“Vertical collaboration in open source business”

Document

Master Thesis Business Information Technology
29/11/2007

Student

Stefan Reijmer Kalter
s0000213

Organisation

eMAXX B.V.
Welbergweg 80-84
7556 PE Hengelo

Institute

University of Twente
Enschede, The Netherlands

Graduation committee

Dr. M. Daneva, *Faculty of Electrical Engineering, Mathematics and Computer Science*
Dr. R.M. Mueller, *Faculty of Business, Public Administration and Technology*
M. Veenhuis , *eMAXX*

"Where the cattle stand together, the lion lies down hungry."

- African Proverb -

Preface

Now the end of my student life is near, I would like to look back once more and evaluate.

After high school, which took me seven years to complete, I started studying economics at college in Amsterdam. Although a change of scenery did me good, the study itself didn't. I started doing *"International management, English style",* which seemed the right thing to do at that moment. When I looked back after one and a half year I noticed that, although I earned most of my credits easily, I had not learned anything... I started to figure out what it was I wanted to do the rest of my life. Now I know that I should have made this decision many years earlier, but who can blame a teenager for not knowing what he wants?!? At high school I had not made the right choice in courses, so when I came to the the conclusion that I wanted to study Business Information Technology at the University of Twente in Enschede, I missed two important courses, one of which was Mathematics B. This proved to be a bigger problem than I could have ever imagined. First of all I had to follow a six week summer course at the James Boswell Institute in Utrecht in order to obtain the certification needed to register for the course in Enschede and second because this summer course can, in my opinion, not be compared with several years of Mathematics at high school. This is probably the reason why I struggled with Calculus and other Mathematics courses during my stay in Enschede.

I had never had real trouble achieving anything until I came to Enschede and that was exactly the reason to go there. Although it was sometimes frustrating, I really think I have learned a lot. For this I would like to thank all of my teachers at the university, my fellow students and my study buddies who have helped me to achieve my goals. You know who you are!

I would also like to thank my parents and my friends for having such patience with me. I think it was in my advantage that most of them haven't had a clue what my study was all about... Empty looks sometimes prevented me to elaborate further, when I tried to explain what Business Information Technology is about. Most of the time I just explained that it had to do with the design and alignment of computer systems and organisations.

I would especially like to thank Marieke for standing by me in times I didn't see light at the end of the tunnel anymore. Without you I would never have made it, I love you!

Last, but not least I would like to thank Michel Veenhuis for his vision and for making this research possible, Maya Daneva and Roland Mueller for their critical comments on my research design, the interviewees in the case studies, and all the respondents of the survey.

Management summary

Introduction and problem statement

The Open Source (OS) principle has been around for several decades now and much research has been done in the economics and psychology behind OS. Research in collaboration in OS, on the other hand, is lacking, so it is unknown what the biggest problems are and why. This research will shed some light on this subject by researching the success and failure factors of collaboration in OS.

Methodology

Because there is not much known about partnering in OS, the first part of this research consists of combining known theory in order to find out what OS and specifically OS business exactly entails. From the theory possible problem areas in OS collaboration have been defined. These findings have been combined with theory on partnering failure and success factors in order to find the highest rated perceived partnering success and failure factors in OS. Parallel to the theoretical research, two case studies have been conducted. These case studies were meant to provide more information about partnering with OS companies in general and about the collaboration between system integrators with OS companies in particular.

After the theory research and the case studies there was enough information to form some hypotheses which lead to the theoretical model for the survey. The survey was conducted among 27 OS companies around the world.

Findings and recommendations

In OS partnerships both partners products should be complementary or both partners will be collaborating and competing at the same time, which is not a good basis for cooperation. If the partners are complementary, mutual dependency increases. A mutual goal, however, doesn't really exist in OS partnerships. It should be clear to both partners that shared goals aren't obligatory in successful partnerships. Both partners can perfectly pursue their own goals in partnerships, as long as the partners are complementary.

From the case studies can be concluded that communication quality suffers from people changing jobs. According to Ellram, having multiple communication lines has a positive effect on collaboration success. This was confirmed by the case study interviewees, but there was no support for this relation, according to the survey .

OS companies should have part of the agreements ready before negotiations with potential partners. In these preparations, success factors per partnership should be defined.

Support has been found for the causal relationship between communication quality and collaboration effectiveness and between communication quality and trust. All the other proposed causal relationships were insufficiently supported by the data of the survey.

Next to the analyzed hypothesis, there was also support found for the causal relationship between the employee turnover rate and the number of communication lines. Partial collaboration between these two variables, controlling all the others, is -0,4265 ($p<0,05$),

which is significant. This can be explained by the fact that communication channels aren't renewed when people change jobs.

In this research problems and opportunities in OS business partnerships have been analyzed from different angles. Different problems and opportunities arose, but one problem comes up every time: communication quality. It should be very clear to OS companies that communication quality works two ways. When it is too low, partnerships will suffer or cease to exist at all, and when it is high it will help the partnership to success. Therefore it is absolutely vital for OS companies to focus their attention to communication quality by implementing communication plans and by training employees. Higher communication quality will lead to higher collaboration effectiveness and higher trust. Higher trust on its turn might affect the flexibility in future agreements.

Table of contents

Lists of figures and tables

List of figures

List of tables

Chapter 1. Introduction

1.1 Introduction

In the introductory chapter of this thesis the problem description for the research will be discussed together with its objectives. At the end of this chapter the research method selection will be explained. The rest of the research design will be discussed in chapter 5. The research will be conducted under guidance of eMAXX in Enschede and Hengelo, the Netherlands. This company has developed software that enables organisations to integrate back and front office systems. Although eMAXX's software has the potential to cover a broad range of markets, its clients are mainly municipalities.

1.2 Research design paradigms

The overall research was designed according to the combination of the paradigm of Verschuuren and Doorewaard [VER00] and that of Yin [YIN03]. Verschuuren and Doorewaard divide research design into two parts. The first part, the so-called conceptual design, describes what will be researched by defining the objective of the research, a research plan, research questions and a description of the used concepts. The second part, the research technical design, describes how the research defined in the conceptual design will be conducted by defining the material consulted and the strategy used.

1.3 Conceptual design

Framework

eMAXX would like to expand the sales of the modules complementing its OS software and the services accompanying that software, such as training, maintenance and support, by increasing the effectiveness of their partnerships. Expansion should take place within the current sectors in which eMAXX is active, as well as other sectors. Selling software and services abroad will not be excluded.

Open source

In general, open source software (OSS) is software that can only be distributed together with the source code. A more detailed description of OS can be found in section 3.2.

Objective

The objective of the research is to further develop the theory on OS business by providing insight into the organizational problems and opportunities for OSS companies engaging in vertical collaboration. The results of this research can be used to identify possible partners, to coordinate partnerships in OS business and to improve the effectiveness of partnerships.

Main research question

In order to meet the objective of the research the following main research question will be discussed in this thesis:

"What are the most important technological and organizational problems and opportunities for open source software companies engaging in vertical collaboration?"

Sub questions

Within this research question exist some terms that need explaining and thus research on their own. By deducing a set of sub questions from the main research question, this can be achieved. The main research question has been divided in researchable components, so it can be answered more easily. These sub questions are:

1. *What is open source business?*
2. *What is vertical collaboration in the open source software field?*
3. *How do open source companies collaborate?*
4. *What organizational factors affect the effectiveness of vertical collaboration in open source business?*

Theoretical model

Widespread adoption of the software is crucial for OSS development companies. Since products and services that are the source of revenue are complementary to the OSS, widespread adoption provides a solid base for earning revenue. In order to achieve an increase in adoption, OS companies could adopt several collaborative strategies:

- companies might collaborate with other companies to increase the perceived ease of use and the perceived usefulness of OSS according to the Technology Acceptance Model (TAM) by Davis [DAV85]. Both factors are known to improve the attitude towards using a specific technology. The Technology acceptance model as a whole models how users come to accept and use a technology. This is shown in figure 1.1. In this figure circles represent constructs, squares represent design features and arrows represent causal relationships.
- OS companies might collaborate with other companies in order to increase their distribution capacity, which in turn will lead to higher adoption.

- OS companies might collaborate with other companies that complement their own OSS, or with companies their OSS is complementary to. A firm's relationship to the network of providers of complementary products determine its value creation, value capture and the durability of its competitive advantage. Creating and managing these relationships is an important part of achieving each of these goals [HAM00], [IAN04].

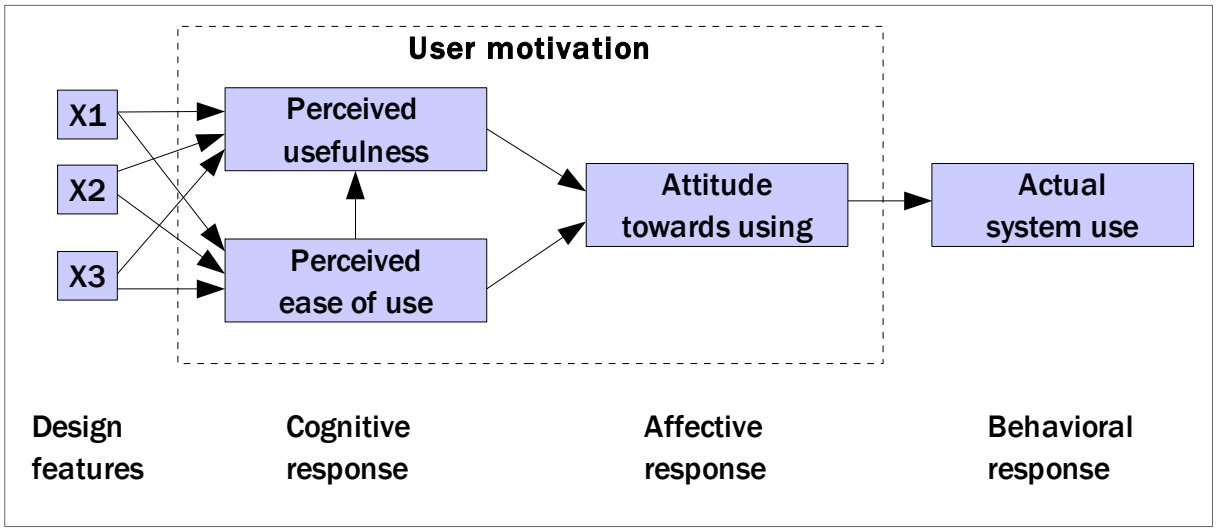

Figure 1.1: Technology acceptance model [DAV85]

Scope and delimitation

In this research the focus will be on the causal relationship between influencing factors and the effectiveness of collaboration which is reflected by the upper left arrow in figure 1.2. There are more ways to influence the attitude towards using specific OSS technology (see figure 1.1), marketing, for example, but the initial assignment included the improvement of collaboration. In figure 1.1 X1 to X3 are design features that influence the perceived usefulness and the perceived ease of use of the technology. The higher the perceived ease of use, the higher the perceived usefulness and the attitude towards using the technology. The attitude is also affected by the perceived usefulness of the technology. The higher the attitude towards sing the technology, the higher the chance that the technology will actually be used.

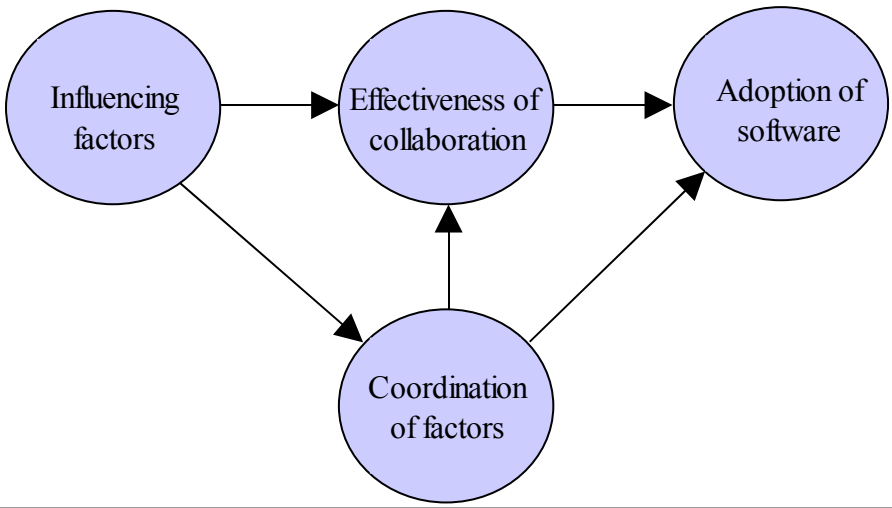

Figure 1.2: High level theoretical model

1.4 Research technical design

Research method selection

This research has been conducted in two phases. Because not much is known about collaboration between OS companies and their business partners, in the first part an exploratory case study has been combined with desk research in order to enhance the comprehension of factors that influence the effectiveness of collaboration in OS business. Case study research is a suitable research strategy to study such unexplored areas. Because one can not generalize from single-case case studies, in this research two cases have been analysed.

After phase one the findings were combined in a theoretical model for testing in phase two. In the second part of this research a survey has been conducted among OS companies in order to rate the factors that originated from phase one and to test the constructed theoretical model.

Before this research started it was not known what the most important factors that contribute to the effectiveness of collaboration in OS business were. There are numerous factors that are known to affect the effectiveness of vertical collaboration in production companies. Results of the research of L.M. Ellram [ELL95] combined with that of M.U. Douma [DOU97] will be used as a starting point for building a list of factors contributing to the effectiveness of vertical collaboration in OS companies. The research of L.M. Ellram is discussed in chapter 2. OS business will be explained in chapter 3 and collaboration in OS business in chapter 4. The case study will be discussed in chapter 5. The theoretical model that was constructed according to chapters 2-4, is discussed in chapter 6. The survey by which the theoretical model will bes tested is described in chapter 7. The conclusions from the whole research can be found in the last chapter, which is chapter 8. Figure 1.3 is a graphical representation of the research model.

