, today's user innovation communities, composed by users and for users, have remarkable abilities, like enabling product development, creating, sustaining, consuming and supporting innovations on their own complex innovative products without manufacturer involvement [19].

However, when FLOSS communities are being analysed as user networks, the impact of innovation costs and competition must also be considered. Manufacturers are capable of developing and diffusing innovative products traditionally because they have the financial incentives, the opportunity to sell and the production distribution capability. Hence they can bear the costs (like the loss of proprietary intellectual property, and the cost of diffusion) associated with revealing an innovation. However, individual user innovators can typically expect to benefit financially only from their own use of their innovations. In order to benefit from the diffusion of an innovation to other users in a marketplace, innovating users would have to obtain intellectual-property protection and set up licensing arrangements, which they are not expected to accomplish as cost-effectively as manufacturers. They will generally expect intellectual property losses to be low if their rivalry with potential adopters is low [17]. Even rivals who would prefer not to reveal an innovation will do so if they expect that others will reveal it if they do not [24]. When the costs of freely revealing an innovation are low, even a low level of benefit can be an adequate reward. Therefore, according to Von Hippel [19] users have sufficient incentive to innovate when they expect the benefits of innovating to exceed their costs. However, user innovative communities can exist only when user manufacture and distribution can compete with commercial production and distribution, as user-led diffusion of innovations that are proven to be of general interest, will be followed by commercial production and distribution.

In the case of FLOSS, innovations can be produced and distributed essentially for free on the Web, enabling them to compete with their proprietary rivals. Aided by the Internet to support collaboration and distribution, the power and pervasiveness of such communities could become enormously amplified [13].

As the innovative characteristic of FLOSS developers and user communities becomes significant, commercial enterprises began to attach to or complement the innovation communities. Red Hat and VA Linux Systems are well-known examples of commercial

involvement in the FLOSS software context [19]. On the other hand, there are user/developer communities of proprietary software products like “Microsoft Site Builders” that serves users and developers of the proprietary software of a specific software producer [33]. However, these communities act as a feedback mechanism for defining bugs and testing new releases for the producer, rather than serving the needs of the users and involving the developers in the innovation processes of the specific software. Microsoft also started the “Shared Source Initiative” where they announce that they are sharing source code with customers, partners, developers, academics, and governments worldwide, encompassing a wide spectrum of technologies, programs, and licenses offered by Microsoft to various communities of customers, partners, developers, organizations, and other interested individuals [34]. In the Shared Source Initiative, Microsoft has announced that they learn from the FLOSS community regarding the benefits of deeper collaboration and increased transparency leading to better communication with customers. However Stallman [47] criticizes this and similar attempts from Microsoft, claiming that it is not possible to check, and hence to trust proprietary programs if they have any hidden codes (e.g. backdoors that bypasses security mechanisms that are not known, hence can not be controlled by the user) or not.

*FLOSS Development and Adoption in Developing Countries*

FLOSS offers unique opportunities for improving the innovation capacity of local software producers [1], it is "a useful and significant tool for developing countries" as well as having the potential to help democratization and find solutions to the most pressing problems faced by developing countries [41]. By utilizing FLOSS, these countries can deploy extensive computerization in their societies while avoiding the high costs of software investments [31]. Ghosh [12] claims that, FLOSS development and adoption provides benefits to developing countries by its three main characteristics: low total cost of ownership, good performance and flexibility for localization, and knowledge base creation/skills development in programming. Also national security and transparency are critical for

software used in government and the public sector [15]. The use of FLOSS also prevents the widespread "unauthorized copying of software" in developing countries, while avoiding proprietary software monopolies and reducing barriers to competition that threaten the local software industries of these nations [48][43]. Hence, support for FLOSS allows developing countries to support the potential development of local software production [31].

The interactive character of FLOSS encourages local amendments and engagement with the technology, and thus, shifts the emphasis from the passive use of proprietary technologies to an emergent culture of the self-development of ICT-related skills. While all software may require specific skills (which can be gained through accredited training), FLOSS allows software professionals to develop skills that are related to their specific local needs, and encourages the development of computer programming, maintenance and development skills within local user communities. Moreover, the deployment of FLOSS allows a form of ongoing apprenticeship in programming communities, with the more experienced programmers helping the newer practitioners [32]. FLOSS communities offer a form of ongoing technological transfer in developing, or developed countries, by funding the initial acquisition of programming skills of individual programmers, who then spread their skills through the community's FLOSS projects [8].

From a technological point of view, since it is an engine for technological innovation at the national level, FLOSS offers the opportunity to free the country from its technological dependence on trans-national software vendors [2].

In order to be able to turn the opportunities that FLOSS offers for improving the innovation capacity of local software producers into strengths in developing countries, the appropriate financial, technical and human resource infrastructures are needed. FLOSS adoption/development must be among the strategic factors for improving the local innovative capacity in software and must constitute a key part of the country's IT strategy [52][51].

Many governments around the world, like China, Spain, South Africa and a number of countries from Latin America, have recognized the empowerment of the IT industry through FLOSS development as an important opportunity, and have initiated the use of FLOSS in their IT strategies. In this context, providing national level strategic support for software developer communities is among the important components of policy as these communities actively take part in the evolution of software technology as an innovator party [1].

*The Adoption of FLOSS and the FLOSS community in Turkey: Linux.org.tr*

Some national level strategic attempts have been made in Turkey regarding the adoption of FLOSS. One example is that the State Planning Organization has emphasized the importance of FLOSS as the national choice for the public sector in the "Information Society Strategy and Action Plan of 2006-2010" [7]. Moreover, in 2007 Turkey's Ministry of Defence, installed Pardus Linux (a national operating system on Linux)[1,] which uses part of a broader national digital archiving and analysis project, on 4,500 desktops and 500 of the servers in its Military Recruitment Division [26].

