HELSINKI SCHOOL OF ECONOMICS (HSE)
Department of Management

EFFECTS OF OPEN SOURCE SOFTWARE ON THE BUSINESS PATTERNS OF SOFTWARE INDUSTRY

Information Systems Science
Master´s thesis
Juho Lindman k73285
Fall 2004

Approved by the Head of the Department of Management 8/10 2004 , and

awarded the grade of 90/100

ABSTRACT

Open source software is a phenomenon that has potential to change the traditional patterns of business behavior. Research committed so far has not evaluated the entire scale of potential changes, which is the purpose of this explorative thesis.

Previous literature on the subject can be divided into history of the phenomenon, explaining the nature of the phenomenon, and a more general discussion about strategies and business models in the software business. By using these theories this thesis provides a framework for analysing the entire phenomenon. The framework is put to use in the empirical part. Data consist of interviews of experts in the field. An analysis of the data is done using narrative methods.

The analysis yield eleven narratives that describe the phenomenon. Four of the narratives reveal effects. On the basis of the responses gathered, open source software can change competition environment, customer expectations, the importance of competence, and platform thinking.

Keywords: Open Source Software, Strategy, Business Model, Narrative

TIIVISTELMÄ

Avoimen lähdekoodin ohjelmistot ovat ilmiö, joka voi muuttaa perinteisiä liiketoimintaympäristön sääntöjä. Tämän laajasti ilmiötä lähestyvän tutkielman tarkoituksena on tarkastella tuota muutosta.

Aikaisempi aihetta käsittelevä kirjallisuus voidaan jakaa kolmeen osaan: ilmiön historiaa tarkastelevaan osaan, ilmiön luonnetta tarkastelevaan osaan sekä laajempaan keskusteluun ohjelmistoliiketoiminnan strategioista ja liiketoimintamalleista. Näitä kolmea eri suuntausta edustavaa kirjallisuutta yhdistämällä esitetään viitekehys, jonka avulla ilmiötä analysoidaan. Empiirinen osuus koostuu ohjelmistoalan ammattilaisten haastatteluista. Datan analyysi suoritetaan narratiivista lähestymistapaa käyttäen.

Analyysi tuottaa yksitoista narratiivia ja niistä neljä tuottaa tietoa liiketoimintaan kohdistuvista muutoksista. Tutkimuksen perusteella havaitaan, että avoin lähdekoodi voi muuttaa liiketoimintaympäistöä, asiakasodotuksia, osaamisen merkitystä ja alusta-ajattelua.

Asiasanat: avoimen lähdekoodin ohjelmistot, strategia, liiketoimintamalli, narratiivi

CONTENTS

1. INTRODUCTION

The open source software phenomenon has affected software industries all over the world. Finland has had a prominent role in open source software due to for example Linus Torvalds, who has initiated the most well known example of open source software: the Linux operating system. Currently software companies are trying to decide on how to respond to this novelty and so they require information about it.

Scientific research on Open source community has often been broadly ideological or overly pessimistic. Scarce studies are focused only on certain elements of open source software and thus lack a big picture of the phenomenon. There are not many studies on the business effects of open source software.

The needs of the actors in the market call for scientific understanding of the phenomenon. This thesis provides some understanging by investigating neutrally open source software phenomenon's effect to software industry.

1.1 Research problem

My research problem is "How does the open source software phenomenon affect business patterns in the software industry?"

This study focuses on the software industry. Part of that industry is the open source community. The concept of a business model helps to understand a company's operations and explain how they might be affected. There is an assumption that business environment determines what kinds of business models are viable. Changes in the business environment thus affect business models and company operations.

My thesis aims at improving the understanding of the business perspectives related to the open source community. I will explain the open source phenomenon's nature and show what the resulting business effects are. I will outline the connection between the open source community and software industry. Finally my aim is to deepen the understanding of how companies view open source software and how open source companies view their surrounding business environment.

1.2 Definitions of key concepts

Open source software is software that is licensed in a special way. There are several licencing ways; as an example, one of them (GPL) gives its users the possibility to use, read, modify, sell and distribute software without paying the original author anything, as long as the licence is not loosened. As the original author received no payment from the product, the secondary author also receives none. The different licencing methods are not the main issue in this thesis.

Open source community refers to people who distinguish themselves as members of this certain community. Most of these people use or contribute to open source software. Whether this one community is an illusion or not, is a question for debate.

Open source phenomenon is a term used to contain all the elements of open software product, service, and possible community. It is a wide concept used to capture all the different elements of this circumstance in history.

Finnish software industry refers to Finland-based companies selling software products or services.

Business model is defined in more detail in chapter 4. As an outline, it includes an action plan of one firm for one product/market (Rajala et al, 2001).

Strategy is defined in chapter 4. As an outline, it includes a large scale of making the decision of which market to enter and how to position in it (Porter, 1985).

Business patterns stand for action – they are commercial transactions of virtual or physical goods. Transactions take place between different agents voluntarily. Agents have different valuations of the traded product according to their expectations. In order to do business both parties must negotiate the products valuation and agree on the transaction. Some of the action concerning open source software is thus business and some of it is not. Patterns indicate that there are some common rules that govern how business is conducted.

Business environment is the environment in which the above mentioned business takes place. It gives the legislative, ethical and competitive framework for the transaction. Actions that are not business can have a direct impact to the business environment.

1.3 Limitations of study

Open source software can be viewed reactively or proactively. This thesis focuses on open source software's reactive effects. Proactive viewpoint would focus on how to use open source - for example how to use licensing with it. Reactive viewpoint focuses on what the effects of an agent using open source software are to all the parties in the market.

This thesis will not make any judgement whether to use open source or not, or whether open source is “good” or “bad”. Some agents in the markets have chosen to use open source software and thus there is a phenomenon to be explained.

This thesis evaluates the effects to the Finnish software industry structure, although this exclusion is not easy to make due to the inherently international nature of the phenomenon. The business effects will probably be similar in other parts of the world, but Finnish software industry has geographical and ideological ties to open source software. It is thus necessary to limit the scope of study to Finland to make the results more valid.

1.4 Outline of the study

Chapters two, three, and four are reviews of previous literature. Chapter two discusses history of open source and the politics shaping it. Chapter three discusses previous research and tries to identify how the phenomenon has previously been categorised. Chapter four discusses strategy and business models in software industries. Chapter five proposes a framework for this thesis.

Chapter six outlines the empirical methods used in the study and thus begins the empirical part of this study. Chapters seven, eight and nine feature analyses of data – they consist of the narratives found. In chapter ten the respondents had their say of the research. Chapter eleven summarises the conclusions, evaluates the process of making this study and outlines interesting future directions based on this thesis.

2. CHARACTERISTICS OF OPEN SOURCE

In this chapter I will outline the colourful history of open source to give some basis for further development in later chapters. It will address political tensions and the open source community of today.

2.1 History of open source software

Prehistory of open source dates back to the creating of Internet and mainframe computers. U.S. military funded the creation of the Arpanet in 1969. It was designed to exchange information between research laboratories, universities and defence contractors, creating for the first time a medium of this kind for heavy users of information technology (Raymond, 1999). In the year 1969 Ken Thompson, a programmer at Bell Laboratories, invented another cornerstone: the UNIX operating system. The following year a programming language called C was invented. These two inventions were quickly combined to port UNIX on different machines. This enabled creating software linked to an operating system, rather than computer hardware that easily becomes obsolete (Raymond, 1999).

In 1980 Usenet was formed using a direct UUCP communications protocol and fixed phone lines. In 1983 it was becoming obvious that microcomputers would sweep the mainframes and in 1982 Richard Stallman, a leading figure in MIT's AI Lab, founded the Free Software Foundation (FSF) to prevent commercialisation of the laboratory's technology on microcomputers (Stallman, 2001). Stallman's aim was to offer a free of charge Unix-clone, the GNU. Heavy users of information technology gave their efforts to this invention. This effort transferred some of the spirit of the earlier hacker communities to the UNIX-connected world as UNIX itself had been commercialised and proprietary (Stallman, 2001).

In the beginning of the 1990’s personal computers had became so cheap that hackers had the possibility to buy them for their homes. The operating system was MS-DOS, but unlike participants in the previous “network nation” using UNIX-clones, most personal computer users did not have any feeling of belonging to a certain networked culture. In the personal computer business there really was no culture of free software source exchange (Raymond, 1999). Different versions of proprietary UNIX faced financial difficulties and Microsoft-based products increased their sales volume all the time. Linus Torvalds continued in GNU’s footsteps and started creating the operating system Linux. Unlike previous open source attempts Linux offered a new sociology: a large group of people could, and would, participate in the production (Raymond, 1999).

According to Raymond (1999) general public discovered Internet in the nineties. Especially WWW boosted Internet’s growth, and it became commercially interesting. At the same time, the Unix culture was shifting its attention to the new Linux-phenomenon and learning a different way to make open source software.

2.2 Perspectives on commercialisation of software

The Open source phenomenon derives some of its values from uncommercial universities and laboratories. Mustonen (2000) has proposed that the values behind open source are closer to the university ideal of free research results than customer value.

Stallman’s FSF has a philosophical foundation: that software is not as much a priced good as it is a right of expression and use (Stallman, 2000). According to Stallman

(2000), this information should be viewed as expression and be freely distributed, modified and copied. In this case proprietary software limits these rights. It is clear that these views offer no foundation for a software industry. These are however views endorsed by many software developers and cannot as such be dismissed when trying to understand what open software is about.

In the year 1998 some proponents of open source software wanted to change its current name "free software" into "open source software", right after Netscape had announced it would open its browsers' code (Raymond, 1999). The name was changed purely for practical reasons: Business environment had gained an interest in the phenomenon, but the name label "free" lacked any business credibility. As Open Source Iniative puts it: *"The winning substance has not changed, the losing attitude and symbolism have"* (www.opensource.org, 2003). This is to say, open source software is a different name for free software, and a name the business world can adopt and sell. Moreover, OSI does not want to take a stand on whether all software should be free or not.

Stallman (2003) does not approve of the use of the term open source, since he would like hackers to focus on freedom as a value in itself, not as a means to some end. The choice of the name for the phenomenon already gives away some indication on what its user thinks about the relation of open software's and commercial usage. This debate on the term has not been resolved yet. Usually commercially orientated newspapers use the term open source software. Later on a term FLOSS (Free/Libre and Open Source Software) was invented to describe the entire phenomenon or the community. In this vocabulary free software and open source software would be movements inside the FLOSS phenomenon. It this study I will refer to open source software as the entire phenomenon. It is notable that Free software was there first and has a longer history than the open source iniative.

This political streak against commercialisation is very interesting. It says that members of the community view themselves as a subculture and are afraid that commercialisation is a threat to that subculture. The succesfull competition of software icons, such as Linux, against commercial software companies such as Microsoft and IBM, offers a joint enemy and appears to create cohesion in the community. This is not a new phenomenon, since the Unix-networks faced similar fears as early as the 80’s from personal computer users (Raymond, 1999). The subculture identity might very well be a necessary elitist side effect that offers the rewards of participating in the network. Should open source become mainstream the cultural dynamics would probably be quite different.

2.3 Current view on open source software

The Open source is in the Internet. As always, Internet is the connecting link in offering exchange of source code and other information. It is a medium used in communication and production (Raymond, 1999). Companies have imitated some of the good aspects of open source software, for example tried to create user communities with variable success, given out some source code for educational purposes, and considered possibilities offered by opening software products and reviewing business models (Raymond, 1999).

Part of the success of open source is probably due to the fact that intellectual property rights legislation is only evolving. Software markets are global and with Internet one can always reach all the countries in the world. National legislation has had some problems trying to keep up with the pace of technological and social development. Rules for the software game have evolved, but international organisations have been able to counter for example software piracy. Recently legislation concerning software patents in the EU has raised some serious doubt in open source community. The community has however been quite successful a lobbyer, preventing laws that would threaten open source software's unique position.

Intellectual property rights are however very important in software industry, since some companies prefer to resolve differences via lawyers. This is probably one of the reasons it is tempting to participate in an open source project – letting some one else take care of the legal fuss.