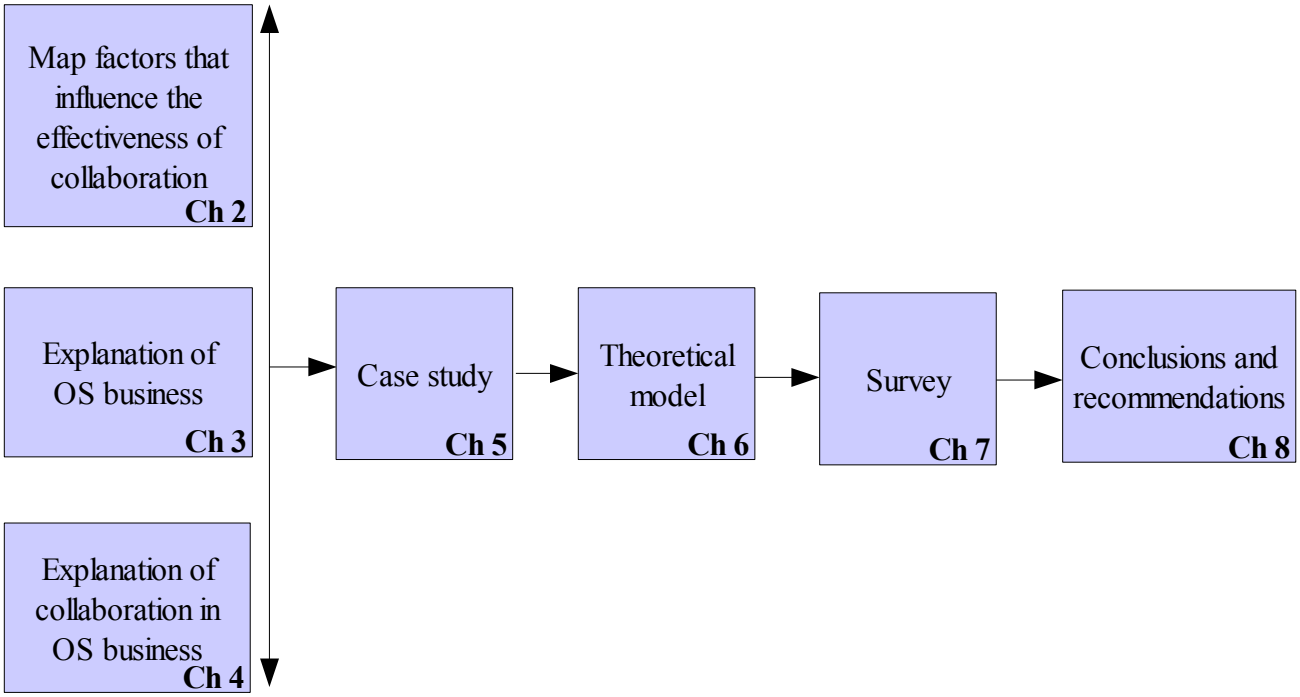

Figure 1.3: Research model

1.5 Chapter summary

This chapter was the introduction for this thesis. At first the framework within which the research was conducted, the research questions and the high level theoretical model were discussed, followed by a description which research methods have been used and why. The next chapter entails the theoretical background for the research.

Chapter 2. Theoretical background

2.1 Introduction

In this chapter the theoretical background for this research will be discussed. Section 2.2 describes research of Ellram [ELL95] about the factors that influence partner relationships. An explanation of measuring the effectiveness of collaboration will be given in section 2.3 and eventually research on the effectiveness of alliances will be explained in section 2.4.

2.2 Partnering Pitfalls and Success Factors

Lisa Ellram [ELL95] empirically researched success factors and factors for failure in vertical partnerships in production companies. The reasons why these partnerships failed and excelled have been studied from two perspectives: the buyers' and the suppliers' perspectives. To increase the internal validity of the research, the sample of companies taken from the Fortune 500 was asked to list the top five factors that lead to success or to failure of vertical collaboration. The factors that are clearly not applicable to the OS field have been excluded in this research, some factors that might be influential according to literature on OS (business) have been added and results from the case studies have also been taken into account. The factors originating from Ellram's research are mentioned in appendices 1 and 2.

Pitfalls

L. Ellram concluded that a significant part of the factors was perceived to be unimportant for ineffective partnerships by both buyers and suppliers. These are the factors with ratings smaller than four, which is the neutral rating (see appendix 1). The most important factor for failed partnerships is poor communication. Lack of trust, poor up-front planning, lack of strategic direction for the partnership and lack of shared goals are also perceived important by both buyers and suppliers. The main differences in perceived importance are low status of the customer's purchasing function, lack of central coordination of the buyer's purchasing function, lack of strategic direction, lack of shared goals, lack of benefit/risk sharing, lack of distinctive value added and lack of total quality commitment by the supplier. It seems that both buyers and suppliers are pointing their finger at the other party, but in fact outsiders might have a real clear vision of inadequacies at the partners while ignoring their own. The fact that both parties have been taken into account in the Ellram research reduces this effect. Some of the interesting findings include the fact that buyers agreed with suppliers that buyer top management support was a greater contributor to partnership failure than top management support of the supplier. Lack of distinctive value added by the supplier and lack of a total quality commitment by the supplier are considered significantly more important factors contributing to partnership failure by buyers. These differences in perception can well be important factors to consider when developing, maintaining, or enhancing partnerships [ELL95].

In order to obtain a different perspective of these results, Ellram asked the respondents to list their top five of factors, independently from the responses in appendix 1. The results are mentioned in table 2.1. The table shows the percentage of respondents that listed the specific factors in their top five. The mentioned ranks are the ranks these factors have in appendix 1.

TOP FACTORS CONTRIBUTING TO PARTNERSHIPS THAT HAVE NOT WORKED OUT OR HAVE BEEN DISSOLVED				
	Buyers' response		Suppliers' response	
	% of respondents	Rank	% of respondents	Rank
Poor communication	64.1%	1	58.7%	1
Lack of top management support by our top management	48.4%	2	31.8%	10
Lack of trust	40.6%	3	33.3%	4
Lack of total quality commitment by supplier	46.9%	4	14.3%	18
Poor up-front planning	23.4%	5	28.6%	5
Lack of strategic direction for the relationship	35.9%	7	47.6%	3
Lack of shared goals	28.1%	8	41.3%	2

Table 2.1: Top factors contributing to partnerships that have not worked out [ELL95]

Ellram noted that the top seven most mentioned factors (see table 2.1) for both buyers and suppliers were the same. Both buyers and supplies most mentioned poor communication as the most important factor contributing to partnership failure. They also rated lack of trust very important. Buyers rate the lack of total quality commitment by the supplier and the lack of top management support as important factors.

Success factors

Ellram noted that all the success factors mentioned in appendix 2 are rated higher than the neutral rating of four, which means that all factors were perceived to be important to both suppliers and buyers.
The five factors with the highest ratings for buyers are:

1. Two-way information sharing
2. Top management support
3. Shared goals
4. Early communication to suppliers
5. Supplier adds distinctive value

The whole list can be found in appendix 2. Ellram found minimal statistically significant differences between buyers and suppliers. Suppliers rated four factors as significantly more important than the buyers:

1. Multiple relationships and points of contact between buying and supplying firms
2. Ongoing relationships between top levels of buying and supplying firms
3. Personal relationships
4. JIT initiatives

Ellram noted that three of these four factors were related to the relationship itself between the partners and that these factors included the term *'relationships'*. In order to

obtain a different perspective of these results, Ellram asked the respondents to list their top five of factors, independently from the responses in appendix 2. The results are mentioned in table 2.2. The table shows the percentage of respondents that listed the specific factors in their top five. The mentioned ranks are the ranks these factors have in appendix 2.

FACTORS BELIEVED TO BE MOST IMPORTANT TO THE SUCCESS OF A PURCHASING PARTNERSHIP				
	Buyers' response		Suppliers' response	
	% of respondents	Rank	% of respondents	Rank
Two-way information sharing	70.5%	1	61.8%	2
Top management support	75.6%	2	84.2%	1
Shared goals	48.7%	3	48.7%	6
Early communication to suppliers of specification changes, new products	34.6%	4	17.1%	11
Supplier adds distinctive value	28.2%	5	39.7%	5
Total quality management initiative	51.3%	7	43.4%	5
JIT Initiatives	10.2%	17	22.4%	5

Table 2.2: Top factors contributing to partnership success [ELL95]

A disparity between the ratings given in appendix 2 and table 2.2 becomes apparent in the fifth ranked factors: supplier adds distinctive value, total quality management initiative and JIT Initiatives. Suppliers rated these factors with an average rating of 5.88 out of 7 in appendix 2, but have widespread results in table 2.2.

2.3 Effectiveness of collaboration

Douma identifies three ways to evaluate the effectiveness of collaboration [DOU97]:

1. Collaboration status: Collaboration status means whether the collaboration is operational or not. Whether it is possible to end collaborations that have succeeded, because of changes in the market for example. In this research a broader definition of collaboration status has been used. (5 point Likert scale from very unsuccessful to very successful)
2. Gained synergy: Synergy occurs when combined results are greater than the sum of its parts. Synergy is hard to measure quantitatively [LUI93] and has therefore not been used as a measure in this research.
3. Degree of goals realized: In the Multiple Constituency Approach effectiveness is defined as the degree to which a company realises the objectives of one or more of its constituencies [WEI94]. There are three types of objectives that might be affected. These are the individual objectives of both partners involved and the shared objective of the partnership.

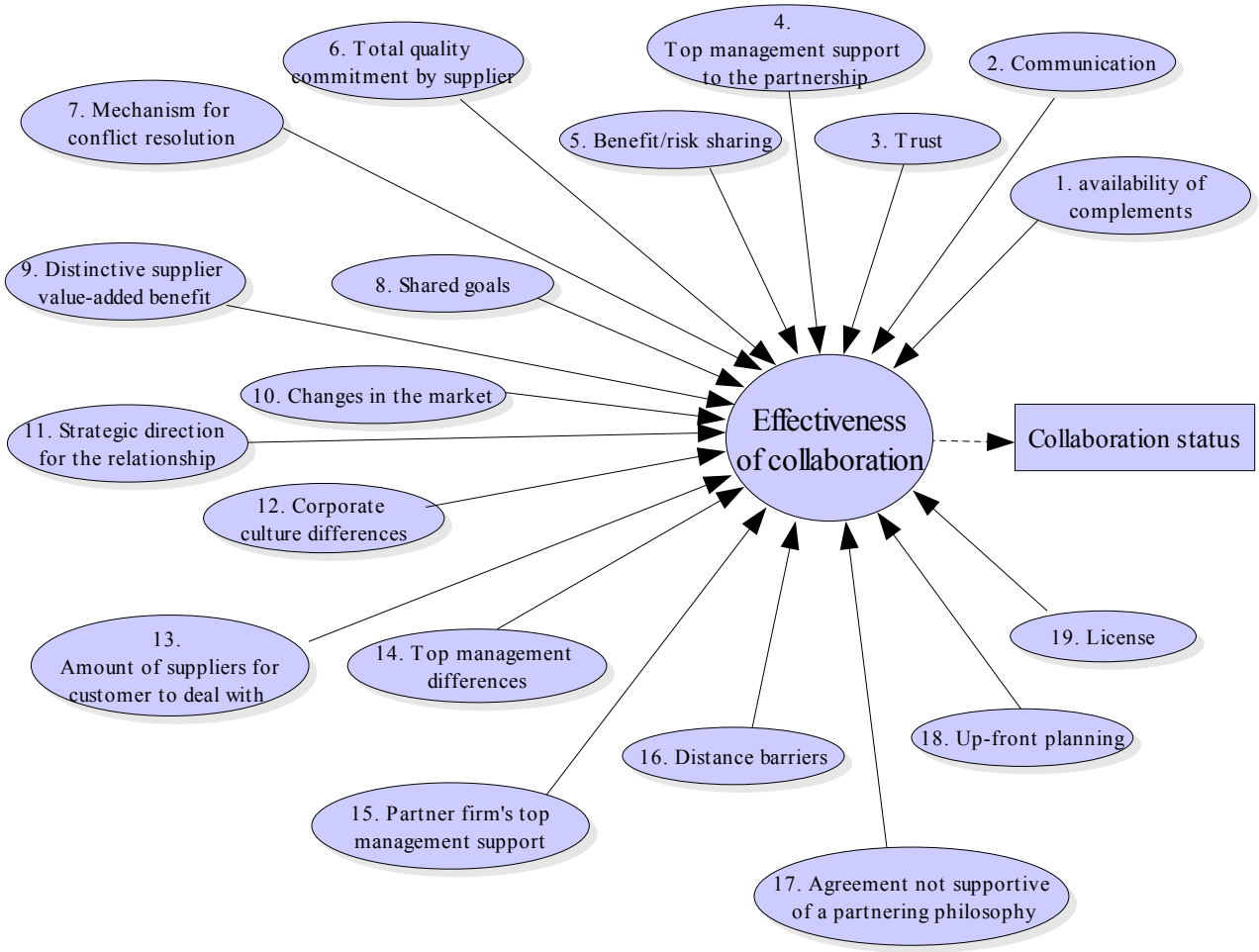

Figure 2.1: Low level theoretical model for case study

Figure 2.1 is a more detailed version of the two upper left circles of the model in figure 1.2. It describes the theoretical model for the case study. Circles represent constructs, the square represents the dependant variable and arrows represent causal relationships. The factors mentioned in figure 2.1 were rated and according to this rating a list of technological and organisational coordination mechanisms have been composed to improve the effectiveness of vertical collaboration in order to achieve higher adoption of the OSS.

2.4 Fit model

In this research the assumption is made that, although the degree of collaboration is different, partnerships are much like alliances. Subjects exclusively concerning alliances will not be used in discussing partnerships. All the factors mentioned in appendices 1 and 2 can be ordered along the fit model by Douma. This model describes a practical framework for structuring the decision making process pertaining to strategic alliances [DOU97].

Douma states that a successful alliance requires a sufficient degree of fit in five areas. These areas can be distinguished by the ovals in figure 2.2, which is a simplification of figure 2.1, as all the factors have been categorized according to the fit model by Douma [DOU97]. Douma states that the degree of strategic fit gives an indication of the alliance potential, so when there is lack of strategic fit, co-operation is not advisable. Feasibility of the alliance is determined by the degree of organisational fit and by implementation risks linked to the alliance. This research will merely focus on strategic and organisational fit, which will be discussed in this section.

