

# Why do commercial companies contribute to open source software?

## Abstract

*Many researchers have pointed out that the open source movement is an interesting phenomenon that is difficult to explain with conventional economic theories. However, while there is no shortage on research on individuals' motivation for contributing to open source, few have investigated the commercial companies' motivations for doing the same. A case study was conducted at three different companies from the IT service industry, to investigate three possible drivers: sale of complimentary services, innovation and opensourcing (outsourcing). We offer three conclusions.*

*First, we identified three main drivers for contributing to open source, which are a) selling complimentary services, b) building greater innovative capability and c) cost reduction through opensourcing to an external community. Second, while previous research has documented that the most important driver is selling complimentary services, we found that this picture is too simple. Our evidence points to a broader set of motivations, in the sense that all our cases exhibit combinations of the three drivers. Finally, our findings suggest that there might be a shift in how commercial companies view open source software. The companies interviewed have all expressed a moral obligation to contribute to open source.*

## 1 Introduction

At the beginning, the open source movement was viewed as a movement of altruist hackers working on their free time to develop code and programs for the common good (Hars and Ou, 2002; Lerner and Triole, 2002, Raymond, 2001). However, the perception of the open source movement as a naïve movement imitating proprietary products is rapidly changing as commercial companies are now engaging in the development of free and open source software and innovative products are coming from the open source movement (Bonaccorsi and Rossi, 2006).

The fact that companies are contributing their intellectual property for free to the open source community seems to contradict common economic theories, which suggest that companies should seek patents on intellectual property in order to grow (Gould and Gruben, 1996). Still, the software industry is full of examples where companies are giving away software and software components for free, by contributing to open source (Capek et al., 2005). While the motivation for individuals contributing to open source has been documented in numerous research articles (Hars and Ou, 2002; Lakhani and Wolf, 2003), the same amount of research has not been aimed at identifying the motivating factors for commercial companies contributing to open source. This is despite the fact that research into the open source community has shown that approximately 40% of the developers participating in open source are being paid by their employers to participate in Free and Open Source Software (FOSS)

projects (Lakhani and Wolf, 2003; Lerner et al., 2006) and that many open source products are now developed and released by commercial companies (Samuelson, 2006).

One notable exception to numerous articles and surveys that are focused on the individual's motivation is the research by Bonaccorsi and Rossi (2006). They compare the motivations of individual programmers and commercial companies and found that there is a significant difference in the motivation of the two. While individuals are motivated by a mixture of intrinsic and extrinsic values, companies are motivated by the technological and economic aspect of open source contribution.

Indeed, an area of research that has received a good deal of focus lately is the increasing professionalisation and monetising of open source, with companies' developing business models based on open source by selling complimentary services (Dahlander, 2005; Fitzgerald, 2006; Watson et al, 2008). An increasing number of companies now have open source as a direct contributor to the revenue by selling complimentary services such as support, consulting, certifications etc.

At the same time, a new phenomenon of user generated content has been described by scholars, researchers and business people. Companies are increasingly exploiting their users' willingness to contribute. Wikipedia, Youtube, Amazon and American Idol are all examples of products that to a varying degree are dependent on user involvement (Cook, 2008; Tapscott and Williams, 2008). Open source software is no different; it relies on the continuous improvement performed by the community to be successful. This has lead to some researchers comparing the open source phenomenon to outsourcing, arguing that by releasing software as open source companies are outsourcing to an unknown workforce (Ågerfalk and Fitzgerald, 2008).

Others again have pointed to the fact that open source is helping companies innovate, by tapping in on the innovative powers present in the community of users. This argument lends support from the theory of open innovation, a theory that claims a company must not originate all research in order to be able to profit from it (Chesbrough, 2003).

Thus, there is no shortage of theories that might explain different aspects of commercial companies' decision to contribute to open source. Still, not much research has been conducted to reveal the motivations leading up to such decisions and further investigation of the commercial companies' motivation for contributing to open source from a strategic standpoint is warranted. The aim of this paper is therefore to explain the commercial companies' strategic drivers behind the decision to contribute to free and open source software, answering the research question "why are commercial IT service companies contributing to free and open source software?"

The structure of this paper is as follows: the next section presents a review of the literature on the subject of commercial contributions to open source software which serves as a basis for an analytical framework. Our method is presented in Section 3. The data collected is then analysed using a pattern matching technique, where data is matched with factors from the framework as well as with rival theories, all detailed in Section 4. Discussion of our results

takes place in Section 5. Finally, conclusions are presented, where the feasibility of the proposed framework is determined and the focus question is answered.

## **2 Research review**

This section will give a short presentation on the history of open source software development. Then three different motivational factors that may influence commercial companies' decision to contribute to open source will be introduced. Finally, we present the framework of motivational factors used to analyse the data in this paper.

Throughout the paper the terms Free and Open Source Software (FOSS), open source and open source software are used interchangeably. All three of these terms are used to describe software released under any of the many licenses approved by the open source initiative (Open Source Initiative, 2009).

### **2.1 The open source phenomenon**

The communal behaviour of the free and open source software movement can be traced back to the 1960s and 70s, when scientists and researchers shared the code they wrote, making it possible to build on each other's innovations. The computer programmers of the 1960s and 1970s were part of the research culture of academic and corporate laboratories, where sharing was considered the norm. However, the sense of communal culture was dealt a blow in the 1980s when MIT licensed away code created by its scientists to a commercial company (von Hippel, 2005).

One of developers working at MIT at the time, Richard Stallman, was very much opposed to the trend of commercialisation of software. To promote computer users' right to study, copy, modify and redistribute software, Stallman founded the Free Software Foundation in 1985 (Free Software Foundation, 2009; Lerner and Triole, 2002). Stallman's idea was to use copyright law, normally used to protect the proprietary nature of ideas and products, to develop open source licenses that would guarantee that open source software remained free and open for everyone (von Hippel, 2005). It is worth mentioning that when Stallman refers to free, he does not mean free as in gratis. Stallman's way of explaining it has become quite famous and much quoted; "Free software is a matter of liberty, not price. To understand the concept, you should think of free as in free speech, not as in free beer" (Free Software Foundation, 2009). Stallman created the GPL (General Public Licence) which is subject to some controversy as it is in sense viral; requiring all software using GPL licensed code to also be licensed under GPL (Fitzgerald, 2006). More commercially friendly and popular open source licenses have since emerged, however a discussion on the different license types are not within the scope of this paper.

The fact that the idea of free and open source software originated in the academic community may be one of the reasons why it is difficult to understand the open source movement from an economic perspective (Lerner and Tirole, 2001). Historically, academic organisations are different from organisations seeking profit for their owners and shareholders. As noted by Baird (1997) in his paper on the conflict between gift and commodity economies, academics

write scientific articles which serve as intellectual gifts to the community. Comparing the commodity economies, in which businesses operate for profit, with the gift economy associated with academic organisations, Baird (1997) provides the following insight: “As commodity economies establish status hierarchies through how much is accumulated, gift economies establish hierarchies through how much one gives” (p. 31).

When looking at individuals’ motivation for contributing to open source, Lakhani and Wolf (2003) found that approximately one third of the respondents in their survey felt a sense of obligation to give back to the free and open source software community. These findings are consistent with the concept of *gift economy*, where one is expected to recycle gifts (Baird, 1997; Baytiyeh, and Pfaffman, 2010). As such, the gift economy may explain certain parts of the open source movement. Both the origin of the free and open source software phenomenon and the individuals’ motivation for contributing suggests that open source software exists within the boundaries of the gift economy. Nevertheless, present-day events show that this is not the case. For example, when IBM released their integrated development environment Eclipse as open source, it was valued at \$40 million (Fitzgerald, 2006). IBM, a multibillion dollar company is surely considered part of the commodity economy (IBM, 2008) and their contribution to open source cannot be explained sufficiently by the theory of gift economy alone.