The FLOSS community in Turkey has rapidly grown in the last 10 years, and there is also a growing interest in participating in international projects. According to Ghosh's survey [11] carried out for European Commission in 2002 covering 2634 FLOSS developers, % 0.4 (around 104 people) of the participants had Turkish nationality, while % 0.3 (around 82 people) of the participants were resident or working in Turkey.

The Turkish Linux Users Group (LUG) was established in 1995, during the '1st Internet Conference' by the Linux Community, as a virtual association. LUG is not only a "developer community" that focuses on the development of FLOSS projects, but also a community that gathers the users and developers of FLOSS in support of free software and Linux/OSC software. It became a chartered association under the name 'Linux Users Association'

[1] Pardus is GNU/Linux distribution developed by TUBİTAK (Turkish Science and Technology Council) according to computer literates basic desktop needs; uses existing distributions' dominant parts as concept, architecture or code; provides easy use, configuration,

bringing together academic, commercial, governmental and non-governmental organizations. Linux.org.tr is the web site that has been administrated by LUG since 1995, and has more than 14 mailing lists on various Linux topics for members at different technical knowledge levels. These lists include linux-programming, linux-servers, linux-networks, linux-core, linux-security, linux-setup, linux-hr, linux-desktop, linux-hardware, linux-chat, etc. This web site is updated via the support of the members, and is in the service of 27.000 visitors monthly. The society's ftp (file transfer) site is accepted as the point of Linux distribution and has been used for assistance and common study field; over 1.000.000 e-mail messages are distributed among members monthly. LUG has established volunteer groups working on various projects such as localization, software development or documentation in Turkish as well as providing training at universities all over the country [30].

**Methodology of the research**

In order to explore the motivation for participating in a FLOSS community and to discover the innovative characteristics of these communities, we conducted a survey among the Linux.org.tr [30] community's members, in Turkey, who are mostly FLOSS software developers. By using the results of this survey, we aimed to understand the type and the level of the contributions of the communities in FLOSS development, and hence the community's involvement in the innovative processes of FLOSS development in a developing country.

The members of the Linux-programming mail list, which is the major Linux developer community in Turkey (operating on Linux.org.tr), were asked to participate in an online-survey that included questions related to the following areas:

1. Main features and characteristics of Linux community members
    - Demographic questions
        - age

installation with configuration environment and tools that can be converted to an autonomous system (pardus.org.tr, 2008).

- education
- foreign language competency
- experience / professional experience in programming (years)
- employment status

- Work place related questions – employer characteristics
  - academy/research institution/ private sector,
  - local/international/
  - software producer/IT Function
  - innovativeness
  - FLOSS support

2. Motivation Factors affecting the participation in FLOSS community

Referring to the classifications of Ghosh [15], and Lakhani and Wolf [25], the survey focused mostly on the extrinsic "social", "product related", "signalling" motivations and on the intrinsic "political" motivation of the community members (direct monetary motivations - like payment for work - are not explored in this survey). Therefore, motivation factors that are explored, are related to:

- Individual benefits
- Individual contributions
- The innovative contribution of community to the employers/corporations of participants (benefits on corporate level)

Additionally, the types of technical problem solving and innovative activities in the FLOSS Community are also questioned:

- Determining/reporting errors (bugs)
- Analysing/fixing errors (bugs)
- Determining/reporting an improvement need (except debugging)

- Recommending solutions on a defined/reported improvement need (except debugging)
- Developing a hands-on solution for the defined/reported improvement need (except debugging)
- Participating in/Getting use of unique/innovative FLOSS Development Projects of the community

*Sample Size*

There is no clear definition of whom a FLOSS developer is (other than someone who writes free/open source software code) and there is no universal or national list and hence no accurate information of the number of developers [13], therefore it is very difficult to determine a representative sample size. Moreover, defining and reaching a satisfactory number of participants is difficult in developing countries where reliable and formal/systematic statistics/data are generally not available and nongovernmental organizations are not mature enough to fill the gap [54].

Also, the administrators of the Linux.org.tr web site [30] do not announce the exact number of their members in their lists, due to the common practice of a member having more than one record because of their use of different e-mail addresses. However, linux.org.tr is also an association that has 900 registered members. 134 FLOSS software developers from the Linux.org.tr community's Linux-programming mail list responded to our online survey. Accordingly, when the number of registered members is accepted as the total population, the participation rate in the survey is roughly 15 %. Therefore the responses may be accepted as indicative of Turkish Linux users and provide reliable data only on the actual respondents.

## Survey Results

*The main features and characteristics of Linux community members*

The findings of the survey have shown that the Linux community in Turkey is rather young, with a high educational level, and a strong professional background in the IT sector:

- The average age of the respondents is 28.2 (median: 26.0). About half of them are between 24 and 28 years old, while a quarter are between 19 and 23 years old; only one fifth of the respondents are between 29 and 33. The tendency in these communities is that members between the ages of 23 and 28 years old have an increased need for assistance from other, more experienced community members as their professional programming activities increase. After this age, and as their expertise increases, the need for assistance decreases. When some of these developers get promoted or move to other administrative positions in their companies, they are no longer involved in programming, or they rarely do hands-on programming.
- The average professional experience period is 3,5 years (median is 4), while 60% of the respondents have 1 to 5 years of experience in professional programming. About one-fifth of the respondents have no professional experience.
- A majority (60 %) of the respondents' university degrees are in IT and programming, of which 20 % have a graduate degree (master or/and PhD).
- Most of the respondents (77 %) are competent in English at different levels (30% are excellent, 27 % are very good, 20 % are good).
- Almost half of the community members are employees, but a relatively high share (11 %) are self-employed. Students constitute 29 % of the sample. The rate of unemployment in the community is 7 %.