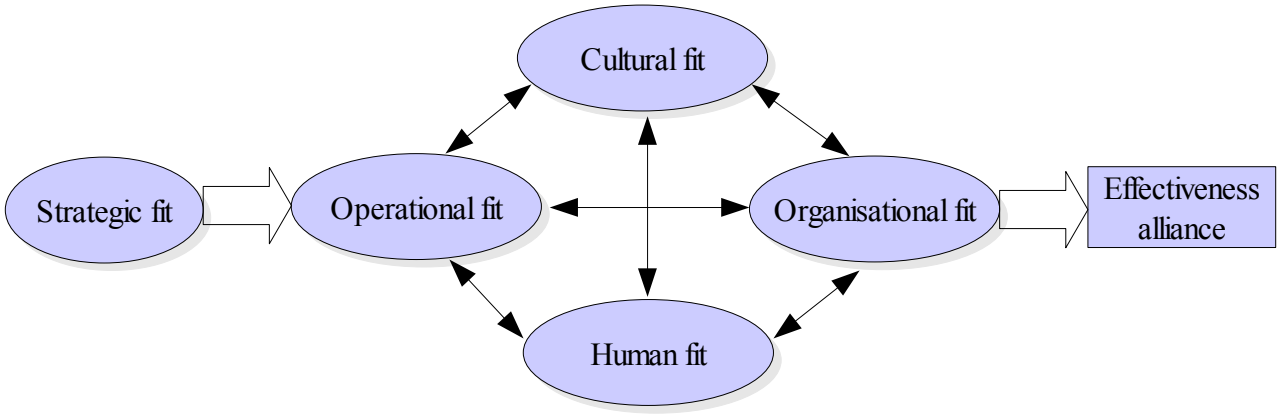

Figure 2.2: Fit model [DOU97]

Strategic fit

If no strategic fit exists between partners and no improvement is expected, cooperation is not desirable [DOU97]. Douma defines strategic fit as:

> *"There is strategic fit if the partners' strategies and objectives are mutually dependant and compatible, and the alliance is of strategic importance to the partners' competitive position"*

Strategic importance

If the alliance lacks strategic importance for both partners, they will probably be less committed to making the necessary efforts for making the partnership work. It should be stated that relative importance increases the effect. When the alliance is relatively more important to one of the partners, this partner is expected to demand a greater degree of control of the policy and functioning of the alliance. The degree of partners' willingness to make concessions depends on two dimensions: pressure of time on establishing the alliance and whether the alliance is offensive (for example entering new markets) or defensive (for example protect market share), which is shown in figure 2.3. The strategic importance, together with goal dependency, and the expected profits, costs and risks, influences the partners' relative bargaining power.

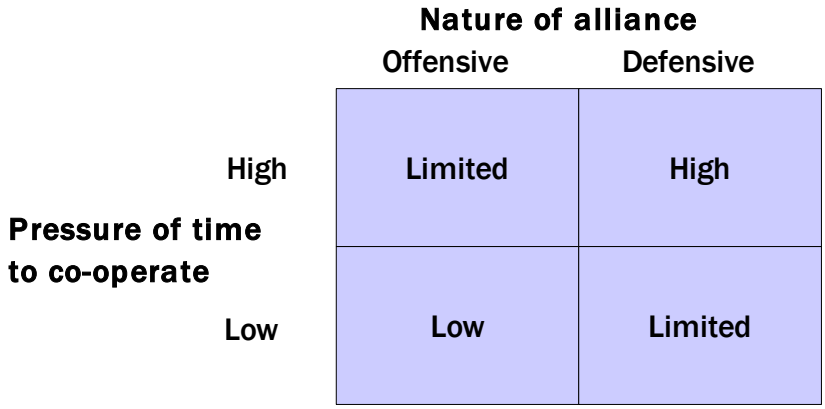

Figure 2.3: Impact of nature and time pressure on decision to co-operate [DOU97]

Compatibility of strategies and objectives

Douma states that strategies and objectives should be compatible at three levels: the corporate, competitive and alliance level. Good compatibility at the competitive and the alliance level results in a positive advice for the alliance. In other cases, there might not be a solid basis for co-operation (see figure 2.4). Compatibility of the objectives is often a result of intensive discussion of basic assumptions underlying the alliance.

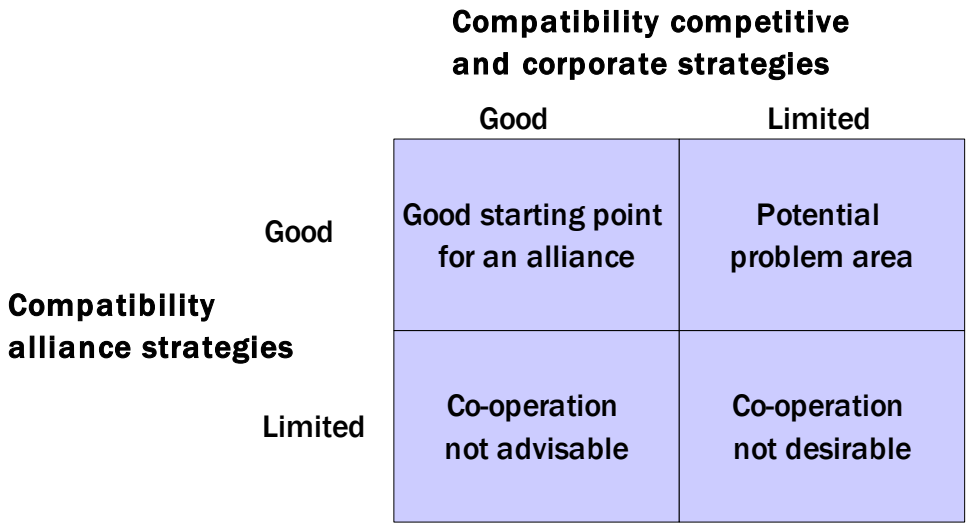

Figure 2.4: Impact of (in)compatible strategies on decision to co-operate [DOU97]

Mutual goal dependency

The existence of mutual dependency of the partners with respect to reaching their individual objectives will lead to commitment to the alliance and willingness to make concessions. Complementary software producers are mutually dependant. Douma states, on the other hand, that collaboration unavoidably leads to knowledge transfer (see textbox below).

> *"In general, it may be stated that the chance of success for an alliance is greater the more complementary the partners are. The partners do have to realise that collaboration unavoidably leads to knowledge transfer. Although this is often cited as one of the advantages of co-operation, there would seem to be a downside to this coin. Transfer of know-how may be undesirable for two reasons. In the first place, if unique knowledge and experience determining the companies' competitive advantage are concerned. Secondly, the basis for co-operation may erode as a result of knowledge transfer. If this flows in one direction, there is a one sided advantage, causing the continuity of the alliance to come under pressure [DOU97]."*

This might lead to a contradiction in OS business. On the one hand OS companies should try to accumulate as many complements as possible (see value network, section 3.7), while on the other hand they should be careful in sharing know how with partners, because sharing of know how might lead to hijacking [LER02] when it is combined with unrestrictive licensing. Hijacking OSS means that someone takes the source and uses it for their own project or company. From this can be concluded that knowledge management should be an important issue in OS companies. It should be widely known throughout the company what information to share, and particularly, what information not to share. The infrastructure used by the company should also enable the separation

of information. This can be accomplished by the use of separate systems or by strict access control. The degree of risk of knowledge transfer are further explained in figure 2.5. The degree of risk of knowledge transfer depends on the nature of the knowledge and experience and how transferable the knowledge and experience is (see figure 2.5).

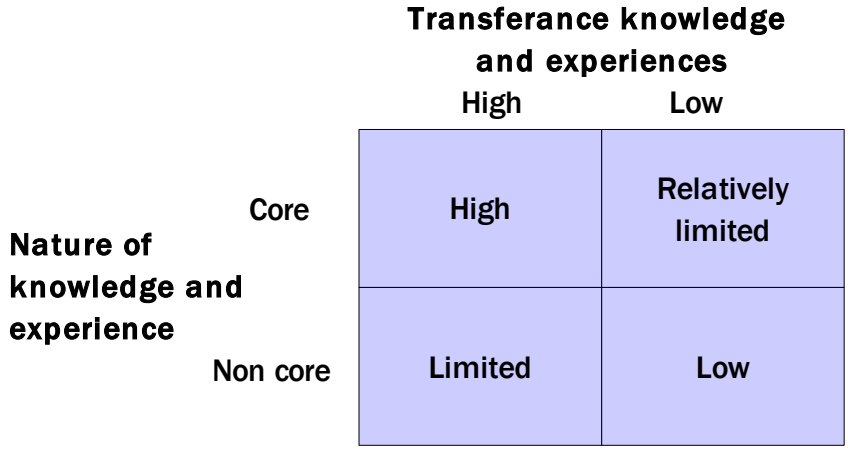

Figure 2.5: Risks of knowledge transfer in strategic alliances [DOU97]

Organisational fit

Organisational fit describes the degree to which organisational similarities or differences either hinder or stimulate successful collaboration, the degree in which the intended alliance design enables the partners to overcomes potential strategic conflicts and whether or not the alliance design enables the partners to realise their alliance objectives.

Organisational fit is determined according to several criteria. Douma uses generic criteria to be able to evaluate tailor made alliance designs, so organisational fit is the degree to which the alliance design meets these criteria. These criteria will be discussed in the next four sections.

Flexibility

From the perspective of the company as a whole, an alliance offers a great degree of flexibility, compared to mergers or acquisitions, because not all resources need to be dedicated to one strategic option. From the alliance perspective flexibility is also important, as alliances that changed the scope in the course of time are more successful than alliances that haven't. [BLE92]

Strategic flexibility, the ability to adapt the alliance strategy to changing circumstances, is about maintaining strategic fit, while organisational flexibility is the ability to adjust the organisation and functioning of the alliance. The main issue in the design and management style of the alliance is to establish clear agreements while retaining flexibility. [DOU97]

The degree of flexibility partners are willing to build into the alliance depends mainly on the mutual trust and strategic fit. Trusting partners discuss potential problems earlier, and therefore react faster to changing circumstances.

Management control

Co-operation means that control is being shared with partners. Management control is about the influence individual partners have on the alliance policy and activities. The need for control is determined by [KIL88]:

- strategic interest of the alliance, the higher the strategic interest for a company, the higher the degree of management control attempted.
- uncertainty surrounding the alliance, the higher the uncertainty of future developments, the higher the need for control.
- fiduciary risk, to keep the fiduciary risk, the risk that the partner will not do what is expected of a good partner, manageable sufficient management control is needed.

Relative management control depends on relative bargaining power and relative ownership. The relative control of the alliance in comparison to relative bargaining power and relative ownership is shown in figure 2.6.

Relative ownership	Relative bargaining power: smaller than partner	equal	smaller than partner
less than partner	Very weak	Weak	Shared
equal	Weak	Shared	Strong
More than partner	Shared	Strong	Very strong

Figure 2.6: Gaining control through ownership and bargaining power [DOU97]

Relative bargaining power is determined by the partners strategic positions and the resources partners are willing to commit. Strong bargaining power should not be too strong, as this might result in unequal advantages for the partner.

Complexity

Alliance complexity is divided in task complexity an organisational complexity. Task complexity should be limited as much as possible which can be achieved by limiting the scope of the alliance or by a strict division of the tasks [DOU97]. A strict division of tasks also limits the chance of unwanted knowledge transfer.

When organisational complexity increases, the chance of alliance failure also increases. Organisational complexity is caused by the amount of organisational alignment and task complexity.

Trust

Buckley states that without trust in the partner's commitment the chance of success is slight [BUC88]. There are two kinds of trust in alliances: rational trust and emotional trust. Rational trust means that both partners assume that the other party will not display opportunistic behaviour, because the interest in the alliance high and emotional trust is based on personal relationships and informal contacts.

2.5 Chapter summary

Consisting of two parts, this chapter described the theoretical background for this research. The first part discussed research by Ellram about partnering success factors and partnering pitfalls. The Ellram research can be ordered according to a theoretical model from the research by Douma, which was discussed in the second part of this chapter. In the next chapter will be discussed what OS business is.

Chapter 3. Open source business

3.1 Introduction

What is OS business? That is the first research question of this research and will be explained in this chapter. This chapter starts with the explanation of the concept of OS by discussing its definition in section 3.2. How OSS is being developed will be discussed in section 3.3. OSS development depends heavily on the license of the software being produced, which will be discussed in section 3.4. In section 3.5 and 3.6 the advantages and disadvantages of OS will be discussed. OS business models differ from proprietary business models, because companies can only profit from products and services complementary to the OSS. OS business models will be discussed in section 3.7. This chapter ends with describing two successful OS companies, Trolltech and Red Hat in section 3.8.

3.2 Definition of Open Source

The term Open source refers to a piece of software from which the source code is obtainable for everyone for use and/or modification, free of charge. This term originates from the Open Source Initiative (OSI), which is a splinter group of the Free Software Foundation (FSF) A part of the free software community splintered off, because of a disagreement with the goals of the FSF, forming the OSI in 1998. The FSF describes free software according to four types of freedom [FSF07]:

1. The freedom to run the program, for any purpose.
2. The freedom to study how the program works, and adapt it to your needs. Access to the source code is a precondition for this.
3. The freedom to redistribute copies so you can help your neighbour.
4. The freedom to improve the program, and release your improvements to the public, so that the whole community benefits. Access to the source code is a precondition for this.

OSS, the term used by OSI, and free software, the term used by the FSF, are nearly the same in the practical sense, but they are based on different goals and values. OS is a development methodology with the practical goal of improving the software, whereas free software is a social movement that considers non-free software a social problem for which free software is the solution. Although the goals and values of both groups differ, they often work together on practical projects, such as software development [FSF07].

The OSI describes the distribution terms the software has to comply to as follows [OSI07]:

1. **Free Redistribution:** The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.
2. **Source Code:** The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost, preferably downloading via the Internet without charge. The source code must be the

preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

3. **Derived Works:** The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
4. **Integrity of The Author's Source Code:** The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.
5. **No Discrimination Against Persons or Groups:** The license must not discriminate against any person or group of persons.
6. **No Discrimination Against Fields of Endeavour:** The license must not restrict anyone from making use of the program in a specific field of endeavour. For example, it may not restrict the program from being used in a business, or from being used for genetic research.
7. **Distribution of License:** The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8. **License Must Not Be Specific to a Product:** The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.
9. **License Must Not Restrict Other Software:** The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.
10. **License Must Be Technology-Neutral:** No provision of the license may be predicated on any individual technology or style of interface.

