

# Case study of company's relationship with open source community in open source software development

Juho Lindman

Helsinki School of Economics, Information Systems Science

*juho.lindman@hse.fi*

Topi Uitto

Accenture, Finland

*topi.uitto@accenture.com*

**Abstract.** Open Source Software (OSS) development success in business context is determined by the relationship between the developer community and the benefiting company. The purpose of this case study is to examine this relationship. We review extant literature on the OSS development, characteristics of the OSS communities and OSS development, moving on to the business rationales of a companies engaging in OSS development. We construct a framework for analyzing a company-community relationship and test our framework with data from semi-structured interviews of corporate representatives who have worked closely with open source software development. As a result, we propose a framework for analyzing company-community relationship in the OSS development context.

# 1. Introduction

Open source software (OSS) has been anticipated to have groundbreaking effects on the software industry (Fitzgerald, 2006). In some industries OSS has already become prominent while in others the use of open source products has become increasingly common. In addition to using OSS, companies can benefit from the external innovation and development resources by engaging themselves with OSS communities in OSS development. Communities on the other hand may provide companies' resources to leverage their software development process. There is however a catch here: in order to benefit from the OSS development the companies need to understand how to mobilize the OSS community. The focal, yet under-researched, phenomenon is the company's relationship with an open source community. In order to work fruitfully together, a company must recognize the important aspects of the relationship and how to address them. This information can support facilitating and managing the OSS development. This might not seem as an easy task, as traditionally OSS communities and for-profit companies have been seen unlikely partners because of their different rationales for existence (Dahlander and Magnusson, 2005, 482). Open source communities claim free use, free distribution and free modification of software while companies have touted for proprietary software and maximizing their profits.

Our goal is to study the role of open source community in companies' open source software development activities by decomposing one relationship of a community and a company into smaller units. Based on an extensive literature review, we construct a framework and test it with one pilot case.

We take the perspective of the company engaging in software development with a community. We would like our research effort to shed light on the primary research question: *1. What is the nature of the relationship between company and a community in OSS development?* Furthermore, we need to elaborate this further by asking: *2. Why company engages with a community in OSS development?* and, *3. How company interacts with a community in OSS development?*

## 2. History of open source software

OSS history is closely tied to the development of computing starting from main frame computers. Originally (years 1945 – 1969) scientists were primarily responsible for development of computer systems mostly used in mathematical problem-solving (Benussi, 2005). Only much later did the business need for computers overcome the scientific need (Feller and Fitzgerald, 2000). Communication capabilities were developed along with the computing capabilities (Raymond, 1999). The emergence of OSS can be traced back to the

late 1960's and 70's of academic institutions and corporate research centers where code sharing was common among the programmers (Rajala et al. 2006; Raymond, 1999). The birth of UNIX operating system and the invention of C programming language are seen as very influential events along the Open Source history (Benussi, 2005).

The concept of proprietary software was somewhat unknown in 60's and 70's because hardware and software were sold together. Therefore there was no need to protect the software with property rights. However, when software and hardware needed no longer to be bundled, the software became "closed source" to protect the business (Dreiling et al. 2005). The 1980's was a decade of rapidly growing popularity of microcomputers (PC's). Undisputedly, it was a time for proprietary software (Cusumano, 1992). At that time many sold computers were bundled with proprietary software such as MS-DOS and Microsoft Windows. The blooming proprietary software market motivated some people to counter-react (Benussi, 2005). Richard Stallman, a researcher at Massachusetts Institute of Technology, founded a Free Software Foundation in 1985. The foundation was "dedicated to promoting computer users rights to use, study, copy, modify, and redistribute computer programs". He codified these rights into Gnu General Public License (GPL), which is nowadays maybe the most common license for OSS. In 1991 a young Finnish programmer Linus Torvalds released source code of infant operating system called Linux and asked for participation. Over the next years the Linux operating system lured thousands of developers to make the system better by power of the Internet (Benussi, 2005). The Linux project was carried out in GPL advocating the computers user's right that Stallman manifested (Benussi, 2005). The wide adoption of the Internet during the 1990's also meant significant possibilities for OSS development. The interaction between developers became easier and enabled developers in dispersed locations to work towards a common goal. The Linux operating system is a noteworthy milestone because it was the first time that the open source movement revealed its full potential by creating a highly complex operating system.

In 1998 the Open Source Initiative (OSI) was founded to promote OSS business credibility. Nowadays OSI is the other influential organization promoting OSS together with the Free Software Foundation. OSS has continued to grow and attract usage. It has also gained the business credibility OSI sought (Fitzgerald, 2006). Companies are seeing the potential of community-type development, but are facing difficulties in accommodating their own operations to the requirements of OSS. There is a growing need for both company understanding and software professionals that would be used to working with communities. We propose a framework to address the first and educational systems, organizations and OSS communities will hopefully provide the latter.

### 3. Framework for analyses

Figure 1 illustrates our conceptual framework for analyzing company – community relationship. The framework’s leading idea is to analyze the company – community relationship from a holistic point of view and *primarily from company perspective*.