- Only 20% of the respondents work for universities, 75 % work in the private sector while 5 % are employed in the public sector. Most (75 %) of community members are employed by local companies.
- Almost two-thirds (65%) of the respondents are employed in software companies, and 5 % work in foreign/international software companies. One-third of the respondents are working in the IT Departments of companies that are not in the software industry.
- Employers of 80 % of the respondents support and/or use FLOSS as a policy.
- Three-fourths of the employers develop innovative products. A majority (60 %) of these companies allocate resources for R&D and new product development activities. 25 % of employer companies have received financial support for their innovative activities from the government and EU Funds.

## Motivation Factors Affecting the Participation in FLOSS Community

### Motivation Factors that are Related to Benefits

As shown in Table 1, all community members mostly agree that they can learn more from the community and other developers. This is one of the major social motivations according to Ghosh [15] and an extrinsic motivation for Lakhani and Wolf [25].

*Table 1. Motivation Factors related to benefits of participating in FLOSS community*

| Motivation Factors | Totally Agree | Quite Agree | Partly agree | Disagree | Totally disagree | Totally Agree | Totally Disagree |
|---|---|---|---|---|---|---|---|
| Learn and develop new skills on programming | 77% | 12% | 11% | 0% | 0% | 100% | 0% |
| Get help for the programmes that I work on and in realizing my ideas for a software | 57% | 43% | 0% | 0% | 0% | 100% | 0% |
| Solve problems that I could not solve about the programmes I use | 76% | 17% | 7% | 0% | 0% | 100% | 0% |
| Distribute software programmes that I developed | 50% | 13% | 25% | 6% | 6% | 88% | 12% |

All of the respondents (totally or quite) agree that they get help for the programmes that they work on and for realizing their ideas for software. These are product-related and extrinsic benefits, in other words benefits that are directly related to the "software production" and the "quality improvement" processes. Therefore, they both have "economic value" and high impact on innovation capability that is strongly linked to "problem solving and realising new ideas". All respondents also believe that the community helps them to solve the problems they faced in the programmes they use. This is one of the motivations that most respondents "totally agreed" on, together with "learning and developing skills". Table 2 also shows that 88% of the respondents agree that they benefit from the community in distributing the programmes that they developed. This motivation is a signalling motivation that enhances the reputation of the developer, providing economic and sometimes even financial benefits in the long term.

Figure 1 shows that the vast majority of participants agree that they "benefit" from their participation in the Linux community, in terms of their four "extrinsic" needs of social, product-related, and signalling, as software developers. Hence, Linux.org.tr community has similar "benefiting" motivations to wide-spread international communities discussed earlier.

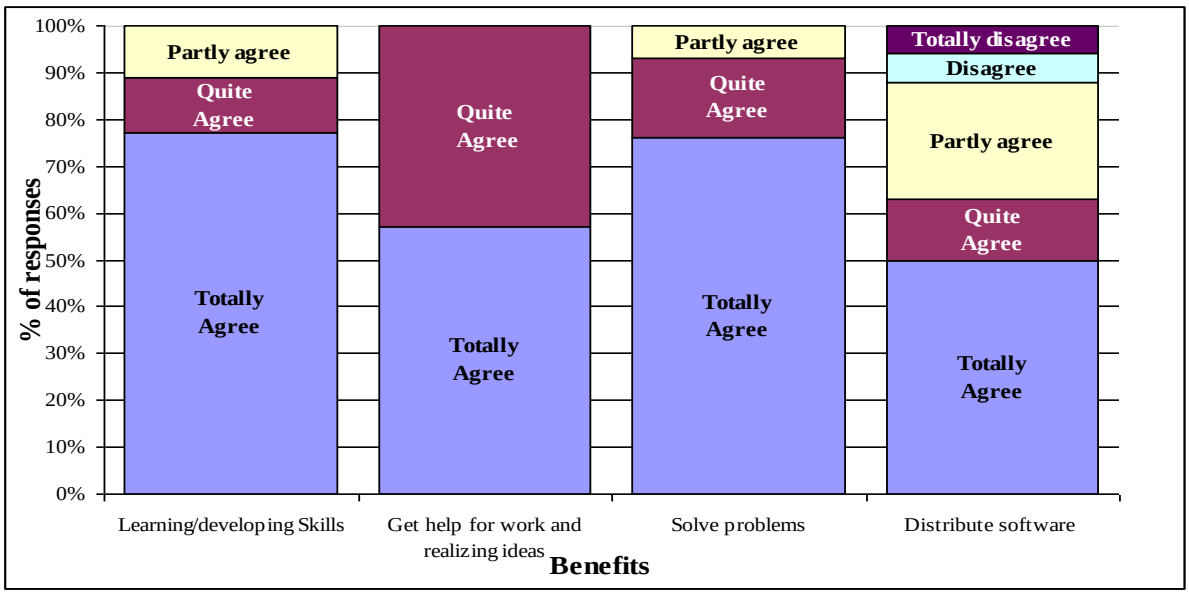


*Figure 1. Benefits of participation in FLOSS community (Table 1)*

**Motivation Factors that are related to contributions**

Regarding motivation related to contributions to the community (or to the work of other members in the community) Table 2 shows that 82% of respondents are motivated by sharing their knowledge and skills with other members of the community. This may be classified as social and extrinsic motivation corresponding to the benefiting motivation of "learning and developing skills" in Table 1.