We conclude this section with the definition of OS used in this research:

> *Open Source Software is software that complies to the distribution terms of the OSI.*

3.3 Open source development

OSS development depends on communities of people working for free. There are three levels of participants in OSS development: non-developers, co-developers and core developers. The activities they are involved in and the levels of participation are further explained in figure 3.1. Transitions between the various levels of participants might occur with several steps at a time.

It might seem trivial that OS companies develop their software according to the OSS development paradigm, but many OS organisations, MySQL AB for example, don't fully profit from the advantages (section 3.5) this development paradigm provides when applied completely.

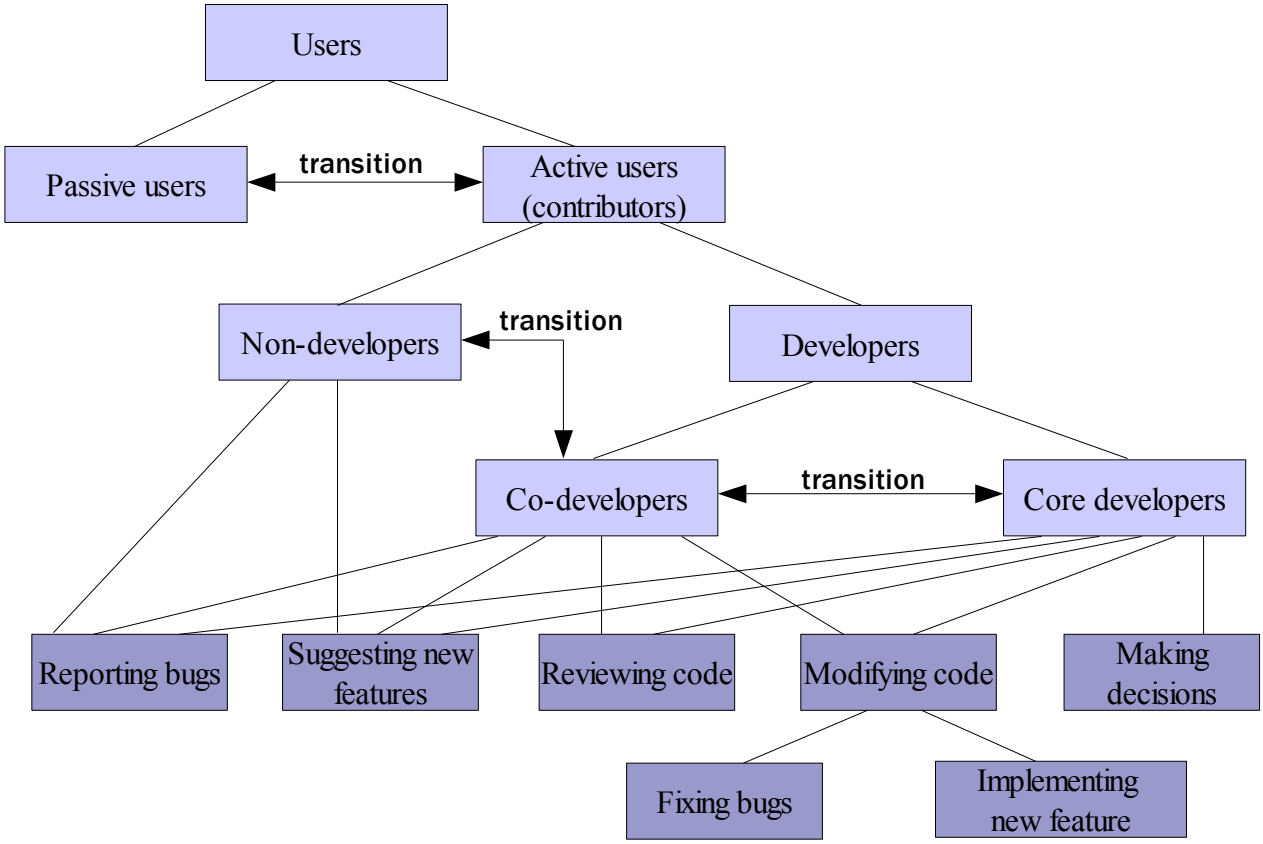

Figure 3.1: Classification of OS users, developers and their activities [BON03]

The OS development paradigm is being tested in other industries than software development. The OSCAR project, for example, uses the OS design paradigm for the development of an OS car [OSC07].

3.4 Open source licensing

Licenses under which software can be released can be divided into two main categories. Software released under a proprietary license is mostly distributed as machine code. This is compiled source code, which is unreadable by humans. The second licensing method, the OS licensing method, provides some way to obtain the source code of the program. There are many OS licenses, some more restrictive than others. Any OS license can be classified according to three qualifications:

1. **Free software** is a matter of users' freedom to run, copy, distribute, study, change and improve the software. It is a matter of liberty, not of price [GNU07].
2. **Copyleft** is a general method for making a program or other work free, and requiring all modified and extended versions of the program to be free as well. To

copyleft a program, it is first stated that it is copyrighted. Then distribution terms are added. These are a legal instrument that gives everyone the rights to use, modify, and redistribute the program's code or any program derived from it but only if the distribution terms are unchanged, making the code and the freedoms legally inseparable [GNU07].

3. The qualification whether the license is compatible with GNU's General Public License (GPL) means whether it is possible to combine a module which was released under that license with a GPL-covered module to make one larger program [GNU07].

Profits

In order to maximize profits from OSS development, several authors recommend to choose “non-copyleft” OSS licenses or to use an OSS and a proprietary license in parallel [BEH99], [HEC99], [RAY99]. Successful companies such as Red Hat and Trolltech, which will be discussed in section 3.8, use this licensing model.

License impacts

The license choice in OS depends on many factors. A license choice that is privately optimal, may not be socially optimal. The choice of a license impacts [LER02]:

- The community of programmers who are asked to work on their project, as its benefits from working on the project may depend on the license.
- The end users, who may for example care about possible incompatibilities among versions or about the number of available applications. The choice of license, by affecting the likelihood of forking or the incentives of application developers, therefor impacts their welfare. The forking of a project means taking the source code and starting a new project. Forking is further explained in section 3.6.
- The other OS projects that later will compete with or complement the project. For example, a GPL program may prove of no use for another OS project licensed under a BSD license that could otherwise have made use of the program. BSD style licenses do not restrict users to redistribute derivative works under the same license.
- Commercial software vendors and support providers, whose opportunities are affected by the license.

Several benefits will be assessed when selecting a license [LER02]:

- The intrinsic motivation that the intellectual challenge provides.
- The signaling benefits.
- The need to solve concrete problems for one's employer.
- The possibility of material benefits.

A case in point is the choice of license by programmers trying to get software established as a standard. Although they involve risks of hijacking, unrestrictive licenses make more sense than restrictive in such a context. This conjecture leads us to anticipate that projects geared toward the Internet, where standard setting has been particularly important in recent years due to the immaturity of key technologies, might be less likely to have highly restrictive licenses [LER02].

3.5 Advantages of open source

The advantages of OSS development can be divided into four categories, according to Krishnamurthy [KRI03]:

Robustness

The OSS development methodology could potentially lead to a more robust product. In OS development a large number of developers and testers can test the product under different kind of conditions, whereas proprietary software companies don't have access to such a community. Krishnamurthy refers to the robustness definition of Neumann [NEU99] which includes meaningful security, reliability, availability and system survivability. The robustness mentioned by Neumann is a subset of the ISO9126 standard for software quality. A summary of the characteristics of this standard can be found in appendix 5. Research has shown that some OSS products are more robust than their proprietary counterpart, but little quantitative research has been conducted on the effect of OSS development on the quality of software according to ISO9126 [SAM03]. Therefore it is not by definition true that OSS has better quality than proprietary software.

Shaikh and Cerone [SHA07] describe three main notions of quality for OSS: quality by access, quality by development, and quality by design. The relationships between factors in these three notions are described in figure 3.2. Boxes represent the factors and arrows represent their dependencies.

1. Quality by access is determined by a suitable format for the purpose of review, development and free distribution, an accessible medium such as the Internet and unrestricted access to the code and documentation.
2. Effective communication and management, the choice of programming languages and the choice of testing strategy are the three most important factors affecting quality, according to Shaikh and Cerone [SHA07]. These three factors form the core of the quality by development measure in their model.
3. Quality by design is defined as a measure of the use of recognised software design and engineering techniques and the production and frequent update of appropriate and explicit documentation.

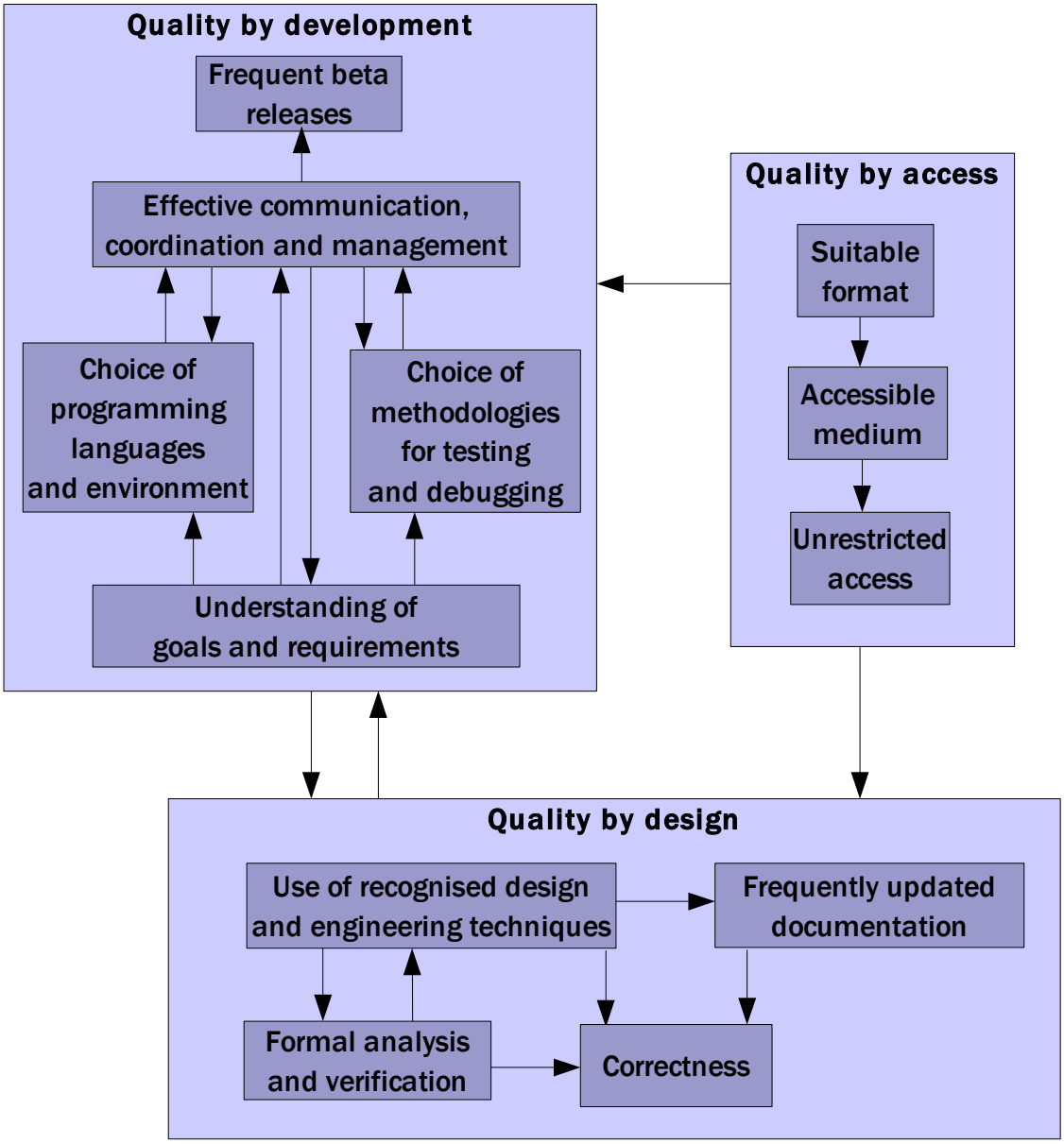

Figure 3.2: OSS quality notions and their dependencies, edited from [SHA07]

Flexibility to the user

Proprietary software producers try to protect their intellectual property by introducing non-standard file formats and other techniques that prevent users to mix-and-match. In this way users are forced to use software products from a selected group of companies. This is also called proprietary lock-in. Because of the extensive use of standards in OSS development and reverse engineering of proprietary file formats and protocols, users aren't tied to commercial vendors. They can mix different software, OS and proprietary, as suits them. An example of this is Microsoft's Windows networking protocol that has been reverse engineered by the OS community. Called Samba, it can connect Microsoft Windows and Linux/Unix operating systems to interact on the file system level.

Support from a community

Slow response rates, paid phone numbers or service and a poor level of quality are common features of support in proprietary software companies. With OSS one has a highly motivated community willing to answer questions [LAK03]. Creation and managing such a community is an important issue for OS companies.

It is also this community support that enables fast bug fixing. Tight schedules and low budgets often limit testing and bug discovery in proprietary software development. To enable fast bug fixing, fast and effective communication between developers, users and developers and users themselves is established through the use of Internet related tools. The use of these tools boost motivation to submit feature requests, bug reports and bug fixes, by treating users as co-developers [FEL02]. Web 2.0 techniques, such as users and moderators rating other users' comments and contributions are used in order to increase the status of users.

No vendor lock-in

Vendor lock-in occurs when customers are dependant on a specific vendor for products and services. Customers can not move to other vendors without making substantial switching costs. Many vendors try to achieve vendor lock-in in order to ensure themselves of clients in the future. Microsoft, for example, uses many proprietary API's in order to ensure themselves that independent software vendors make software compatible to their products, so users encounter high switching costs when moving to another platform. In some cases the European Union doesn't allow vendor lock-in. Another example is Apple Inc., which has profited a long time from iTunes, because it enabled users to upload music to their popular iPod music player. By using OSS, vendor lock-in can be avoided, because the OS community tries to use standards where possible and because of the fact that anyone can modify and distribute OSS.

3.6 Disadvantages of open source

Version proliferation

The version release structure used for Linux was meant to satisfy two groups: developers and enterprise customers [SPR00]. Even-numbered releases represent relative stable version targeted at corporate users while odd-numbered releases are developmental versions with new functionality. This creates a mass amount of different versions, making it hard for potential users to choose the right version and making it hard to support by companies.

