

# How Can Companies Work with Open Source Communities for Open Innovation?

Mikko Ahonen<sup>1</sup>, Maria Antikainen<sup>2</sup>, Marko Mäkipää<sup>1</sup>

University of Tampere<sup>1</sup>, VTT <sup>2</sup>Affiliation(s)

*mikko.ahonen@uta.fi, maria.antikainen@vtt.fi, marko.makipaa@uta.fi*

**Abstract.** Innovation activity in companies is more and more focused on integrating internal research with external development groups. However, researchers and practitioners have had difficulties in explaining the value creation methods of the Open Innovation paradigm using analogy of open source initiatives. Similarly, companies often approach open source with insufficient level of knowledge, which may cause problems. In this paper we examine how a company can integrate itself systematically to open source communities and utilize viable and sustainable business model for commercial exploitation. Case examples are provided from open source software context to illustrate the mechanisms of utilizing external research and development resources and how open source is integrated into a business. As a result, we present models and lessons learned from building open source communities to facilitate innovation activity.

**Keywords:** open source, open innovation, community, toolkit, trust

## Introduction

Successful technological innovation in firms is strongly dependent upon managing sufficient knowledge resources to support continuous discovery, knowledge creation and technical development. Often, a strategy and a process are needed to collaborate with external groups and networks. Open Innovation paradigm (Chesbrough, 2003) stresses the importance of using a broad range of sources for a firm's innovation and invention activities, including customers, rivals, academics, and firms in unrelated industries while simultaneously using different methods to protect firm's Intellectual Property (IP). Open Innovation needs also open business models and therefore Open Innovation researchers (e.g. West and Gallagher (2006), Chesbrough (2006) and West (2007) have had interest in open source.

However, within the Open Innovation literature there are not many descriptions, how the co-operation with innovation communities takes place, especially when users are innovating by themselves. In the literature there are some descriptions like the one from von Hippel (2005) who explains the nature of these communities:

Innovation communities can have users and/or manufacturers as members and contributors. They can flourish when at least some innovate and voluntarily reveal their innovations, and when others find the information revealed to be of interest. ... these conditions do commonly exist with respect to user-developed innovations: users innovate in many fields, users often freely reveal, and the information revealed is often used by manufacturers to create commercial products—a clear indication many users, too, find this information of interest. (von Hippel, 93, 2005)

Open source software (OSS) communities are good examples of these innovation communities, where free revealing is an important element. However, the view of von Hippel (2005) stretches to wider areas than software, like hardware and service development.

In this paper we inspect how a company can integrate itself systematically to open source communities. First we will discuss about the open source 2.0 phenomenon and then we will make distinctions between Open Innovation and open source. Information systems research topics are viewed through innovation communities and their use of toolkits. The research question here is: *How should a company work with open source communities to support its innovation strategy.* Since we do not have comprehensive data available from companies, we base our findings in the literature and present some models (Grand *et al.* 2004, West and Gallagher 2006) which explain innovation activity within innovation communities.

## Open source 2.0, communities and companies

*Much of OSS research has focused inward on the phenomenon itself, studying the motivations of individual developers to contribute to OSS projects, or investigating the characteristics of specific OSS products and projects, for example. Far less has been done in looking outward at the process of OSS adoption and implementation in organizations. (Fitzgerald, 2006, 588 )*

In order to work with open source communities, companies need to learn what open source (software) is. Therefore, the first step is to use OSS within the company and learn lessons about its adoption and usage. Only thereafter the company is fully capable of integrating itself and its employees to open source communities. The first generation of OSS was about enthusiastic amateurs while the second generation of OSS is about companies working closely and integrating themselves to open source communities. Fitzgerald (2006) uses the OSS 2.0 term meaning that open source software phenomenon has metamorphosed into a more mainstream and commercially viable form.

In order to explore the firm-level implications of OSS, it is important to put OSS innovation in a broader context of models of technological innovation.

Innovation strategy can be divided into two basic choices. First, private innovation assumes that returns to the firm result from private goods and efficient regimes of intellectual property protection taking place within firm boundaries or as part of a firm's web of partnerships. Second, collective innovation assumes that firms collaborate with public institutions such as universities to produce a public good, where public access to the results and transparent communication of the research and development process are inherent (Grand *et al.* 2004).

Another division is provided by von Hippel and von Krogh (2003). Their model is called the private-collective model. In this model economic actors invest their private resources to produce a public good. Open source software development is an exemplar of a compound "private-collective" model of innovation that contains elements of both the private investment and the collective action models and can offer society the "best of both worlds" under many conditions. (von Hippel and von Krogh (2003, 209).