*Table 2. Contribution Related Motivation Factors for Participating in FLOSS Community*

| Motivation Factors | Totally Agree | Quite Agree | Partly agree | Disagree | Totally disagree | Total Agree | Total Disagree |
|---|---|---|---|---|---|---|---|
| Share knowledge and skills | 64% | 17% | 11% | 0% | 8% | 92% | 8% |
| Improve FLOSS products of other developers | 40% | 7% | 27% | 20% | 6% | 74% | 26% |
| Take part in FLOSS development projects of the community | 29% | 0% | 48% | 17% | 6% | 76% | 23% |
| Contribute to the improvement and diffusion of FLOSS | 58% | 29% | 7% | 0% | 6% | 94% | 6% |

Improving others' work is a hybrid motivation, as it is extrinsic for having some signalling attitude (enhancing reputation in community) and it is also intrinsic for having some enjoying/self-satisfying attitude (intellectual stimulation to write codes, like to work with other developers etc.) Only 40% of the respondents totally agree that they are motivated by improving the FLOSS products of other developers, while almost one-fourth disagree about being motivated by such contributions. As that motivation is the one least agreed on (74% as shown in Total Agree column in Table 2), the demographic characteristics of these respondents are also analysed to find out if there is a correlation between the age or experience level of the developer and his/her attitude towards contributing to others' work in the community. The respondents who "disagreed" are aged between 19 and 23 years, with an average of 1 to 3 years experience in the profession. Hence, the contribution to others' work seems to be dependent on the level of expertise.

One of the critical motivations related to innovation is the degree of contribution to or taking part in FLOSS projects in the community. Only 29% of respondents totally agree that

they are motivated by taking part in FLOSS projects in the community, while almost half of the respondents (48%) partly agree with this. The demographic characteristics of the respondents who "totally agreed" show that they are aged above 34, with an average of 6 to 8 years experience in the profession. Hence, the level of contribution to the projects in the community is related to the level of expertise as well.

A vast majority of the respondents (94%) believe that they are motivated by contributing to the improvement and diffusion of FLOSS projects. This is an intrinsic and political motivation that indicates that political motivations are as important as the "economic" or product related motivations. Hence, it can be concluded that the participation in the FLOSS community is not a short cycled game of "give and take" or "win-win", it is rather being a part of a "techno-professional frontier".

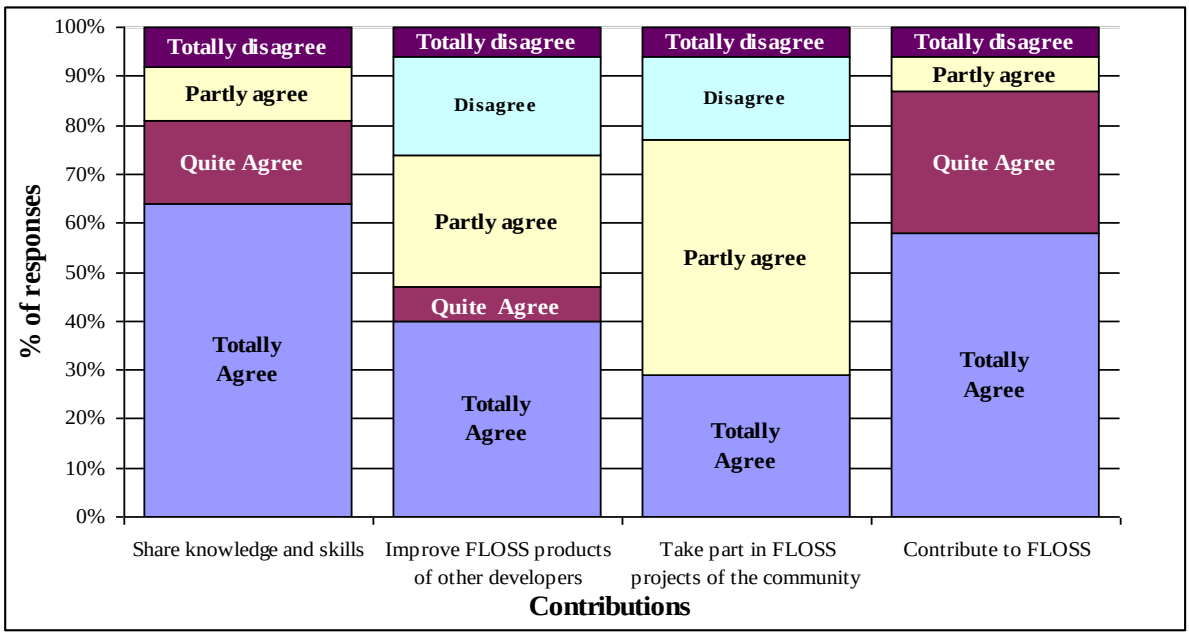


*Figure 2. Comparison of Contribution Factors of FLOSS community Participation (Table 2)*

Figure 2 illustrates that the vast majority of participants agree that they are motivated by contributing to the Linux community, in terms of their four basic professional/productive needs as software developers.

**Motivation Factors that are related to the innovative contribution of community to the employers/corporations of the participants (benefits on corporate level)**

The survey also explored "the perceptions of the participants about the contribution of their community membership to their jobs, and hence on the companies they work for".

*Table 3. Motivation Factors related to the innovative contribution to the professional corporate life*

| | Totally Agree | Quite Agree | Partly agree | Disagree | Totally disagree |
|---|---|---|---|---|---|
| 1) I believe that developer and user communities may help for simplifying and fastening the process, decreasing the costs of the product development processes of software developer companies, etc. | 28% | 45% | 18% | 9% | 0% |
| 2) I share the knowledge and information which I get from developer/using community with my colleagues in my workplace. | 45% | 37% | 15% | 3% | 0% |
| 3) I believe that participating in the community had helped me in improving my knowledge, skills and competencies in programming that I use in my work place. | 28% | 54% | 18% | 0% | 0% |

As can be seen from Table 3, most of the respondents (91%) believe that developer and user communities may help software developer companies in improving their product development processes. Nearly half of the respondents (45%) totally agree that they share the knowledge and information that they gain from the community in their workplace, indicating that they form a kind of knowledge transfer link between the community and companies.