Forking

When users don't agree with the goals or some of the functionality of an OS project, the OS paradigm allows him to start their own project based on the other project. This phenomenon is called forking. Instead of working together on some large project, users start smaller projects that might be far less successful. There are many Linux distributions to choose from, (Red Hat, Debian, SuSe, Xandros, Knoppix, Fedora, Gentoo, Slackware and many others) all with their own specific functionality. Communities might divide among different forks and choosing the right fork might become more difficult.

Usability

Many OS projects that have been badly structured and lack resources, suffer from poor usability. Other causes for poor usability are lack of usability expert support to the project, the fact that usability problems are harder to specify than functionality problems, higher incentives for development of functionality and bloated code [NIC03]. Nicols describes several ways usability problems can be dealt with:

- involve companies in the development of better interfaces
- automated evaluation of interfaces
- academic involvement
- end user involvement
- creating a usability discussion infrastructure
- fragmenting usability analysis and design
- involving the experts
- education and evangelism
- interface specification method

Usability is a part of the technology acceptance model (TAM) by Davis [DAV85], which models how users come to accept and use a technology (see figure 1.1). OSS projects aiming for high adoption should understand this and take measures to coordinate the usability of the OSS.

3.7 Open source business

Open Source business definition

By now it should be clear what OS is, so the OS business model can be explained further. The definition for OS business in this thesis is:

> *An Open Source company is a for-profit company that produces Open Source Software, be it by itself or through a community that it coordinates. The company generates the main part of its revenue from products and/or services complementing the Open Source Software.*

Motives

The motives for firms to contribute to OSS development can be grouped in five categories [HEN04]:

1. Setting a standard and enabling compatibility
2. Increasing demand for complementary goods and services
3. Benefiting from external development support
4. Signaling technical excellence and/or good "OSS citizenship"
5. Adapting existing OSS to the firm's needs

Most OS companies combine several of these motives, but the most important motive for full OS companies, such as Red Hat, Trolltech, MySQL AB and Alfresco is increasing the demand for complementary goods and services. Companies want to make profit and because OSS is by definition free, companies must make money from the complementary goods and services.

Business models

In order to analyze OS business models, one should first define business models in general. According to S.M.Shafer [SHA04] the definition of business models is:

> *"A business model is a representation of a firm's underlying core logic and strategic choices for creating and capturing value within a value network."*

This definition derived from twelve other definitions, placing forty-two components into four categories. (core logic, strategic choices, creating and capturing value, value network). The first category is core logic. A well-described business model articulates cause-and-effect relationships and the (internal consistency of) strategic choices made. The second and third categories are creating and capturing value, which describe two functions that are indispensable for conducting business. Successful value creation by differentiating from competitors does not mean a company can successfully convert this value in monetary value. (value capturing) Both value creation and value caption occur within a value network, which could include suppliers, partners, distribution channels and coalitions that extend a company's own resources. Figure 3.3 is a schematic reflection of the business model definition.

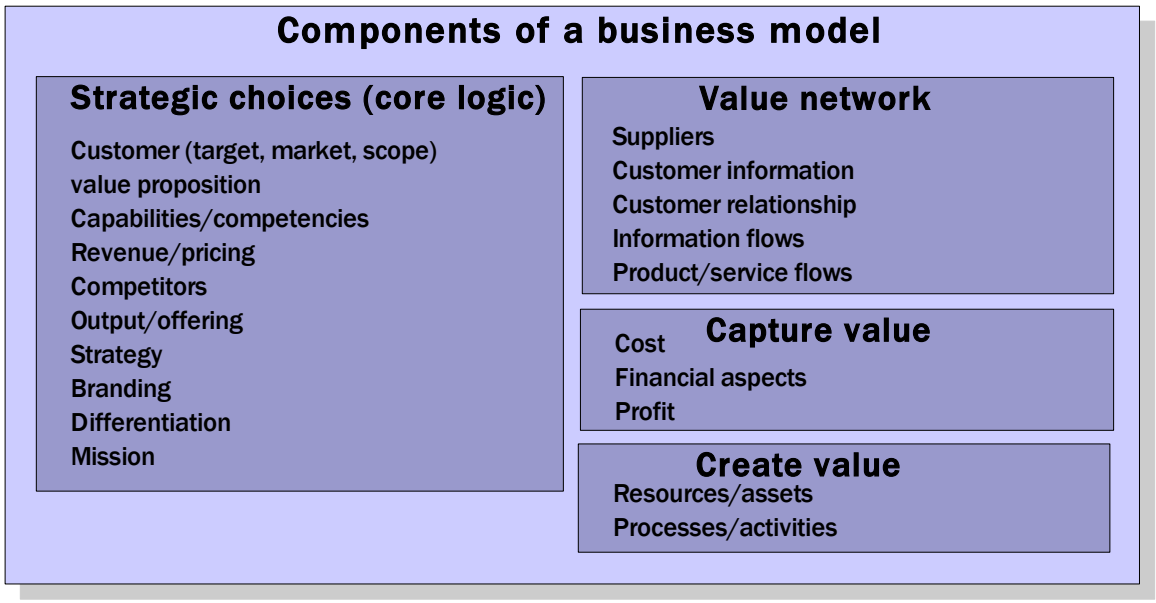

Figure 3.3: Components of business model affinity diagram [SHA04]

Strategic choices affect the creation of value, the capturing of value and the value network and vice versa. This thesis describes some of the strategic choices to be made in order to enable effective collaboration in an OS business model.

Value creation

Resources

Firms create value for different purposes, but all these purposes diverge to one reason, namely to generate revenue. There are several kinds of value. Use value is the specific quality of a new job, task, product or service as perceived by users in relation to their needs, such as the speed or quality of performance on a new task or the aesthetics or performance features of a new product or service. Exchange value is the amount paid by the buyer for the perceived use value [BOW00]. The greater the perceived novelty and appropriateness of the task, product or service under consideration, the greater the potential use value and exchange value to the user [LEP07].

Value creation consists first of all of resources and assets and the secondly of processes and activities. In OSS development the main resource is people.

Next to the OS community, there are other groups that can add significant value to any OS project [PAL02]:

- **Domain Experts**
  Domain experts essentially are industry experts with significant experience, and help bring in the perspective of the end-user/customer. They add a significant value in terms of developing a high level design of the software.
- **Technology Gurus**
  Technology gurus are the technology stalwarts, who have spent considerable amount of time working with technology, and have a significant expertise. They

are invaluable when it comes to fixing a critical bug, or designing new features over existing code base.

Processes

When the organization is the source of the value creation, issues regarding innovation knowledge creation, invention and management gain prominence [LEP07]. The project leader is responsible for the OSS development process, which can be organized into five major parts [MAR07]:

1. Communication and documentation
2. Revision control
3. Build management
4. Testing
5. Release creation

Developers in OSS development achieve status by developing quality source code, not by writing documentation. Many software tasks, such as documenting, testing, internationalization/localization, and field support- are tedious, but vital, particularly as new cohorts of developers join and maintain projects [FIT04]. Communication and documentation is specifically important to OS companies, because sharing too much information might result in worse value capturing possibilities and sharing too little information results in worse usability and thus in lower adoption probability.

Value capturing

Value capturing is the realisation of exchange value [BOW00]. It is determined by the bargaining relationships between buyers and sellers. The presence of viable substitutes combined with low switching costs enhances the bargaining position of the buyer and reduces the value capturing capabilities of the seller. The amount of captured value depends on the perceived bargaining relationship between the seller and the buyer [BOW00].

Value network

A firm's relationship to the network of providers of complementary products determine its value creation, value caption and the durability of its competitive advantage; thus creating and managing these relationships is an important part of achieving each of these goals [HAM00], [IAN04].

Because partners might become competitors, literature about coopetition, cooperation with competitors, is relevant.

In comparing vertical and horizontal relationships, it can be stressed that vertical relationships are often build upon a mutual interest to interact, whereas competitors are often forced to interact with each other, giving rise to rivalry and mutual dependence between them. Even though similarities can be found, vertical and horizontal relationships are, in many senses, totally different relationships, and it is obvious that the trade-offs between cooperation/harmony and the competition/conflict in vertical and horizontal relationships, respectively, are of different nature and accordingly have to be managed differently. The two different types of interaction are not divided between

counterparts but between activities, as it is impossible to compete and cooperate with the same activity [BEN00].

The value created and captured by a firm is determined by its position in the value chain. Therefore a firm's business model must define its role in the value chain [AMI01], [CHE02], [MAR02].
Many firms also depend on the supply of third-party complements, and thus the firm's strategy depends on the total value network of suppliers, buyers, complementers and other allies [AMI01], [CHE02], [AFU00].

For OS initiatives to succeed it is important that the firm devise multi-usage alliances. Each of the partners should find the OS complementary and offering network externalities [PAL02].

In order to map the value network for abstraction purposes, one can analyse it in five steps [PEP06]:

1. Define the network
2. Identify and define network entities
3. Define the value each entry perceives from being a network member
4. Identify and map network influences
5. Analyse and shape

Value created by one source or at one level of analysis may be captured by another (value slippage) [LEP07].

OSS initiative multiplicity and market multiplicity

A two-dimensional model describes the consequences of choices made concerning entering multiple markets and the amount of OSS initiatives sponsored [PAL02].

Type	Type of company	Consequences	Example
Single OSS Initiative, Single Market	smaller companies, Single-product Company, or having a portfolio of related products for a single vertical market	easy to operate	
Single OSS Initiative, Multiple Markets	Large organizations, that have a portfolio of products or services for multiple vertical markets, with any single technology focus	leverages the investment made in the OS initiative across multiple products for different vertical market segments or domains (probably most profitable)	MySQL AB (MySQL database server)
Multiple OSS Initiative, Single Market	world leaders in a particular product/technology, and dominate the markets	the company spends considerable bandwidth and resources to initiate and manage multiple OS initiatives	lat/lon (OS Geo Information Systems)
Multiple OSS Initiative, Multiple Markets	typically large companies	Sustain multiple OS initiatives, and leverage them across multiple markets	SUN Microsystems (JAVA, Solaris) Red Hat (Red Hat Enterprise Linux, Jboss) Trolltech (Qt, Qtupia)

Table 3.1: OSS initiative and market multiplicity [PAL02]

In the *"Single OSS Initiative - Multiple Markets"* model, the OS initiative is chosen such that, the output of the initiative will provide a 'platform' which can be used to build various commercial products for different market needs. To adopt this model, the company should be operating in multiple markets with specific offerings in those markets. It should have significant engineering resources to support an OS initiative and collaborate with the public-domain community [PAL02].

3.8 Successful open source companies

Red Hat

Introduction

Red Hat is a global leader in providing OSS to the enterprise. Red Hat employs an OSS development and licensing model that uses the collaborative input of an international community of contributors to develop and enhance software. Red Hat actively participates in this community-oriented development process, often in a leadership role, and leverages it to create Red Hat-branded enterprise technologies.

Red Hat offers a choice of operating system platforms and other infrastructure enterprise technologies, such as technologies for the development and deployment of JAVA-based web applications. Furthermore Red Hat's integrated management services, provided via the so-called Red Hat Network (RHN), permits these technologies to be updated, configured, monitored and managed.

Red Hat enterprise technologies are provided under annual or multi-year subscriptions. Through the life of the subscription, the customer is entitled to specific levels of support as well as security errata, bug fixes, functionality enhancements, and upgrades to new versions of the technology via Red Hat's integrated management services, generally without additional charges.

Red Hat enterprise technologies are sold through both direct and indirect channels of distribution. Red Hat sells to customers directly through its sales force and its web store. Red Hat's indirect sales channel includes distributors and resellers. In addition, leading global server and workstation hardware vendors support and pre-load Red Hat enterprise technologies on various servers and workstations and also sell their hardware together with Enterprise Linux as part of pre-configured solutions.

Business model

Red Hat offers and provides their Enterprise technologies, and its related services, to their customers in the form of annual or multi-year subscriptions, generally on a per installed system basis.

In contrast to the lock-in of traditional proprietary technology, through the life of the subscription Red Hat provides the customer security errata, bug fixes, functionality enhancements and upgrades to new versions of the technology, which they provide through RHN, as well as specific support levels.

Red Hat believes the success of its business model is predicated on:

- the acceptance and widespread deployment of Enterprise Linux as a significant platform by the large enterprise
- the ability to generate subscription revenues on a per installed system basis for Red Hat enterprise technology

- the ability to increase annual average subscription revenues per customer by providing additional value to our customers in the form of additional technology infrastructure
- providing customers with additional services

Red Hat's subscription business model defers revenue when they bill customers and recognizes revenue over the life of the contract. Under a proprietary license business model revenue is typically recognized when software is licensed/sold.

Revenue distribution

Category	2004	%	2005	%	2006	%
Subscriptions	82,408	60,3 %	151,125	68,4 %	230,444	74,7 %
Services	42,329	31,0 %	45,341	20,5 %	47,886	15,6 %
Other revenues	11,866	8,7 %	24,431	11,1 %	30,053	9,7 %
Total	**136,603**	**100 %**	**220,897**	**100 %**	**308,383**	**100 %**

Table 3.2: Red Hat analysis of sales by category in USD 1,000 [RED05]

As shown in table 3.2 Red Hat total sales have grown 125% in two years. All sources of revenue have grown, but the biggest growth comes from subscriptions (180%), followed by other revenues (153%). The 13% growth in services is small compared to subscription and other revenues, so it is obvious that Red Hat doesn't focus on this area. The growth rates have been visualized in figure 3.4 and 3.5.