With economic theories seemingly unable to explain the open source phenomenon accurately, researchers have been trying to explain what motivates developers to contribute to open source. Most of the research available has focused on individuals motivations for contributing (Baytiyeh, and Pfaffman, 2010; von Krogh and Spaeth, 2007;). However, the literature identifies different advantages of open source, and the companies that contribute to open source might be motivated by one or more of these apparent benefits. Analysing this research we identified three different motivational factors that are relevant for commercial companies that are contributing to open source, shown in table 1. We take as our premise (in line with Bonaccorsi and Rossi, 2006) that commercial companies use FOSS not for altruistic reasons, but for business reasons, i.e. to achieve competitive advantage (Porter 1985).

| <b>Motivational factor</b>             | <b>References</b>  | <b>Competitive advantage</b>  |
|--|--|---|
| Building greater innovative capability | von Krogh, 2003; Dahlander and Wallin, 2006; von Hippel and von Krogh, 2003; Bonaccorsi and Rossi, 2006; Chesbrough, 2003; Kline 2003; von Hippel, 2001; Ebert, 2007 | Being able to offer new products, better or faster than competitors                   |
| Selling complimentary services         | Watson et al., 2005; Dahlander, 2005; Dahlander and Magnusson, 2005; Dahlander and Magnusson, 2008; Pykäläinen, 2007; Watson et al, 2008; Lerner and Triole, 2002    | Specializing in a market niche, to sell services to companies that use FOSS products. |
| Cost reduction                         | Ågerfalk and Fitzgerald, 2008; Hawkins, 2004; Cook, 2008; Pisano and Verganti, 2008; Tapscott and Williams, 2008; Howe, 2008   | Using FOSS to lower development and maintenance costs                                 |

Table 1. Motivational factors for companies contributing to FOSS

## **2.2 Building greater innovative capability**

One advantage that many argue open source software has over proprietary software is greater ability to innovate (von Hippel, 2001). While many used to view FOSS as merely an imitation of proprietary offerings, people today are acknowledging the FOSS movement as a major source of innovation (Ebert, 2007, Bonaccorsi and Rossi, 2006). The innovative capability of FOSS may be explained by FOSS' position somewhere between the worlds of the gift and commodity economy. The two different economies are associated with two different innovation models. In the collective action innovation model, associated with the principles of gift economy, innovations are freely revealed in order to benefit the common good, with the risk of motivating free riding. Another innovation model known as the private investment model, more closely associated with the commodity economy, does not have this problem with free riding. In the private investment model, innovations are protected by intellectual property laws allowing the innovators to benefit from their innovations. However it makes it impossible for others to improve on the innovative design, as it is kept secret by the inventor and protected by law. von Hippel and von Krogh (2003) argue that the innovation process observed in the free and open source software community is best explained by a new theory of innovation, the "private collective innovation model", a best of breed of the two previously mentioned models. The argument is that even though code is revealed for free, the people contributing are getting rewards that are not available to free riders, such as the joy of learning. Other rewards such as corrections and critiques from the community are only possible if the code is freely revealed. By opening up the source code, the developers are also making it possible for others to improve and develop the code further, enabling a cumulative innovation process (Henkel, 2004). Developers are allowed to "stand on the shoulders of giants", an important concept in technological innovation and scientific research (Baird, 1997, Hauge et al., 2010; Scotchmer, 1991).

By participating in the development of free and open source software, companies are in fact applying an extreme form of open innovation (Dahlander and Wallin, 2006). Adopting open innovation is a strategic choice, where the company acknowledges that not all smart people are working for the same company, and that one does not have to originate all research to be able to profit from it. Adopters of open innovation favour building the best business model over being first to market (Chesbrough, 2003). von Hippel (2001), a strong advocate of open innovation, argues that software innovation through open source makes sense as the manufacturers will never know what the users want as well as they do themselves. By tapping in on the innovative capability of the open source community, companies that are releasing products as open source or in other ways contributing to the development of FOSS are speeding up the innovative process beyond what they would be capable of doing by themselves (Tapscott and Williams, 2008). The theory of open innovation has also lead to the rethinking of the definition of competitive advantage, with some arguing that when an innovation is shared the market for that innovation grows faster, benefiting everyone (Kline, 2003). Bonaccorsi and Rossi (2006) have found that smaller firms with limited or no R&D funds are able to take advantage of the R&D spillover present in open source. Suggesting that the innovative capability of open source is an important consideration for both small/medium sized companies as well as large. It is however important to note that according to the

“private-collective innovation model” (von Hippel and von Krogh, 2003), small firms will not benefit from free riding of the community innovations. They should instead contribute themselves to harvest the fruit of the open innovation model (Bonaccorsi et al., 2006; Henkel, 2009).

While one can certainly argue that the open innovation seen in open source development is a new form of outsourcing (Quinn, 1999; 2000), there is a vital difference between open innovation and innovation outsourcing. In open innovation the companies are not reducing the amount of R&D done internally, instead they are using the community to increase their R&D resources (George et al., 2005). This increase in R&D resources makes it feasible that the products will be developed faster and become better than if all the ideas and solutions originated inside the company. By employing an open innovation strategy through open source, the company is actively pursuing a differentiation strategy with better products than the competition.

Lerner and Triole (2002) have argued that open source software is created for “smarter” and more technically savvy users. The inherent innovative capability of open source software means that the product will add new features faster, and makes it a good fit for users that are looking for the latest innovative solutions to their problems. One can thus argue that by contributing to open source commercial companies are adopting a niche strategy (Mintzberg and Quinn, 1996) aimed at the technically savvy users. Hence, it stands to reason that the community innovation present in open source may motivate a commercial company either as part of a differentiation or a differentiation focus strategy, depending on the market being targeted.

Dahlander’s (2005) research has also shown that companies relying on innovations from the community are proactive in releasing code. This is a viable strategy, as it has been shown that the innovation process in the software community is cumulative (Henkel, 2004), therefore the sooner code is released the sooner it will serve as a foundation for new innovations. Hence, it is expected that companies motivated by innovation are frequently communicating with the community through bug trackers and messages boards/forums, and that they are quick to contribute their own innovations to the community. The strategic importance of the product is likely to be high in the case of the innovation motivational factor. If the decision to open source was taken by someone with an important position within the company, it is an indication that the importance of the product is also high.

Companies motivated by the innovation factor will to some extent expect the community to suggest new features and improvements to the product. To allow the community to undertake these intellectually challenging tasks, it is expected that employees of the commercial company pursuing the innovative properties of open source to perform routine tasks such as checking code quality and testing (Dahlander and Magnusson, 2008).

## **2.3 Selling complimentary services**

The fact that the software in free and open source software is free (gratis) does not mean that commercial companies are unable to profit from FOSS. The literature is full of examples of

companies that have created new business models that enable them to profit from FOSS by selling complimentary services (Dahlander and Magnusson, 2005; Watson et al., 2005). In fact, most research on motivations for companies contributing to open source focus on this factor alone. The selling of complimentary services has been identified as the dominant trend for companies trying to appropriate returns from FOSS (Dahlander, 2005).

There is a wide range of complimentary services offered for many FOSS products today, such as training, technical support, consultancy and certifications (Fitzgerald, 2006). Companies are succeeding with this approach as the adopters of open source are willing to pay for professional services (Fitzgerald, 2006). However, a business model based on the sales of complimentary services is not limited open source products; most proprietary software companies also sell complimentary services such as consulting and training (Cusumano, 2004). Still, it may be argued that companies selling complimentary services for open source software have a competitive advantage over their proprietary counterparts. As the FOSS product itself is free of charge, the amount of potential clients may be greater for companies selling FOSS complimentary services as software becomes a commodity (Pykäläinen, 2007). Lerner and Triole (2002) has likened this to giving away the razor to sell more razor blades. Assuming that the client of the company selling the complimentary service will need training et cetera regardless of the product being open or proprietary, the total acquisition cost and total cost of ownership will be lower with open source products as they are now of equal quality as proprietary products (Watson et al., 2008).

Porter's (1985) theory of competitive advantage would suggest that companies selling complimentary services for open source products have the competitive advantage of cost leadership. By exploiting the zero acquisition cost of open source, the company is delivering the same service at a lower price than competitors selling complimentary services for proprietary products with associated acquisition costs. However, as Cusumano (2004) argues, it is very easy for new entrants to compete in this area as the product one is selling complimentary services for is open. Cancelling out the threat of new entrants may be accomplished if the incumbent company has superior experience or access to raw material (Porter, 2008). Thus, it makes sense for the company to engage in the development of the open source product. The rationale is that having committers on a FOSS project is an advantage in both experience and access to resources that are difficult to replicate (Riehle, 2007).