Distinction between private and collective innovation is useful, but Private-Public division to represent them is inadequate. In case of OSS, collective innovation is many times a result of group of individuals without any intervention of private or public organizations. In OSS context community innovation is better metaphor than collective-public innovation. However, in OSS 2.0 this division is not anymore this simple as mixed forms are emerging and developed.

West (2007) describes two different types of communities organized around a particular shared product (open source software): those sponsored by an (usually corporate) organization, and the more traditionally studied autonomous communities.

According to West (2007) one of the characteristics of sponsored communities is that they are different in that they face conflicting goals between those of the sponsor and those of the community. The sponsor would like to control the community, but creates the community in hopes of attracting participation and growth. At a minimum, the sponsor must provide transparency. In most cases, the sponsor retains privileged (monolithic) rights for some portion of the community's decisions. In some cases, the sponsor shares control with the community and in rare cases, bestows upon the community full autonomy.

Furthermore, communities sponsored by government or non-profit actors are more likely to favor public good ahead of the sponsor's pecuniary gain, but face the same tension between maintaining control and attracting community participation and growth (West, 26, 2008). For private companies these challenges are even more difficult to manage, if they seek to transform the results to profitable business. Company can lose its face and trustworthiness very quickly if it acts against the community's expectations. Legal rights can be very different from moral rights when participating in or sponsoring open source communities.

## Open Innovation versus open source

The OSS phenomenon has intrigued a lot of interest and enthusiasm to extend the ideology also to other areas than software. Open Innovation term has been coined by Henry Chesbrough (2003) and originally has described the changes in business models and innovation processes of large companies. In other words, the origins of Open Innovation and OSS are originally far away from each other.

The Open Innovation paradigm is often contrasted to the traditional vertical integrations where internal R & D activities lead to internally developed products that are then distributed by the firm. West & Gallagher (2006) therefore define Open Innovation as:

“systematically encouraging and exploring a wide range of internal and external sources for innovation opportunities, consciously integrating that exploration with firm capabilities and resources, and broadly exploiting those opportunities through multiple channels”

Piller et al. (2004) see certain limitations in Open Innovation. Therefore, they suggest a new approach:

“...Open Innovation in our understanding goes one important step further: Open innovation aims at transferring the ideas and approaches from open source software development to the domain of other product categories and services. The idea is to build and operate platforms where (communities of) customers and users create, develop, and discuss newly customized products and services.” (Piller *et al.*, 2004, 5)

Similarly, von Hippel (2005) discusses about toolkits in the innovation activity of a company.

Firms that understand the distributed innovation process and users' roles in it can change factors affecting lead user innovation and so affect its rate and direction in ways they value. Toolkits for user innovation custom design offer one way of doing this. ...The resulting co-location of sticky information and problem-solving activity makes innovation within the solution space offered by a particular toolkit cheaper for users. It accordingly attracts them to the toolkit and so influences what they develop and how they develop it. (von Hippel, 16, 2005)

These toolkits are in essence information systems and software tools. Toolkits and communities around them are described by Piller *et al.* (2004) and Ahonen *et al.* (2007). *In line with the open source concept, but as an extending feature to the toolkit approach, the presented solution is embedded in an online community. Thus, contributions by innovative users can be stored in a library leading to a continuously growing information pool of available components.* (Piller *et al.*, 2004).

So, the actual toolkits can be free, even developed in open source communities but the communities and web-based services can be commercial.

How does Open Innovation paradigm (Chesbrough 2003, Chesbrough *et al.* 2006) then integrate with open source development methodologies and the activities in communities? According to Chesbrough (2006):

“Open Innovation is sometimes conflated with open source methodologies for software development. There are some concepts that are shared between the two, such as the idea of greater external sources of information to create value. However, Open Innovation explicitly incorporates the business model as the source of both value creation and value capture. This latter role of the business model enables the organization to sustain its position in the industry value chain over time. While open source shares the focus on value creation throughout an industry value chain, its proponents usually deny or downplay the importance of value capture.” (Chesbrough, 2006, 2)

Similarly, West and Gallagher (2006) list various open source projects and see that for example Project GNU and Mozilla projects are not Open Innovation because their business model is missing. From the point of view of OSS 2.0, however, Mozilla can be seen Open Innovation because employees of companies like IBM work in the Mozilla community to provide the Firefox browser that supports IBM's products and may even be more functional than the competitor's (Microsoft's) browser. According to West and Gallagher (2006) IBM's Eclipse (<http://www.eclipse.org>) is both Open Innovation and open source since IBM still has hundreds of developers committed to Eclipse projects although it Eclipse is nowadays governed by an OSS foundation. Thus, the OSS 2.0 provides many mixed and sometimes even very complex commercial opportunities for companies,

similar to Open Innovation. The complexity of OSS 2.0 and Open Innovation business models introduce new management challenges which differ from those of proprietary innovation models. West and Gallagher (2006) emphasise three management factors, namely motivating, incorporating and maximising.