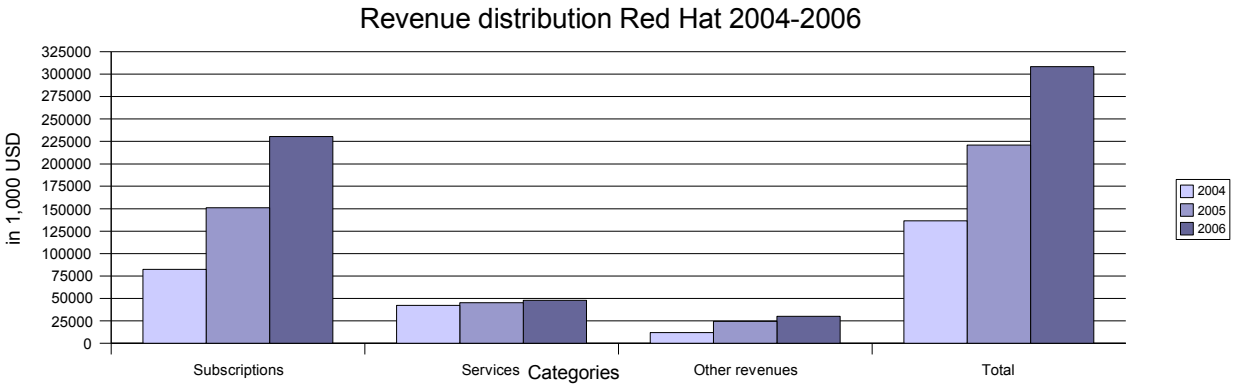

Figure 3.4: Red Hat analysis of sales by category in USD 1,000 [RED05]

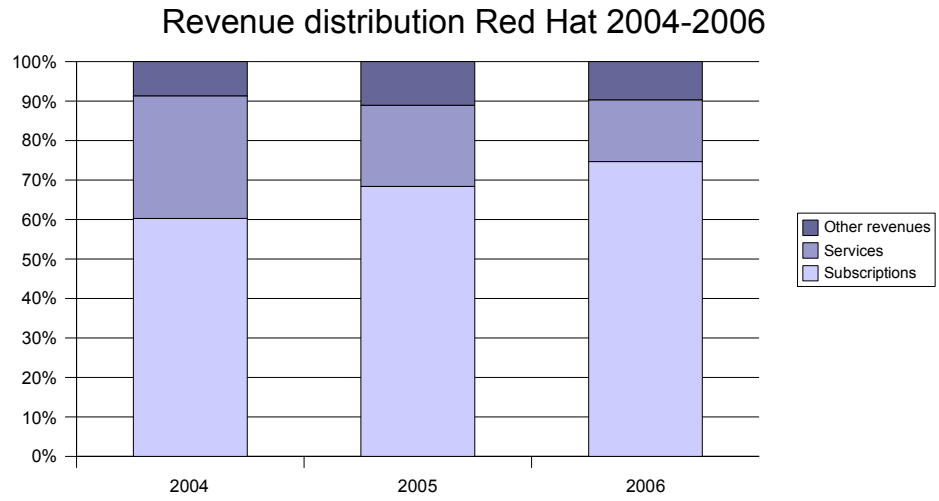

Figure 3.5: Red Hat revenue distribution per year [RED05]

Trolltech

Introduction

Trolltech ASA was established in Oslo, Norway in 1994 with the vision to build the best cross-platform C++ GUI tools around.

The company's family of products include Qt, which sets the standard for high-performance, cross-platform software development; and Qtopia, the unrivaled application platform for the efficient creation of Linux devices.

Trolltech has successfully established itself as a key player and major contributor in the OS world, with thousands of OS projects, most notably KDE, which is built with Trolltech technology.

The company provides three kinds of services: support, training and consultancy. Support comes in two variants, standard and premium.

License and post contract service (PCS)

Fees for software licenses are fixed and non-refundable. Once the license is delivered a Group entity has no remaining obligations to perform. Fees for software licenses in that case are recognized when a group entity has delivered the software license to customer. PCS, which gives the customer a right to upgrades and e-mail support, is recognized ratably over the service period.

Other services

Revenue on fixed price projects, such as e.g. engineering service, is recognized based on percentage of completion method as work progresses and service is performed. However, if outcome of the contract cannot be measured reliably, revenue is only recognized to the extent of the expenses recognized that are recoverable. Estimated losses on fixed-price service arrangements are recognized as an expense when it is probable that total

contract costs will exceed total contract revenue. Revenue from training and consulting service elements is generally recognized as the services are rendered.

Business model

Trolltech released its desktop developer tools under a dual licence. The dual licensing business model allows software companies to provide their products for two distinct uses: commercial and OSS development. In return for the advantages realized from using a Trolltech product to create an application, there are two options:

1. a commercial license is purchased from Trolltech in order to contribute to the continued development of the product. This option secures the right to distribute the application under the license terms of the developer's choice.
2. The application is placed under an OS license (e.g. the GPL) in order to contribute to the OS community. This option secures all users the rights to obtain the application's full source code, modify it, and redistribute it.

Customers that use Qt software to develop software in an embedded device pay a price per unit sold. Fixed non-refundable run-time fees paid for a pre-defined period are recognized ratably over the contract period. However, if the customer reports sales of the embedded device that exceeds the number of embedded devices already paid for as part of the fixed fee, the incremental run-time fee is recognized based on sales reports received from the customer. Other run-time fees that are paid for an unspecified period are recognized based on sales reports received from the customer.

Revenue distribution

Category	2004	%	2005	%	2006	%
Sales of licenses	48,329	56,7 %	57,817	48,8 %	87,672	51,4 %
License subscription and support	33,104	38,9 %	48,138	40,6 %	65,087	38,2 %
Engineering and consulting	3,147	3,7 %	7,311	6,2 %	14,734	8,6 %
Other revenues	627	0,7 %	5,273	4,4 %	3,123	1,8 %
Total	**85,207**	**100 %**	**118,539**	**100 %**	**170,616**	**100 %**

Table 3.3: Trolltech analysis of sales by category in NOK 1,000 [TRL05]

As shown in table 3.3 Trolltech total sales have grown 100% in two years. All sources of revenue have grown, but the biggest absolute growth comes from the sale of licenses. The Trolltech growth rates have been visualized in figure 3.6 and 3.7. The exchange rate of USD to NOK is 1 to 7.3921411 (per 30/10/2007)

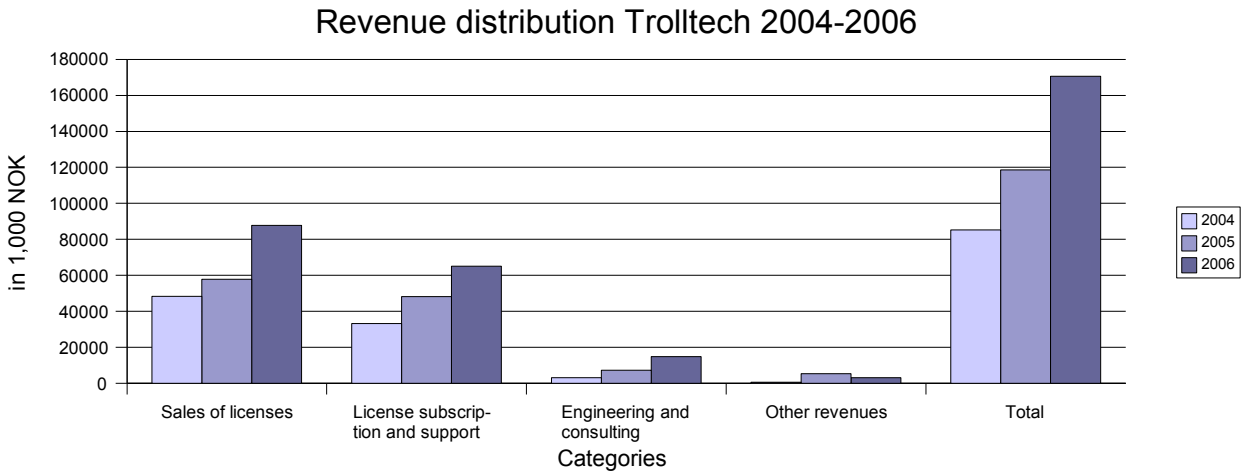

Figure 3.6: Trolltech analysis of sales by category in NOK 1,000 [TRL05]

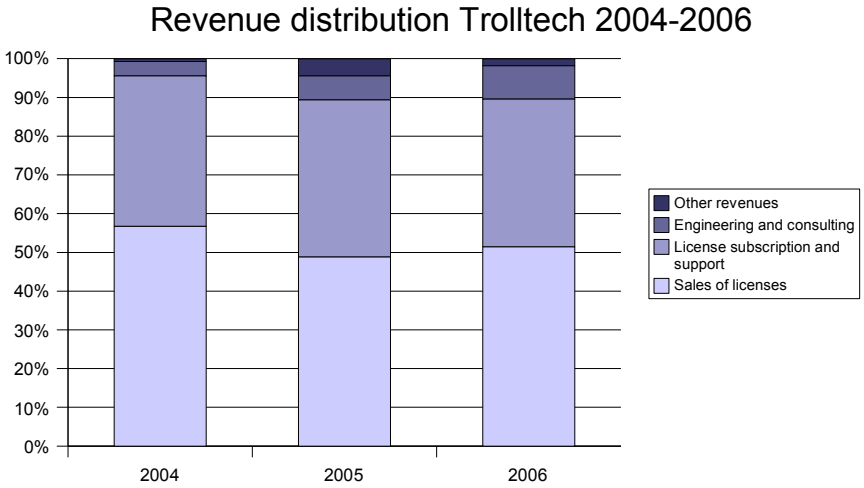

Figure 3.7: Trolltech revenue distribution per year [TRL05]

Conclusions from successful open source companies

Both Red Hat's and Trolltech's financial figures show that OS provides a solid base for generating revenue. Although these companies use different business models, both show healthy financial figures. While the Trolltech example confirms the statement by several authors to use "non-copyleft" OS licenses or to use an OS and a proprietary license in parallel [BEH99], [HEC99], [RAY99], the Red Hat example shows that solid branding can well be combined with the subscription model.

3.9 Chapter summary

In this chapter the definition of OS and OS business was explained. It was explained when software is called OS, to what terms the OS license must comply to and what the advantages and disadvantages of OSS are. The focus in this chapter was on the OS business model. Eventually the business model and the figures of two major successful OS companies and why they are successful have been discussed.
In the next chapter will be discussed how OS companies collaborate with other companies.

Chapter 4. Collaboration in OS business

4.1 Introduction

This chapter will shed some light on vertical collaboration in the OSS field. First vertical collaboration in OS is defined in section 4.2. Then the OS collaboration business model is discussed. In sections 4.4 to 4.7 the different types of collaboration with OS companies are explained and in section 4.8 the partnering model of Alfresco is discussed.

4.2 Definition

Vertical collaboration is collaboration in the value chain. The software value chain (see figure 4.1) consist of [FLO02]:

- Software development
- Software documentation
- Software packaging
- Marketing and sales
- Consulting
- Implementation/integration
- Training
- Support
- Application management

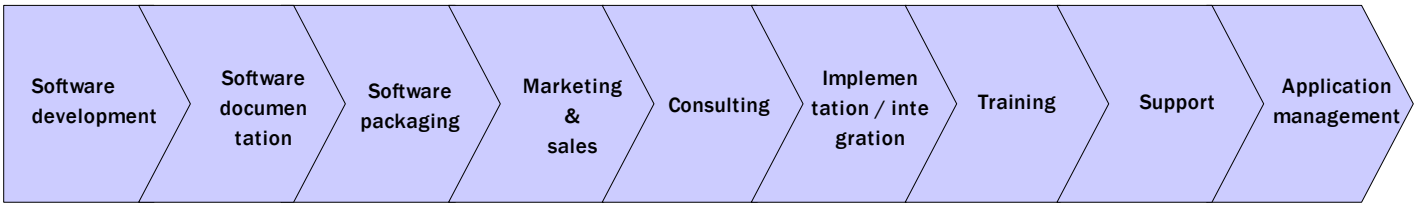

Figure 4.1: Software value chain [FLO02]

Companies might collaborate with others in for example software development, distribution and training to reduce costs. Collaboration could provide the OS company with a competitive advantage over others. Within the scope of this research, the definition of vertical collaboration has been widened to include collaboration in the software stack and with hardware providers, which leads to the definition of vertical collaboration in OS:

> *Vertical collaboration in Open Source is collaboration in the value chain, collaboration in the software stack and collaboration with hardware providers.*

4.3 OS collaboration business model

Value creation

R&D is the main source of value creation in OS. The use value of OSS is very high. Because of the use of open standards the flexibility to the user is high. The software is highly robust most of the time and its community also provides some extra value. R&D mostly takes place in the OS community, but when companies initiate the projects, most R&D is done within the company. Within the community or company domain experts have most expertise in required functionality of the OSS. Usefulness according to the TAM will increase when domain experts are involved. System integration partner companies might contribute in R&D depending on the importance of the OS to the company strategy and the usefulness of the OSS to the company software portfolio. Whenever the OS company that is the main OS project contributor ceases to exist for whatever reason, the system integrator will take over their role in the OS project. The clients of the system integrator are insured of continuation of the project in this way.

In order to implement desired functionality into the OSS, system integrators can first of all simply pass a feature request to the project. This is the easiest way, but it might take a long time before the feature is added to the branch. Another possibility is to find a specific project member to build the functionality for the integrator. Project members have more knowledge of the OSS and will therefor develop the functionality quicker. The system integrator might even temporarily hire the project member in order to develop the desired functionality. The last option for system integrators is to develop the functionality themselves, inside or outside the OS project.

Although OSS is in fact free, organizations don't always install the OSS themselves, but hire large system integrators to do the job for them. System integrators create value for their customers by combining experience, knowledge of the products, large resource pools and the ability to offer support, with the certainty of continuity.

Value capturing

Value capturing in OS is done by means of license choice, protection of value adding information and branding. Some very successful OS companies, such as Trolltech and MySQL AB, use a dual licensing business model in order to capture value. The principle behind dual licensing is quid pro quo. Under dual licensing, OS companies offer their products under both an OS license and a commercial license. Companies redistributing the OSS as part of their own commercial products can purchase a commercial license, which releases them from the obligation to publish their source code, while on the other hand OS projects and individuals can use the software at no cost. This creates value capturing opportunities for OS companies developing framework software that attract closed source complements. The exchange value of OSS is very low, because the OSS is free most of the time. Customers actually pay for the services that are combined with the OSS. Red Hat uses branding together with a subscription model in order to capture value. The Red Hat Trademark may only be used by official Red Had distributors. See section 3.8 for financial results of Trolltech and Red Hat.

Value network

The value network in OS business consists of seven actors, the community, the customer, the partners and the company itself. The values exchanged between these actors are represented by arrows, the actors by circles and optional value exchanges by dotted lines in figure 4.2.

OS companies collaborate with different types of partners differs in values that are being exchanged. The whole product comprises the OSS, under an OS or under a proprietary license, together with some value adding product or service, so the exchanged money, the exchange value, is being paid for the whole package.