Also, all respondents believe that they use their knowledge, skills and competencies in workplace programming activities which are improved through their participation in the community. This indicates that companies benefit from the participation of their employees in the FLOSS community, since 65% of the respondents work for local software companies, Employers of 80% of the respondents support and/or use FLOSS. In this context, even the companies that do not support FLOSS may benefit from the participation of their employees in the FLOSS community. When the responses of participants from non-supporter companies are examined, it is found that they share their improved knowledge, skills and competencies in programming gained by participating in the community.

**Types of Technical Problem Solving and Innovative Activities in FLOSS Community**

The nature of innovative and product development related work in the FLOSS community is analysed in terms of contributing or benefiting activities related to technical problem

solving, innovative FLOSS development projects and hence the building of technological learning/capability. The types of activities included are as follows:

1. Determining/reporting errors (bugs)
2. Analysing/fixing errors (bugs)
3. Determining/reporting an improvement need (except debugging)
4. Recommending solutions for the defined/reported improvement need (except debugging)
5. Developing hands-on solutions for the defined/reported improvement needs (except debugging.)
6. Participating in/Using the products of unique/innovative the FLOSS Community Development Projects

These "technical problem-solving" activities are categorized based on their relations to "contributions/assistance made" and "benefits/assistance received". Table 4 shows the frequency of the innovative activities in terms of the "contributions" of participants to other community members, while Table 5 explores the level of "innovative benefits" or the assistance that participants received from the community by the participants.

The most significant finding that can be derived from Table 4 is that the main contribution of the members to the community is in "determining/reporting errors (bugs)" of the programs –with 10 or more programs- that were developed by others. However, other than "determining/reporting errors", no community member contributed to 10 or more programs. The average number of contributions to the programs/projects per each community member is about three. 18 % of the participants have never determined or reported any errors on the programs that were developed and distributed by the other members of the community. Analyzing the demographic features of these respondents showed that 70% of them have less than 1 year experience in professional programming.

*Table 4. Innovative contributions/ assistance made (frequency of contributive activities)*

| | Number of Projects/Contributions | | | | | |
|---|---|---|---|---|---|---|
| Question / Criteria | 0 | 1-3 | 4-6 | 7-9 | >=10 | Average |
| 1. Determining/reporting errors (bugs) | 18% | 50% | 15% | 12% | 5% | 3,2 |
| 2. Analysing/fixing errors (bugs) | 8% | 62% | 19% | 11% | 0% | 3,1 |
| 3. Determining/reporting an improvement need (except debugging) | 38% | 50% | 12% | 0% | 0% | 1,6 |
| 4 .Recommending solutions on the defined/reported improvement need (except debugging) | 23% | 51% | 18% | 8% | 0% | 2,6 |
| 5. Developing a hands-on solution for the defined/reported improvement need (except debugging) | 74% | 21% | 3% | 2% | 0% | 0,9 |
| 6. Participating in a unique/innovative FLOSS Development Projects of the community | 86% | 11% | 3% | 0% | 0% | 1,0 |

As can be seen from the Table 4, half of the participants contributed to others' work by determining/reporting the "errors/bugs" or "an improvement need" or "recommending solutions on the defined/reported improvement need" in 1 to 3 programs that were developed and distributed by other members of the community.

A vast majority of community members had never developed a "hands-on-solution for an improvement" for another developer in the community. "Developing hands-on-solution" is the contribution type with the lowest number of average (0,9) per community member. Hence, "developing the program for an improvement need (which includes system design and coding that are more time consuming and relatively harder work)" is a rare type of contribution that community members provide to others. "Fixing and analysing errors" are also coding work, but it is a kind of routine task, requiring less time, and less creativity; so community members mostly contribute to others work in this context (average nr. of contribution per member is about 3, similar to "determining/reporting bugs")

The majority (86 %) of community members do not contribute to unique/innovative FLOSS Development Projects of the community. This is due to the fact that 60 % of the community members have an average of 1 to 5 years professional experience, and this kind of contribution is a process of new product development, and innovative work that requires a high level expertise. However, none of the community members had contributed to seven or more FLOSS projects.

Table 5 shows the frequency of the benefiting activities that the participants of the survey received from the community. Hence, it is possible to see how community members actually benefit from the community in solving technical problems or achieving innovative software.

*Table 5. Innovative benefits/assistance received (frequency of benefiting activities)*

| | Number of Projects/Contributions | | | | | |
|---|---|---|---|---|---|---|
| Question / Criteria | 0 | 1-3 | 4-6 | 7-9 | >=10 | Average |
| 1. Determining/reporting errors (bugs) | 9% | 36% | 28% | 9% | 18% | 4,6 |
| 2. Analysing/fixing errors (bugs) | 60% | 17% | 11% | 7% | 5% | 2,0 |
| 3. Determining/reporting an improvement need (except debugging) | 37% | 34% | 18% | 8% | 3% | 2,7 |
| 4 .Recommending solutions on the defined/reported improvement need (except debugging) | 29% | 44% | 18% | 0% | 9% | 2,8 |
| 5. Developing a hands-on solution for the defined/reported improvement need (except debugging) | 47% | 32% | 14% | 7% | 0% | 2,0 |
| 6.Getting use of unique/innovative FLOSS Software that were developed by Projects of the community | 73% | 27% | 0% | 0% | 0% | 0,9% |