Innovation Model	Management Challenges	Resulting Management Techniques
Proprietary (or internal or “closed”)	<ol style="list-style-type: none"> <li>1. Attracting “best &amp; brightest”</li> <li>2. Moving research results to development</li> </ol>	<ol style="list-style-type: none"> <li>1. Provide excellent compensation, resources, and freedom.</li> <li>2. Provide dedicated development functions to exploit research and link it to market knowledge</li> </ol>
External	<ol style="list-style-type: none"> <li>1. Exploring wide range of sources for innovation.</li> <li>2. Integrate external knowledge with firm resources &amp; capabilities</li> </ol>	<ol style="list-style-type: none"> <li>1. Careful environmental scanning</li> <li>2. Developing absorptive capacity, and/or alliances, networks, and related consortia</li> </ol>
Open	<ol style="list-style-type: none"> <li>1. <b>Motivating</b> the generation &amp; contribution of external knowledge</li> <li>2. <b>Incorporating</b> external sources with firm resources &amp; capabilities</li> <li>3. <b>Maximising</b> the exploitation of diverse IP</li> </ol>	<ol style="list-style-type: none"> <li>1. Provide intrinsic rewards (e.g. recognition) and structure (instrumentality) for contributions.</li> <li>2. As above</li> <li>3. Share or give away IP to maximise returns form entire innovation portfolio.</li> </ol>

Table 1. Models of innovation and resulting managerial issues (West & Gallagher, 2006)

Open Innovation model through motivating, incorporating and maximising means constant communication between a company and a community. Next a closer look is taken at this integrative activity.

## Costs and benefits of integrating to OSS 2.0 community

Grand *et al.* (2004) have suggested a four level model of resource allocation for firms ranging from a modest investment in developing knowledge about OS to a compatible business model. According to this model:

Level	Description
1	OS software is used in order to replace or supplement existing software within the firm, without implications for the user-firm's business model. Any firm in any technological or industrial context can become involved in OS innovation as a user of the software - however, the fact that substantial expertise may be needed to run OS code within existing IT infrastructures is partly responsible for the emergence of business approaches for specialized software/IT firms at the other three levels.
2	OS software is used as a complementary asset, which is combined with other software or hardware to deliver integrated solutions for level 1 OS users. The firm integrating OS software as complementary asset must invest substantial development resources to adapt the code to its particular services and turning it into a stable technological solution. In addition, the software thus developed must be publicly released, according to the licensing agreement of the OS software used.
3	OS software is not only seen as a complementary asset, but as core for the entire software developed by the specialized firm. At this level, companies both complement their own software with OS 'building blocks', but essentially build their own software based on the freely available code. As, in either case, the entire developed code must be freely released back to the OS community, this clearly implies a substantial commitment a firms' own business model to OS.
4	The packaging, bundling and distribution of stable services and solutions for OS software becomes the business model itself, complementing the interests of the many level 1 firms. The emergence of Level 4 firms can be seen as the natural outcome of the success of the OS software community.

Table 2. Four-level management model of increasing private resource allocation (Grand et al., 2004)

Like mentioned in the table above, the fourth phase of Grand *et al.* (2004) means that activities within the OS community becomes a business model itself.

Grand *et al.* (2004) have also analysed costs and benefits of firm involvement in OS innovation. Both the risks and benefits increase towards the higher levels of the model. In their study they emphasize the importance of knowledge in firms to be able to make the decisions how to integrate OSS in their innovation process.

## Business models, open source and Open Innovation

Open source offers one possibility for companies that are continuously seeking new opportunities to organize their business activities and to increase the amount of value they produce. Furthermore, the concept of a business model is considered as a tool for exploring new business ideas and capturing the essential elements of each alternative. Within Open Innovation literature the term open business model is used (Chesbrough, 2003, 2006).