Simon Phipps, the Chief Open Source Officer at Sun Microsystems, put it this way when speaking at the Community One conference in Norway in April 2009: "There are two types of open source support out there. One is where you call a number and a guy answers the phone at the other end and asks you to reboot your server. The other kind is where the guy who actually wrote the code is flown in on a helicopter, landing on your roof, fixes the bug and promise that the fix will be included in the next release of the open source product. One of these is worth paying for". Although Phipps is talking about extremes, it illustrates the advantage a company selling complimentary open source services which employs committers has over new entrants which are unlikely to have this valuable resource.

Selling complimentary services for open source software may also be considered a differentiation strategy, as the company is creating buyer value through good quality and innovative products that has no proprietary counterpart (Bonaccorsi and Rossi, 2006). At the same time, the cost inflicted for the company by releasing the product as open source as opposed to keeping it proprietary is considered low (Harhoff et al., 2003). The pursuit of uniqueness in product at a low cost is a good strategy for a company going after the generic strategy of differentiation (Porter, 1985). Depending on the market being target the strategy may be labelled as either differentiation or differentiation focus.

There is a wide range of literature that supports the notion that companies are contributing to open source software because it enables them to sell complimentary services. The leading trend seems to be selling consultancy hours (Dahlander, 2005), but selling support is also a common way of monetising open source software. For both consultancy and support it is important that the company selling these services has the required know-how (Pykäläinen, 2007). To get the know-how needed to enable the sale of complimentary services, the company employing this strategy is expected to encourage its own employees to contribute to open source. This will improve the employees' competence which again allows the company to sell their know-how. In addition to allocating resources to the open source development, it is also expected that the company is working hard to retain the employees that are actively contributing to the open source community. Nonetheless, the contributing employees' time is likely prioritised to serve paying clients before the community. The company may also recruit employees from the community to better serve their clients (Watson et al., 2005). It is expected that the company is contributing bug fixes which will improve the overall quality of the program. However, it is not expected that all add-ons to the product is released to the community if it is feasible to sell them as part of a complimentary service.

## **2.4 Cost reduction through opensourcing**

In a journal article in 2008, Ågerfalk and Fitzgerald coined the term *opensourcing*. The article describes a software development model similar to that of outsourcing, except that instead of outsourcing to an outsourcing partner one is outsourcing to the open source community (Ågerfalk and Fitzgerald 2008). By releasing a product as open source, the potential gains for the commercial company are several. One obvious gain is the potential for development cost savings occurring when developers who are not paid by the company are contributing code. However, users are also contributing by identifying and reporting bugs in the software, serving as testers etc (Henkel, 2004; von Krogh et al., 2003; Lerner and Triole, 2002). Since the source code is accessible for users of the product, it also makes it easier for the users to pinpoint where the product is flawed. An argument for increased quality in open source software has been that "given enough eyeballs, all bugs are shallow" (Raymond, 2001, p. 30).

Hawkins (2004) concludes that companies are contributing extensions to open source products, which they are entitled to keep private, to open source, in order for it to become part of the code base that is maintained by the community. By doing this the code's compatibility with future releases is assured without the company itself having to do it. Hawkins' (2004) findings suggest that the opensourcing strategy may also be a motivational



factor for companies that are not contributing entire products, but instead are contributing additional functionality to existing FOSS products.

Both Ågerfalk and Fitzgerald (2008) and Dahlander and Magnusson (2008) have shown that establishing an open source community, a vital part of opensourcing, is no easy task. Thus, some might consider the opensourcing strategy not as merely a new flavour of the outsourcing operational effectiveness scheme, however opensourcing should instead be viewed as a strategy where the company in question is getting comparable value at a lower cost than its competitors.

## **2.5 Framework of motivational factors**

Summing-up, the literature review has shown that there are three main factors impacting the decision to contribute to open source. Which factor is the most important?

The main focus in the IT literature has been on the sale of complimentary services. As such, one should expect to find a strong association between the sale of complimentary services and the decision to contribute to open source. On the other hand, there are ample reasons to investigate the other two factors.

The theory of open innovation has been used by academics to describe the open source development process. The idea of superior innovative capability through open source development seems to be catching on, and it is expected that while the innovation will not be as strong a motivator as sale of complimentary services, companies will factor innovation into their decision to open source.

Regarding cost reduction, the business literature has been advocating the use of the *unknown workforce* (Cook, 2008; Tapscott and Williams, 2008). However, with the exception of Ågerfalk and Fitzgerald's (2008) article, there has not been much mention of the benefits of open source that is comparable to those of outsourcing in the IT literature. Yet, the business literature has described this phenomenon as a production revolution and new economic model.

## **3 Methodology**

### **3.1 Research approach and design**

A case study was chosen for this research. Research questions that aim to answer *why* a contemporary set of events (over which the investigator has little or no control) happen is a good fit for the case study methodology (Yin, 2009). Yin further states that the purpose of a case study is not to do a statistical generalisation, but to do an analytic generalisation (Yin, 2009).

According to Yin (2009), having two or three cases makes it possible to do literal replication, where similar results are expected. This literal replication enables the establishment of a theoretical framework that can later be generalised for new cases. As a framework of

motivational factors was suggested in this paper, a multiple-case study design with three cases was considered a good fit when trying to answer the chosen research question.

The unit of study was the commercial company contributing to open source, hence three different companies was analysed. The companies chosen were three companies of very different sizes working in the IT service industry, i.e. selling IT services such as training, consulting and support. Companies from the IT service industry, and not the software industry, were chosen as they were the most obvious candidates to be motivated by the sale of complimentary services. As the sale of complimentary services has been touted as the main motivator for companies contributing to open source, interviewing these companies would be a good way to see how the innovation and opensourcing motivational factors compare to the conventional motivator of selling complimentary services.

At each company, the decision to contribute to open source was made by a limited amount of leaders and developers. Key personnel involved in each company's open source initiative were interviewed.

### **3.2 Data collection**

The three companies were at the time actively contributing to open source software. Interviews were conducted with key individuals at each of the three companies. The role of the individuals ranged from decision makers involved in the decision to go open source to developers involved in the development and contribution of software code.

A semi structured interview guide was created to assist in the investigation of the factors motivating each company. The semi structured interview guide contained questions aimed at revealing the motivational factors in the analytical framework, but also potential rival theories identified. The interviews were conducted by the first author.

The number of people interviewed at each company in this case study (Accenture(3), Arktekk(2) and Redpill Linpro(3)), is a limitation. It is worth mentioning that the number of people interviewed at Arktekk actually amounted to 50% of the employees at the time. However at Accenture and Redpill Linpro, the number of people interviewed was just a small sample of the total amount of employees.

Still, it should be noted that the sample at Redpill Linpro and Accenture was not chosen by random; in fact the people interviewed were employees in strategic open source leadership roles as well as lead developers on the OSS projects.

With regards to the number of respondents for each case, it is important to be aware that for case study research, as opposed to quantitative research, theory is developed through analytic generalization instead of statistical generalization (Yin, 2009). The case study also utilized other sources of information, such as e-mail communications and the open source bug databases to strengthen the study's construct validity (Yin, 2009).

The interviews were conducted in person or over the phone and all interviews were recorded with the interview subjects' consent. After each interview a summary was written and sent to

the subject to get a final approval of the accuracy on the facts reported. A summary of the questions asked and the answers given is supplied in appendix A.

### **3.3 Data analysis**

The data analysis was conducted using a pattern matching technique, where a set of expected patterns was compared to the observed patterns. Prior to any data being collected a literature review was done and a set of expected patterns were identified. Identifying the patterns before any data was collected was done to secure the internal validity of the case study, as suggested by Yin (2009). The expected pattern was that the main driver in the companies open source strategy would be to sell complementary services.

After the data had been collected, a review of the interview summaries was done and the indications for each case study was organised in a matrix according to motivational factors. The matrix showed which motivational factor had weighed the strongest for each company. Using the data from the matrix, a radar chart was created to show how each motivational factor had contributed the companies' decision to open source. Pattern matching was then done to compare the expected pattern with the observed pattern at the three companies.

In addition to developing indications for the three motivational factors of the analytical framework, indications of rival theories was also developed, the rival theories identified were; public relation, recruiting and null hypothesis. During all interviews queries were done to investigate if there was support for the rival theories.

## **4 Analysis and findings**

This section provides a short description of each company and their involvement in open source software development. Then we present in some detail our findings.