Similar to the contributions shown in Table 4, community members mostly benefit from the community in determining/reporting errors (bugs) of the programs they developed. The average number of contributions received by each member from the community in this context is almost 5 per each community member. It is also notable that 18% of the participants stated that they had received help for more than 10 projects. Therefore, the average number of received contributions is higher than the average number of contributions made. This gap is caused by the performance of community members who contributed to 10 or more programs/projects. In addition, the different levels of contribution in determining and fixing bugs are worth analysing. When a bug is reported, the developer can generally fix it himself/herself more easily than any other developer. When an unexpected problem arises that requires a higher level of expertise, the developer may ask for help from others. But it must be noted that debugging is generally a “correcting activity” rather than an “innovative activity or preventive development”. Innovative work is often needed if correcting the bugs requires exceptional restructuring of the basic design.

Another important finding is that the average number of benefited programs/projects is about 3 per community member in terms of “determining/reporting or recommending

solutions on a defined/reported improvement need". These benefits include "commentary contributions" rather than the "hands-on" work of contributors. Accordingly, 60 % of members did not receive any help in analyzing/fixing errors, and 47 % could not benefit from the community in terms of "developing hands-on solution for an improvement need".

On the other hand, an important portion (63%) of community members does not use unique/innovative software developed in the projects of the community. The maximum number of that kind of software that community members use is only 2. Moderators of the community stated that the community had launched a maximum of 4 projects up to the survey date, and a maximum of 2 of them were completed and distributed to the community members. This points out the major weaknesses of the community in terms of innovativeness, because the frequency of developing and implementing unique and innovative projects (that will produce innovative products) is the main performance indicator for innovative capability of a network or organisation. To evolve to an innovation network, community has to develop and improve its capacity for launching and implementing higher numbers of innovative unique projects.

**Summary and Conclusion**

A sample of the Linux community was studied as an example of FLOSS communities in Turkey to determine whether these user/developer communities utilize FLOSS only as knowledge sharing platforms or benefit from it as an innovation networks.

The members of the Linux user/developer community in Turkey are young, highly educated with a strong professional background and mostly work for local software companies that support or use FLOSS. There is a positive correlation between the age or experience level of the developer and his/her attitude towards contributing to others' work in the community and taking part in the community's FLOSS projects.

All members of the community are motivated by the social and product-related extrinsic benefits of the community, like learning and improving their skills, receiving help in solving their problems in their programmes, and realizing their ideas for software. These benefits are related to the "software production" and "quality improvement" processes, and have both an "economic value" and a high impact on innovation capability that is strongly linked to "problem solving and realising new ideas". Distributing the programmes that they developed is also a motivation that enhances their professional reputation and marketing abilities.

"Sharing their knowledge and skills with other members of the community" is the leading motivation and can be defined as a social extrinsic motivation, while improving others' work and taking part in FLOSS projects of the community is a motivating factor that has both extrinsic and intrinsic characteristics. Contributing to the improvement and diffusion of FLOSS is an intrinsic and political motivation factor for the vast majority of the respondents. Existence of this kind of political motivations, that are as important as the "economic" or product related ones, shows that participation in the FLOSS community is not a short cycled "give and take" or "win-win" game, it is rather being part of a "techno-professional frontier".

The innovative contribution of the community to professional corporate life also motivates participants, as they believe that participating in the community may help software developer companies improve their product development processes. Most of the community members build an informal kind of knowledge transfer link between the community and their companies by sharing their knowledge and information that they receive from the community in their workplace. Moreover, in their workplace, community members use their knowledge, skills and competencies in programming that are improved by their participation in the community. Hence, their companies indirectly benefit from the FLOSS community since two-thirds of the participants work for local software companies that support and/or use FLOSS as a policy. One-fifth of the participants' employer companies that do not support or

use FLOSS also benefit from the participation of their employees in the FLOSS community in similar ways, proving the "infectious" characteristic of FLOSS.

The community is perceived by its members as a strong learning and collaboration environment. The members receive and provide help in solving technical problems in software. By developing their technical problem solving skills, community membership offers opportunities for improving their innovative capabilities. The main contribution of the members to the community is in determining/reporting errors (bugs) of the programs developed by others. Similarly, community members mostly benefit from the community in determining/reporting errors (bugs) of the programs they developed. However, contributions to technical analyzing and/or debugging processes are rare; rather developers act as ordinary users in defining the debugging or improvement need. This contribution should not, however, be undervalued, because making user or performance tests, and reporting the results in terms of correction/improvement needs in programming terminology significantly ease the task of the developer, and help in the technical problem solving processes. Additionally, members contribute to other's programmes by suggesting improvements or developing a hands-on solution (a process that require creative work) rather than correcting errors.

Benefits from the community are higher than contributions to the community in terms of technical problem solving and improving innovative capabilities. This gap is caused by the extraordinary performance of some community members who are more experienced and have "higher expertise". These members (mostly leaders/moderators/founders of the community) contribute to the activities of the community much more than other members, and hence they increase the overall "beneficiary attitude" of the community.

Innovative Projects, launched and completed by the community, are insufficient in number and quality. The vast majority of community members do not contribute to the community's unique/innovative FLOSS Development Projects. On the other hand, an important portion of the community members does not use the products of the community's

projects. This is one of the major weaknesses of the community in terms of innovativeness, as the frequency of developing and implementing unique and innovative projects (that will produce innovative products) is the main performance indicator for the innovative capability of a network. However, this kind of FLOSS community still has the potential to evolve towards being an innovation network if the community develops and improves its capacity in conducting a higher number of innovative/unique projects with higher quality.