3. KEY ACTORS IN OPEN SOURCE SOFTWARE COMMUNITY

In the previous chapters I have outlined the history of open source. In this chapter previous scientific studies of open source are discussed. These previous studies focus only on some aspect of open source since they view the phenomenon only from one angle - only as developers, end-users, bug testers, recruitment pool, source of rivals, or content network. By combining these different aspects the phenomenon can be viewed with more precision.

Basically the fundamental two questions this thesis seeks to answer are: Who are the members of the community and what is the phenomenon fundamentally about? The scientific field that attempts to answer these questions is discussed in more detail below.

3.1 Developers of open source software

Developers are the people who make the open source software programs. Previous research on the community is focused around the question why anyone would participate in an open source project. This motivation has puzzled especially economists, since it is difficult to show a tangible economical motivation for this activity (Lerner et al, 2000).

One of the ways used by some economists to explain this “irrational” behaviour is by counting the resources needed to participate in an open source software project and then show that this countable input yields some countable output. This countable output could be for example complementary income in form of reputation or heightened career opportunities (Lerner et al, 2000).

The role of the developers can be integrated to the role of the end-users. Von Hippel (2002) summarises these two roles when talking about user innovation networks. These networks, according to Hippel, do not follow business rationality, but develop products as they go along and then offer them freely to all the members of the community. Hippel names windsurfing as an example of this kind of behaviour prior to Internet. Kollock (1999) continued to develop Hippel's ideas and proposed that Internet would have accelerated these kinds of network developments, since it is very cheap to communicate the information through Internet.

Another viewpoint is that open source community activity should not be compared to commercial software development, but to something like scientific activity (Mustonen, 2000). The difference between the two would be the same as it is with publicly funded research versus R/D activities of firms. Taking into account the history of open source phenomenon and its connections to the scientific community, this approach is probably quite correct. Holrtgrewe (et al., 2001, 43-65) have suggested that the open source phenomenon could be viewed as a counterforce to commodification. This commodification refers to a process where individual actors' efforts are turned into digital products. Decommodification would thus be an effort to prevent turning individual actors' inputs into products.

Hars and Ou (2001) have conducted a survey on some of the developers. First they divided motivations into inside and outside motivations. Inside motivations stem from the person and outside motivations are in the outside world. Inside motivations include intrinsic, altruistic and community identification motivations. Outside motivations include future monetary returns and personal software needs. Survey's results were the following: 42% were amateurs, 34% were professional programmers and 16% were paid to develop open source code. The most common motivation was the outside motivator of investment to intellectual capital (88.3%) seconded by intrinsic motivation (79.7%). This finding could offer some insight to motivational issues, if the survey is valid and reliable. The question is whether those taking the questionnaire see investment to intellectual capital in the same way as the ones making the survey.

An important thing to notice is that open source developers are not in all cases motivated mainly by consumer markets, and thus rarely have any incentive to market their software (Mustonen, 2000). Motivation for the development thus probably has little to do with mass-market end-users. On the other hand, software companies usually receive revenue from these users, their customers. This approach is not correct if the developers are working in a company that sells their product. In Finland, business users have been the largest source of revenue in software industry, and historically they have been more interested in business value than in usability (Rajala et al, 2001, 30). Since this is changing, in Finland it would probably be necessary to take more interest in end-users (Rajala et al, 2001, 30).

Looking at a company from the business model perspective, software development is usually a part of product development. It can be viewed as one way to take users or a third party into account. From this perspective open source programmers or communities could be more or less strategic partners. Althought when operating with this partner it is necessary to keep in mind that the rules of the game are different.

Developers of open source software are also end users (Raymond, 1999). Their opinion has strong marketing value, since they are respected members of their community. This community is used to operating in the Internet, so word of mouth travels fast and has a good reach. Especially in issues that are somehow related to software projects the developers have participated in.

Development of open source software is done in groups and in most cases essentially in the Internet. Programmers also try to get first versions out quite fast, so that the community of users can be used for testing for bugs. This kind of programming is potentially open for anyone to follow. The Open source community can basically aid software development in three ways: direct programming, testing, and orientation for development (Feller, 2002).

The educational aspect is two-sided: since many universities use open source environments in training, these are already familiar to developers and thus easy to approach. Another part of the educational aspect is that learning to program happens by programming and reading code. Since there is a lot of open source code out there, this would make open source easy to approach if one wants to learn how to program (Raymond, 1999). There is also a communicational aspect where developers communicate through code and make decisions on how to proceed and what is important (Raymond, 1999).

There are also other reasons, why the open source phenomenon is more than just a group of developers. The end-user role is one and the role of the powerful community is another. Also in some cases these developers are working in or can be recruited to companies. Developers also offer their source code for use and modification. Focusing entirely on the motivation of developers or on explaining the phenomenon by referring to it only as software development fails to take into account its other aspects.

Economists tend to view open source software as something where programmers can demonstrate their skills in order to get a real job (Lerner et al, 2000). It is true that open source software programmers can demonstrate their skills in this environment that enhances meritocracy (Raymond, 1999). It would however seem strange to claim that one would choose between going to college and learning C++.

One of the explanations could be that some software developers also work in software companies and thus are aware of the demand and needs of the labour market. In this way it would be very wise for companies to have a presence in the open source movement in order to find capable programmers that already know what they are doing (Mustonen, 2000).

If the open source phenomenon is viewed only as a pool, its special characteristics are ignored. Some developers already work in software companies. Most open source products do not really have any commercial potential and do not even function as good examples of clever programming. Some of the programmers do not even want to work in software companies as they consider it unethical. Some of them consider programming to be only a hobby. Most importantly, this approach sees open source as a playground and software markets as the grown up business, underestimating open source's commercial potential.

3.2 Testers of open source software

Another way to view open source communities would be to think of them as beta testers for new products. One of the remarkable findings is that some open source products are tested for bugs very fast – much faster than any commercial applications are (Raymond, 1999) .

Raymond (1999) has introduced a division into two kinds of software products: cathedrals and bazaars. The names speak for themselves, but the main difference is that in a cathedral model a developer prepares the product with proprietary care, and in a bazaar everyone is allowed to participate. The bazaar model is much faster, but the debuggers need a plausible promise of software in order to be motivated (Raymond, 1999).

This is product development from the business model perspective. Bug testing is necessary for all software products. It is an after-sales service to upgrade the product in case of deficiencies or security problems. It is also a service to let users know about these problems. After-sales services have gained momentum after it has become apparent that faulty products are launched into the markets. Open source community has probably encouraged this after-sales development by offering examples of product-to-market times that proprietary code companies cannot match.

This way of fixing bugs is quite recent. It became the norm only in the beginning of the 90’s, when Linus Torvalds presented a new sociological way to approach coding: taking many people in his project by releasing unstable versions often (Raymond, 1999).

The idea of the open source community as a way to bug fix products probably would have some commercial potential. The community could be a partner used by a coordinator as a pool of users fixing bugs.

The open source community does more than just bug tests - it develops and uses the software. Viewing a community as a way to search for bugs one ignores its cohesive role.

3.3 Rivals in open source software

Open source products can be direct rivals to software companies. In this situation, Porter (1985) would advise a software company to differentiate. This might not be necessary, if there are strong network externalities that favour the software company, or it is able to block entry by other means (Shapiro, 1999).

It is also possible that an open source software product could in the future become a rival in a certain industry. To prevent this, a software company should try to alter the market conditions to more favourable ones. One way to accomplish this would be destroying the software product before it has a chance to enter the market. Especially in the US lawsuits might accomplish this. Another way to prevent entry is to try buying the key developers of the potential rival product to one’s own company.

In some cases companies seem to have incentives to support development of complementary or even substitute products or services (Mustonen, 2002). According to Mustonen (2000) complementary products widen the market, so existing products yield better benefits. Mustonen (2000) also points out that substitute products can be an option if supporting them hurts the company a bit, but an existing rival much more, or there is some factor in the business environment that favours supporting open source software.

The open source community can also be a source of potential rivals. In this case, a company might have incentive to participate in open source development in order to control and prevent these kinds of uninvited entries.

3.4 End-users of open source software

The community has one characteristic that marketers should take into account: it has a history different from other software users. This subculture with different fractions is undoubtedly only a niche in the entire software market. Also those individuals that have contributed to the development of a product are probably not willing to substitute it with another one lightly.

Entering the customer market, licensing becomes very important. In this thesis I will not address that problematic area. It is enough to say that there are several different licenses under which software can be made and sold. The choice of licence has consequences to the business model (Rajala et al., 2001).

3.5 Content network of open source software

Von Hippel (2002) argues that open source networks are also content networks. Content networks are a communication tool. Users can post content and answers in hopes of getting feedback and advice from other users. This kind of communication could potentially offer high value network externalities and end-user value. This way of using the community has potential and is also applied by companies.

In many cases companies have not been able to control all the information posted to content networks, so also negative aspects of their products have come out. This puts pressure on company communications.

The notion of content networks is interesting, but the concept is not very rigorous. This cannot be the only way to view the community, since it focuses only on innovation – not on the total effect on business environment.

4. STRATEGY AND BUSINESS MODEL THEORIES

In this chapter I define the concept of business model, and its relation to strategy. This viewpoint is necessary to understand what the business effects of open source software are. My assumption is that business models take parameters from business environment. My theoretical framework is Porters (1985) strategy thinking applied to the business model and information economics.

4.1. Concept of strategy

The strategy tradition can be divided into two paradigms: Industrial Opportunities (I/O) and Resource Based View (RBV). The I/O-view assumes that external industrial forces in an industry affect a firm’s strategy via its managers. Strategic choices are affected by substitute products, customers and suppliers, and above all competitors and potential competitors. There are only two generic strategies that can create competitive advantage: differentiation or low-cost (Porter, 1985). The I/O-based view has its roots in micro-economics and thus gives market conditions deterministic power (Porter, 1980). RBV is also interested in competitive advantage, but is more focused on a firm's resources (for example Minzberg, 1983). The main idea is that a company should build core competencies in order to get competitive advantage (Minzberg, 1983). This view and its historical predecessors stress more clearly that, a company’s own actions determine its success (Drucker, 1954). According to Barney (1991), the following resource qualities affect competitive advantage: value, rareness, and imperfect imitability and substitutability. The difference between I/O and RBV is that the former focuses on a company’s position in the market and the latter to development of a firm’s resources.

According to Porter (1985) strategy is a tool a firm uses to choose and position itself in a market. It is a way of trying to generate and reap a part of the revenue. Strategy is also an end-product of the strategy process (Porter, 1985). This process evaluates market conditions and gives the firm a suggestion on which market to choose and which role in the market to pursue. Although strategic thinking has its roots in micro-economics, it is apparent that it is pragmatic rather than strictly analytical and scientific. It proposes frameworks and simplifications that have appliances, rather than economics models.

Porter (1980) has elaborated that first a firm needs to decide which markets to participate in and after that which strategies to pursue in those markets. There are two generic strategies that stem from the five-forces of any given industry. These strategies are low-cost and differentiation. Porter’s five forces are entry of new competitors, threat of substitutes, bargaining power of buyers, bargaining power of sellers, and rivalry between existing competitors. It is also possible to choose the scope of implementation for these two strategies.

Porter (2001) gives an example of the five force analysis by applying it on the Internet in his obituary to the new economy. Porter is able to show some trends caused by the Internet. Positive trends include increased bargaining power to channels because of new routes to customers and increased size of the market. Porter states that negative trends are problematic: the bargaining power of consumers has increased because of more easily accessible information on products, barriers of entry are reduced because of decreased sales force, more intensive rivalry is a fact because of decreased proprietary offerings, the geographical market has become larger, and pressure to engage in price competition has increased since variable costs have gone down (Porter, 2001).

When trying to understand how open source software affects basic assumptions of strategy, it is necessary to investigate the market decisions and positioning. Porter (1985, 176-177) admits that technological development can shift industry boundaries and change industry structures. Furthermore, Porter claims that it is possible that development decreases industry attractiveness by affecting the five forces. Porter gives three examples of ways technology might affect industry: it could decrease logistic costs, enhance product performance and thus widen the market, or it could increase interrelations between industries. Open source seems to have done all of the three and in order to fit it in Porter's framework it could in this sense be viewed as a new kind of technology.