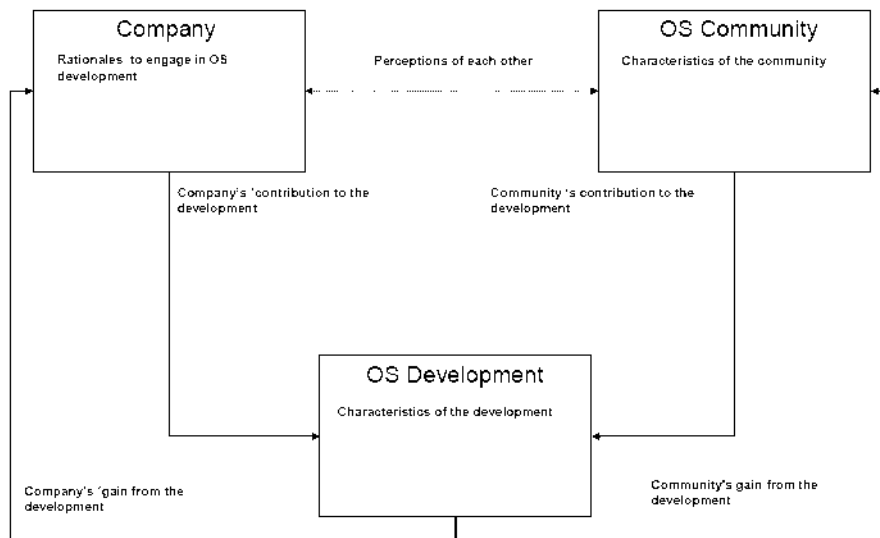


Figure 1. Proposed framework for analyzing company – OSS community relationship

The proposed framework consists of three boxes: company, OSS community and process of OSS development. The framework assumes that a company has rationales to engage in OSS development. The community has also reasons for existence and certain characteristics that describe it. Moreover company and community contribute to the OSS development in some way. Their contribution is illustrated with arrows which lead from the company and OSS community boxes to OSS development box. The OSS development emerges from the contribution and interactions of both company and community. The OSS development process creates something which is then sent back to the company and the community. This is illustrated with arrows leading from the development box back to company and community boxes. Process is iterative and dynamic.

The dotted line between company and community presents the conjecture that neither the company nor the community can directly influence each other, therefore they can only a perception of each other. The interactions between the two take place in the OSS development context.

### 3.1 OSS Company

OSS company is defined as a company engaged in OSS development. The fundamental question is why companies engage in OSS development? Moreover; what are their expectations of OSS development and what they have actually gained? It seems that companies' should have clear business incentive(s) in mind when they enter the OSS field (Fink, 2003). These reasons for engagement can be divided into the economical, social and technological rationales (Bonaccorsi&Rossi, 2006). Business model concept is the culmination of company's business (Hecker, 1999). The question main question is; how the OSS engagement relates to the company's entire business model and especially its revenue model (Rajala et al., 2006). Companies may take different industrial roles in OSS development: OSS provider, OSS integrator, OSS participant and finally ISS participant (Hauge et al., 2007).

Useful dichotomy is also that of inbound OSS, which means OSS projects are adapted to proprietary setting, and outbound OSS, which means proprietary software projects can be made open source (Fink, 2003). Another term for outbound is company sponsored or spin-off project (West and O'Mahoney, 2005). In these spin-off projects companies release internally developed software's code under OSS license inviting involvement of external community (West and O'Mahoney, 2005). There are at least two reasons: for a company to release their code for open source project: to win adoption or to gain development assistance (West, 2003).

### 3.2 OSS community

OSS community means here a community engaged in OSS development. It should be noted that individual developers often have several different kinds of motivations to participate to OSS development (Lakhani&Wolfe, 2005; Raymond, 1999). Our framework focuses more on the similarities between developers (homogenous concept of community) rather than differences between them (heterogeneous concept of community).

When analyzing the community, it is important to understand how a community has emerged; the entire birth process and the initiator. The initiator is important to recognize because often they still possess a lot influence over the development (Raymond, 1999; Fink, 2003). It is necessary to recognize the unique features OSS community has because these characteristics influence community's contribution to the OSS development process (O'Mahoney&West, 2003).

The organizational structure and working practices differ from one OSS community to another. Understanding how the OSS community operates may in fact help companies to co-operate with them better in OSS development (Crowston and Howison, 2005). Nakakoji (et al. 2002) provide useful typology for classifying open source communities and their rationale for existence. West and O'Mahoney (2005) have approached the paradigm of adoption of OSS projects to corporate domain. They argue that "community founded projects remain community managed and are usually structured to prevent a takeover by a firm or other organization." Further they claim that: "A firm may sponsor contributors but firms typically can not become members". This highlights the clear distinction between a company and the community.

### 3.3 OSS development

Companies can undertake different actions in order to contribute and influence the OSS development. We divide these actions into: 1) communication, 2) stimulation, 3) control and 4) coordination. The general rationale for this division is to take into account the rich information about the interactions between the company and community in OSS development process.

The rationale for studying *communication* between company and the community is that it is one of the basic necessities for a company – community relationship. The examination of communication is focused on types of communication and frequency of communication. Mockus et al. (2002) and Feller and Fitzgerald (2002) have described the pattern of communities in their case studies concerning different OSS communities. Also Crowston and Howison (2005) provide useful readings on how the communication differs between different members and on the frequency of the communication. We presuppose that communication also has a strong link to the other areas under examination; stimulation, control and coordination.

Companies *stimulate* community involvement. Simply releasing the outbound software does not mean that users and developers become interested in it (Fink, 2003). The examination elaborates on literature of developer motivation; Bonaccorsi and Rossi (2006) provide a useful taxonomy of OSS participants' motivations. Moreover the examination reflects on Dahlander and Magnusson (2005) and Fink (2003) on marketing and creating incentives for developers to join the development.