### **4.1 Arktekk**

Arktekk is a small Norwegian company made up of five consultants which offers consultancy services to clients implementing solutions based on open source software. In addition to performing traditional system integration and development work, several of Arktekk employees are also working as instructors in training focusing on open source software such as Maven, Spring and Hibernate.

The company was founded by four consultants who had a shared passion for open source development. Arktekk employees have contributed to well known open source software such as Maven (Mileva et al., 2009). Recently Arktekk has also contributed an enterprise application configuration tool, Constretto, which is being used by several large Norwegian companies. Constretto was developed by Arktekk at one client, who was then persuaded to release the product as open source to enable Arktekk consultants to maintain the code after the project had ended. Since then, the several Norwegian companies have adopted Constretto.

### **4.2 Accenture**

Accenture is a global consultancy with approximately 177.000 employees and offices in 52 different countries around the world. The company is delivering management, technology

and outsourcing services and with more than 30 years' experience with developing custom solutions, Accenture are at any time involved in many of the most challenging system integration projects in the world.

In 2006, one of Accenture's senior managers had noticed that a lot of work was being done at different projects to develop batch processing frameworks for clients. While many open source frameworks existed for developing web applications, there was no open source de facto to assist developers of batch processing applications. The result was that each project ended up implementing its own batch processing framework, one project had spent more than 8 months developing such a framework. Having identified the gap in the plethora of available open source frameworks, the manager came up with the idea of creating an open source offering to stop the redundant writing of similar code. The background for this idea became clear in an interview with ACC2, where he said:

*"I found six projects creating their own batch architectures from scratch (...) The vision behind Spring Batch was, shoot, let's stop the craziness and stop re-inventing the wheel every project. Let's create something that projects can reuse. Oh, and by the way, the best way to get people actually actively using it will be to get the market to accept it as well."*

ACC2's statement clearly shows that there was a great need for the product. More interestingly it shows that Accenture from the very beginning appreciated that greater adoption and acceptance would be possible through open source contribution.

Much of Accenture's development efforts were utilising open source frameworks developed by the renowned open source company SpringSource. At the time, SpringSource had a large portfolio of open source products ranging frameworks for core java and web development to frameworks for security and developer tools. By 2007, Accenture had formalized an alliance with SpringSource to develop Spring Batch, and in June 2008 it announced the first production ready release of Spring Batch, an open source batch processing framework released under the Apache license 2.0.

Spring Batch helps developers reduce the time it takes to develop batch applications. Batch applications are process intensive applications which handle a large amount of transactions, typically scheduled to run outside working hours without human interaction.

### **4.3 Redpill Linpro**

Redpill Linpro employs 180 people and is a Nordic supplier of open source solutions and services. These services include support, training and consultancy. In addition to offering support for many well known open source products such as JBoss and Sugar CRM (Watson et al., 2005; Brydon and Vining, 2008), they are also developing and maintaining four different open source products, with the web accelerator Varnish being their commercially most successful open source product. The initial development of Redpill Linpro's open source initiatives have started as custom solutions for clients, but have been released open source with the clients consent as the ability to generalise the product to make it usable for more than one client has become evident.

Varnish was developed by Redpill Linpro in collaboration with the Danish developer Poul-Henning Kamp for the Norwegian newspaper Verdens Gang (VG). VG wanted the product to become open source as this would increase the chance of the product flourishing and evolving, with the help of the open source community. After the initial 1.0 release, which was paid for by VG, Redpill Linpro took ownership of the product by dedicating developers to work on the product at their own expense. At moment three developers employed by Redpill Linpro is dedicating between 50% and 100% of their time to develop Varnish further.

Varnish is currently being used by clients in all parts of the world and increases the performance of web servers and reduces the need for new hardware.

#### **4.4 The motivational factors**

This section reports the findings related to the three motivational factors identified namely complimentary services, opensourcing and innovation in addition to the rival theories of public relation, recruiting and the null hypothesis.

##### **4.4.1 Innovation**

Innovation is a key motivator for all three companies. An important indicator of this is the expectance that the community will help develop and identify additional features for the open source product present at both Redpill Linpro and Accenture. It is worth noticing that the companies are not expecting these contributions to surface by magic. Both companies are actively involved in stimulating the community, something which makes it more likely that major contribution will emerge from the community.

Redpill Linpro is stimulating the community by arranging an event in London for users of Varnish, where users are invited to share their knowledge and help create a road map for the future development of Varnish. The importance of user involvement is emphasised by Redpill Linpro employee RED1 who offers the following insight into Redpill Linpro's motivation for setting up this user group meeting:

*“Arranging such a meeting is relatively inexpensive when you take into account the potential gain we may harvest.”*

It is clear that Redpill Linpro is hopeful that the community interaction will ultimately lead to a better product.

Accenture has a history of listening to the community when developing Spring Batch. By taking the community seriously and adopting their design propositions Accenture is stimulating the community, increasing the likelihood of further contributions. Accenture employee ACC2 stressed that the community was very much involved in the design of Spring Batch, and at one point changed the course of development:

*“I remember ACC1 going; if we haven't been doing it like this [open source], we never would have thought about that [use case] and we probably wouldn't have considered this approach (...) and [now] we ended up with a more flexible framework.”*

This clearly shows that by listening to the community, and incorporating suggestions, Accenture allowed the community to innovate and acknowledged that not all clever programmers could be employed by Accenture. The fact that the idea originated outside the group of Accenture and SpringSource developers did not mean that it should not be adopted. This is a vital point in open innovation and showed that Accenture's intent to utilise the ideas of open innovation.

The partnership Accenture struck with SpringSource should also be seen as a move to further stimulate community innovation. Accenture hoped to benefit from SpringSource's momentum in the open source community. Through the partnership with SpringSource, Accenture got an instant community with a bug tracker, community forums and thousands of potential users and early adopters. Having the means to communicate with the community is an obvious requirement for any company aiming to benefit from community innovation.

Accenture expected that they had to do most of the initial work on Spring Batch, although SpringSource added a couple of resources to the project some months into the development. This attitude towards contribution is often seen by companies hoping for community innovation. The idea being that by contributing code, the company is making it easier for others to build innovations on top of the available code thereby motivating cumulative innovation.

Contrary to Redpill Linpro and Accenture, Arktekk is not as definitive in their expectations from the community, consistently using the word "hope" in lieu of "expect". However, Arktekk is adamant that open source software does give better quality and that it seems that most software related problems have an open source solution. As such they are cognisant of the open source movement's innovative capability. The need to contribute to open source to stimulate innovation is also seen at Arktekk, with ARK1 stating:

*"While you can't expect contributions from the community on each project you are involved in, you will certainly not get any contributions if you don't share your thoughts and your code. I fully expect others to share their ideas through open source software in the same way that we share ours."*

Arktekk, like Redpill and Accenture, are paying its employees to contribute to open source. This approach to open source contribution is seen as a vital activity to stimulate community innovation as well. A vibrant community is a requirement for cumulative innovation. Employees are actively monitoring different channels to interact with the community, helping out beginners as well as discussing major design changes.

A major difference between Arktekk and the other companies interviewed is their stance when it comes to bug fixing and improvements of existing open source software. While Redpill Linpro and Accenture encourage giving back to the community, Arktekk demands that the code is given back to the community, otherwise they will not write it. While such an attitude could be explained by ideology, it is also an indication that Arktekk acknowledges that without contributions to the community the innovative process will suffer. This attitude suggests that Arktekk's main motivation for contributing to open source is innovation.

#### **4.4.2 Complimentary services**

All three companies interviewed are profiting on open source software by selling complimentary services, and as such it is no surprise that the sale of complimentary services is a factor for all companies. However, there are some differences between the three companies in the emphasis they put on this factor.

Redpill Linpro has a clear policy where they are increasingly trying to monetise their open source portfolio. The employees who are paid to make open source contributions are sometimes reallocated to support paying customers, additionally bug reports from customers are prioritised at the expense of non-customer bug reports. Moreover, Redpill Linpro is offering a proprietary add-on for its paying customers only, as a differentiator to persuade companies to buy Redpill Linpro's services. According to RED3 one is also expecting the community to contribute to Redpill Linpro's commercial offerings by posting favourable blogs and articles about Varnish.