When Porter discusses technological changes, there is an assumption of a company or a state making the decision to develop and implement (1985). In the case of open source, neither has the real proprietary right to do so. Open source can enter a market via a firm or some other independent agent. If a firm launches a software product to market, it has to have an incentive to do so. Porter would probably claim it would be a move to change the industry structure by changing the rules. This substitution would probably be a very good strategy to destroy rival's revenue - it would be a price war driven to extreme, and take competition to the next level.

Another option is that a software product could be complementary, in increasing market size or boosting consumer value expectation (Mustonen, 2002). Complementary products and services are those that can be used with each other, but are not direct competitors. A good example of complementary products is the Intel and Microsoft alliance. These kinds of alliances can create extra value for customers and enable new technology (Shapiro, 1999). Open source software offers good possibilities for these kinds of complementary products. The software can also be used as a threat to gain power in negotiations (Shapiro, 1999).

Open source could affect the five forces also by increasing the bargaining power of the buyers or the sellers. This could happen for example by a threat of changing the license on some product. Open source could also be seen as a source for substitute products. In this case it would mostly be seen as destructive technology that creates a new market (Porter, 1985).

Software industry produces digital goods that have some distinct characteristics controversial to classical analysis: reproduction and distribution costs are close to zero (Shapiro, 1999). This fact has a tremendous effect on the utilization of the five force analysis on the software industry. Software product companies sink their costs into product development and sales. Logistics and reproducing are not usually very costly in the age of the Internet. This means that information commodity markets that rely solely on price are not feasible. Basically, it is not possible to compete with a product by relying solely in price, if there are two or more companies using the same strategy (Porter, 1985). Product differentiation is needed to isolate profitable markets (Porter, 1985).

Shapiro (1999) has pointed out, that positive feedback or network externalities are forces strongly affecting competitive situations. The concept means that a product's value is dependent on the number of other such products in the network. An example used by Shapiro is a mobile or a fixed line telephone. These kinds of products usually have demand side economies to scale. Shapiro's theory predicts that managing the expectations of customers becomes very important, since network dominance often leads straight to industry structure re-engineering and thus to revenue. A factor that makes standards important is the switching cost. It is a cost associated with changing the product or the service provider. It is possible to entirely prevent competition by increased switching cost. This is a possibility offered by network externalities, and is favored by which companies that have lacking technology but strong market and legislative position.

4.2 Business model concept

Paul Timmers (1998) took a different perspective to strategic issues in 1998 by defining the concept of business model in an electronic commerce context. Timmers claimed it was a concept designed for electronic commerce to present the organization of product, service and information flows and the sources of revenue and benefits. Furthermore, Timmers claimed that business models rise from the empirical findings about the ways companies do business.

It is easy to see that this approach is somewhat different from Porter's. Porter made his response to this concept a few years later (Porter, 2001). Essentially he claimed that there is no such thing as a business model and that theorists use the term loosely. Porter urges to go back to business fundamentals, that is, to the strategy and positioning. Porter's main argument for back to basics thinking is the end of the "new economy". After the hype, it is necessary to focus on value and profit.

Critics (for example Tapscott, 2001) of Porter's thinking point out that Porter's companies are vertically integrated units, whereas in the Internet companies view their surroundings and industries very differently. Rajala (et al, 2001) limited the definition of business model in two categories: business model applies only to a single company and only on one market situation/product. According to Rajala (et al.) it is first necessary to outline an overall business idea to answer the questions about the product, the market and the structure of operations, only after which the right business model can be chosen (Rajala et al, 2001, 20).

4.3 Strategy versus business model

Both parties of the dispute between business models and strategy have presented good arguments for their case. In my opinion, the difference between these schools is how they view business, value, and Internet. Porter views Internet as an enabling technology and a company as a vertically integrated isolated economic unit (Porter, 1985). Business happens when companies operate on a fixed market to create monetary countable exclusive profit. These two assumptions stem from the long micro-economical tradition (Coase, 1937). Value is defined as something a company produces in the "value chain" and is based on customer needs (Porter, 1985). In this analysis value has a direct connection to the process that puts the product into customer's possession.

Business model is more focused on different aspects of value and on win-win situations and partnering. In digital goods, value does not have to have a direct link to manufacturing process, but to the expectations of consumers (Shapiro, 1999). This is because copying costs are close to zero after the first product (Shapiro, 1999). In this study, this wider definition of value is used from now on, because it offers a better framework to understand the dynamics of the industry and several aspects of value.

4.4 Business models in the finnish software industry

Rajala (et al, 2001, 38) proposes four elements for software business models:

1. Product development approach
2. Revenue logic

3. Marketing and sales approach
4. Servicing and implementation approach

According to Rajala (et al.), several things affect a company's choice of business model: timing, organisation, and the level of integration. Actors that have an effect to company's level of integration are: vendors, affiliates, customers and communities. The four elements have direct linkages and some actors contribute to several of them (Rajala et al. 2001).

In Finland this business model framework has been previously used to classify different business models. Four generic business models were found in the software industry: software publishing, application service provisioning, customized solution provisioning and open source software based models (Rajala et al, 2001). The revenue logics of open source software based models were the following eight: support sellers, loss-leader, widget frosting, accessorizing, service enabling, freeing it after the sale, brand licensing and software franchising (Hecker, 2000). So far the two most successful models have been the most traditional ones, loss-leader and support selling (Harwood, 1999).

Rajala (et al., 2001) has defined the different models: support sellers get their revenue from selling products like books and manuals or selling services like support and training, loss-leaders hope to stimulate demand for other company's products, widget frosters free their drivers and additional upgrading software products to keep them current and valid, companies that use accessorizing do not perform R/D, but offer open source product to help using the main product, service enablers use open source software to help customers use their online services, sell-it, free-it firms change the license after the sale, brand licensers free the product, but hold rights to the brand, and software franchisers sell the right to use the product name on different market areas. Obviously widget frosters and service enablers are similar to loss leaders.

It is not enough to investigate only those companies that reap revenue from open source to discover the full effect of the open source phenomenon to the software industry. Industry structures and proprietary software companies are also affected if a rival or some third party introduces a open source software product to the same market. The effect of the use of open source software is usually quite difficult to make into numbers, since there are no “normal” methods of counting sales volume or the usage reliably.

User networks can be viewed straitforwardly as a source of possible customers and thus revenue. They might also be something more. Especially open source users have historically belonged in user communities. These communities offer possibilities for communication and marketing (Shapiro, 1999). This free communication enabled by the Internet is an essential part of the open source phenomenon. A community might even participate freely in, for example, software development (Shapiro, 1999). In any case, it is crucial to understand the dynamics of the community.

5. FRAMEWORK

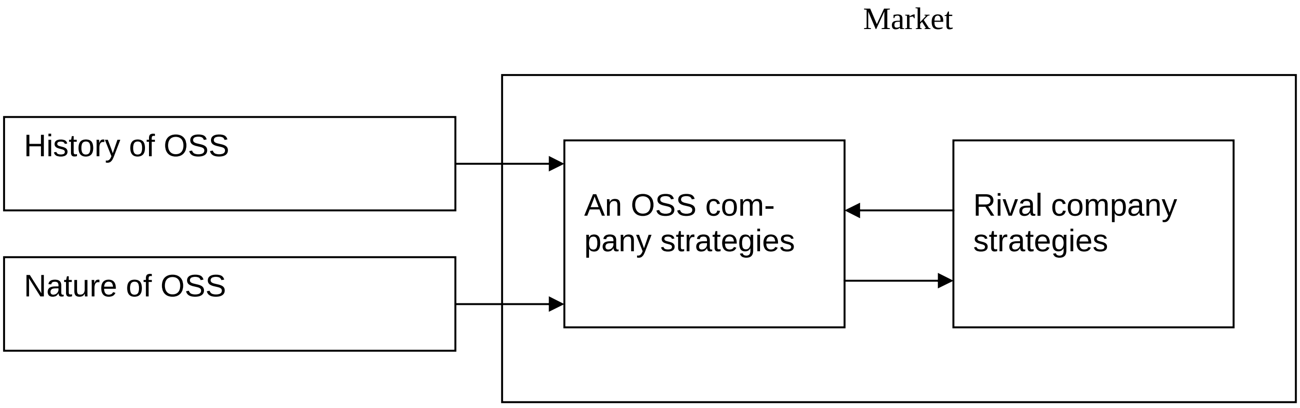
Open source software has many facets. By now it should be apparent that these characteristics have consequences to business. To give a trivial example: Who is the customer in open source software development? Is it the developer, his peer group, an “open source community”, visitors to certain web pages, the corporation that mimics functionalities and ultimately employs the coder, a different company that sells consulting services for the software, the end-user, or who? The right answer to the previous question cannot just be “all of the above” since that would not really explain the business logic or give managerial implications.

To investigate open source software’s effects, it is necessary to understand open source software. This understanding can be gained via two routes: the history of the phenomenon and classifications of open source software. These two lay the basis for understanding what the phenomenon is about. History explains of how open source software happened and previous classifications offer examples how this phenomenon has been explained before. It is notable that these categorisations have been previously defined in a way that highlights certain elements that fit current research agendas.

In the previous sections we have outlined the rules governing open source, and now we can move into exploring what the phenomenon’s business effects are. The framework is interactive – it presupposes that there is interaction in the market. This interaction is based on explaining other actors’ motives and goals. Interaction affects strategies. To give a simple example: if company A lowers price by X per cent, company B will view their future strategies and actions in this light. Probably company B would be very interested to know what the other company is trying to accomplish with this move. It is supposed that these explanations and expectations govern companies’ actions and plans.

Thus companies' actions are results of their own aims and capabilities, but the market environment interacts with these aims.

The framework is as follows: the Open source phenomenon's history and nature first have effect only to open source company and its strategies. As one of the actors in the market opts to deliver open source software, this decision has effects on all the players in the market since it affects the market structure. There may or may not be attempts to hinder this delivery. These effects are the research agenda of this thesis.



I hope to investigate what has happened and what has not happened when companies havee thought of opting for delivery of open source software. What are the strategic decisions and consequences of these decisions? The aim is to explore and gain understanding on the changes, that open source software brings into business patterns – an understanding that is based on facts and real-life experience of the experts.

6. METHODOLOGY

In the four recent chapters I have outlined the theoretical field and proposed a framework. I have also highlighted theories that provide answers to the research question. In the following five chapters I will put these theories to use. This chapter is about the methodology and the following three about the analysis of data.

6.1 Philosophical foundations of the study

The entire open source software phenominon is actually a very recent point of interest in business information systems research. It is science at the frontier: there is a phenomenon but no good theory to describe it. Much of the research done in the area is poorly documented and committed by non-academics. Thus it is impossible to use a framework without examining the underlying assumptions.

This perspective is different from the one used in most previous research since it has tried to capture a part of the phenomenon inspected in the research. Thus the results underline for example the perspective to writing code or bug testing programs. I do not hope to propose a framework applicable to larger parts of society or in analyses of our way of life. I am only interested in open source in all its variety and hope to help understand how it changes the business patterns that limit companies' choice of business model and ways of doing business.

My analogy is that there is some kind of a more or less static market for products and companies that compete in the market. The market is defined widely, but essentially there is demand for products and companies that choose whether or not to produce and on what price, according to customer valuations. This way a new software product can satisfy some demand and possibly create revenue to its producer in return. There are different agents with different business models on this market. As defined earlier, the business environment comprises of the factors dependent on the market structure that affect the choice of business model. Namely, they depend on competition and ultimately on individual customer preferences.