The issue of *control* in open source development is somewhat controversial. It is questionable if it is necessary or even possible to control the OSS development. The examination of control identifies the methods that companies possibly use to influence and steer the development. Moreover, the control examination looks at companies' actions to influence the quality of input from the community (Sharma et al., 2003).

*Coordination* means exploring what are companies' efforts to coordinate the development process (Mockus et al., 2002; Egyedi et al., 2003). What type of actions and roles do companies take in the OSS project coordination?

### 3.4 Relationship

The concept of relationship is defined as description of the whole system of interactions between a company and OSS community in OSS development. In order to understand the relationship we are required to understand company's business rationales, community's characteristics and OSS development. In addition we need to consider how the nature of the relationship influences company's business? Can a company work in favor of a fruitful relationship with OSS community?

Companies and OSS communities have fundamentally different motives for existence (Dahlander and Magnusson, 2005, 482). Therefore, they also have different motivations to engage in OSS development (Bonaccorsi and Rossi, 2006). Companies' rationales for OSS development tend to be related to economic and technological factors while members in the OSS communities emphasize the social motivations (Bonaccorsi and Rossi, 2006). One can aggravate that companies seek for maximizing their profits while OSS communities cherish altruistic values (Dahlander and Magnusson, 2005). Consequently it seems somewhat challenging to build a relationship between a company and community which would align the interests of both actors.

Dahlander and Magnusson (2005, 487) have studied Nordic OSS companies and have come up with a typology for company-community relationship. Their typology includes symbiotic, commensalistic and parasitic approach. The authors warn about considering these categories as distinctive or exclusive. Rather, they should be seen as "a steps on a continuum regarding the benefits for the communities deliberately searched by the OSS firms."

Bonaccorsi and Rossi (2005) studied OSS companies' motivations and actions in OSS development and found significant discrepancies. Based on findings they identified four types of company approaches to OSS community and OSS

development: 1) Non-community oriented-firms, 2) Incognito Community Oriented firms, 3) Community Oriented-Firms and 4) Opportunistic Firms. This classification is based on declared intrinsic motivations such as conforming values of OSS communities and then comparing those motivations on actions. Non-community oriented firms do not share the intrinsic values of OSS communities and they act accordingly. Incognito community oriented firms act as if they would conform to OSS values but have stated that they do not. Community oriented firms have strong intrinsic motivations and they act in line with these motivations. Finally opportunistic firms claim to agree with OSS values but actually their behavior shows that they are not.

## 4. Research methodology

We conducted a case study to test the framework derived from literature for company-community relationship analysis. A descriptive case study method was selected as research method since it would reveal how fitting the framework would be and what issues still need to be addressed.

Case is: “a specific, a complex and functioning thing” (Stake, 1995). Mentioned notion of specific is considerably controversial. Stake (1995) considers a case rather as an object than a process. Yin (1994, 13) defines a case study as empirical inquiry that “investigates a contemporary phenomenon within its real-life context especially when the boundaries phenomenon and context are not clearly evident.” Case study method is a preferred method as the research question aims for understanding a phenomenon by asking questions of the type “how” or “why” when the researcher does not have the control over related event, when the phenomenon is changing over time and is closely tied to the actual context (Yin, 1994). As a research method it emphasizes understanding one phenomenon (one case) without prejudice (Stake, 1995).

We used this method by selecting one specific case and tested our framework to see how it explains the elements of the relationship. While we hesitate to compare the explanations it offers to other explanatory frameworks, we can argue to be able grasp quite many of the necessary interactions.

Obtaining the data from different sources and analyzing it from various angles enables the triangulation which can help to improve the case study validity and reliability (Yin, 1994). Data for the case study was acquired from two main sources; 1) from semi-structured interviews and 2) from secondary sources. Secondary sources were used to obtain data from the company and its products. These sources included corporate web pages, annual reports and also the initial

case description provided the companies themselves as a contribution for participating in three year European industry research project.

Semi-structured interviews were conducted in the case firm. The semi-structured interviews were seen as a suitable way of conducting the data gathering because it gives the interviewee and respondent ample freedom and also at the same time ensures that relevant information is collected (Corbetta, 2003, 271). The original interview format was based on business model concept by Rajala (et al. 2006). This model has four general parts which are offering, resources, relationships and revenue model.

The semi-structured interviews are a challenging way of making data collection, since the researcher might have to be able to “read between the lines” in the interview (Yin, 1994). Therefore the interviews require understanding of the topic from the researcher because researcher’s ability to control the interview situation is great. Misunderstandings are common problems in such semi-structured-interviews. Also in our research in some cases it was obvious that the respondent had difficulties in understanding the question. Keeping the interview in the topic was a challenge; there were interviews in which the respondent answered outside the question, however sometimes those answers provided valuable information to the overall case.

The interviews were recorded and transcribed briefly after the interviews. The data was kept in a safe place. The respondents were made aware of the purpose of the interviews. After the interviews were transcribed, they were sent to respondents so that they could comment on the content and correct possibly misunderstandings. The respondents represented different backgrounds within the company in order to provide rich data. We requested the interviewed, but company selected them. In order to become a respondent there was a prerequisite that the respondents had to be familiar with the open source practices.

The analysis is based on finding themes from the data and reflecting the data to the study framework and to the relevant theoretical propositions. Moreover the analysis tries diligently to build explanations about causal relationships. The analysis method is therefore two-folded; on one hand it tries to match patterns by reflecting on existing theoretical propositions and on the other hand tries to be descriptive and build explanations when the theoretical propositions are absent. The pattern matching is done to theoretical propositions that were identified in the framework. Some issues in open source context have a quite strong theoretical background. Therefore it is justified to reflect on those propositions. However some of the issues in the examination have vague scientific background and therefore a more interpretive analysis method was used.