Accenture initiated Spring Batch to fill a gap in the enterprise development stack, and Accenture believed that by going open source the product would have a greater chance of becoming a commodity, thus increasing the number of adopters and potential clients. This motivation is typically observed by companies motivated by the sale complimentary services (Raymond, 2001).

Accenture is currently gearing resources towards the sale of complimentary services, with a wide range of offerings available. Proprietary add-ons for open source products, support for a stack of open source products as well as consultancy is being offered to clients. As Accenture has reduced the number of people working directly with Spring Batch, these people have been used as experts on client projects implementing solutions using Spring Batch. This shows that Accenture is able to profit from its dedication to open source by selling know-how, as suggested by Pykäläinen (2007).

Arktekk acknowledges that their involvement in open source software development is helping them secure new clients, with ARK1 admitting that they don't have enough resources to do unpaid OSS work. The know-how gained by Arktekk consultants is an asset for Arktekk, as ARK2 puts it:

*"Code is worthless; knowledge is the only real value."*

However, there are not many indications that Arktekk is engaged in open source development as a mean to sell its consultants. As such, the sale of complimentary services as a motivator is not particularly strong for Arktekk.

#### **4.4.3 Opensourcing**

For opensourcing there is a significant difference between the three companies, with Accenture being the one that considers opensourcing the most prominent driver for open source contribution of the three companies interviewed.

Accenture always expected that the amount of resources spent on developing Spring Batch would decline with time, according to ACC2. After some initial work was done by Accenture

alone, SpringSource assigned two people to the project, which let Accenture offload some of its knowledge to SpringSource while at the same time being confident that someone would help support Spring Batch in the long term. ACC3 said in his interview:

*“Let’s make it open source to share costs and give others a platform to participate and contribute.”*

This clearly shows that cost saving was a motivator for Accenture. Avoiding being stuck with maintenance cost for a proprietary product was also underlined as an important motivator by ACC1. ACC1 further revealed that the community was vital to get help with regards to testing on different platforms.

The actions taken by Accenture after SpringSource got developers on the project further strengthen the argument that Accenture was motivated by opensourcing. Accentures developers were gradually rolled off the project in favour of working on projects for paying customers. The partnership with SpringSource also indicate an opensourcing motivation, as ACC3 revealed that Accenture wanted to avoid having to maintain a community and believed that this job should be left to a professional open source vendor like SpringSource.

Similarly to what was observed at Accenture, using the community as testers was an important factor for Redpill Linpro when they developed Varnish. As Varnish is expected to perform under heavy load getting test data from community users with much traffic has been vital. However, Redpill Linpro accepts that the community is presently not self sustained and Redpill Linpro must be involved, otherwise the product would not prosper.

As with the innovation motivator, Arktekk were once again different from Redpill Linpro and Accenture. While Arktekk as a company is not motivated by the opensourcing factor they are using this feature of open source to convince clients to contribute to open source. ARK1 reveals that clients that are undecided on open source are often persuaded to contribute when they are told that by contributing they are increasing the likelihood that the community will adopt and maintain the code. Furthermore, Arktekk does expect users of open source to report bugs as they surface, helping improve the overall quality of the product.

#### **4.5 Key findings**

The complimentary services is without a doubt the strongest motivator for Redpill Linpro who is adopting the business model seen by most professional open source companies. The business model is based on selling support, training and consultancy for renowned open source products. This, however, was not the case for Arktekk and Accenture. Accenture to some extent was motivated by the sale of complimentary services, by expecting more clients through adoption. For Arktekk the sale of complimentary services was only a limited factor as open source contribution was seen as helpful when getting new customers.

All cases show a tendency towards expecting innovative contributions from the community. While this in itself might not be surprising, the interviews revealed that these expectations are backed up by actions taken by each company in an attempt stimulate community innovations.



All three companies are stimulating the community by donating code and actively interacting with the community.

The importance of opensourcing is different at each company, with Accenture's contribution being the one that is motivated the most by this factor. The partnership with a noted open source vendor and the gradual offloading of knowledge to this vendor is a strong indication that opensourcing is a major factor for Accenture. Redpill Linpro is also exhibiting some community expectations associated with opensourcing motivation, such as bug reporting and testing. However, their realisation that the community is not self sustained decreases the importance of opensourcing as a motivational factor for Redpill Linpro. Arktekk's stance on opensourcing is that it is a good way to motivate other companies to contribute to open source. As such the factor does impact Arktekk's contribution to open source in the way that it helps them argue in favour of open source with their clients. However, internally the indications of this motivation are not as strong at Arktekk as the case was at Accenture.

As is shown in Figure 1 the three different cases are exhibiting three different major motivators. Moreover, it shows that all the factors identified have had an impact for the companies in this analysis.

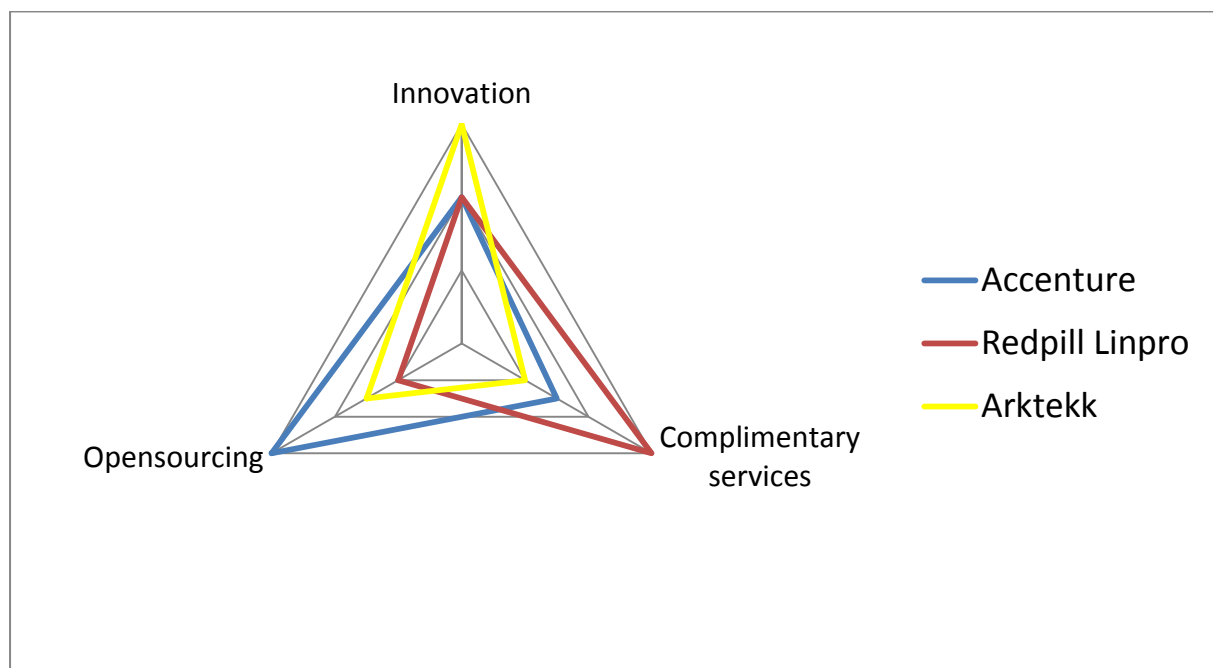


Figure 1. Graphical representation of motivational factors for each of the case studies

#### 4.6 Other findings

A finding that is worth mentioning is that there was a sense of moral obligation to contribute to open source at all three companies examined. At Accenture one informant said:

*“Be a good citizen, contribute to open source.”*

An informant at Arktekk had a similar view:

*“If you are using open source software, you should also contribute.”*

Redpill Linpro also exhibited proof that they see real value of being part the open source community. One informant said:

*“We are honoured to contribute to open source, and we don’t want to be ‘just another open source consultancy’ that happens to install Linux instead of Windows. We want to be a contributor.”*

The moral obligation to contribute to open source was not something that was specifically investigated, but something which came up during many of the interviews. While it was not covered in depth, it is contradictory to the research of Bonaccorsi and Rossi (2006) which concluded that social factors were not a motivation for commercial companies. Such findings are not totally unexpected for smaller firms like Arktekk, where it is difficult to draw a line between company and individuals. However, it is surprising that the bigger firms like Redpill Linpro and Accenture also show a sense of moral obligation to give back to the community.