I assume that prices are negotiated in the market, that there are customers in the market with different tastes and expectations and that these expectations may change, that companies and customers have asymmetric information of the market and that they make individual decisions. I do not assume perfect competition, rational customer preferences or thus profit maximising agents. I assume that companies in the market need to make profit in the long run and that they select their business models to fit the market environment. There are also agents that do not act rationally by maximising their gain – they do not seek revenue at all or seek revenue by different means. Their aims may be different: some may be in for political reasons, others view their actions as goodwill, scientific research, hobby or personal ambition. These different agents change the market conditions and their effect to firms that compete must be examined carefully. I assume that it is possible to gain knowledge of historical circumstances and thus alter agents' behaviour. In this sense agents and markets are real. This information can be gained in the form of narratives that describe how companies have operated, operate and how they should operate. These narratives are also real, but in addition to describing they categorize, predict and maintain themselves – they create the history, organise the present and recommend future.

6.2 Narrative approach to the phenomenon

I have chosen the narrative approach to shed light on my research question. Since narrative studies are quite rare in information systems research, it is necessary to explain what the method is and what it supposes. Case studies are the most common approach in information systems. They are especially useful in investigating contemporary phenomena when the context and the phenomenon are difficult to separate (Yin, 1994, p.13).

Narrative research is any research that uses stories loaded with meaning as a tool to explain phenomenon (Burr, 1995). Stories are reports that describe events. These stories are expressed via language, that is, signs that transfer meaning. But these narratives are not just objective reflections of reality - they also shape the events they describe (Burr, 1995). They create order and load stories with values (Burr, 1995).

Narrative research is interested in the world of meaning. It relies on the assumption that by analysing how stories are told and what they say and do not say, we can discover something about the world they describe (Burr, 1995). There is an assumption that the world consists of social constructs. These constructs are by nature social – they are created and upheld in social interaction and practices. They are used to explain and to give recommendations. Examples of social constructs include open source, the open source community, companies, and market structure.

On a more detailed level, stories are a very good way of explaining companies' actions, since stories shape the social world of the decision-maker. Decisions are made based on how the agent understands the situation and thus acts. Basically, companies are not themselves actors, but their acts consist of the acts of the people who belong to the organisation. In this study there is an assumption that companies also have some kind of an understanding on the surrounding business environment and that companies are agents. When trying to explain how companies understand their surroundings and act, it is necessary to study how their key decision-makers view the world and how they see their options.

In social sciences, especially in social psychology and sociology, there has been a growing interest towards narrative research. This means that social scientists have been influenced by liberal arts such as philology and sociolinguistics (Polkinghorne, 1988). The pioneers of narrative research came from ancient Greece, but cultural studies and literature studies are the modern equivalents of this tradition. The most influential branch of this science was the French scholarly movement of narratology. Followers of the movement such as Claude Levi-Strauss and Alan Greimas searched for hidden preliminary constructs and the universal grammar of stories (Polkinghorne, 1988).

In literature studies the researcher is interested in the narrative's relationship to genre and tradition, not to the surrounding reality. But the basic assumptions and methods of narrative research are taken from this field of research. Sociolinguistics are interested in how the stories told by ordinary people reflect the way these people understand the world. Social psychology has been interested in 1) how the narratives are formed during social and interactive processes and 2) how the actors produce their identity using these narratives .

In the sixties the historians became interested in autobiographies. In their book Metahistory Hayden and White (1973) claimed that historians always view the historical events as having a narrative form. These forms would have been evaluative and thus history would always include ideological aspects. Later this critical view has been introduced to other fields of science – for example some members of the movement of social constructionism view scientific theories more as rhetorical constructs than objective accounts of reality. Also philosophy and psychology have been interested in narratives. Continental philosophy focuses on a wider relationship between a story and real life. Psychology is focused on narrative thinking. They view stories as a fundamental way of gaining information to explain and predict human behaviour.

In social sciences autobiographies have for decades been a very influential method of gathering information. Originally these stories were viewed only as referential presentations of the social world subjects had participated in. In the eighties the focus changed: the question was now how these stories were ordered according to narrative form. The order could be viewed as limiting the realistic explaining power of the accounts. On the other hand, these accounts can be seen as cultural reality; the norms and values of the social organisation can be brought to light by pointing out what kind of narrative constructs these autobiographies contain. (Bruner, 1986)

There have been attempts at narrative approach in several other scientific fields - in information systems research not very many yet. Qualitative case studies in information systems research have traditionally been divided into three groups: positivist, interpretative, and critical (Chua, 1986). The last two cannot be outlined very clearly, but in this study elements are taken from the last two perspectives. This study is interpretative, as it aims at understanding a complex social phenomenon, and critical, as there are clear connections to power, resistance and dominion. The main point of this study is to understand, not to emancipate.

Narratives have been used in information systems research to explain social phenomena. One of the pioneering articles on using several qualitative research epistemologies is Lynne Markus' (1983) article Power, Politics, and MIS Implementation, where she points out why there are implementation problems and how resistance could be understood. The article opened up a new branch of research and had several followers that applied critical and interprative methods (for example Myers, 1994).

Another example of a textual approach is Lee (1994). He showed how electronic mail was a medium used for rich communication by some users. It was not inherently a poorer way of communicating, but the richness depended on the interaction between the email medium and the social and organisational context. Although in email some cues were lost due to the lack of face-to-face interaction, it contained elements that made richer communication possible. Management science has also produced a number of interesting studies that use narratives. For example Palvia (1995) has written about information systems and outsourcing by using a dialectic approach.

Much of the research involving complex social dynamics benefits from narrative approaches. It is actually quite surprising that there have not been many serious attempts in the field of information systems science, when both information and systems are so deeply rooted into text in our society.

In this thesis the approach is first down, then up. First I look at the details of the phenomenon. Only after that it is possible to try saying something general about it. The way to achieve this is to look for common narratives. These narratives will be produced by the actors in case companies. I will try to find if there are narratives similar to those used to explain open source software in scientific literature. The aim is to find the business effects of open source software.

The way to accomplish this is to look at some cases very closely. It would seem impossible to gather numerical data purely in laboratory form. The explanation for the phenomenon would thus include some qualitative elements. The main question is whether to use some quantitative data also. In this study the essentially pragmatic nature of the theories relied on supports using qualitative methods. The scope of the phenomenon is such that it would be difficult to explain by statistical methods. The explorative nature of the study also recommends qualitative approach. In this study, I will thus gather and base conclusions on qualitative data only.

The method of data gathering to get information on the narratives is structured interviews. Respondents are people who have both business and open source software experience and who currently work in software companies. This results in them having the necessary background knowledge of the phenomena and industry. There is the risk that the respondents might be biased to favour open source software, so to ensure the neutrality of the study, this possibility has to be taken into account in the selection process of the respondents and in the interpretation phase of the study.

6.3 Research questions

The question I seek to answer is “How does the open source software phenomenon affect business patterns in software industry?”. To explain open source software’s business effects, it is necessary to group the questions raised by existing literature. In this paper those questions are grouped into three categories: history, nature of the phenomenon, and business effects.

The historical part is necessary to understand how open source software has evolved. I will ask the respondents questions that help to discover how they see the history and the current and future trends of open source software. I am also interested in whether the answerers talk about the streak concerning the debate on open source software and software business.

The earlier literature on the nature of the phenomenon raises the question: What is the phenomenon about? How is it described? How do the respondents talk about it? What is their relation to the community - are they members? I will investigate how they limit the scope of the phenomenon and how they view their relation to it.

The effects on the business environment are the main concern in this study. In order to clarify what this question is about, I want to investigate a certain circumstance: What happens when an open source software product enters a market that has competition. How does it change the business patterns of the industry? What are the business rules of open source software? These rules can be described through answers to following fundamental questions: Why does product enter the market and what does it aim to do? How do incumbent agents react to the product initially and later in the long run? How do their strategies change? How do consumers react to the product?

It is not plausible to ask these questions directly, since it would give quite direct cues on what answers the researcher expects and how the answers would be used in the study. Therefore, it is necessary to ask these questions indirectly. An open discussion also offers a higher possibility of revealing the narratives. In order to be ethical it is necessary to give the general subject of the study and the business viewpoint.

In addition to this, the respondents will have a possibility to comment on the findings before they are published. These comments will also be taken into account when writing the final version of the study, since they are the best criteria on whether the study has revealed the narratives correctly.

6.4 Data collection

I chose to make three different interviews in three different companies. These companies would have to have nothing in common, but some experience on open source software. Thus the workload would be manageable, and if everyone would share the common narrative the results would be viable.

The selection of respondents was critical, since only few interviews were made. That means inferences of the entire population were made based on only a very small sample. The outside validity of this study could be questioned if this process was not handled properly.

I selected three companies and approached them. All the companies are partly Finnish in origin as determined by my exclusion of the scope of thesis. First, I was only going to focus on software sellers, but after the first interview I expanded the range to include also software service selling, where open source software would serve as a complement. The companies operate in different branches and use different revenue logics.

Company A was a small Finnish software company, Company B was a quoted international company and Company C was a large international company. There were also other options for the choice of companies, but since these companies use open source software in different ways and apparently have nothing in common, they were selected. All three companies answered favourably to my approach.

Interviews in companies A and B were held in person, the third “interview” happened via email. It was considered whether the differences in data collection had an influence on the results. To make sure that this did not happen all of the interviews had to be in line before conclusions could be drawn. One of the interviews was held in a group and the others as single-person interview. These three different modes of data gathering are together stronger than if all the interviews would have been made with same formula, since the mode of gathering the data does not seem to effect the results.

The interviews took place in a quiet environment and were taped. They lasted for about 45 minutes. For preparation, the respondents were told that I was looking for people who have experience in both open source and software business. Furthermore, I disclosed that I was searching for business effects and that I was interested in open source software. I asked the tematical questions in a non-leading way in order to guarantee that it would result in the respondents’ discourse, not the researcher’s. I was also prepared to ask for clarification, if something interesting would pop up.

The group interview consisted of three people and the interviewer. It was held in company A. The respondents participated lively. In interview B there was only one person, but this interview lasted over 45 minutes. To make the interview in company C I prepared open questions in a form and sent them to the respondent. The cover letter and the form can be found in the attachments (APPENDIX1 and APPENDIX2).

In the beginning of the two interviews I introduced myself and gave the above mentioned information on the research. Then I asked the interviewees for some personal history or positions in the company and their experience in open source, if these were not already known. After that I asked questions of company background and history. I was especially interested in open source software and asked for clarification when the topic came up. Then I moved to their point of view on open source and tried to find out how they see possibility of competition with it and how their customers see it. I asked for their conceptions of their competitors' views on open source. In the end I asked if something was left out and if there were other comments. Since the respondents had a lot to say about the business effects, most of the time was used to talk about them.The introduction, company background and general history parts were primarily used to gather data on the first two research topics and the rest of the questions were used to elaborate the third question on business effects.

The interviews were conducted in Finnish and where translated into English word for word according to the taped interview. The translation was done rigorously – the meaning of the words had to be translated also. It was considered whether this would have an impact on the results, but as Finnish was the native language of most of the respondents, it would have been biased to interview them in English.

The three interviews succeeded and no obvious threats to reliability were noticed. The only possible problem to internal validity was that all the respondents already had experience on open source. The other possible sources of errors are discussed in the last chapter. The interviews yielded about 100 minutes of tape and three pages of answers to the questionnaire. This data was the basis for further analysis.

6.5 Grouping of narratives

The narratives I found can be grouped into three categories the same way as in the literature review: narratives concerning the history, narratives concerning the nature of the phenomenon, and narratives concerning effects to the business. The analyses were conducted by writing down the interviews on a computer and after that selecting when the different respondents used same kinds of narratives. All the eleven narratives found are described in the next chapters. After the analysis a draft version of this text was given to the respondents and their comments to it were included in chapter ten.

The respondents were familiar with the history recited in the history books and hypertexts for example by Raymond (1999). They also had views on the theme of commercialisation of open source software. There were striking similarities on how the respondents told the story of open source software. These elements concerning the history and status of open source were grouped into three categories: utopian, hostile, and heroic. Respondents used utopian narratives philosophically and these narratives concerned freedom of speech, different kinds of rights and ideals regarding future society. Utopian narratives were interesting, since they tell how respondents see the world and how they would like it to be (Kottak, 2000). Respondents used hostile narratives when talking about the “enemies of open source”. Heroic narratives were used to tell the history of the heroes of open source software. These two narratives were derived from Claude-Levi Strauss’ (Kottak, 2000) theory of binary opposition, which claims that humans understand the world by dividing it into two camps, and by using a narrative where the villain disrupts a balance causing a conflict and hero is trying to resolve this conflict.