Values are exchanged between OS companies and:

- customers: the OS company provides a well tested software package together additional hardware, software or services to the customer. The customer pays a subscription fee, a one time payment or both.
- the community: the OS community contributes to the development of the OSS by means of bug reports, feature requests, code reviews, bug fixes and the implementation of new functionality. The OS company provides the design of the software, roadmaps, development environment and the management of the overall project. The community can use the OSS under its OS license.
- system integration partners: system integration partners implement the OSS in their customer's infrastructures. The services they receive from the OS companies are related to implementing the OSS, such as training, certification and access to the OS company sales department. OS companies expect their system integration partners try to sell the supported edition of the software to their clients. In some partnering models, like Alfresco's, the partners might get a percentage of the subscription fee.
- platform partners: Platform partners provide their own product together with that of the OS company and sell it as a package deal. The partnership between Red Hat and Dell is an example of such a partnership. Dell sells computer hardware with Red Hat Enterprise pre-installed
- complementing partners: complementing partners provide soft- or hardware solutions based on the OSS. OS companies might earn revenue in two ways:

- The complementing partner provides the total package under a proprietary license in exchange of a license fee. For every package sold, a license fee will be paid to the OS company.
- The complementing partner sells its own product together with a subscription for the OS company product. In this case the complementing partner might get a percentage of the subscription fee.

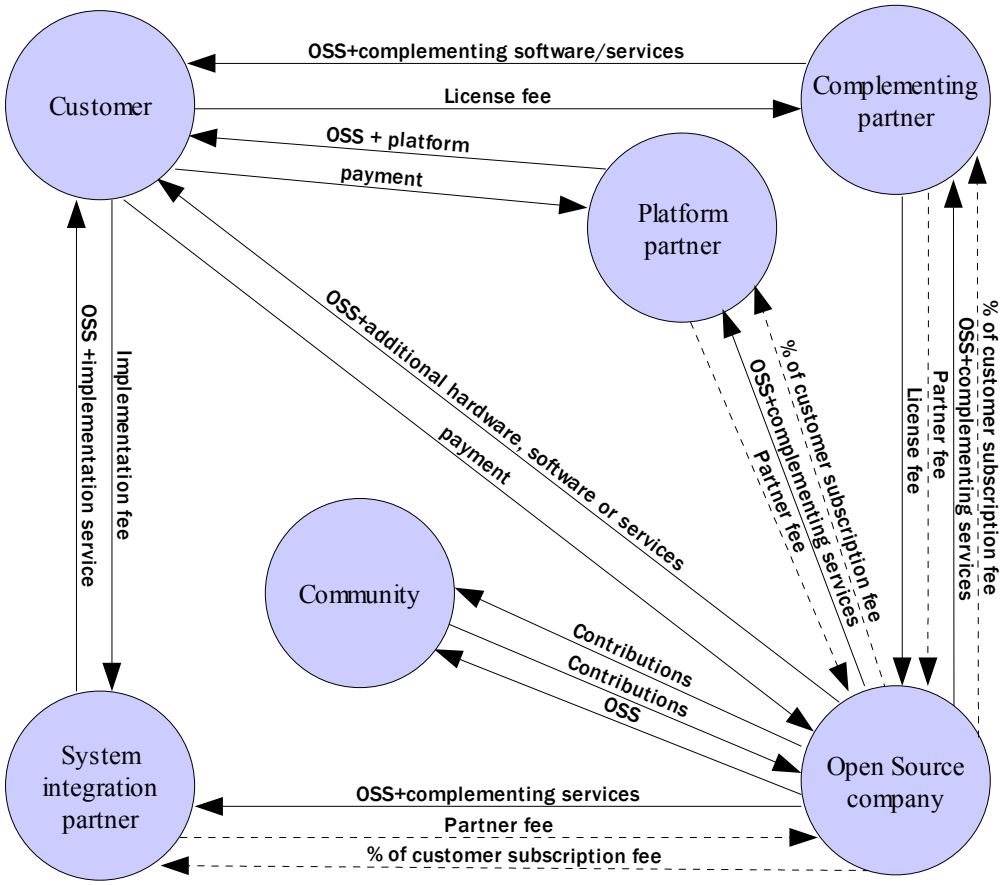

Figure 4.2: High level value network for OS companies

4.4 Collaboration typology in open source

In order to enable value capturing, OS companies must try to achieve widespread adoption of the OSS. The more the OSS is perceived as the standard in its field, the more complementing products and services can be sold.

Some factors are known to affect the adoption rate of software and in this research we will try to improve the adoption rate by improving effectiveness of collaboration to change negative effects of several of these factors and to amplify positive effects.

Portability is achieved by making the software compatible with platforms layers down the soft- and hardware stack while software complementing the OSS is situated up the

software stack. Partnering up and down the soft- and hardware stack assures OS companies of compatibility with the hard- and software and it enables improvements, which is in favour of the adoption of the OSS. Partnering in the software value chain is triggered by a combination of a lack of resources, a lack of knowledge and a focus on core business. While marketing might also influence the perceived usefulness of the technology, this research only focusses on R&D and distribution.
In order to distinguish different kinds of collaboration in OS companies, we will use the software value chain and the software stack as dimensions. A software stack is a collection of *n* pieces of software, where *n* is dependent on *n-1*, *n-1* is dependent on *n-2*, and so forth. A well known OS software stack is LAMP, which stands for the Linux operating system, Apache webserver, MySQL database server and the Perl or PHP interpreter. These dimensions derived from literature on:

- the quality of software (portability according to ISO 9126 [CHU04])
- the ease of use of technology [DAV85]
- OS (the existence of complements [HAM00], [IAN04])

According to these dimensions, we identified four types of vertical collaboration in OS:

1. collaboration with system integrators
2. collaboration in R&D
3. collaboration with complementers
4. collaboration with platform providers.

These four types will be further explained in de next four sections. It should be mentioned that partners may collaborate with OS companies at more than one type of collaboration at one moment in time.

4.5 Collaboration in research and development

Description

One of the most successful and widely employed approaches to improve R&D performance and productivity in companies has been a cooperative strategy: R&D alliances [INK98].

Goals

The goals for partners collaborating in R&D can be divided into two categories. Shared goals and private goals. Shared goals are the improvement of R&D performance and productivity and private goals may vary from market penetration to company espionage.

Communication

Companies collaborating in R&D will protect their intellectual property. In order to minimize chances for opportunistic learning by partners companies might divide R&D across multiple sites.

Collaboration in R&D might occur across multiple sites in the world, which reduces (if not eliminates) informal, unplanned and ad hoc communication, which in turn is extremely important in supporting collaboration [CUR88], [HER99], [KRA95], [PER94]. In the

absence of this type of communication, R&D work across multiple sites should be thoroughly coordinated. This can be done through [GRI99]:

- functional areas of expertise: expertise for a specific functional area involved in development of the product is located at a single site. Improved load balancing and development and enhancement of expertise are benefits of this approach. Projects spanning multiple sites, on the other hand, are harder to manage than projects at one site. To avoid problems in this approach, detailed development processes and sound project planning is necessary.

- product structure: divides the organization across sites along lines suggested by the product architecture. This approach enables independent operating environments so individual components don't have to follow the same processes and tools. Testing individual components shouldn't produce extra problems, but testing the components manager that links all the individual components is harder, because of the dissemination of experts. Problems testing the components manager can be avoided by bringing component experts together when their expertise is needed. Another problem relates to features that might span components which requires extra coordination in the early design phase. Coordination across components is achieved by defining clear standards and interface specifications as well as identifying when and how components become available for testing and integration.

- process steps: work is broken up into process steps such as systems engineering and testing. These steps are then used as hand-offs among various locations. By locating parts of product development close to potential customers, local knowledge can be used. Another positive effect might be the better use of scarce resources. A negative consequence of using this approach is the hand-off point. It should be clear when, how and what is handed off. Furthermore, different priorities among sites may result in delays, even when process steps further along the line might have finished their part on time.

- customization: one geographical site owns the core code for the product and other sites involved in the project make changes to the code base for a specific customer base. By locating the customization part of production near the customers it is possible to obtain valuable information from them. The use of different standards in different locations might call for expertise that can only be obtained in that specific location. Knowledge about local infrastructure is also easier to obtain from the local site itself.

4.6 Collaboration with system integrators

Description

In the context of this research, system integrators implement third party software for their clients or advice them on software to install.

Goals

Goals of collaboration differ between OS companies and system integrators. System integrators provide their customers with solutions that best fit their problems. System

integrators don't get paid for the OSS itself, but for the consultancy and implementation services concerning the software and sometimes a percentage of the support delivered by the OS company (see the advantages of being an Alfresco partner in appendix 6). OS companies have access to larger markets through system integrators, because these markets don't want to deal with many suppliers. System integrators gain knowledge and expertise from the collaboration.

Communication

System integration partners mostly communicate by email and web-based tools with OS companies. Integration of these tools is non-existent, which has some downsides. First of all everyone using one of the many existing systems should be taught how they work. Second, administration of the different systems, for example user names, passwords and locations, takes time and money to maintain. Currently a SOAP webservice is available for the OS bug tracker Mantis, which means that the first steps to integrate bug trackers have been made [FUT07].

The IT business is changing rapidly all the time, which manifests itself in communication problems. There are several reasons for these communication problems. First of all, after initial projects there is no need to continue communication channels but the communication channel itself. Many organisations don't recognise this and cease the continuation of the communication channel. Second, agreements made between people, even documented ones, don't get passed on when people change jobs. These problems don't only occur in partnerships, but also between companies and their clients. Possible solutions for these problems include solid function handover, the creation of communication channels in multiple layers of both organisations and the appointment of a Customer Relationship Executive (CRE) that is in charge of all communication to large clients and important partners.

Effectiveness

System integrators measure the effectiveness of partnerships by a couple of factors. First of all system integrators want to make money, so the more profit a collaborative project makes, the more it is perceived effective. While money is important, the image of the system integrator is at least as important. Projects that have started will be finished even though they might not be profitable in the end. The collaboration is partly perceived successful if the image of the system integrator has not been damaged. The most important factor for evaluating collaboration effectiveness is client satisfaction, which is measured (yearly) by means of a survey. After evaluation of the survey by the department of quality management, measures are taken to increase client satisfaction where needed and possible. This department is also present in (large) projects to keep an eye on client satisfaction continually.

4.7 Collaboration with complementers and platform providers

Description

For platform OSS, OS companies depend on complementary software. These complements may either be OS or proprietary. OS companies may earn revenue by dual licensing the OSS in order to enable proprietary software vendors to develop complements without releasing the source code. Hardware providers are also considered platform providers.

Goals

In order to increase the demand for the OSS, OS companies collaborate with their complementers and with platform providers. They provide Application Programming Interfaces (API), documentation about the functionality of the OSS and. In the collaboration with complementers scenario both parties share the same goal: collaborate in order to generate more revenues from the sales of the own soft- or hardware.

Communication

Complementers need information about the functionality of the OSS and its API. The earlier they know about future changes and how these are implemented, the earlier their software is compatible with that of the platform. By publishing documentation, API's and roadmap on their (project) website, OS companies satisfy the information needs of their complementers.

4.8 Partnering according to Alfresco

Alfresco is the OS Alternative for Enterprise Content Management (ECM), providing Document Management, Collaboration, Records Management, Knowledge Management, Web Content Management and Imaging [ALF07]. Alfresco's solutions are divided into a community and a supported enterprise version.

Alfresco divides its partners into six categories:

- system integrators
- solution partners and OEMs
- technology partners
- hosting partners
- training partners
- resellers.

These categories will be shortly explained in table 4.1. There are three levels of partnerships, bronze, gold and platinum, each with its own costs and benefits.

System integrators	System Integrators provide expertise consulting, integration and migration services for Alfresco content management solutions. All System Integrators have trained and certified Alfresco consultants on staff.
Solution partners and OEMs	Solution Partners either provide complementary solutions for Alfresco content management solutions or embed (OEM) Alfresco directly into their products. In both cases, Alfresco certifies these solutions to ensure compatibility.
Technology partners	Technology Partners provide the infrastructure or platforms required for deploying Alfresco solutions. Customers can deploy mission critical solutions with confidence knowing that solution partners' products are certified and validated to work seamlessly in an Alfresco environment.
Hosting partners	Hosting Partners provide the managed infrastructure necessary to deploy Alfresco solutions. Many partners also provide direct support for the Alfresco Enterprise edition.
Training partners	Training Partners deliver Alfresco classes based on course material developed by Alfresco itself. Partners have certified Alfresco trainers on their staff.
Resellers	Resellers provide Alfresco branded support and services for a specific region. All reseller partners have been trained by Alfresco and can deliver support services to end customers.

Table 4.1: Alfresco partner types [ALF07]

Alfresco profits in three ways from the way they deal with their partners:

1. By letting partners share in revenue Alfresco creates a shared goal (to sell subscriptions for the supported Enterprise edition), so partners become more involved.
2. Alfresco profits from the fact that partners agree to abide by the Alfresco Network Certification Procedures. This way Alfresco is assured of a certain level of service delivered by their partners to third parties which at least maintains the strength of the Alfresco brand. Alfresco has API's and Javadoc available for other parties that want to make their software compatible with Alfresco's.
3. Alfresco partner program pays for itself. Bronze partners pay an annual fee of $2995, gold partners pay $9995 and platinum partners have to make a revenue commitment (per September 2007). The higher the level, the more benefits partners will have from the partnership. More supported deployments of the Enterprise edition, higher discounts on training and certification, higher royalties on support and access to some of Alfresco's resources are among these benefits. The whole list of benefits can be found in appendix 6.

Value slippage might occur at Alfresco when system integrators decide to install the community version of Alfresco software. Alfresco tries to prevent this by providing system integrators with royalties on support.

4.9 Chapter summary

In this chapter we first defined vertical collaboration in OS. Then we discussed the companies model of OS collaboration. The OS collaboration typology with a description for each of the types were explained in the next sections. This chapter ended with a description of the partnering model of Alfresco. In the next chapter the case study will be discussed.

Chapter 5. Case study

5.1 Introduction

In this chapter the research methodology of the case study will be explained. This research process includes two phases. During the first phase a case study has been carried out to gain more insight in vertical collaboration in OS business and in the second part a survey has been carried out in order to quantify the success and failure factors in OS collaboration and to investigate the effects of the quality of communication, the type of collaboration and trust on the effectiveness of the collaboration. This chapter will describe the methodology of the case studies in section 5.2.