One explanation is that the open source development model has matured since Bonaccorsi and Rossi’s (2006) research, and that companies are beginning to develop a sense of how open source development is dependent on mutual contribution, and as such a moral obligation is starting to grow inside the companies and not just on an individual level. These initial findings, which contradicts existing research, indicates a shift in corporate view of open source and a revisit into the research area of moral obligation and company contribution to open source is warranted.

## **5 Discussion**

As our research review revealed, most previous research on commercial open source software has focused on the sale of complimentary services as the way to appropriate returns from open source software development (Dahlander, 2005; Watson et al., 2006). Evidence found in this paper indeed supports the claim that selling complimentary services is an important way of appropriating monetary rewards from open source development. All companies exhibited indications that the sale of complimentary services was a motivation, nevertheless the emphasise they put on this was varying. The case of Redpill Linpro lends strong support to the vast amount of publications (Dahlander, 2005; Fitzgerald, 2007; Pykäläinen, 2007; Watson et al., 2008) that argue that direct appropriation through sale of complimentary services is the reason why commercial companies are pursuing open source development. By comparing the expected pattern in with the observed patterns it is evident that Redpill Linpro’s motivations are very close to what was expected.

However, by drawing on the observations at Accenture and Arktekk, the view proposed by the literature that commercial companies are contributing to open source as a way of making money is moderated. Arktekk seem to be more motivated to contribute to open source by the open source movement’s strong tendency to be innovative than any of the two other factors. An important aspect of open innovation is the realisation that not all talented people can work at one place, and that one don’t have to originate all research to profit from it (Chesbrough,

2003). This may explain why Arktekk, the smallest company in the study, is the company that is seen as the one that is the most motivated by innovation when contributing to open source. Small companies, as opposed to the bigger ones, are more likely to realise that they simply do not have enough resources to originate all research, and that their limited R&D resources should go to innovate in unison with external R&D resources. This combination of external and internal resources is a principle in the open innovation model (Chesbrough, 2003).

Contrary to Arktekk, who do not have enough resources to do open source development without getting paid, Accenture and Redpill Linpro have dedicated resources to do open source development. Being able to afford a full time R&D staff, explains why these two companies are not as strongly motivated by the innovation factor as Arktekk. As they are able to originate research without help, the collaborative R&D exhibited in open source development is recognised as motivator, but not a main driver. However, recent literature suggests that being able to participate in knowledge flows that exists outside the boundaries of a company is crucial to compete in the knowledge intensive economics of the new world (Hagel III et al., 2009). As such, it is expected that bigger companies such as Redpill Linpro and Accenture will see innovation as a more important factor in the future if they are to have continued success.

Innovation is a strong or moderately strong factor for all three companies analysed, in fact, innovation is the factor that has the strongest presence when combining all three companies. This shows that there is a gap in previous research with regards to motivations for contributing to open source. While open innovation and open source software development has been the subject of academic analysis (von Hippel, 2001; von Hippel and von Krogh, 2003), IT researchers have not gone on the inside of companies to observe the innovative forces at play in open source software contributions. This paper provides empirical support for the theory of open innovation and findings show that companies are aware of the benefits of collaborative innovation present in open source software development and that this is in fact an important aspect when they decide to engage in open source software development.

Porter (1996) has argued that outsourcing is simply an improvement of operational effectiveness, and not a source of competitive advantage. This argument suggests that opensourcing (outsourcing) should not be considered as a motivational factor when trying to gain competitive advantage through open source contribution. However, to succeed with opensourcing one has to have a self sustained community of outside developers (Ågerfalk and Fitzgerald, 2008). Previous research has shown that establishing such a community is difficult, and that firms have failed both to establish and maintain such communities (Dahlander and Magnusson, 2008).

The reliance on a self-sustainable community for any company pursuing the opensourcing strategy might explain why both Arktekk and Redpill Linpro saw this motivational factor as secondary. Neither company were involved in projects where the community was big enough to be self-sustained. At Accenture however, opensourcing was seen as the main motivational factor for the company's decision to contribute to open source. The step taken by Accenture

to partner with a professional open source vendor to get an instant community is interesting in this aspect. An informant at signified the importance of this partnership, saying:

*“To make it successful, we had to engage with a strong, well established and managed community. A commercial open source vendor seems to be the best option for this purpose.”*

Accenture’s actions show that it is possible to reduce the time it takes before one may reap the benefits of cost reduction through opensourcing. Findings at Accenture also go a long way in identifying a gap in the existing literature on commercial contribution to open source, which has not focused on the opensourcing motivational factor. It also shows that careful consideration must be done with regards to the size and momentum of an open source community if a company is to succeed with the opensourcing strategy.

Critics of outsourcing point out that outsourcing will gradually drain a company of innovative capabilities (Pisano and Shih, 2009). Yet, for open source software development, this study has shown evidence that companies might be able to both reduce development cost through opensourcing and at the same time increase the innovative capability through open innovation.

No evidence was found to support the proposition that companies are participating in open source development to improve public relations (Lerner and Triole, 2002). This is probably due to the fact that participating in open source development is resource intensive, which makes it easier to pursue other ways of improving a company’s reputation. It may also be explained by the “less glamorous” (ARK2) software developed by some of the companies, which makes is unlikely to receive much attention. Another explanation may be found in the fact that not many people are able to read and evaluate source code, and as such there are more efficient ways of impressing the public and potential customers than releasing source code (Bonaccorsi and Rossi, 2006)

While there are examples of companies that are recruiting employees from the open source community (Watson et al., 2005; Watson et al., 2008), this was not seen as a motivation at any of the three companies. There are probably several explanation why this is not an important motivation. First of all, the open source community is dispersed around the globe and as most companies favour working in the same location, reallocation of recruited resources is likely to be costly and difficult. Second, a community has to be of a certain size before recruiting personnel becomes feasible. Only Accenture were interacting with a community of that size, but were they to recruit from the community, the value of the community as a cost saving outsourcing partner would decrease (Ågerfalk and Fitzgerald, 2008).

## **5.1 Limitations**

We acknowledge that this study has limitations. The study focuses on commercial companies offering IT services, as opposed to software companies. As such, the area of application is limited. Specifically, one area of motivation that is observed at some software companies contributing to open source, that of the loss-leader (Raymond, 2001), has limited application

for IT service companies. Hence, further research should be done to test the framework proposed in this paper, before findings can be applied to OSS contributing software companies.

While both Redpill Linpro and Arktekk are relatively young companies, Accenture has been working with system integration for more than 30 years. As such, one may ask if Accenture's contribution to open source is anything more than an outlier. However, there are several points that suggest this has more to do with a change in corporate culture than a mere one-off.

First of all, commercial contributions to open source have not been around for as long as Accenture. In fact, the first open source business can be traced back to Cygnus Solutions in 1989, whilst Redhat, arguably one of the most successful open source businesses, was not founded before 1995 (Raymond, 2001; Lerner and Triole, 2002). Still, the major breakthrough for open source did not come until 1998. This was the year when Netscape, a fortune 500 company, bet its future on open source by releasing its browser as open source and creating the Mozilla Public License (Raymond, 2001). IBM, which is now renowned for their open source contributions to both Linux and the Eclipse foundation, only pledged their commitment to Linux in 2001 (Capek et al., 2005; Fitzgerald, 2006; Samuelson, 2006). Coincidentally, 2001 was also the year that JBoss inc was founded, the company that brought major open source contributions to the Java platform through the JBoss application server (Watson et al., 2008).

Second, since Spring Batch, Accenture has released another product as open source. Cloud Map Reduce, released as open source by Accenture in 2009, is an implementation of a framework patented by Google to support distributed computing over large data sets using clusters of computers (Ranger et al., 2007; Cloud MapReduce, 2011). Accenture's implementation is in some cases 60 times faster than Hadoop, one of the most widely adopted map reduce implementations (Cloud MapReduce; 2011, Hadoop, 2011). Thus, this contribution along with Spring Batch shows that Accenture is contributing high quality products to open source.

Hence, while Accenture were certainly not on the forefront of commercial open source contribution, their contribution should not be written of as an outlier, but instead seen as a reaction to a trend in systems integration and development.

## **6 Conclusions**

This paper investigated why commercial companies contribute to open source software. Building on an extensive research review and a case study of three IT service companies, we offer the following conclusions.