Narratives concerning the nature of the phenomenon took a stand on two variables: who is a member of the community and what the community fundamentally is about. These narratives were grouped into four categories: ingenious, elitistic, participatory, and common cause. Ingenious narratives featured one coder-genius, who is able to design and control superior software development. In elitistic narratives a small elitistic group knows best what software is and should be. Participatory narratives had the development process open to all who are willing and able. In common cause narratives members of the community participate in the development to be part of and create something greater. These narratives told directly how the respondents view their community, who the members are, and what the respondents want to identify with (Kottak, 2000). The different earlier research categorisations of the community had selected some stand to this basic question. The answers provided by the respondents show that this previous picture is a simplification.

The narratives the respondents' used in describing the business effects were a mixture of scientific theories, consultant jargon and tried-and-true business practice, but clearly showed how the respondents view their business environment and what their decisions were based on. Porter's (1985) ideas mixed with Timmers' (1998) in the answers when the respondents used words like “strategy” and “business model” as a part of their everyday language in describing how their companies operate and how actual decisions are made. The narratives were grouped as: competitive effects, customer expectations effects, competence effects, and platform effects.

Respondents used the competitive effects narrative in describing how open source software interacts with companies, restricting and enabling different options. This narrative was found in some cases to follow Porter´s (1980) idea of generic strategies: differentiation and low-cost. The narrative was firmly rooted in the thinking of industrial opportunities school. The customer expectations narratives depicted changes in the expectations of the customers and how these changes determined the alternatives companies had in their disposal. This line of thinking followed quite directly from Shapiro's (1999) remarks on the characteristics of digital goods. Respondents used the competence effects narrative to tell what the changes in the necessary resources of the companies are. This line of thinking seemed to follow the resource based view on a company. The platform effects narratives were used to describe open source software's role in creating new markets by acting as a platform. These views seemed to follow the thinking of the business model theories, since they did not assume strict fixed industry structure or the vertically integrated company.

7. NARRATIVES OF HISTORY

I was looking for three different historical narratives: Utopian, hostile and heroic. Utopian narratives were known to all the respondents, but they had different stands on how relevant they were. Hostile and heroic narratives were used by all the respondents. Linux and Microsoft were given as the typical hero and villain of open source and proprietary software.

The colourful history and the conflicts between commercial and uncommercial were known to all the respondents. There was no clear separation between free software and open source software – it would appear that they are used interchangeably. All the respondents made a separation between open source software and proprietary software – furthermore there was an agreement that they were qualitatively something different. The nature of this difference was however not agreed on. All the respondents agreed that open source software had considerable merits.

7.1 Utopian narratives

Utopian narratives concerned philosophical questions about freedom of speech, rights and an ideal of the future society. This utopian narrative was not present in all the interviews, so it is impossible to draw conclusions based on it. The development of this kind of thinking was described by a respondent of company A: *“Originally Linux had been just a tool, but later we became interested in the philosophy behind the free software movement.”* The content of this philosophy was described as follows:” *Information and the access to information should be free. Since we develop products for public sector that enable access to information, it is imperative that our customer institutions can offer free access. This access cannot be proprietary to any one company, since it would affect the* *content of the information."* Later on there was a statement: *"However we are not selling our ideology, we are selling our product – and we are selling them because they are good. They are good for example because of our ideologies."*

In company B the meaning of open source was described as follows *"The main point in open source software is working together. By working together we are able to create tools that each individual wants to use. I do not see proprietary software and open source software as ideological competitors – they are just different ways of doing things".* Respondents in company A would probably also accept the claim that working together is what open source is about. The respondent in company C claimed that the meaning of open source community has changed: *"It used to be a gathering of software developers who wanted to share their work with others. Today it is a collection of people willing to take benefit of the community and the original developers."*

7.2 Hostile narratives

Hostile narratives concerned the "enemies" or "villains" of open source. There were clear value judgements that were not in favour of these villain companies. In this study, the author makes no value judgement against any company, but the respondents' common disaffection of some companies, namely Microsoft, is a very important characteristic in understanding what open source software is about. It is not a trait of an individual – it is a trait of a community.

All the respondents were familiar with the same enemies. Some of the enemies were direct rival firms, but most of them were not. However, the disaffection to these companies was shared. The respondent in company C stated: *"What would be better market-*

ing than the Microsoft guy saying: "That software is nothing." In company A the same company's marketing efforts were discussed in more detail:

-If for example some schools think of starting to use Linux, then Microsoft arrives giving away Windows. And in a few years, after you are hooked, they'll hit you with licence fees.
- In my opinion, that resembles closely the way drugs are sold.
- The first one is always free. By the way, the same tactic is used by our proprietary competitors.

7.3 Heroic narratives

Heroic narratives described the heroes of open source software. Linux was a common example to be used. These narratives were in most cases linked directly to Microsoft. The Cathedral and the Bazaar by Eric Raymond (1999) was mentioned in two occasions and it seems that it has had quite an effect to open source community.

The role of media came up in all the interviews. The company C described open source being a "pet of the press", but warned that this might change in the future. Company A was a bit more cautious, as they said that they had faced prejudices when selling their products. All in all, the company had adopted a media strategy that always underlined that they were an open source company, as thus they were viewed as a separate group against all the companies that offer proprietary software. *"This distinction is good, since a small segmented group gets a lot more attention compared to a bigger group."* Respondents agreed that there had been media campaigns against open source software, *"of the hearts and minds of the users"*, but we return to these when we talk about the business effects.

All the respondents knew very well the story of Linux and it was apparent that it had touched all of them. It is quite a good story about how a very small but smart community can challenge the world's biggest company in a battle for a "market share". The story is quite sympathetic, although it is probably the oldest story in the world. It is very important to understand that many of the members of the community feel like they are in league with Linux – and thus against many international software corporations. Not all the respondents viewed themselves this way but this narrative was present in all the responses.

8. NARRATIVES OF NATURE OF THE PHENOMENON

The nature of the phenomenon means the different aspects of open source software. The main idea was highlighted by two questions: who is a member of the community and what the community is like? All respondents viewed themselves as members of the community. However, they had different ideas on who is a member of this community: company C viewed itself as a member of the community, whereas Company A would probably not have viewed company C as a member of the community. The relationship between the developers and the community was not unclear – all agreed that developers were part of the community. The question was who else was a member and on what grounds?

The different narratives were ingenious coder, elitistic group, participatory development and common cause community. These are simplifications but as such quite interesting – they were also in one way or another all used by the respondents.

8.1 Ingenious narratives

Ingenious narratives were narratives that focused solely on coding as work of a single genius. There were some remarks were the respondent followed thinking of this kind – although there was also the idea that the work should be shared among other users. For example, a respondent in company A stated: *“As a software developer, sometimes there is a specific need. And that need is very personal. If you have the skill and the source of the software you want to change is open, then you can go and change it”.* The respondent in company B put it in a similar manner: *“The classical open source developer does things from his/her own interest and for his/her own use.”* This is a very interesting narrative for future study, since it seems to be bit different from the narrative that focuses on communal information.

Some of the ingenious narratives were probably used because the interviewed were experts in their field and thus had better understanding of the matter than regular members of the community or their customers. The respondents were able to solve technical problems in their field easily and thus had insight on what is possible, necessary and worth the effort. On the other hand, they were also at the same time users of the software and developers, so they could tell what piece of software is good and what is not.

8.2 Elitistic narratives

All the respondents agreed that there was an elitistic or achievement based meritocracy in the community. The company C respondent identifies this group, and states in his response about philosophy of open source software that he believes that there will be changes:

“Open source will probably divide into “Business Guys” and “The Original Developers”. As OSS companies evolve, they will hire more conventionally thinking people. Thus the community will move further away from the original ideals into a “grey area”. These companies will be despised by pure open sourcists.”

There is however an interesting counter-narrative to this narrative. company A’s respondent states that *“Often in proprietary projects software is developed in secrecy. The coders have some great vision of who they are selling the software to and what it is for.”* There is a clear idea that the users know best what makes a program good – not the company that makes it. Thus, open source would better take customers wishes into account and adapt to them, unlike as noted by the same respondent, *" some murky corporation from Redmond".*

8.3 Participatory narratives

All the respondents agreed that some of the open source software invites participation. The problem present in company A was that users' companies were initially not very interested in participating. Especially the most necessary feedback of the "normal user" was often very difficult to get. This was problematic, as it would have been very valuable to the company. A respondent in company A stated:*" It is possible to create this kind of a joint project only if you let people see that their response has some effect on the software".*

Another respondent in the same company continued by telling about participation in their open source software product: *"There was a lot to do with our software before it would have been ready, but we opened in a very early stage. We were able to give plausible promise and thus received a lot of valuable feedback. This resulted in a quite different end-product."*

8.4 Common cause narratives

Common cause narratives told a story of the members of the community participating in creating something together. The respondent of Company C did not talk much about this, but from his/her answers it was possible to determine that at least there had been this line of activity. It was apparent that there was this kind of a spirit in the community – and that all the respondents genuinely believed in open source having a lot of potential in the future. This is quite a good sign, since as there is a community of people who think this way, there will probably be some more upcoming commercial applications as well.

Company B's respondent calls common cause narratives *"classical open source development"* and admits that he has contributed to several open source projects according to his personal needs and preferences. The ready-made software did not include the necessary elements in these cases. He also states that in the future open source software might have some possibilities in multifirm software development.

A respondent in company A states that currently this concept behind open source is not very well understood: *"People still have a need for electronic property, for example Microsoft Windows, Microsoft Office and so on. People want to own their part of this allmighty Microsoft thing. When I install Linux, I do not think I am installing Linus Torvalds Linux, but Our Linux. I view it as common property of all of us who have participated in coding it."* These narratives are interesting, since they contain a very different view on digital ownership. My material is too limited to make inferences this large and it is not the aim of this study. These, as well as previous narratives, support the proposition that there are direct effects to a business environment. In the next chapter I will review them.

9. EFFECTS ON THE BUSINESS PATTERNS

To understand the rules of the open source software game, it is imperative to understand how the decision makers view their business environment and how this environment changes when there is the possibility of an open source software entry. The narratives describing open source software's business effects were grouped into competitive effects, customer expectations effects, competence effects, and platform effects.

All the respondents agreed on open source software having characteristics that have the potential to severely change business patterns. Whether or not open source software is able to change the patterns depends on the characteristics of the industry and the actions of other companies. Respondents outlined their opinions on what companies have done and could do in these situations. When asked about the effect to industry structure, respondent from company C put it quite well: *"(Entrant) will either totally change the industry or have no effect at all. It depends on the state of the industry."*

There was a clear tendency to talk about services, not products, among all the respondents. Partly this can be attributed to the fact that their companies sold services – but it is also apparent that they did see the future of software business in services rather than in commoditised products.

9.1 Narratives of competitive effects

Competitive effects narratives explained business effects by competitors' actions. There were clear market orientated narratives – in fact, these were the prevailing ones when talking about business effects. All the respondents talked at some point about open source software that is in competition with proprietary software. This underlines the importance of understanding the changes in the rules of the business environment. It also verifies that this phenomenon is worth the research effort.

All the respondents agree that proprietary software cannot compete very long in the same market with similar open source software. There are several clear reasons. The obvious one is, as respondent C put it, *“the business will have a fierce price war, where profits disappear.”* The respondent from company B describes the dilemma:

”I have looked at that (an entry of open source customer software) from a proprietary company’s viewpoint and it was a quite difficult situation. The question was: How could we differentiate, when the open source software was able to mimic all the features on the long run. Basically we could have put more money in R/D and try to develop more features and such or then we could have gone with the open source movement. Neither option was a very good one.”