## 5. Case: Philips Medical Systems DICOM Validation tool

Research effort is a part of a ITEA's (Information Technology for European Advancement) COSI (Co-development with inner and Open Source in software intensive products) project. Consequently, the empirical part of the study has been conducted under this project; the interviews were done in COSI project partner company Philips Medical Systems (PMS). The case study is Philips Medical Systems DICOM validation toolkit (DVTK). Some earlier work has been conducted and published on the same case as a part of the earlier research project CALIBRE (Ågerfalk, 2008).

In order to understand the corporate context it is necessary to introduce the Philips Medical Systems and the whole Royal Philips Electronics in brief. The interviews for the case study were made at Philips Medical System (PMS). The Interviews were done at Philips Medical Systems premises at Best in the Netherlands in November 2006. Altogether three interviews were conducted and analyzed. The individual interviews lasted from about 40 minutes to one hour.

Royal Philips Electronics N.V. is a multinational conglomerate operating on the electronics field. The company was established and is headquartered in the Netherlands. The company is divided into four business areas which are domestic appliances and personal care, lighting, medical systems and consumer electronics. In 2006 the group turnover was approximately € 27 billion and the group employed over 125000 people in more than 60 countries. Philips Medical Systems is one field of business of the Philips group. It provides healthcare services to professional and patients. Philips Medical Systems (PMS) employs approximately 31,000 people worldwide and operates in three main business areas: Diagnostic Imaging Systems; Customer Services and Clinical Solutions. In 2006 the sales were about €6.7 billions. PMS maintains sales and service organizations in more than 63 countries and runs manufacturing operations in the Netherlands, Germany, Finland, Israel and the USA.

The abbreviation DICOM stands for Digital Imaging and Communications in Medicine. DICOM is a standard in medical domain for handling, storing, printing and transmitting information. The standard enables the integration of scanners, servers, workstations and network hardware from different manufacturers into communication and archiving systems. The DICOM standard has been widely adopted by hospitals around the world.

DICOM Validation Tool Kit (DVTK) is software that assists in testing product's conformity with the DICOM standard. In other words it tests if a scanner for

example fulfils the DICOM protocol's requirements and can therefore be used along with other products. In general the toolkit will improve the interoperability of medical equipment of different vendors. DVTK provides a graphical user interface (GUI) to assist the user performing the conformity tests. Initially, starting in 2000, DVTK was developed jointly by Philips Medical Systems and Partner, a multinational company working in imaging field. In the beginning these two companies managed the development process. Today, DVTK has and is still moving away from Philips-Partner cooperation. The closed source way of working has changed to more open source way of working. At present, the tool kit is developed and distributed in SourceForge.net under GNU Lesser General Public License (LGPL).

DICOM validation toolkit was initially developed as proprietary software package. The primary business rationale in the beginning for Philips Medical Systems was to develop a tool that could be used in the company in DICOM validation tool. Such a program did not exist before therefore Philips together with a Partner wanted to create a de-facto standard that could also be used by other companies and it would therefore reduce the problems related to integrating products (interoperability) from different vendors. The software was put on "open source" in 2005. By putting the software "open source" was believed to increase the trustworthiness and the general adoption of the software leading to reduced number of interoperability problems in the field. This would in turn reduce the field support costs for the vendors.

After putting the application outbound the project has been able increase the number of users and developers. Despite of the growing participation, the community is still heavily influenced by the original initiators especially Philips Medical Systems and despite of the increased interest, the community has not still grown very large. Much of the contribution still comes from the original corporate partners. The three corporate members (PMS, Partner and DVT) which comprise an active steering committee around the project. Two or three times a year a DVTK steering committee meeting is organized and there the managers discuss about the long-term goals for the project. From the Philips Medical Systems point of view it is crucial to sustain some level of control, maintaining the project road map for example. The steering committee accepts contributions from developers and they can influence the development road map. It was believed that too much coordination however harms the independent status of the project and may decrease the participation in the project. Currently, the community has been mainly coordinated and controlled by one or two persons working in Philips.

## 6. Interviews

The rationale of the company was to create a de-facto standard for DICOM validation. De-facto standard would decrease the interoperability problems between products of different vendors. This would benefit also other medical equipment manufacturers, which might help to gain external development resources to share the development burden. Company was motivated by open source development because it enables sharing of resources and costs of the development with the OSS community.

Despite of going open the DVTK software's development has controlled by a steering in which PMS is member. Figure 2 presents the general findings of the case study from interviews and from the secondary data sources.