First, we identified three main drivers for contributing to open source are a) selling complimentary services, b) building greater innovative capability and c) cost reduction through opensourcing to an external community.

Second, while previous research has documented that the most important driver is selling complimentary services, we found that this picture is too simple. Our evidence points to a broader set of motivations, in the sense that all our cases exhibit combinations of the three drivers. Building greater innovative capability through an open innovation approach may be an equally strong driver, and the same applies to the motive of cost reduction through opensourcing to an external community.

Finally, our findings suggest that there might be a shift in how commercial companies view open source software. The companies interviewed have all expressed a moral obligation to contribute to open source. Further research should investigate whether this is true, or merely lip service. Further research should also investigate whether outsourcing to an external community affects negatively the innovative capabilities of a company. We also believe that outsourcing networks should be investigated more holistically as developing ecologies, rather than separate actors.

## 7 References

Ågerfalk, P. J. and Fitzgerald (2008) 'Outsourcing to an Unknown Workforce: Exploring Opensourcing as a Global Sourcing Strategy', *MIS Quarterly*, 32 (2), pp. 385-409.

Baird, D (1997) 'Scientific Instrument Making Epistemology, and the Conflict Between Gift and Commodity Economies', *Society for Philosophy and Technology*, 2 (3-4), pp. 25-45.

Baytiyeh, H. and Pfaffman, J. (2010) 'Open source software: A community of altruists', *Computers in Human Behavior*, 26 (6), pp. 1345-1354.

Bonaccorsi, A., Giannangeli, S. and Rossi, C. (2006) 'Entry Strategies Under Competing Standards: Hybrid Business Models in the Open Source Software Industry', *Management Science*, 52 (7), pp. 1085-1098.

Bonaccorsi, A. and Rossi, C. (2006) 'Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement: From Community to Business', *Knowledge, Technology & Policy*, 18 (4), pp. 40-64.

Brydon, M and Vining A. R. (2008) 'Adoption, Improvement, and Disruption: Predicting the Impact of Open Source Applications in Enterprise Software Markets', *Journal of Database Management*, 19 (2), pp. 73-94

Capek, P. G., Frank, S. P., Gerdt, S. and Shields, D. 'A history of IBM's open source involvement and strategy', *IBM Systems Journal*, 44 (2), pp 249-257.

Chesbrough, H. W. (2003) 'The Era of Open Innovation', *MIT Sloan Management Review*, 44 (3), pp. 35-41.

Cloud MapReduce (2011) *Cloud MapReduce*, Available at:  
<http://code.google.com/p/cloudmapreduce/> [Accessed 9th of January 2011]

Cook, S. (2008) 'The Contribution Revolution – Letting Volunteers Build Your Business', *Harvard Business Review*, 86 (10), pp. 60-69.

Cusumano, M. (2004) 'Reflections on Free and Open Software', *Communications of the ACM*, 47 (10) pp. 25-27.

Dahlander, L. (2005) 'Appropriation and appropriability in open source software', *International Journal of Innovation Management*, 9 (3), pp. 259-285.

Dahlander, L. and Magnusson, M. G. (2005) 'Relationships between open source software companies and communities: Observations from Nordic firms', *Research Policy*, 34 (4), pp. 481-493.

Dahlander, L. and Magnusson, M. (2008) 'How do Firms Make Use of Open source Communities', *Long Range Planning*, 41, pp. 629-649.

Dahlander, L and Wallin, M. W. (2006) 'A man on the inside: Unlocking communities as complementary assets', *Research Policy*, 35, pp 1243-1259.

Ebert, C. (2007) 'Open Source Drives Innovation', *IEEE Software*, 24 (3), pp. 105-109.

Fitzgerald, B. (2006) 'The Transformation of Open Source Software', *MIS Quarterly*, 30 (3), pp. 587-598.

Free Software Foundation (2009) *Free Software and the GNU Operation System – Free Software Foundation*. Available at: <http://www.fsf.org/about/> [Accessed at 14<sup>th</sup> June 2009].

George, M. L., Works, J. and Watson-Hemphill, K. (2005) *Fast Innovation – Achieving Superior Differentiation, Speed to Market, and Increased Profitability*. New York, NY: McGraw-Hill.

Gould, M. D. and Gruben, W. C. (1996) 'The role of intellectual property rights in economic growth', *Journal of Development Economics*, 48, pp. 323-350.

Hadoop (2011) *Hadoop*, Available at: <http://hadoop.apache.org/> [Accessed 9th of January 2011]

Hagel III, J, Brown, J. S. and Davison, L. (2009) 'The Big Shift – Measuring the Forces of Change', *Harvard Business Review*, 87 (7/8), pp. 86-89.

Harhoff, D., Henkel, J. and von Hippel, E. (2003) 'Profiting from voluntary information spillovers: How users benefit by freely revealing their innovations', *Research Policy*, 32 (20), pp. 1753-1769.

Hars, A. and Ou, S. (2002) 'Working for Free? Motivations for Participating in Open-Source Projects', *International Journal of Electronic Commerce*, 6 (3), pp. 25-39.

Hauge, Ø, Ayala, C., and Conradi, R. (2010), 'Adoption of open source software in software-intensive organizations – A systematic literature review' *Information and Software Technology*, 52 (11), pp. 1133-1154.

Hawkins, R. E. (2004) 'The economics of open source software for a competitive firm - Why give it away for free?', *Netnomics*, 6 (2), pp. 103-117.

Henkel, J. (2004) 'Open Source Software from Commercial Firms – Tools Complements, and Collective Invention', *Zeitschrift für Betriebswirtschaft*, 4, pp. 1-23.

Henkel, J. (2009) 'Champions of Revealing – The Role of Open Source Developers in Commercial Firms', *Industrial and Corporate Change*, 18 (3), pp. 435-471.

Howe, J. (2008) *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. USA: Crown Business.

IBM (2008) *Complete 2008 IBM Annual Report*, Available at: <http://www.ibm.com/annualreport/> [Accessed 24<sup>th</sup> May 2009]

Kline, D. (2003) 'Sharing the Corporate Crown Jewels', *MIT Sloan Management Review*, 44 (2), pp. 89-93.

Lakhani, K. R. and Wolf, R. G. (2003) 'Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects', *MIT Sloan Working Paper* No. 4425-03. Available at: <http://ssrn.com/abstract=443040> or DOI: 10.2139/ssrn.443040

Lerner, J. and Tirole, J. (2001) 'The open source movement: Key research questions', *European Economic Review*, 45, pp 819-826.

Lerner, J. and Tirole, J. (2002) 'Some Simple Economics of Open Source', *The Journal of Industrial Economics*, 50 (20), pp 197-234.

Lerner, J., Pathak, P. A. and Triole, J (2006) 'The Dynamics of Open-Source Contributors', *American Economic Review*, 96 (2), pp. 114-118.

Mileva, Y. M., Dallmeier, V., Burger, M. and Zeller, A. (2009) 'Mining trends of library usage' *Proceedings of the Joint international and Annual ERCIM Workshops on Principles of Software Evolution (Iwipse) and Software Evolution (Evol) Workshops* : held in Amsterdam, The Netherlands, August 24 – 25 2009. New York, NY : ACM, pp. 57-62.

Mintzberg, H. and Quinn, J. B. (1996) *The Strategy Process – Concepts, Contexts, Cases*. Third edition. New Jersey, USA: Prentice-Hall.

Open Source Initiative (2009) *Open Source Licenses*. Available at: <http://www.opensource.org/licenses> [Accessed 14<sup>th</sup> June 2009]

Pisano, G. P. and Verganti, R. (2008) 'Which Kind of Collaboration Is Right for You', *Harvard Business Review*, 86 (12), pp. 78-86.