This quotation speaks for itself: a company cannot go into a price war and mimicking makes differentiation in some cases impossible. The software companies try to do everything in their disposal to not get into this situation. The most obvious way is to block entry. There are several ways to accomplish this. Respondent from company C described some options: *“Hire the best developers from that rival to kill the project. If that cannot be done, then prepare to start an uphill battle. Microsoft and other large US companies have used FUD (fear, uncertainty, doubt) to diminish the entrant, but that has actually given entrants more visibility”.* A company A respondent forecasted similar approaches: *“Some people have believed into these FUD-buy talks”.* Another respondent continued: *“Microsoft has announced that open source is like a cancer that is spreading.”* This FUD-process is directed at customer expectations and will be dealt with in the following chapter.

The respondents agree that the development towards open source software products will not happen in all business environments, since it puts quite heavy demands on the open source community. This kind of development requires a commoditised product and many developers – all the respondents did however agree that, for example, operating systems and some other most used programs will probably be open source in the future. Respondent of company C predicted that *“Open Source will erode the great margins of the widely used software packages and will force the providers of those packages into service business.”* On the other hand, all the respondents agreed that proprietary software products would not disappear, since open source software products need some special circumstances – at least quite many users and developers.

The customers’ needs affect whether there is a possibility of an open source software entry. For example, a respondent of Company B stated that since his company operates in an environment where trust to the company is the key, products cannot fail, and price is not that important, there was a very small chance of an open source entry. The only possible threat would be that some of the competitors together or alone would open their software, but there would have been no gain for anybody should they have done so. They could only lose their customers and revenue.

The respondent of company C stated that the probability of an open source software entry depends on how the companies in the market create their revenue. If their revenue creation is based only on the product, then there is a high probability of entry but if their revenue creation is based on services, then an entry is less likely. This is because a product can be more easily commoditised than service.

The respondents use the concepts locking-in and locking-out quite freely – it was not apparent what things they meant by them or what exactly caused the locking-in or locking-out. All the respondents had in mind that locking means forming a relationship between the buyer and the supplier, and that there are consequences to the buyer making similar acquisitions next time. These consequences could be for example financial, compatibility issues or expertise issues.

One of the strengths the incumbent companies have and will use is the lock-in by expenses - once you buy something from one supplier, it becomes more costly to buy them from some other vendor the next time. Incumbent companies often try to lock the entrant out. Even by giving away some of their software in order to take down rival companies. A respondent of Company A describes this process against their own open source software company:*" We tried to create our own market with open source software and selling service. Our competitors responded by starting to give away their software."* The company had anticipated this and was able to survive – it had spotted an existing market and entered it, forcing incumbent companies to react. And when the situation has gone this far, the incumbent companies do not really have very many good options left.

Locking-out is a good barrier of entry, but it is not unproblematic. As all the respondents agreed, there is less lock-in also in open source software. Since the source code is open, it is possible to continue the development of the program without the supplier or with a different supplier. This makes open source software easier to sell to customers who are becoming more and more cautious of lock-in. Company A also tells the other side of the story: customers become locked-in also with open source software – once the client has selected open source software, it is difficult to change back to proprietary. The users do not get locked in to a certain product – they become locked in to open source software.

But basically it is possible to evade lock-in to one company, if there are several open source vendors operating on a same platform. There is one catch here too: assuming that software product is evolving and there are several developers that sell customisation and one company that coordinates the project. If a company does not buy from the coordinator, but some of its rivals suppliers, then *"how can he be sure that he can use the bought customised features also in the future versions?"* underlines the respondent from company A.

Company C's respondent also urged all the companies in software business to take their rivals seriously: *"And make sure that your software is of high quality and your company is considered ethical, so people don't want to take you down. If your software quality is low and development cycle long, then there will be someone challenging you, and that might be an open source competitor".* All the respondents agreed that no companies are above competition, no matter how well they think they have locked-out their rivals.

9.2 Customer expectations effects narratives

As derived from features of the digital products, customer expectations became very important in the answers. All the respondents agreed that customer expectations determine what solutions are commercially viable. These narratives were called customer expectations effects narratives. The respondents took customer expectations as granted – they implicitly assumed that expectations direct customer behaviour and are thus important to companies. This is true also for FUD already discussed in chapter 7.3.1.

All the respondents agreed that media coverage has some effect on the sale and thus expectations of the potential users become important. Respondent C even stated that *"Today Open Source is pet of the press".* The role of images and media is important, since it takes competition to a new level. Products and services do not compete only with differentiable features and solution quality, but also with images and reputations. This viewpoint is especially interesting when talking about network externalities; as claimed in previous literature (Shapiro, 1999) plausible promises of the future directly affect the sales of today. FUD can, in this sense, be viewed as a way of discrediting the open source software in the beginning and deliberately block its development. All the respondents agreed that incumbent software companies have a good motivation to stop "the cancer from spreading".

Respondents had differences of opinion on who knows best what the customer want. Most of the respondents assumed that customers knew what they wanted in most cases and that this information should guide the development of software products. The way to accomplish this was not discussed very clearly – there were some cases when the elitistic narratives came up. In these cases the customer needed guidance in knowing what he wanted. The helpless customer and the omniscient customer were used in several occasions and in some cases only few sentences apart (from company A): *"If the customers do not understand what the benefits of open source software are, and you bring up that we have this open source software, then they don't understand what you are talking about. They go: What is this, is it not 1-2-3 and we have the operational software?"* as opposed to *"Open source software works in such a way that straight from the beginning, we release it. The users, I mean the real users, can take part in it, and see it."*

In some industries the proprietary companies were seen to have best information on the different customer needs. These needs were categorised differently by different respondents: some talked about technological needs and others about business needs. In some other industries the global reach, high level of penetration and the possibility to develop the product according to tastes were seen to answer better the customer needs.

Another aspect all the respondent brought up was quality. All the respondents used the word quality, but they used it in several different meanings and they apparently had different opinions on what this quality is. All the respondents agreed that quality was something a customer perceived.

A respondent from company A saw that response from the clients made quality possible. He told: *"Often in proprietary software development they live in a basement and tell nobody anything. They possibly take some guys to the basement, quickly sign the NDA's, even without telling what is happening. Like it was something revolutionary they are making. Then, after years of careful craftmanship, when the product is ready, they climb up and release a ready product to the market. The attitude is called We Know What is Good For You. And then they'll sell it, or shove it if necessary, since now they have something great – no matter what the customer really wants".* Open source software development is thus done differently and is able to satisfy the needs of the customer better, as it is done much faster and feedback loops are much shorter.

The other view given by respondent B is that software companies basically know what their customers need and are thus able to produce products of better quality. This is true for those software companies that have a large contact network – their salespeople see the problems of the customers from a larger perspective. The respondent from company B urged however the proprietary companies to keep up this competence – in the long run, open source rivals could very well catch them up. There is also the question of testing – it is much more demanding to do open source software development on a critical system, since it really cannot be allowed to crash in order to produce the necessary feedback to programmers. Respondent B also highlighted the need for product quality, but also the support services quality.

In company A there was some controversy about communicating with clients – although the company underlined in its media relations that it was open source software company, it did not automatically tell this to its customers. In some cases it was thought that it does not really matter that a customer is buying an open source product – *“it is no big deal”.* Company B that uses an open source platform in its main product, is quite hushed about it. This is quite interesting, as it tells that there is some prejudice or at least lack of information concerning open source software. Perhaps it means that some of the FUD has some basis or at least it has been thought to be credible by the customers. And the customers buy the products or the services that create the revenue for these companies.

Credibility in the eyes of the customer was another thing the respondents were interested in. Company A stated that originally one of the reasons they started making open source projects was that it provided them with credibility. Their competitors were many times larger and had a lot of resources to use in marketing. *”We got credibility from the open source community. It was a fact that we were international and our product was used all around the world by thousands of users. The product was backed up by companies and institutions.”* On the other hand, a respondent from company B stated that in their product market, the potential open source companies would have a very hard time becoming credible. *“We are facing all the time the most important issue – the issue of credibility. The company behind the product has to be credible”.* Since the systems sold are critical, the companies sell security. The community could not guarantee this security, since the customers trusted much more the companies, according to the respondent. The credibility thus depends heavily on the characteristics of the product, the credibility of competitors and what is necessary to become credible in some market.

Several respondents took up the question about the price of open source software. These narratives as such were not very interesting, unlike anticipated. The potential, or even analytical, price difference was not seen as being crucial – the respondents agreed that it was only a small part of the total cost of ownership. All the respondents seemed to agree, that the price was not the key motivator for the customers, at least not in their industries.

However, respondent from company B claimed that one of open source software's strengths is that in most cases it is the most cost efficient solution. *"Some proprietary software companies claim, and in some cases quite correctly, that the total cost of ownership is less for proprietary products. In some cases it is, but in some others the price favours very clearly open source software."* Also in this case the price was seen as a method of marketing, not as a direct determinant of customer behaviour. But as an example, this proprietary company is quite interesting, since it would seem to have been quite sure in its marketing that price would interest the buyers and thus the total cost of ownership would be imperative. Why would it otherwise take up the subject of price at all?

Three factors might explain why the price was not seen very interesting by the respondents. First, all the respondent sell services and it is a common practice to sell services on the maximum price the business environment enables. Thus, price would not be the dividing factor between different companies. Secondly, the respondents sell to companies and thus their customers probably value also other things besides price. Price would then not really be the relevant determinant of customer behaviour. The third possibility is that pricing is an issue in the community, and that all the respondents agree software or especially software services cannot be too cheap or free, if someone is trying to make living out of them.

9.3 Competence effects narratives

Respondents used competence effects to explain some business effects. The main point was that open source software development needs some special competencies. The number of people competent to make open source software is growing rapidly and this makes open source software development more rapid. The environment where services are becoming more important than products also favours competencies – locking-out is based on competence of the developer company rather than on copyrights or direct expenses.

Especially in Finland there are a lot of people that can be recruited and who know about open source software. This is a factor that supports the development of open source programs. Respondent B claimed that *“Five years ago, it was very hard to find competent people for open source software. Today you can find more specialists in Finland for any 100 most popular open source systems than for any proprietary systems, excluding 10 most popular software products.”* Especially in Finland, IT-specialists who write code very close to operating systems have also actively taken part in the development of open source software programs. Later on, the respondent pointed out that above mentioned numbers were based on intuition, and could not as such be trusted without checking. Respondent B believes that since Finnish universities are using open source software to teach how to code, the know-how will diffuse rapidly – the more know-how there is, the more the software companies will find uses to that know-how.

The competencies are also becoming more important in the competition. A respondent from company A stated *“If the client wonders who is the best provider of some customised service to the software, then it has to be the original coder, the developer. If there is a problem, he is the best one to fix it – not some forker from Internet. We can compete* *with the fact that we are the experts of this software"* This way the company really does not have any other assets than know-how. The code is out, so basically anyone can study it, modify it a bit and start selling it. The competitive advantage a company has or has not thus comes only from the competencies of their experts – not for example from copyright.

Competency also gives some protection from competition. A respondent from company A continues: *"That is why we are not so worried about competition even if we have open source software products. We have half a year's head start compared to anyone who would try to do what we are doing."* This head start is partly technical as the company has coordinated the development of their software products and can offer superior technical support. But the company also has a reputation that can be used in marketing. An entrant that would start from scratch would not have these.

It is necessary to note that these figures and trends towards open source software and there being more specialists in the field are not objective – but they are opinions of the experts, who also have observed the open source community. They should not however be taken as the truth as such, especially since all the respondents were at least a bit open source orientated.

What is fundamentally interesting is the way the respondents saw copyrights as protection for some more traditional companies. These incumbent companies have the option of using copyrights of software to block newcomers that have technologically superior products. Some of the respondents considered this quite problematic, but there is not enough material to say anything more about this copyright issue.

9.4 Platform effects narratives

The respondents used platform effects narrative when they talked about creating new markets. These platforms are technical and judicial, but the main point was that they are something that enables business. When solutions share the same platform, there is a market for goods. All the respondents viewed open source as a good way to create platforms rather than as a way to create products. The open nature of platforms means that they are more effective in some circumstances – especially when it is critical to know how the platform operates.