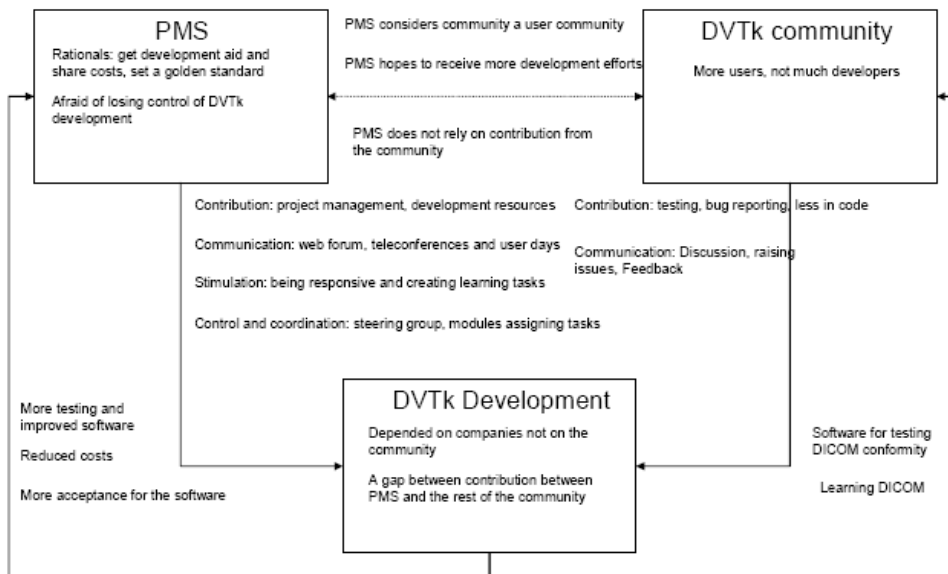


Figure 2. Case study findings

After going “open” the company has remained active in the development. They have contributed in many ways; they have assigned developers to create code and people to work on control and coordination activities related to the development in terms of project management. Moreover they have thought of assigning development tasks to the community and they have also supported modular software architecture as mean of coordination. Furthermore the company has communicating with the community via the established web site DVTK.org website. Moreover the company is organizing a user to day to communicate with the existing and potential community participants face-to-face. The company

addressed stimulation of the community involvement by been being responsive towards the community and their needs and ideas. Moreover they have approached the stimulation by providing learning tasks etc.

There appears to be a gap between company's expectations and the actual contribution from the community. Until now it appears that the company has contributed disproportional to the development compared to what it has received in return from the community. The company clearly wants to receive more development resources and capabilities from the community. So far the community has been able to provide testing, bug-reporting and feedback. Only occasionally the community has been able to produce actual code.

## 7. Results

Philips Medical System had a clear economical motivation when it entered the OSS field; it needed to sustain the development of software that would test its medical equipments' conformity to DICOM standard. Currently their business model is based on selling products that have been tested for conformance to DICOM standard. Therefore in a way one could say that their economic motivation is to obtain indirect revenues by selling related products and therefore their revenue model could be classified as "widget-frosting". Considering other economic rationales for engagement with the community, one is that by creating the software on their own and developing it open source they could avoid paying for costly licenses. Therefore being independent of license policies of the large software companies has been one of the economic motivators. In addition the open source community has provided the company with source of new ideas and innovations. Thus, one can also identify the motivation of trying to afford the innovation by exploiting the research and development activity of the community.

PMS has trained people for example students to be competent in DICOM standard and in developing the DVTK, and they might be looking for employing these people at some point. Thus, the motivation of trying to hire good IT-specialist can be identified. From the perspective of social rationales for PMS it is important to recognize that PMS appears conform to the values of open source communities. There are visible signs of this conformity. First, the company has initiated the whole community by licensing the DVTK software under open source license. Moreover, it has actively contributed to the development and shared knowledge with the DVTK community. Despite of conforming to the OSS phenomenon, it is likely that in the long run the company probably wants receive more contribution from the community. Technological motivations can also be identified. The interview responses highlighted that the company is trying to attract development efforts from the community. Moreover the respondents

stressed the possibility of cutting down the development costs and improving software. Moreover the respondents emphasized Linux-Law, by stating that the increased number of users would lead to more bug reporting and eventually increased quality of the DVTK software.

Despite of the drive for more efficient development it appeared evident that the contribution that they have got so far has not been actual code development. The contribution has primarily been testing, bug reporting and feedback. Moreover from the technical perspective it seems that by participating in DVTK development PMS is looking for creating a de-facto standard in DICOM validation. This can be seen an effort to promote standardization. Moreover it appears that PMS aims at increasing the number of products that are interoperable with other vendor's products.

## 7.1 Communication

DVTK development people work from dispersed locations and face-to-face meetings are rare. Electronic communication devices carry a lot of weight in communication between PMS and DVTK community. It appears that DVTK website is the main communication channel between PMS and rest of the community. The website enables bi-directional communication; the company can inform the community on actual matters and the members of the community can use the web forum to submit problems reports and raise issues. PMS appears to be using the DVTK website quite frequently. It for instance releases information about steering group meetings on fixed basis. In addition to web forum, also face-to-face meetings and teleconferences have been organized to support communication and coordination. People present in these meetings and teleconferences have usually been employees of the corporate partners. However these meetings and teleconferences have been broadcasted over the Internet and they have been free for everybody to monitor. Moreover PMS has participated in fairs and exhibitions in order to communicate and interact with the members in the DVTK community

## 7.2 Stimulation

Until now PMS has primarily tried to stimulate community involvement by indicating to the community that they appreciate community contributions. The concrete measures have included being responsive and making bug-fixes quickly.

In general they have tried to make the DVTK software as good as possible in order to attract users. Moreover if the company has people working for them who have good reputation are they more likely able to attract stimulation. PMS has been present at the fairs and exhibitions where they tried to raise awareness about the DVTK software. Creating incentives for developers is one thing that the company has addressed. They have offered learning tasks for people interested in DICOM standard and who want to learn more.

Despite of the mentioned stimulation efforts the community still appears to be lacking participation especially in development and more particularly in programming. The OSS literature provides a vast number of studies aiming at understanding the motivation of individual programmers. The question is; could the company use some of the ways presented in the literature to stimulate people to get involved and be more active in development?