A business model performs two important functions: it creates value, and it captures a portion of that value. It creates value by defining a series of activities from raw materials through to the final consumer that will yield a new product or service with value being added throughout the various activities. The business model captures value by establishing a unique resource, asset, or position within that series of activities, where the firm enjoys a competitive advantage. An open business model uses this new division of innovation labor – both in the creation of value and in the capture of a portion of that value. Open models create value by leveraging many more ideas, due to their inclusion of a variety of external concepts. (Chesbrough, 2006, 2)

What are those elements of business models? An important consideration is the context-specificity of a business model. Is there a need for specific models that are targeted to a particular industry and particular product/service combinations? Regardless of the several industry-related papers devoted to these (such as business models for e-business – referring to Rappa, 2007; Rayport & Jaworski, 2001; Weill & Vitale, 2001), Seppänen *et al.* (2007) propose that all business models share common elements. A generic business model should involve the same elements regardless of the industry where the model is used. Then, a context-specific model – perhaps with a prefix – should be seen as a local application of a general business model. (Seppänen *et al.*, 2007). Since the question of relationships and partner networks are essential in business models, communities and their building will be next inspected.

## Building communities

Fogel (2005) has pointed out difficulties in establishing an Open source project:

One of the most common mistakes is unrealistic expectations about the benefits of open source itself. An open license does not guarantee that hordes of active developers will suddenly volunteer their time to your project, nor does open-sourcing a troubled project automatically cure its ills. In fact, quite the opposite: opening up a project can add whole new sets of complexities, and cost *more* in the short term than simply keeping it in-house.

Opening up means arranging the code to be comprehensible to complete strangers, setting up a development web site and email lists, and often writing documentation for the first time. All this is a lot of work. (Fogel, 2005, 51)

Fogel continues:

The hardest part about launching a free software project is transforming a private vision into a public one. You or your organization may know perfectly well what you want, but expressing that goal comprehensibly to the world is a fair amount of work. It is essential, however, that you take the time to do it. You and the other founders must decide what the project is really about—that is, decide its limitations, what it *won't* do as well as what it will—and write up a mission statement. This part is usually not too hard, though it can sometimes reveal unspoken assumptions and even disagreements about the nature of the project, which is fine: better to resolve those now than later. The next step is to package up the project for public consumption, and this is, basically, pure drudgery. (Fogel, 2005, 52)

If people feel that they are involved in the decision-making process and that their viewpoints are heard and respected, then the community will generally accept whatever decision is made. If people feel that a decision is made without hearing their opinions, then they will object and, in the worst case, go elsewhere, possibly forking the source code and starting a competing project.

The exact decision-making process varies from one open-source project to the next, but in many it is based on the idea of a meritocracy: Those who have demonstrated their competency through their work on the project are the ones who make the decisions. In many cases, the project lead, often the originator of the code, has the final say. Likewise, module owners make decisions that affect their module. This works only as long as the "benevolent dictator" can maintain the respect of the developer community; otherwise, the community will call for a replacement. (Goldman and Gabriel, 2005)

## Conclusions

Previously very company centric innovation process has experienced many changes during the last decades. Globalization and developments in information and communication technologies have enabled companies to distribute R&D processes globally both inside the firm boundaries and outside. Outsourcing and strategic alliances have attained a sustainable position in management strategies. Today, the Open Innovation paradigm is challenging companies' innovation strategies provoked by success of open source software communities.

OSS offers a fascinating model for future innovation and knowledge creation and brand new challenges for companies. Previously companies have concentrated on value creation and have received income from that activity. In network economy, based on core competencies and outsourcing, the absorptive capacity has been raised to be in a key position in firms' innovation strategies. In an emerging business environment building on open innovation and OSS 2.0 like business models companies no longer are the sole creators of value, but instead find themselves from the centre of complex network of value creators; both individual and organizational, both private and public. In this kind of environment only those companies that learn new value harnessing activities and are able to optimize their value creation/harnessing activities for total offering portfolio will survive. Companies need to start building their networks and communities and persuade others to join in and contribute.

## Discussion

Most of our examples and references describe open source software development. However, models from OSS are more and more applied to services and hardware development. Both OSS 2.0 and Open innovation may offer remarkable benefits for companies, yet they include many challenges.

If the company succeeds in integrating itself to OSS 2.0 communities, it does not automatically provide help with its innovation activities. According to West and Gallagher (2006, 29) when open source software is used as part of a complex system, a firm still faces the fundamental issues of coordinating the systemic innovation, assuring overall value creation and capturing the firm's portion of that value. While open source has been celebrated as a new and different approach to software development, its emergence ironically has coincided with the emergence of stronger intellectual property protection for patents and other IP (Chesbrough, 2006, 48) For this reason Chesbrough recommends that companies should consider *both* advancing towards open source *and* simultaneously heading towards stronger IP protection. Companies also face certain risks while embracing open source as part of their business models: the OSS community's contribution may not remain active forever and when OSS becomes mainstream there is always risk for Intellectual Property Rights (IPR) infringement (Fitzgerald, 2006).