The level of individual contributions to the community is strongly related to the level of expertise and professional knowledge. Hence, as members' expertise and knowledge improves, innovative development and technical problem solving activities will take place more often and more effectively. Then, community would be able to launch and implement more unique and innovative projects with success. The support of the local software industry and academic institutions would help the community to evolve in this context.

In short, the improvement of local FLOSS communities will help local software developers increase their level of expertise, technical problem solving skills and, in the end, innovative capabilities. The evolution of these communities to innovation networks will provide an infrastructure for knowledge creation that is needed for building the local innovative capacity in software development that is crucial for developing countries.

**References**

[1] Ansal, H., Yildirim N. and Yıldırım, H. ; "Foresighting the Open Source code from a developing country perspective" , IAMOT 2007 (International Association for Management Of Technology) Conference, Miami/USA. May, 2007.

[2] Bruggink, M. ; "Open Source Software: Take it or leave it?" International Institute for Communication and Development Report, 2003.

[3] Carr, N.G. ; "The Ignorance of Crowds*",* in *Strategy+business Journal*, No:47, Booz Allen Hamilton, 2007.

[4] Chesbrough, H.W., Vanhaverbeke, W. and J. West ; "Open Innovation: Researching a New Paradigm", Oxford University Press, 2006.

[5] Corti, E. and Storto, C. ; "Knowledge creation in small manufacturing firms during product innovation: an empirical analysis of cause-effect relationships among its determinants" in *Enterprise and Innovation Management Studies*, 1 (3), 245-263, 2000.

[6] Daffara, C. and Barahona, J.B.; Free Software / Open Source: Information Society Opportunities for Europe? Working group on Libre Software1, 2000 , Version 1.2 (work in progress), 2000.

[7] DPT-State Planning Organisation High Planning Council,; "Information Society Strategy and annexed Action Plan" in *Official Gazette Nr. 26242 28th July 2006.* Decision number 2006/38- July 11, 2006, Ankara, 2006. Available at: <www.bilgitoplumu.gov.tr/yayin/OECD_ITPolicyQuestionnaire2008_Turkey.PDF>.

[8] Dravis, P. ; "Open source software: Perspectives for development" in *Information for Development Programme/World Bank*, Washington DC, 2003.

[9] Free Software Foundation, 2009. Available at: <www.fsf.org>.

[10] Gacek, C., and Arief, B. ; "The Many Meanings of Open Source", in *IEEE Software,* Volume: 1, No:21, 34-40, 2004.

[11] Ghosh, R.A. "Understanding Free Software Developers: Findings from the FLOSS Study" Survey of Free/Open Source Developers, FLOSS project, European Commission/IST, 2002.

[12] Ghosh, R.A., 2003. Licence fees and GDP per capita: The case for open source in developing countries. First Monday 8 Dec., Available at: < http://firstmonday.org/issues/issues8_12/ghosh/index.html>.

[13] Ghosh, R. A., Glott R., Krieger B. and Robles, G.; "Community above Profits: Characteristics and Motivations of Open Source and Free Software Developers" in *MERIT/Infonomics Working Paper Series*, 2003. http://floss.infonomics.nl/papers/,

[14] Ghosh, R.A.; "Towards an Innovation-Driven Economy Through Industrial Policy-making" in *The Innovation Journal: The Public Sector Innovation Journal*, Volume 10(3), article 34.pp.23, 2004.

[15] Ghosh, R.A. "Understanding Free Software Developers: Findings from the FLOSS Study" in *Perspectives on Free and Open Source Software, (*Eds.) J. Feller, B. Fitzgerald, K. Lakhani & S. Hissam, pp. 23-46, MIT Press, 2005.

[16] GNU.org, 2009. Available at: <www.gnu.org>.

[17] Harhoff, D., Henkel, J., von Hippel, E. "Profiting from Voluntary Information Spillovers: How Users Benefit by Freely Revealing Their Innovations", in *MIT Sloan School Working Paper,* No: 4125, 2000.

[18] Hars A, Ou S.; "Working for free? Motivations for participating in open source projects". In *International Journal of Electronic Commerce* 6(3): 25–39, 2002.

[19] Hippel, E. von; "Innovation by User Communities: Learning from Open Source Software*",* in *Sloan Management Review*, Vol. 42 (4), 82-86, 2001.

[20] Hippel, E. von.; "Horizontal innovation networks—by and for users," in I*ndustrial and Corporate Change*, Vol. 16 (2), 293-315, 2007.

[21] Hippel, E. von., Krogh, G.; "Open Source Software and the 'Private-Collective' Innovation Model: Issues for Organization Science" in *MIT Sloan Research Paper No. 4739-09* , 2009.

[22] Krogh, G. ; "Open Source Software Development", in MIT Sloan Management Review, No: 44, 14-18, 2003.

[23] Lakhani, K., Wolf, B., and Bates, J; "BCG Hacker survey of free software/open source developers", The Boston Consulting Group, In Cooperation with OSDN, 2002.

[24] Lakhani, K. R., and Hippel, E. von.; *"*How open source software works : « free » user-to-user assistance", in *Research Policy*, Vol. 32, 923-943, 2003.

[25] Lakhani, K., and Wolf, B.; "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects" in *Perspectives on Free and Open Source Software, (*Eds.) J. Feller, B. Fitzgerald, K. Lakhani & S. Hissam, pp. 23-46, MIT Press, 2005.

[26] Lewis, J., 2007. Government open source policies. Report. Center for Strategic and International Studies, Washington D.C., USA.