### 7.3 Control

The company does not support strict membership management regarding users and developers. This means that everybody is free to join the development. However it seems that the company still favors to uphold the membership management in decision making; in the steering. Important decisions about the development directions are still done within the steering which at presents only consists of corporate members. The study did not reveal how one can become a member in the steering group. The DVTK community has well established institution, the core group or the steering group, to control the development. The group makes most significant development decisions. Within the group the decisions are made in a democratic spirit, however it is not clear to what extent the rest of the community can influence the decision making.

Monitoring and sanctions has been considered one way to control the OSS development. It appears that PMS has a good position to monitor community members' behaviors and contributions. By acting in the steering group they have a word to say about the appreciated level of contributions and moreover they can ensure the control contributions compliance with the norms. Efficient peer review and control by reputation are basic characteristics of OSS development. Within the interactions of PMS and DVTK community it appeared that the peer review did not play a significant role. It appears that the consortium structure does support efficient peer-review. The peer review does not have clear role when the consortium is charge of development.

All in all, despite of the rather free participation to development it seems apparent that the company and the steering group as a whole have control over the development process. They decide on new features and the schedules for new releases. Moreover they have their word to say about the quality of the software. The project leader at PMS is responsible for different versions of the DVTK software and finally decides on what is the stable version of the software which will put on the website for everybody to download. One of the interesting findings that are related to the control issues are that the respondents PMS stated as significant downside in the OSS development that they might lose the control over the development.

## 7.4 Coordination

Committee standardization appears to be one of the coordination mechanisms in the case. PMS together with the other companies in the steering group decide on the technical specifications that support the compatibility of software. Moreover it appears that PMS uses as one of the ways to increase coordination to support modular software architecture. One of the respondents expressed that PMS could designate parts of the software to certain members in the community.

Bandwagon coordination does not appear to be one of the coordination mechanisms in the case. The reason for this is that the community has not attracted developer pool which would be large enough, neither has the level of activity been high enough. One could argue that because of lack bandwagon coordination the other ways of coordination are needed.

Regulatory coordination does not have great significance in the coordination. Maybe the most important factor is the license under which the DVTK software is licensed, the Lesser General Public License. The LGPL license requires that changes to the software libraries have to be “open” while the applications using these libraries can be licensed under proprietary license.

Operational coordination such as manuals and revision tracking systems have role in the coordination. It came evident that the company supports SourceForge revision tracking system. It appeared that the DVTK web site has central role in the coordination. Via the web site the company can inform the community about latest changes and also what; are the current issues in the development. Also it could be interpret that by communicating the latest discussions and development trends the company signals the rest of the community where they are heading now. By communicating latest status the members are kept posted on what has been done and what not.

## 7.5 Relationship between community and company

The fundament for the relationship is that the company clearly conforms to the open source norms and values. It has demonstrated this by licensing the software in LGPL license and actively contributing resources and capabilities to the development. It has acted as an open source provider. Moreover it has tried to attract users and developers to join the development. And furthermore it has tried to provide coordination in the development process. In addition it appears to have the acceptance of the DVTK community use the software as part of their business. All in all, the community conforms to OSS values and has acted accordingly and therefore the company can be identified as a community oriented company.

DVTK community has characteristics that resemble a service-oriented community. The community aims at providing a stable and robust service to all stakeholders i.e. the software to test DICOM conformity. The community is not so much focused on created something new and pushing the boundaries of the development further as in exploration-oriented community.

Furthermore the community's control structure resembles is to some extent hierarchical because of the steering group driven decision making and is thus close to town council. One can say that the relationship that Philips Medical Systems has with the DVTK community is close to symbiotic relationship. The company seems to have clear effort to make a fruitful and long-term relationship with the community. However, despite of the seemingly symbiotic relationship there are several issues which hinder the relationship.

Firstly, even though it appears that both parties (PMS and DVTK community) are both contributing to the development there is clear gap between what the company expects to receive from the OSS development and what it actually receives. The gap is in the contributed development resources. At presents a majority of the actual development (e.g. lines of code) come from Philips Medical Systems or from Partner or DVTK while the contribution coming from rest of DVTK community is vague. There seems to be a significant challenge to unlock and foster the resources and capabilities that reside the DVTK community.

Secondly, it appears that the lack of development contribution from the community side could be related to control and ownership of the DVTK software development. So far PMS and the steering group have kept tight control of the development. This has maybe lead to a situation in which the community has not gained a sense ownership and felt responsible of the development. Figure 3 summarizes the findings and illustrates the explanations offered by the framework.