Another common example of a platform was an operating system. A respondent from company B said explicitly that they needed an open source operating system under their product so that they could guarantee that it would be reliable. A proprietary operating system simply would have been too high a security risk, since they needed to know how the system works and participate in the development if the solutions were not elaborate enough.

The respondents talked about open source software products, but in such a way that open source software was a part of the service or a platform to the service. All the respondents agreed that open source software enables selling some other product. The respondents did not go into the different business models, although all seemed to be well aware of them. They all agreed that there are several of them and the applicability would depend on the industry characteristics.

Respondent B also saw open source software as a potential mode of co-operation between companies in order to save costs. *“Several companies could work together to create open source software for existing needs. This would be more efficient than if all* *the companies bought the same work and told nobody about it"*. The respondent from company C agreed with this idea of co-operation and also saw the potential for savings.

The respondents considered it a real possibility to create a market with open source software and invite entry to the market. The companies would do this deliberately – and they would probably not care if some other field of business would suffer in the process. The company that originally created the open source software would also participate in the same market with a clear advantage: they would have the original developers in their payroll. Company A viewed this as their real strength – they had the original developers, so they had the know-how to compete with. Respondent C also saw a clear problem in this market making: *"Anyone can compete with services, when there is no vendor lock-in by software. An entering company could hire 10-15 best people from the market. Then there would be a service provider, who has less overhead and less expenses, who could offer same service with lower price or even outsource some services to cheap labour countries. The original service provider would be in trouble."*

All the respondents thought software business will go towards Application Service Providing (ASP). This is no news as ASP has been on its way for quite a long time already. ASP does however offer interesting possibilities to open source software, as all the respondents had noticed. As a respondent from company A stated: *"More and more of the software will be bought as ASP – and then it will not matter if it is open source or not from the customer's point of view"*. Since all the respondents favoured services, this was seen as a good possibility: *"traditional software houses will have to really alter their business models"* said a respondent from company A.

10. RESPONDENTS' COMMENTS

After writing this research paper it was given to the respondents for comments. The main aim was to make sure that it was the respondent's voice speaking, not the researcher's, and that the research had been done according to their opinions and preferences. This was the real test of internal validity, since the respondents were the best people to judge whether their ideas were reported correctly.

All the companies sent comments to this thesis. All agreed that they had been quoted correctly and that the narratives, and conclusions based on them, were real. The interpretation of the researcher was seen as justified and it was accepted. This unanimous acceptance supports heavily the validity of the thesis. All the respondents had some comments on some issues raised in the study. The respondents only had two week period to give comments due to external constraints, but this was seen long enough by researcher for respondents to give their comments.

The respondent from company B stated that his speech was quoted correctly. He pointed out that his quote on the large number of software developers in 100 most used open source programs were based on intuition rather than statistics and therefore should be taken with a grain of salt. As this was not a major issue, I did not correct it to the original text. The respondent B also wanted to stress that open source as a platform enables finding niches in the markets, since the necessary investments can be lower. The respondent wanted to underline that thus it is not necessary for a starting software company to conquer an already inhabited marketplace.

The respondent from company C stated that he would have stressed some things differently, but that there was nothing ultimately incorrect in the thesis. He also stated that he did not agree with all the results, but that his speech was quoted correctly. The respondent also corrected one error on the original paper considering licensing conditions.

The respondent from company A accepted the interpretation and did not see anything wrong about the conclusions, although he admitted that some of the opinions had changed after the interview, since analysis took about six months. Respondent corrected some spelling and detail errors adding for example some of the FLOSS-discussion into the thesis.

Respondent A also clarified his views on the membership of the community: *"Members are the 1) developers, 2) users, who acknowledge the existence of developer community, and 3) service providers who acknowledge the developer community and admit to being a part of the community (like IBM)."* Thus *"service providers who simply use an open source product as a part of their products or sells open source software products without informing their customers and taking part of the discussions shaping the software, are really not members of the community". "The community members are those agents that recognise and admit belonging to the community and who have influence on what the community is."* This view is different from the other respondents. This notion is quite interesting, but unfortunately there is not enough data to draw further inferences about it. The respondent of company A however makes it clear that there is some disagreement on who is a member and who is not.

The respondent from company A suggested that interviews from companies that use accessoring as a business model might have had interesting and different viewpoints. Respondent thought that approach and methods of the thesis were interesting.

11. DISCUSSION AND CONCLUSIONS

In this final chapter there is a summary of the results and the conclusion, an evaluation of this study and some possible future directions. The narratives of history of the open source phenomenon showed that the respondents used several ways of telling the story of the open source software. There were philosophical disagreements as there was disagreement on what direction the community should develop to. All the respondents considered themselves to be members of the community and highlighted some of the good things they considered open source might yield. The common enemies were agreed on as were the heroes of the movement.

The media's powerful role in open source software came up already in the beginning – the media's strong role in creating expectations, but also in creating coherence to the open source community by telling the tale of Linux with an interesting plotline. The media's role was also one of interaction – the respondents did not settle for what it told them, but questioned the stories. Respondents also considered it important to shape the way the tale of open source software is told in the future.

The history and the "must" books of the movement were known to the respondents and they could tell the tale spicing it up with their own experiences. They also thus proved that they considered the history commonly told about open source, for example in the chapter two of this thesis correct. Characteristics of the history of open source software also help us to understand why there is disagreement for example over how and why to commercialise the software. The ideological foundations can come up in different situations. The powerful notion of freedom also comes into play and has a direct effect to company strategies and policies – what and whose freedom to offer and respect.

The narratives on the nature of the phenomenon answered to the question about who is a member of the community and what the community is like. This was a very explorative way to approach the phenomenon and yielded only general results about the disagreement. This thesis ultimately did not answer this question very well – in fact it appears that the respondents mixed different kinds of narratives in their speech. They apparently did not agree very well on who is a member of the community and who is not – the only criteria would probably be that members of the community are those people that distinguish themselves as members. This combines quite well the community aspect of the phenomenon into the other aspect of freedom. Also, as a consequence it really is not so important exactly who is a member at a certain point – more important is is the community and the coordinators exist.

The previous classifications of open source were thus not verified or denied. But since the respondents used many narratives at the same time, it is apparent that some of the previous attempts to understand the phenomenon were overly simplistic.

The narratives of effects and the change in business patterns yielded the most interesting results. Basically the four narratives in themselves provided the main verified results.

1) The competitive environment can change as forecasted by Porter (1980). This is because proprietary products cannot compete very long in the same market with open source products and thus try to prevent their entry. This blocking out of potential rival companies becomes imperative and it can be accomplished by using media targeted marketing tactics, different kinds of lock-ins or copyrights strengthened by lawsuits to shield the market from the entry.

2) As theorised by Shapiro (1999), customer expectations become most imperative. This is linked to the first result, but also highlighted by the necessity to become credible and produce quality. These words were used quite loosely, but all agreed that they were important. Surprisingly price did not seem to be very interesting to the respondents.

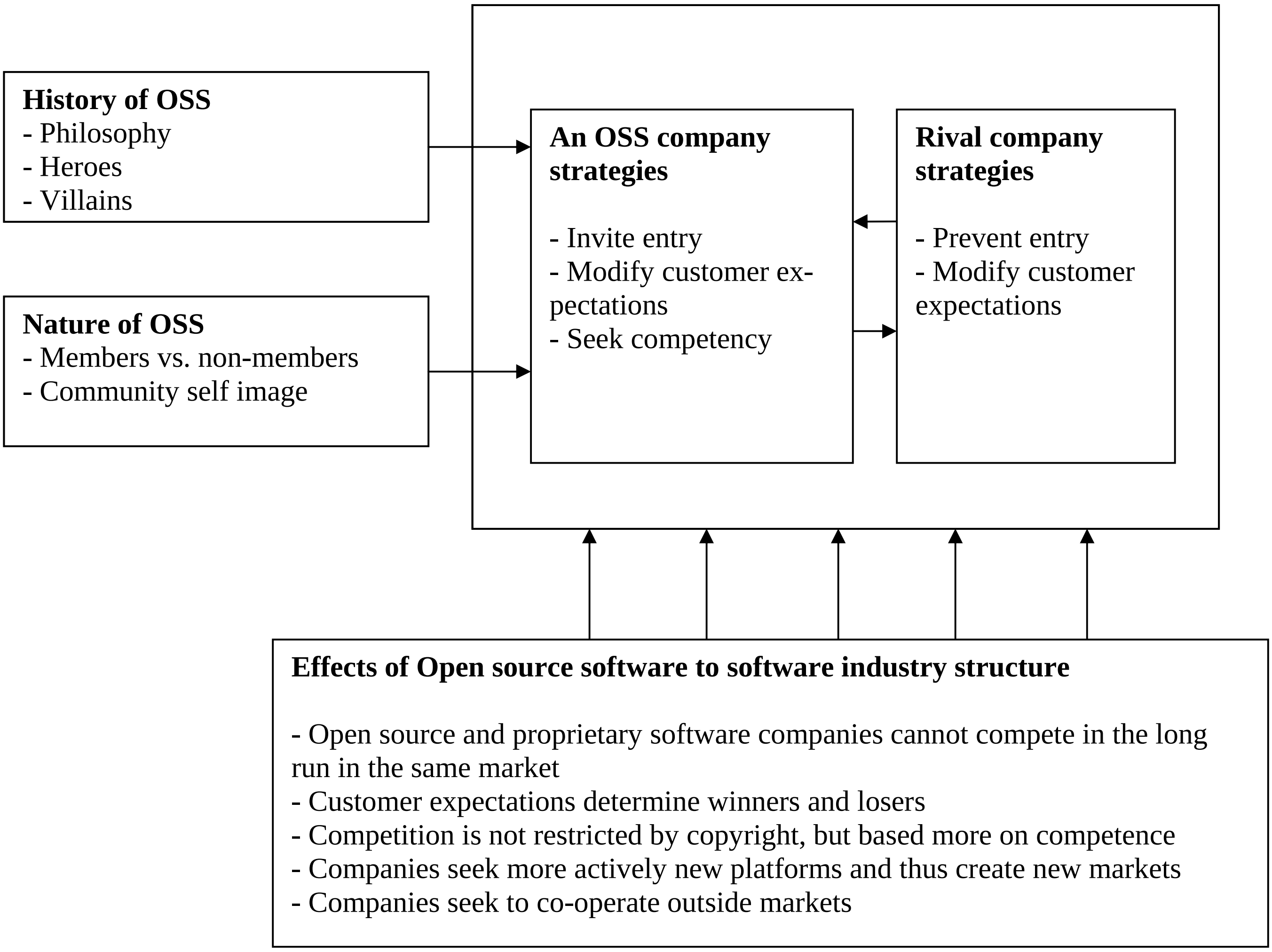
3) The respondents agreed, as some RBV-theorist predict, that competencies will become more and more necessary, since the markets will turn from products to services and copyrights will no longer protect the companies making open source. It was also noted that such competencies are much better available today than for example five years ago.

4) In line with business model theories, the respondents agreed that the role of platforms as market makers is evolving. It will be possible to create new markets and necessary to invite entry to them, but also to use common platforms as a way to co-operate.

These findings mean the following managerial implications to open source companies. Actively seek markets and niches you can enter. The media's role and customer expectations should be one of the primary concerns, and available competencies should be valued while being on lookout for new competencies. The platform thinking of creating markets and inviting entry, remembering the lock-in to open source software, could potentially yield very good results. Also the possibilities for co-operation outside the market structures should be taken seriously.

The following managerial implications are for non-open source companies according to this study. First of all, be on the look out for potential open source competitors – once the product is in the market, it is too late to react. It is not yet certain which industries are more in danger, so better prepare. Prevent the entry before the product makes competition in the market impossible by using media to modify customer expectations.