Therefore, much more research is still needed to better understand the emerging role that OSS 2.0 and Open Innovation will have in companies' innovation strategies and practices. This study will continue by participating a project from Autumn 2007 till Spring 2009 where companies systematically integrate into OSS communities and use open source software as part of their industrial, hardware products. This opportunity allows us to continue developing the body of knowledge of OSS 2.0 and Open Innovation by collecting empirical data from real company-OSS community collaboration.

## References

- Ahonen, M., Antikainen, M., & Mäkipää, M. (2007). *Supporting Collective Creativity within Open Innovation*. European Academy of Management Conference. Conference proceedings. EURAM: Paris.
- Antikainen, M. and Aaltonen, T. (2007). In Helander, N. & Antikainen, M. (eds.), *Essays on OSS practices and sustainability*. e-Business Research Center Research Reports 36. Tampere University of Technology & University of Technology
- Antikainen, M. and Aaltonen, T. (2007, forthcoming). *The role of trust in OSS communities – Case Linux Kernel community*. Accepted for publication in The Third International Conference of Open Source Systems, Limerick, June 13-16, 2007.
- Chesbrough, H. (2003). *Open Innovation: The New Imperative for Creating and Profiting from Technology*. Boston: Harvard Business School Press.
- Chesbrough, H.W. (2006). *Open Business Models How to Thrive in the New Innovation Landscape*. Boston (Mass.): Harvard Business School Press.
- Chesbrough, H., Vanhaverbeke, W., & West, J. (2006). *Open Innovation: Researching a New Paradigm*. Oxford: Oxford University Press.
- Fogel, K. (2005). *Producing Open Source Software: How to Run a Successful Free Software Project*. Sebastopol, CA: O'Reilly.
- Gallivan, M. (2001). Striking a balance between trust and control in virtual organizations. *Info Systems*, 11, 277-304.
- Goldman, R., & Gabriel, R.P. (2005). *Innovation Happens Elsewhere: Open Source as Business Strategy*. Morgan Kaufmann.
- Grand, S., von Krogh, G., Leonard, D. & Walter Swap. (2004) Resource Allocation Beyond Firm Boundaries: A Multi-Level Model for Open Source Innovation. *Long Range Planning* 37, 591-610.
- Hertel, G., Niedner, S. and Herrman, S. (2003). Motivation of software developers in open source projects: An Internet-based survey of contributors to the Linux Kernel. In J. Feller, B. Fitzgerald, S. Hissam and K. Lakhani (eds.), Taking stock of the bazaar. *Proceedings of the 3rd Workshop on Open Source Software*.
- von Hippel, E.V. (2005). *Democratizing Innovation*. Cambridge, MA: MIT Press.
- von Hippel, E.V., & von Krogh, G. (2003). Open Source Software and the "Private-Collective" Innovation Model: Issues for Organization Science. *Organization Science*, 14(2), 209-223.
- Luhmann, N. (1979). *Trust and power*. New York: J. Wiley.

- O'Leary, M., Orlikowski, W. and Yates, J. (2001). Distributed work over the centuries: trust and control in the Hudson's Bay Company (1670-1826). In P. Hinds and S. Kiesler (eds.), *Distributed work*. Cambridge: MIT Press.
- Osterloh, M., & Rota, S. (2007). Open source software development--Just another case of collective invention? *Research Policy*, 36(2), 157-171.
- Rappa, P. (2007) Business Models on the Web. <http://digitalenterprise.org/models/models.html>. Last accessed: 10th June, 2007.
- Rayport, J. and Jaworski, B. (2001). *Electronic Commerce*. New York: McGraw-Hill.
- Rotter, J.B. (1971). Generalised expectancy for interpersonal trust. *American Psychologist*, Vol. 26 pp.443-52.
- Seppänen, M., Helander, N. and Mäkinen, S. (2007) Business models in open source software value creation. *Open source software Handbook*.
- Timmers, P. (2003). Lessons from E-Business Models. *Zeitschrift für Betriebswirtschaft. ZfB – Die Zukunft des Electronic Business* (1): 121-140.
- Weill, P. and Vitale, M. (2001). *Place to Space: migrating to ebusiness models*. Boston: Harvard Business School Press.
- West, J. (2007). Sponsored open source communities. *Managing Open Innovation through Online Communities*.
- West, J. & Gallagher, S. (2006). Patterns of Open Innovation in Open Source Software . In Chesbrough, H., Vanhaverbeke, W., & West, J. (Eds.) *Open Innovation: Researching a New Paradigm*. 82-106. Oxford University Press: Oxford, UK.