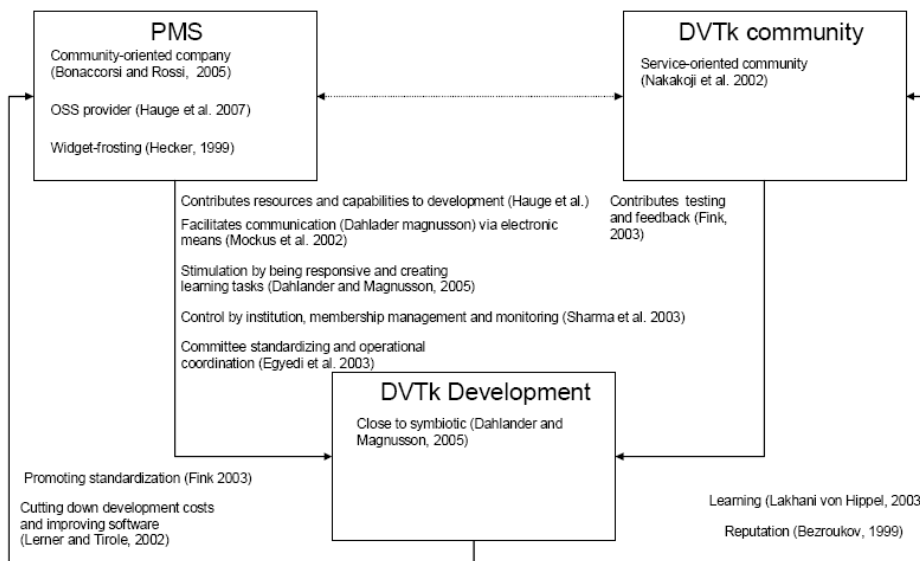


Figure 3. Case study mapped into the proposed framework

## 8. Discussion and conclusions

The aim of this effort has been to *1. What is the nature of the relationship between company and a community in OSS development?* Furthermore, we elaborated this further by asking: *2. Why company engages with a community in OSS development?* and, *3. How company interacts with a community in OSS development?*

The study supports that engaging in open source development can benefit companies' in acquiring resources and capabilities (Rajala et al. 2006) used in company's business operations. Consequently the study supports that by engaging in OSS development companies can reach development capabilities and afford innovation. Results of the case do not support that companies would actively engage in OSSD in order to find to talented labor force (Fink, 2003) even though PMS had tried that in their operations. This supports that companies' rationales to engage with OSS communities are not completely economic. Companies may also have "softer" rationales behind their actions. Findings support that companies enter the OSS field because they genuinely conform to the OSS norms and values. Moreover, the study supports that companies' actively share code and reciprocate with the OSS communities in order to sustain long-term cooperation. The study did not find evidence that companies engage with OSS companies in

order to fight against proprietary software vendors as Feller and Fitzgerald (2002) suggest.

Findings support that companies want to benefit from the feedback and contributions from open source developer communities in order to cut down development costs and improve the software (Hecker, 1999). Moreover it supports that companies seek to benefit from the user community in testing and improving software (Fink, 2003). Company rationales for engaging in OSS development can be related to promoting standardization (Fink, 2003). Moreover it supports that companies perceive that OSS development can enhance security of software products what Fink (2003) proposes. The study does not give evidence that cutting down hardware costs would be a reason to enter OSS field and engage in development what Feller and Fitzgerald (2002) claim.

This case study subscribes the claim that in open source software development a major part of the communication is conducted through electronic channels (website, user groups and e-mail) and face-to-face communication is rare (Mockus et al. 2002). Moreover it supports the claim that companies can actively contribute by facilitating communication (Fink, 2003) e.g. by building communication channels for development. In the case study Philips medical systems had a very role in facilitating communication in the community.

This study found that the case company stimulated the community by being responsive and producing good quality code. This is in line with Dahlander and Magnusson (2005) who have claimed that maintaining reputation is an important mean of attracting new participants. Moreover this study supports that companies actively market promote open source activities in order to raise awareness and attract new participants (Fink, 2003). Furthermore this study supports that companies stimulate developers by creating incentives (Fink, 2003) by providing them intellectual challenges and by learning tasks (Raymond, 1999).

This study supports that OSS communities use institutions (Sharma et al. 2003) to control the OSS development. In the case study the OSS development was heavily controlled by the steering committee which was one kind of institution in the development. This study supports the claim by Mockus et al. (2002) that companies can not directly control the open source development they can only facilitate and stimulate the OSS community. This is also in accordance with Dahlander and Magnusson (2005) who have written about companies' subtle means of control in OSS development. For example this study supports that companies can try to "sell" development tasks (Dahlander and Magnusson, 2005) to the community in order to get suitable development contribution.

All in all this study supports Dahlander and Magnusson (2005) that a solid and sustainable relationship between a company and OSS community requires reciprocal interactions. In addition the study is in line with Dahlander and Magnusson (2005) who claim that companies which respect OSS values and reciprocate with OSS communities are in favorable position to influence OSS development and gain business benefits. This study however does not confirm the claim that the nature of relationship would have strategic implications for company's business that Dahlander and Magnusson (2005) claim. Moreover this study proposes that company's a close relationship with the OSS community does not necessarily unlock the communal resources that reside in OSS communities.

We identified four types of challenges in a company – OSS community relationship:

1. Results support that a basic necessity for a fruitful relationship is that companies truly respect the norms and values of OSS communities (Dahlander and Magnusson, 2005). Moreover this study supports that companies should demonstrate their commitment to OSS communities in order to get approval for business use of the software. When engaged in open source development companies should obey OSS norms and rules and not try to impose their way of working.
2. Lack of communication can cause problems in company – community relationship. This study proposes that companies should actively communicate with the OSS communities in order to better articulate their own expectations and needs and moreover to understand communities' motivations. In general the notion of communication has not been addressed in the OSS literature extensively. In order to communicate better companies can contribute by building facilities for communication (Fink, 2003) such as web forums or by organizing events. Furthermore this study suggests that by having a central position in the communications of the OSS development increases company's possibilities to influence the development.
3. We propose that too much company control in OSS the development may harm the relationship. This in line what Dahlander and Magnusson (2005) have suggested; they highlighted the managerial challenge of resolving ambiguity about control and ownership in OSS development. The case presented an example in which the company had a strong control over the development activities through a steering group. It appeared that too much company control over the development can obstruct the community from adopting the responsibility of the development.