If the product is already in the market, the options are much more problematic: in some cases it is possible to buy off the developers of small software product or use FUD. These might very well backfire, so better advice would be to explore the options of trying to control the development by for example opening some source code or changing the company revenue logic profile for example towards service business or ASP. Below are the same findings in a table:


History of OSS
- Philosophy
- Heroes
- Villains
Nature of OSS
- Members vs. non-members
- Community self image
An OSS company strategies
- Invite entry
- Modify customer expectations
- Seek competency
Rival company strategies
- Prevent entry
- Modify customer expectations
Effects of Open source software to software industry structure
- Open source and proprietary software companies cannot compete in the long run in the same market
- Customer expectations determine winners and losers
- Competition is not restricted by copyright, but based more on competence
- Companies seek more actively new platforms and thus create new markets
- Companies seek to co-operate outside markets

11.1 Validity

Outside validity is based on previous research and respondents' comments. It should be quite high since the results of the study are in line with previous literature concerning the phenomenon's history and business effects. The problematic part about the nature of the phenomenon shows that some of the previous research is based on different socio-dynamical assumptions than this study.

Inside validity of the data should be good, since the respondents did not know each other, and the only thing they have in common seems to be their age, occupation and interest in open source software. All but the latter probably reflect the Finnish software industry very well. The interest towards open source software is a thing that could potentially put to question the data's validity. However, since all the respondents are professionals in software companies and provide the same kinds of answers, this is not very probable. Since this is a qualitative study, the results would still stand if these five respondents were the only five people on earth who view the world similarly. This is also very unlikely, as there are too many similarities in their narratives.

The collection of the data from natural and tematical interviews fits in with the narrative interpretation method, so this should not yield problems to validity. There is enough data that the main issues and the side issues regarding the research question can be separated.

Since the respondents agreed on the interpretation with the researcher, it can be assumed to describe their social world and guide their actions. In this sense, the conclusions and implications are correct according to the respondents.

11.2 Reliability

The use of the respondents' own voice in many occasions provides the reader with the possibility to evaluate if what is said really supports the claims. The reorganising of the respondents speech was done in the following way: first it was translated correctly from Finnish to English, secondly grammatical errors were fixed and thirdly in some cases sentences were shortened. The meaning of the sentence was not altered in any occasion. The original data including the tapes is stored in a safe place should another researcher wish to check that the sentences have been exported correctly from raw data. Thus there should not be any problems in the reliability of the study.

11.3 Ethics

All the respondents were told what they had participated in and their names or companies were not disclosed to anyone but the researcher and a representative of the university. Not at the beginning nor during the interviews did the researcher direct the conversation except tematically and by asking for clarification if there was a possibility of misunderstanding. The respondents were not told exactly what the researcher was looking for before the interview, but it was discussed after the interview. The respondents also had the possibility to comment on the findings and correct them if they thought them wrong. No one was misled during the interview or afterwards.

The tapes and other interviewing material are kept locked up in a safe place and, if necessary, checked should there be some doubts about the reliability of the findings.

11.4 Future research

This explorative study opened several interesting routes to continue the research. The exact nature of the business effects is one the most interesting. The locking-out and locking-in mechanisms could have interesting managerial implications. Different media strategies of software companies, including FUD, would be quite uncharted territory for investigation. The competence issues have many sides, and this thesis has only scratched the surface of the issues related to the changing competence needs of software companies. The platform thinking and more general discussion about software products versus software services would also probably yield interesting results. The discussion was too large to be taken into this thesis.

It would also be interesting to hear the proprietary software company's voice in the issue of the open source software phenomenon and to compare it to the narratives used by the open sourcists. Would there be similarities or not, and why?

The nature of the community was not agreed upon, and more research into the issue could be fruitful. The disagreement on the membership could serve as a good starting point to investigate the community. The philosophical differences between OSI and FSF might cause some differences in the way the two and the entire community acts. Narrative study concerning only the stories of this community and division might produce interesting results. Especially the literature methods that classify stories applied to the saga of Linux would show how it is told to create cohesion, show the enemy, and most importantly to give directions and transfer values.

On a more general level, there has been a discussion concerning a networked way of life and changing the requirements of work, economy, and society. These views where not explicit in this thesis, but they came up– especially in a discussion about the digital ownership, co-operation outside market structures, and in the premises of the business model theory. The changing business patterns caused by open source companies could be included in the field of science that investigates the networked economy.

REFERENCES

Barney, J. (1991). Firms resources and sustained competitive advantage. Journal of management 17, 99-120.

Bruner, J. (1986). Actual minds, possible words. Harvard University Press.Cambridge.

Burr, V. (1995). An introduction to social constructionism. Routledge. London.

Coase, Ronald H. (1937). The Nature of the Firm. Economica. The London School of Economics and Political Science. London.

Drucker,P. (1954).The Practice of Management' . Harper & Row. New York.

Feller,J. and Fitzgerald, B. (2002). Understanding Open Source Software Development, Pearson Education Limited.

Harsh, A. and Ou, S. (2001). Why is Open Source Viable? – A Study of Intrinsic motivation, Personal Needs, and Future Return. 34th Hawaii International conference on System Sciences, Hawaii. Adapted from Väisänen, 2003: Avoin lähdekoodi ohjelmistoliiketoiminnassa.

Harwood, Kate, Presley Diana, Tsuchida Yasuaki, Wolf Lisa. The Open- Source Revolution. http://cci.bus.utexas.edu/research/white/open-source.htm (21.11.2003)

Hecker, F. (1999). Setting up Shop: The Business of Open-Source Software, Open Resurces: http://www.openresources.com/documents/setting-up-shop/index.html, 17.06.1999. (21.11.2003)

Von Hippel, E. (2002). Horizontal innovation networks – by and for users. MIT Sloan School of Management. June 2002.

Holtwere, U and Werle, R. (2001) “De-Commodifying Software? Open Source Software Between Business Strategy and Social Movement. “. Science Studies, vol. 14, 2, p. 43-56.

Kottak, C. P. (2000) Cultural Anthropology. McGraw-Hill. Boston.

Kollock, P. (1999). The Economics of Online Cooperation: Gifts and Public Goods in Cyberspace. Chapter 7, Communities in Cyberspaces, M.A. Smith and P. Kollock eds. Routledge, London.

Lee, A.S. (1994). "Electronic Mail as a Medium for Rich Communication: An Empirical Investigation Using Hermeneutic Interpretation" MIS Quarterly vol. 18, 2, pp. 143-157.

Lee, A,S. (1989). ”A scientific methodology for MIS case studies”. MIS Quarterly, vol.13 n.1, p. 33-50.

Lerner, J.&Tirole, J. (2000). The simple economics of open source. NBER Working Paper, no. 7600.

Levi-Strauss, C. (1949). The Elementary Structure of Kinship.

Lynne, M. (1983). “Power, politics, and MIS implementation”. Communications of the ACM, vol.26, 6, p.430-444.

Minzberg. H. (1983). Structures in Fives. Prentice Hall. New York.

Mustonen, M. (2000). Copyleft – the economics of Linux and other open source software.In (2003) Essays on the economics of information and Communication Technologies: Copyleft, Networks and Compatibility. Department of Economics, University of Helsinki.

Mustonen, M. (2002). Why do firms support the development of substitute copyleft programs? In (2003) Essays on the economics of information and Communication Technologies: Copyleft, Networks and Compatibility. Department of Economics, University of Helsinki.

Myers, M.D. (1994). "A disaster for everyone to see: an interpretive analysis of a failed IS project" Accounting, Management and Information Technologies vol. 4,4, pp. 185-201.

Open Source Iniative. (1998). http://www.opensource.org (21.11.2003)

Palvia, P.C.. (1995). "A Dialectic View of Information Systems Outsourcing: Pros and Cons" Information & Management, Vol 29, pp. 265-275.

Polkinghorne, D. (1988). Narrative knowing and the human sciences. State University of New York Press. New York.

Porter, ME. (1980) Competitive Strategy. Free Press. New York.

Porter, ME. (1985) Competitive Advantage. Free Press. New York.

Porter, ME. (2001).Strategy and the Internet. Harvard Business Review, March 2001; www.hbsp.harvard.edu/hbr

Rajala, R., Rossi,M., Tuunainen, V., Korri,S. (2001). Software Business Models: A framework for Analyzing Software Industry. TEKES, 108. Helsinki.

Raymond, Eric S. (1999). The Cathedral and the Bazaar, Musings on Linux and Open Source By an Accidental Revolutionary. O'Reilly & Associates, inc. Sebatopol. CA, 1999.

Shapiro, C& Varian, H. (1999). Information rules: A Strategical Guide to the Network Economy. Harvard Business School Press, Boston Massachusetts.

Stallman. (2000). http://www.gnu.org/. (21.11.2003)

Stallman. (2001). http://www.gnu.org/fsf/fsf.html. (18.08.2004)

Tapscott D (2001). Rethinking strategy in a Networked World. Strategy and Business. Fall 2001.

Timmers P. (1998) Business Models for Electronic Markets. Electronic Market 8 (2), 2-8.

White, H. (1973) Metahistory: The historical imagination in nineteenth-century Europe. Johns Hopkins University Press. Baltimore and London

Yin, R.K. (1994). Case Study Research, Design and Methods(2nd ed.). Sage Publications. Newbury Park.

APPENDIX 1

Juho Lindman
Kunnantie 4 B 8
00700 Helsinki
09-2247827
juho.lindman@iki.fi

SAATEKIRJE TUTKIMUKSEEN

Hyvä vastaanottaja,

Käsissäsi on tällä hetkellä ohjeistava saatekirje kyselyyni, joka kulkee tutkimukseni liitteenä. Olen laatimassa pro gradu tutkielmaan Helsingin kauppakorkeakoulun tietojärjestelmätieteen laitokselle. Tutkielman aiheena ovat avoimen lähdekoodin ohjelmistojen vaikutukset liiketoimintaympäristöön. Vastaajat on valikoitu siten, että teillä on kokemusta avoimista lähdekoodeista ja ohjelmistoteollisuudesta.

Toivon, että vastaat liitteenä olevaan kyselyyn. Pyydän varaamaan kyselyn täyttöön noin 30 minuuttia, jotta ehdit vastata rauhassa myös avoimiin kysymyksiin. Mikäli et syystä tai toisesta koe voivasi vastata johonkin kysymykseen, niin jätä kohta tyhjäksi. Halutessasi voit kertoa syyn kohdan tyhjäksi jättämiselle. Mikäli mieleesi tulee jotain muuta aiheeseen liittyvää mielenkiintoista, niin senkin voit kirjoittaa vastaavaan kohtaan tai loppuun Muita kommentteja kohtaan.

Mahdollisesti antamasi yhteystiedot pysyvät tutkimuksen tekijän tiedossa. Vastauksia säilytetään luottamuksellisina eikä luovuteta eteenpäin. Tutkimuksesta vastaajien henkilöllisyys salataan. Voit vastata kyselyyn myös täysin nimettömänä. Voit vastata kyselyyn suomeksi tai englanniksi.

Jos sinulla on kysymyksiä mihin tahansa tutkimukseen liittyvästä seikasta, niin älä epäröi ottaa yhteyttä allekirjoittaneeseen.

Ystävällisin terveisin,
Juho Lindman

APPENDIX 2

QUESTIONNAIRE

Attached you will find information on filling this questionnaire.

1. PERSONAL INFORMATION (OPTIONAL)

Name (optional) :____
Educational background :____
Position within company :____
Branch of industry :____
Have you taken part to open source software development? How?

____

____

____

Why did you participate?

____

____

____

2. OPEN SOURCE SOFTWARE
In your opinion, what is open source software?

____

____

____

____

What is the meaning of open source community?

____

____

____

____

3. BUSINESS AND OPEN SOURCE SOFTWARE

How is it possible to compete with open source software?

____

____

____

Does open source software offer any other business benefits?

____

____

____

If a rival would bring (or has brought) an open source software in the market, what might be the immediate reaction of your company?

____

____

____

How new entrant would effect the industry structure?

How would industry’s clients react?

Is your company a direct rival of an open source software firm?

What is your opinion about open source software’s publicity?

What do you think about ”heroes” and “villains” of the open source?

What are the open source software’s future opportunities in business?

What are the open source software’s future threats in business?

What are the open source software’s future strengths in business?

What are the open source software’s future weaknesses in business?

Other comments?

____

____

____

____

Thank you for your reply. Your company and personal information will not be disclosed.