[27] Lehenkari, J.; "The Networks Of Learning In Technological Innovation" in *Research Report - 10 for Center for Activity Theory and Developmental Work Research*, Helsinki University Press, Helsinki, 2006

[28] Lerner, J., and Tirole J. ; "Some Simple Economics of Open Source", in *Journal of Industrial Economics*, Vol. 50, 197-234, USA, 2002.

[29] Lin, Y.; "Hybrid Innovation: The Dynamics of Collaboration between the FLOSS Community and Corporations" in *Knowledge, Technology, & Policy*, Winter 2006, Vol. 18, No. 4, pp. 86-100, 2006

[30] Linux.org.tr; "Official Web site of Linux Users Association-Turkey", http://www.linux.org.tr, 2009.

[31] May, C. ; "The FLOSS alternative: TRIPs, non-proprietary software and development", Journal Knowledge, Technology & Policy, Springer Netherlands, Volume: 18, No: 4, December, 2006

[32] May, C.; "Opening other windows: a political economy of 'openness' in a global information society" in *Review of International Studies*, 34, 69-92. Cambridge University, 2008.

[33] Microsoft Site Builders; Corporate Web Site. Available at : < http://www.sitebuilders.org>, 2009.

[34] Microsoft.com; "Shared Source Initiative" Available at : < http://www.microsoft.com/resources/sharedsource/initiative/faq.mspx>,2009.

[35] Mockus, A., Fielding, R., and Herbsleb, J.D.; "Two Case Studies of Open Source Software Development: Apache and Mozilla", ACM Transactions on Software Engineering and Methodology, Volume: 3, No: 11, 309-346, 2002.

[36] Netcraft.com ; "September 2009 Web Server Survey", 2009. http://news.netcraft.com/archives/2009/09/23/september_2009_web_server_survey.html

[37] Open source Initiative;. Available at: <www.opensource.org>, 2009

[38] O'Reilly, T.; "Open Source Paradigm Shift", In *Perspectives on Free and Open Source Software*, (Eds.) J. Feller, B. Fitzgerald, K. Lakhani & S. Hissam, pp. 461-482, MIT Press, 2004.

[39] Page, S.; "The Difference: How the Power of Diversity Creates Better Groups, Firms, Schools, and Society," Princeton University Press, 2007.

[40] Rasch, C., 2000. A brief history of Free/Open Source Software Movement. Available at: <http://www.openknowledge.org/writing/opensource/scb/brief-open-source-history.html >.

[41] Rajani, N., Rekola, J., and Mielone T.; "Free as in Education.: Significance of the Free/Libre and Open Source Software for Developing Countries." Report for Ministry for Foreign Affairs – Finland, Helsinki, 2003. Available at : < http://www.maailma.kaapeli.fi/FLOSSReport1.0.html>

[42] Raymond, E. S.; "The Cathedral and the Bazaar", O'Reilly & Associates Inc., 2001.

[43] Reed, M.; "eGovernance and free software: How they are changing developing countries" in Panel Report of United Nations University International Institute for Software Technology (UNU-IIST) Event, New York, USA, 2006.

[44] Rosenberg, N.; Innovation and Economic Growth, Paris: OECD, 2004.

[45] Scacchi, W.; "Socio-Technical Interaction Networks in Free/Open Source Software Development Processes". In *Software Process Modeling. In International Series in Software Engineering.* S.T. Acuña and N. Juristo *(*Eds.) Vol. 10, pp. 1-27, Springer Science+Business Media Inc., 2005.

[46] Solow, R.; "Technical Change and the Aggregate Production Function" in *Review of Economics and Statistics*, 39:312-320, 1957.

[47] Stallman, R.; "Your Freedom needs Free/Libre Software" in *Richard Stallman's ZSpace.* http://www.zcommunications.org/your-freedom-needs-free-libre-software-by-richard-stallman, 2007

[48] Stoltz, M.; "The Case for Government Promotion of Open Source Software" retrieved November 2, 2003, from NetAction.org database, 1999.

[49] Storto, C.; "How Small Firms can Learn from Planned and Random Technical Problem-Solving" In: *Knowledge Ecology in Global Business: Managing Intellectual Capital: Learning Organizations or Organizations for Learning?*, Eds: Miltiadis Lytras; Patricia Ordonez de Pablos, Premier Reference Source, Pages 108-131, 2009.

[50] United Nations Conference on Trade and Development [UNCTAD]; "E-commerce and development report-2003", New York, 2003. Available at : <http://www.unctad.org/en/docs/ecdr2003_en.pdf>.

[51] Weber, S. ; "The Success of Open Source", Cambridge: Harvard University Press, 2004.

[52] Weerawarana, S., Weeratunga J. ; "Open source in developing countries" SIDA Report: Swedish International Development Cooperation Agency, Stockholm, Sweden, 2004.

[53] Wheeler, D.A. ; "Secure Programming for Linux and Unix , HOWTO", Free Software Foundation, 2003. http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO/history.html

[54] Yildirim, N., Ansal H.; "Challenges of Foresighting in a developing country," In *IAMOT (International Association for Management Of Technology) Conference*, Vienna, 2005.

[55] Zimmermann, J. B. ;"L'industrie du logiciel: ébauche d'une approche prospective," *In Terminal Hiver 97 - Printemps 98,* Nr. 75., 1998.

[56] Zimmermann, J.B , Hapke, M. and Jullien, N.; "Does using libre software imply contributing to its development?" In *CALIBRE Workshop*, Paris, 2005.

[57] Zimmermann, J.B., Jullien, N.; "Free/Libre/Open Source Software (FLOSS): lessons for intellectual property rights management in a knowledge-based economy." In *DIME London Conference on Intellectual Property Rights for Business and Society*, London, UK, 2006.