4. We identify a major challenge of attracting participation in open source projects. It conforms to Dahlander and Magnusson (2005) who have claimed that attracting developers and users can be a major managerial challenge in a company-community relationship. This challenge is especially significant in open source projects which require special domain knowledge such as in DICOM standard. The fundamental question is; what can companies do to encourage people to take part in the development? Fink (2003) has emphasized creating incentives for developers. This study suggests that in order to attract more participation companies should better understand developer and participant motivations.

## 9. Summary

The purpose of this research effort was to show a way to analyze community-company relationship. We have elaborated a framework to be used in further studies through extensive literature review and testing the framework in one company case. We acknowledge that one such test for a framework does not make it valid, but only serves a call for further testing. This is the path we intend to follow in the ongoing research. Our sincere hope is that this understanding might also ease companies still pondering the OSS leap by showing the range of issues connected to the management of OSS communities.

## References

- Ågerfalk, P., Fitzgerald, B. (2008). Outsourcing to an Unknown Workforce: Exploring Open sourcing as a Global Sourcing Strategy. *MIS Quarterly*, 32: 2, pp. 385-409.
- Benussi, L. (2005). Analyzing the technological History of the Open Source Phenomenon. Stories from the Software Evolution . Working Paper. Available at: <http://opensource.mit.edu/papers/benussi.pdf>, 2005. Accessed: June 3rd 2007.
- Bonaccorsi, A., Rossi, C. (2006). Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement: From Community to Business. *Knowledge, technology and policy*. Vol. 18:4 p.40-64.
- Corbetta, P. (2003). *Social research: theory, methods and techniques*. Sage publications.
- Crowston, K., Annabi, H., Howison, J., Masango, C.(2005). Effective work practices for FLOSS development: A model and propositions, in 'Proc. of Hawaii International Conference On System Sciences (HICSS).
- Cusumano, M.(1992). Shifting Economies - from Craft Production to Flexible Systems and Software Factories. *Research Policy*, 1992 Vol. 21:5. p. 453-480.
- Dahlander, L., Magnusson, M. (2005). Relationship between open source software companies and communities: Observations from Nordic firms, *Research Policy*, 2005. Vol. 34. p. 481-493.
- Dreiling A., Klaus H., Rosemann M., Wyssusek B. (2005). Open Source Enterprise Systems: Towards a Viable Alternative. Proceedings of the 38th Annual Hawaii International Conference on System Sciences.

- Egyedi, T., van Wendel de Joode, R. (2004). Standardization and Other Coordination Mechanisms in Open Source Software. *International Journal of IT Standards & Standardization Research*. Vol. 2:2. p. 1-17.
- Feller, J., Fitzgerald, B. (2000). A Framework Analysis of the Open Source Software Development Paradigm. *Proceedings of 21st International Conference on Information Systems (ICIS)*, ACM Press, 2000. p. 58-69.
- Feller, J., Fitzgerald, B. (2002). *Understanding Open Source Software Development*, Pearson Education Limited.
- Fink, M. (2003). *Business and Economics of Linux and Open Source*. Prentice Hall, New Jersey.
- Fitzgerald, B. (2006). The Transformation of Open Source Software. *MIS Quarterly* 30:2:587-598.
- Hauge, O. Sørensen. Røsdal. (2007). Surveying Industrial Roles in Open Source Software Development. *Proceedings of The Third International Conference of Open Source Systems*, Limerick, Ireland.
- Hecker, F. (1999). Setting Up Shop: The Business of Open-Source Software. *IEEE Software*. Vol. 16:1. p. 45-51.
- Lakhani, K., Wolfe, R. (2005). Why Hackers do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects. In (eds) Feller, J., Fitzgerald, B., Hissam, S., and Lakhani, K. (2005). *Perspectives on Free and Open Source Software*. MIT Press, Cambridge, Massachusetts.
- Nakakoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K., Ye, Y. (2002). Evolution Patterns of Open-Source Software Systems and Communities. *Proceedings of the International Workshop on Principles of Software Evolution*, Orlando, Florida.
- Mockus A., Fielding R., Herbsleb J. (2002). Two case studies of open source software development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology*. Vol 11:3.
- Rajala, R. & Westerlund, M. & Nissilä, J. (2006). Software for free? Revenue models in the opensource software business. *Proceedings of the 5th Global Conference on Business & Economics*, Cambridge University, Cambridge, UK, June 6-8.
- Raymond, Eric S. (1999). *The Cathedral and the Bazaar, Musings on Linux and Open Source By an Accidental Revolutionary*. O'Reilly & Associates.
- Sharma, S., Sugumaran, V., Rajagopalan, B. (2002). A framework for creating hybrid-open source software communities. *Information Systems Journal*, 2002. Vol.12:1. p. 7-25.
- Stake R. (1995). *The Art of Case Study Research*. Sage Publications.
- West, J. and O'Mahony, S. (2005). Contrasting Community Building in Sponsored and Community Founded Open Source Projects. *Proceedings of the 38th Annual Hawai'i International Conference on System Sciences*, Waikoloa, Hawaii.
- West, J. (2003). How open is open enough? Melding proprietary and open source platform strategies. *Research Policy*, Vol. 32:7 p. 1259-1285.
- Yin, R. K. (1994). *Case Study Research: Design and Methods*, 2nd edition, Sage Publications, California.