- Pisano, G. P. and Shih, W. C. (2009) 'Restoring American Competitiveness', *Harvard Business Review*, 87 (7/8), pp. 114-125.
- Porter, M. E. (1985) *Competitive advantage*, New York: Free press.
- Porter, M. E. (1996) 'What is strategy', *Harvard Business Review*, 74 (6), pp. 61-78.
- Porter, M. E. (2008) 'The Five Competitive Forces that Shape Strategy', *Harvard Business Review*, 86 (1), pp. 78-93.
- Pykäläinen, T. (2007) 'Model for profiting from software innovations in the new era in computing', *Technovation*, 27, pp. 179-193.
- Quinn, J. B. (1999) 'Strategic Outsourcing: Leveraging Knowledge Capabilities', *MIT Sloan Management Review*, 40 (4), pp. 9-21.
- Quinn, J. B. (2000) 'Outsourcing Innovation: The New Engine of Growth', *MIT Sloan Management Review*, 41 (4), pp. 13-28.
- Ranger, C., Raghuraman R., Penmetsa, A., Bradski, G. and Christos Kozyrakis (2007) 'Evaluating MapReduce for Multi-core and Multiprocessor Systems', *Proceedings of the 2007 IEEE 13th International Symposium on High Performance Computer Architecture*: held in Pheonix, Arizona, USA, February 10 - 14, IEEE, p.13-24.
- Raymond, E. S. (2001) *The Cathedral & The Bazaar – Musings on Linux and Open Source By an Accidental Revolutionary*. Revised Edition. California, USA: O'Reilly Media.
- Riehle, D. (2007) 'The Economic Motivation of Open Source Software: Stakeholder Perspectives', *Computer*, 40 (4), pp. 25-32.
- Samuelson, P. (2006) 'IBM's Pragmatic Embrace of Open Source', *Communications of the ACM*, 49 (10), pp. 21-25.
- Scotchmer, S. (1991) 'Standing on the Shoulders of Giants: Cumulative Research and the Patent Law', *The Journal of Economic Perspectives*, 5 (1), pp. 29-41.
- Tapscott, D and Williams, A. D. (2008) *Wikinomics – How Mass Collaboration Changes Everything*, USA: Portfolio, member of Penguin Group.
- von Hippel, E (2001) 'Innovation by User Communities: Learning from Open-Source Software', *MIT Sloan Management Review*, 42 (4), pp. 82-86.
- von Hippel, E (2005) *Democratizing Innovation*, Cambridge, Massachusetts: MIT Press.
- von Hippel, E. and von Krogh, G. (2003) 'Open Source Software and the "Private-Collective" Innovation Model: Issues for Organization Science', *Organization Science*, 14 (2), pp. 209-223.
- von Krogh, G. (2003) 'Open-Source Software Development', *MIT Sloan Management Review*, 44 (3), pp. 14-18.

von Krogh, G. and Spaeth, S. (2007) 'The open source software phenomenon: Characteristics that promote research', *Journal of Strategic Information Systems*, 16, pp. 236-253.

von Krogh, G., Spaeth, S. and Lakhani, K. R. (2003) 'Community, joining, and specialization in open source software innovation: a case study', *Research Policy*, 32, pp. 1217-1241.

Watson, R. T., Wynn, D. and Boudreau, M-C. (2005) 'JBoss: The Evolution of Professional Open Source Software', *MIS Quarterly Executive*, 4 (3) pp. 329-341.

Watson, R. T., Boudreau, M-C., York, P. T., Greiner, M. E. and Wynn, Donald Jr. (2008) 'The Business of Open Source', *Communications of the ACM*, 51 (4) pp. 41-46.

Yin, R. K. (2009) *Case Study Research – Design and Methods*. Fourth edition. California, USA: SAGE Publications

## Appendix A

Below is a table with the list of questions asked in the semi structured interview conducted with individuals at each company. As the nature of the interviews was semi structured, not all questions will have a direct answer. The answers are presented in a synthesized form, formulated by the authors of this paper to give an overview of how the different companies answered these questions.

| Questions/Company  | Accenture  | Repill Linpro  | Arktekk   |
|--|--|--|---|
| - How did you expect the community to contribute                                 | <ul style="list-style-type: none"> <li>- Expected sophisticated contributions from the community as well as help with testing on different platforms</li> <li>- Expected much community feedback.</li> </ul> | <ul style="list-style-type: none"> <li>- Expected some major functionality, such as a port to solaris.</li> <li>- Expects bug reports, feature requests and help with documentation</li> <li>- Could have done a better job at facilitating contributions</li> </ul> | <ul style="list-style-type: none"> <li>- Expecting more “fancy” features</li> </ul>   |
| - What sorts of contributions are you expecting from the community?              | <ul style="list-style-type: none"> <li>- More eyeballs on code will help identify and resolve bugs</li> </ul>  | <ul style="list-style-type: none"> <li>- Community assisting in performance testing</li> </ul>   | <ul style="list-style-type: none"> <li>- Expect bug reports from community</li> </ul> |
| - How do you facilitate community contribution? How did you build the community? | <ul style="list-style-type: none"> <li>- Outsiders were able to steer the direction of the development (user-driven QA).</li> <li>- Got an instant</li> </ul>  | <ul style="list-style-type: none"> <li>- Bug reports used to steer innovation</li> <li>- Mailing lists and IRC set up to interact with community.</li> <li>- Set up user</li> </ul>  | <ul style="list-style-type: none"> <li>- Community used to review code</li> </ul>     |

|   |  |   |  |
|---|--|---|--|
|   | community through the SpringSource partnership.  | group meeting in London to engage community.  |  |
| - Do you consider the community a major contributor of business value to the product? | <ul style="list-style-type: none"> <li>- Community will allow better, faster and cheaper development.</li> <li>- OSS will prevent us from being stuck with the maintenance cost of proprietary product.</li> <li>- Product will grow with the help of the community</li> </ul> | <ul style="list-style-type: none"> <li>- Two people not employed by Redpill Linpro granted committer status and supply patches.</li> <li>- Expected community to assist with word of mouth marketing</li> </ul> | <ul style="list-style-type: none"> <li>- OpenSource has a solution for most problems</li> <li>- It is a way to learn from others and to gain know-how</li> <li>- Software becomes better as OSS</li> <li>- Software that is open for modifications becomes better if everyone contributes</li> <li>- Long term maintenance only available by opensourcing</li> <li>- Community maintenance is a selling point to convince clients to give back.</li> </ul> |
| - How many resources are you assigning to open source development                     | <ul style="list-style-type: none"> <li>- Much resources dedicated at beginning</li> <li>- Number of allocated resources reduced after</li> </ul>   | <ul style="list-style-type: none"> <li>- Employees are paid to contribute</li> <li>- Resources are sometimes reallocated to support paying customers</li> </ul>   | <ul style="list-style-type: none"> <li>- Employees are paid to contribute, but company is not big enough to have full time contributors not doing client</li> </ul>  |

|   |  |  |  |
|---|--|--|--|
|   | 1.0 release-<br>Transferred<br>knowledge to<br>SpringSource<br>resources                           | - Contribute<br>majority of code<br>- Contributors<br>also used as<br>third line<br>support for<br>paying<br>customers | work.<br>- If you don't<br>share, no one<br>will ever give<br>back.<br>- If you use<br>opensource,<br>you should<br>contribute           |
| - How are your<br>contributions of<br>strategic importance  | - Partner with<br>SpringSource<br>to avoid having<br>to maintain the<br>community.                 | - Always<br>looking for<br>opportunities<br>to opensource.   | - Code is<br>worthless,<br>knowledge is<br>the only thing<br>of real value.<br>- Opensource<br>was the reason<br>Arktekk was<br>founded. |
| - What is the nature of<br>the company's<br>contribution (new<br>features, fixes,<br>documentation etc)   | - Much of the<br>initial code<br>contributions<br>came from<br>Acenture, this<br>was expected.     | - Routine tasks<br>performed by<br>Linpro<br>employees<br>- Frequent<br>contributions                                  | - Demand that<br>fixes developed<br>for customers<br>for existing<br>products is<br>given back to<br>community                           |
| - At what stage in the<br>opensourcing process<br>did the idea of<br>complimentary<br>services surface?   | - After initial<br>release, some<br>resources were<br>re-allocated to<br>do client facing<br>work. |  |  |
| - How do you separate<br>between what should<br>be given to the<br>community and the<br>things that you are<br>able to sell as<br>complimentary | - Be a good<br>citizen,<br>contribute to<br>open source.   |  | - Not opposed<br>to making<br>proprietary add-<br>ons for<br>customers   |

|   |   |  |   |
|---|---|--|---|
| services?   |   |  |   |
| - How are you able to profit from your contribution to open source? | - Higher adoption through open source may produce more work for us. | - Developing proprietary add-ons as a differentiator for paying customers<br>- Using OSS to gain credibility and attract paying customers. | - Resources are dedicated to opensource development to get “billable” know-how. |