5.2 Case study methodology

The amount of confounding variables, the complexity of the problem and the fact that data is needed that can only be obtained from the field, are the reasons for doing a case study during the first part of the research. Two cases were analysed in order to be able to replicate the results. Researchers should aim for what Yin terms Level Two Inferences [YIN03]. Level Two Inferences occur when two cases support the same theory, but do not support a plausible rival theory. This research is exploratory and no theory rivalling that of Ellram is used, so no Level Two inferences are present.

Goal

The purpose of the case study is to gain more insight in and and to formulate hypotheses about the problems concerning collaboration in OS business. The research of Ellram is used as a guide line during the case study. The factors that originate from the Ellram research were used as context and to abstract important subjects. The case study was also used in order to categorize types of collaboration.

Unit of analysis (case selection)

The case study will be conducted at two large, influential IT/consultancy companies in the Netherlands that will be selected by their openness to OS, knowledge of partnerships and their willingness to cooperate.

The companies that fit these requirements and their type of partnership that will be used in the multiple-case studies are Unisys and Bull.

Material

The research material to be consulted can be divided in two categories. In order to enable methodical triangulation during data analysis, interviews with management will be combined with documentation. An overview of the research material that will be consulted during this research (when available and allowed access) is shown in the list below.

Documents:

- collaboration directions
- strategy descriptions
- planning documents

Interviews:

- Technical implementers
- Project management

The main goal of the expert interviews was to check whether the Ellram research was transferable to OS collaboration. The expert interviews were conducted in a semi structured way according to the questions listed in appendix 3.

5.3 Case study design quality

Quality of case study design can be evaluated by construct validity, internal validity, external validity and reliability [YIN03]. Construct validity refers to whether measures accurately reflect the underlying construct. In this research construct validity is increased by:

- triangulation; using multiple sources, in this case interviews and documents
- a chain of evidence, which can be used to follow the derivation of the evidence in this research. The chain can be constructed with a case study database (all evidence), a case study protocol and the research questions, which can all be found in this document.
- A review of the case study reports by key informants.

Internal validity is a non-issue during the case studies, because the case studies are exploratory. External validity is the extent to which the results of this case study can be transferred to other OS partnerships. By analysing multiple cases, results can be generalized more. That is why two cases are analysed in this case study. The chain of evidence mentioned before also increases the reliability of the case study.

5.4 Case 1: Unisys

Introduction

Unisys is a global company with clients in more than one hundred countries. Unisys focusses on five vertical markets worldwide (financial services, public sector, communications, transportation and commercial) with expertise on consulting, systems integration, outsourcing, infrastructure and server technology [UNI07].

Strategic fit

Strategic importance

The objective of Unisys is to make money while meeting client expectations, whether this is achieved by OS or by proprietary solutions. Unisys does not prefer OSS over proprietary software, but chooses the best solution according to the clients' situation. The strategic importance for Unisys to prefer OSS over proprietary software is moderate and depends on the business model used by the OS company and the functionality of the OSS. The business model used by the OS company might award Unisys with a percentage of the subscription fee, which is a motivator to prefer OSS over proprietary software. Alfresco, among others, use this model. The strategic importance is higher when the OSS and proprietary are functionally the same, but the OSS generates more revenue (see figure 5.1).

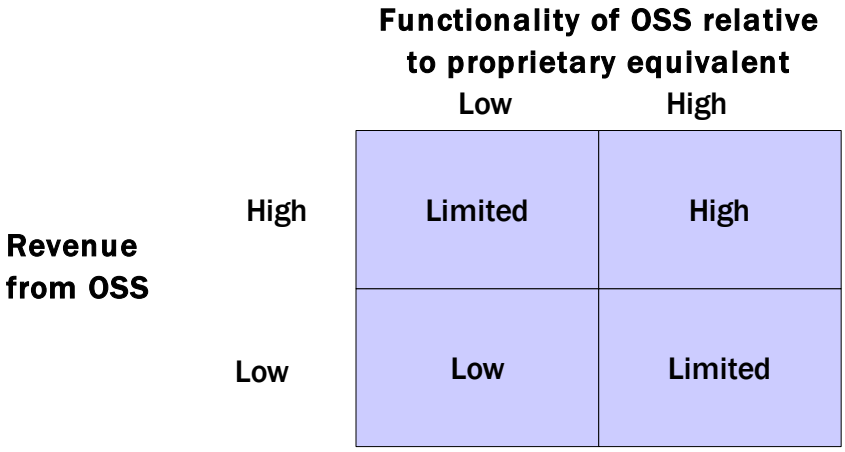

Figure 5.1: Strategic importance to prefer OSS over proprietary equivalent

The core business of OS companies using system integrators for the implementation of their OSS is mainly support of the implementations, according to the subscription model.

Compatibility of strategies and objectives

On the Unisys corporate strategy level, there are not many changes to be expected. The only plausible change might be that OSS will be preferred over proprietary software because of the higher (returning) financial results, which of course, would be advantageous for OS companies.

Unisys is not fully compatible with OS companies on the competitive level, when the OS company provides the same services from the software value chain as Unisys does. But when both parties aren't fishing in the same pond and thus are complementary, they are compatible.

Mutual goal dependency

The service Unisys provides to their customers is complementary to that of the OS companies most of the time. Some OS partners implement the OSS next to offering support, so in that case the partners are less complementary. Complementing can occur at different levels. At the software value chain level (see figure 4.1), Unisys focusses on consulting, system implementation/integration and application management [FLO02], while OS companies mainly focus on support. Both parties are also complementary when knowledge and resources are concerned. Unisys is a very large company with much resources, but sometimes lacks the knowledge of the OSS. It can be stated that both partners are very complementary, so Unisys and OS companies are mutual dependent.

There aren't many system integrators of the same scale of Unisys in the world, but OS companies do have alternatives to choose from. Some alternative system integrators might be small, but are totally committed to implement OSS and eventually implement more than the large system integrators..

Organisational fit

Flexibility

Strategic flexibility is about maintaining strategic fit. The strategic fit between Unisys and its OS partners is very good. It could worsen if the division of tasks or the main strategy of one of the partners is changed.

Management control

Unisys has management control in all partnerships. Unisys is protective of its clientèle. Unisys' customers are served by both Unisys and its partners, but the customer relationships are 'owned' by Unisys.

Complexity

The partnerships by Unisys are not thought to be complex. The division of tasks and responsibilities is clear, depending on the agreements made. This means that both task complexity and organisational complexity are low.

Trust

Even though trust is important in partnering agreements, not much trust can be accumulated. Emotional trust is hard to accumulate, because in the IT sector people change jobs regularly. Rational trust exist between the partners as long as primary goals are being met.

Most important

The interviews showed that Unisys rates communication as the most important factor contributing to partnership failure. In the fast-moving IT industry people change jobs often, so communication channels are renewed or discontinued. Agreements might get lost and personal relationships can not be replaced quickly.

5.5 Case 2: Bull

Introduction

Bull is an IT company, dedicated to helping corporations and public sector bodies develop open and secure information systems to sustain their business strategies. The company operates in more than a hundred countries worldwide and is particularly active in the defense, finance, healthcare, manufacturing, public and telecommunication sectors [BUL07].

Strategic fit

Strategic importance

The objective of Bull partnerships is to make money and to enter new markets. In some cases entering new markets is more important than making money in the first place. Although Bull hopes to provide other products and services to markets they approach this way, some of these partnerships weren't supposed to be unprofitable in the first place. Bull has not had clear objectives going into the negotiations, otherwise other agreements had been made.

Compatibility of strategies and objectives

Compatibility of the strategies and objectives on the competitive level of Bull and its OS partners can worsen if Bull decides to provide the same products and/or services as the OS companies. At the moment moderate compatibility exists, because OS companies sometimes do implementations of their own software too.

Mutual goal dependency

In partnerships with the goal of entering new markets, Bull is more dependent on the OS company than the other way around. This is specifically the case when the demand for the OS product in the market is high. In other partnerships where the products of both partners are complementary, the partners are mutual dependent.

Organisational fit

Flexibility

In some of its partnerships Bull is flexible because of the low coverage of the formal agreements. In these agreements only the most important issues have been discussed. The gain in flexibility comes with a downside though. Everything that is not clear by the contract, must be agreed upon separately, which might cause trouble later in the partnership.

Management control

Management control in Bull's partnerships should lie with Bull, but this is not always the case. In one partnership Bull acts as the main communication channel and implements most of the software, but management control is divided between both parties.

Complexity

Task complexity of the tasks concerned in Bull's OS partnerships isn't different from other partnerships and depends on the agreements made. The division of the tasks involved in the partnership and knowledge about the specific OSS are responsible for task complexity in OSS partnerships with system integrators.

Organisational complexity in the partnerships of Bull and its partners is higher in the partnerships where the formal agreements are clear on the division of tasks. In partnerships with high trust and flexibility the organisational problems can be resolved more easily than in inflexible partnerships with low trust.

Trust

Perhaps the most important advantage of trust is that the need for procedures and rules declines. [DOU97] At least one of the agreements between Bull and its partners had not been made clear enough, so the division of tasks and responsibilities was not clear. This lead to unnecessary discussions and negotiations, which had a negative effect on both emotional and rational trust.

Most important

The interviews showed that Bull rated communication and clear agreements as the most important factors contributing to partnership success. The agreements should be two-

way, which means that both parties must profit from the partnership. The intent of the partnership should also be translated into the agreement.

5.6 Conclusions from case study

In OS partnerships both partners products should be complementary or both partners will be collaborating and competing at the same time, which is not a good basis for cooperation. If the partners are complementary, mutual dependency increases. A mutual goal, however, doesn't really exist in OS partnerships. It should be clear to both partners that shared goals aren't obligatory in successful partnerships. Both partners can perfectly pursue their own goals in partnerships, as long as the partners are complementary.

From the case studies can be concluded that communication quality suffers from people changing jobs. According to Ellram, having multiple communication lines has a positive effect on collaboration success. This was confirmed by the case study interviewees.

OS companies should have part of the agreements ready before negotiations with potential partners. Define success factors per partnership before going into negotiations.

5.7 Chapter summary

This chapter explains why this research was divided up into two pieces and how the two pieces were carried out. The first part explains the reason for the case studies and its basis. The second part explained the detailed theoretical model, its options and its measures that was used during the survey part of the research. Then it was explained how the survey has been carried out and how we tried to achieve a high response rate.
In the next chapter a theoretical model that derived from literature and the case study will be explained.

Chapter 6. Theoretical model

6.1 Introduction

In this chapter the theoretical model will be discussed that will be tested with the survey. This chapter starts with defining several hypotheses deriving from the theoretical research and from the case study in section 6.2. In section 6.3 the actual theoretical model will be discussed and its alternatives will be explained in section 6.4. How the models are connected to the survey will be explained in section 6.5.

6.2 Hypotheses

According to the literature study and the case study these hypotheses have been tested in the survey:

H1. Poor communication is perceived as the most important variable contributing to partnership failure in OS business by OS companies.
H2. Two-way information sharing is perceived as the most important variable contributing to partnership success in OS business according to OS companies.
H3. The more an OS company trusts its partner, the more effective the collaboration will be.
H4. The longer a partnership exists, the more the OS company trusts its partner.
H5. The higher the employee turnover rate, the lower the communication quality.
H6. The lower the number of communication lines between partners, the lower the communication quality.
H7. The higher the communication quality, the more effective the OS collaboration.
H8. The higher the communication quality, the more OS companies trust their partners.
H9. Depending on the type of collaboration, the amount of trust will vary.
H10. Depending on the type of collaboration, the communication quality will vary.

Explanation H1

According to the research by Ellram [ELL95], the highest rated factor for failing partnerships in production companies is poor communication. The case study confirms that poor communication is the number one perceived problem in OS collaboration.

Explanation H2

The most important success factor for partnerships in production companies, according to Ellram, is two way information sharing. The case study confirms that communication is the most important success factor in OS collaboration too.

Explanation H3

There is less need for strict agreements between partners that trust each other [DOU97]. With unstrict agreements partners are able to react to changes quickly, which leads to more effective collaboration compared to partnerships with strict agreements.

Explanation H4

The longer a partnership exists, the more both partners trust each other.

Explanation H5

From the case study it became apparent that the IT industry is a fast-moving industry. As a consequence people change jobs regularly and the communication channels they were involved in, are dissolved or renewed. In both cases the communication quality suffers.

Explanation H6

The number of communication channels is likely to affect the communication quality. When one channel is dissolved or renewed, communication can still continue through the other channel. Communication channels don't last long in the IT industry, because people change jobs regularly.

Explanation H7

The most important factor contributing to collaboration effectiveness, according to both Ellram and the case study, is communication quality.

Explanation H8

The effect of communication on trust has been found by Butler and Cantrell [BUT94]. They found that communications related to task, career, and responsiveness were related to trust, whereas personal communications were not. Trust is sustained by contact and regular dialogue [BRA89], [POW96]. This dialogue ensures the mutual understanding of the wants of stakeholders and promotes commonality, mutual trust and organizational learning [PHI97], [SCH93].

Explanation H9

From the case study it became clear that trust in OS collaboration depends partly on the type of collaboration. In partnerships where there's an overlap in the service offerings by both partners, trust will be lower than in partnerships where this is not the case. In partnerships it is impossible to compete and cooperate with the same activity [BEN00]. This is the case in collaboration in R&D and in collaboration with system integrators where the OS company implements their own software too.

Explanation H10

The amount and difficulty of communication differs between the different types of OS collaboration. In collaboration in the software stack, communication is limited to compatibility and roadmap issues, which is stable and thus easily manageable. Communication with system integrators is different, because they need implementation information that might be different every implementation.

In the high-pace IT business people change jobs regularly and handing over communication channels doesn't get enough attention by one or both sides of the channel. It is also possible that agreements might get lost. The independent variable that quantifies people changing jobs is the employee turnover rate.

As a result of lower trust, some OS companies try to protect part of their intellectual property by not disclosing how (part of) their software works.

Trust is more important to OS companies than their partners, because OS companies might be afraid that others duplicate the project and OS companies want system integrators to install the supported edition of the OSS.
The goals of collaboration with system integrators, for example, differ for both partners. OS companies probably expect more from the collaboration than the system integrators, so they will rate the lack of shared goals higher than system integrators.

6.3 Main model

Figure 6.1 describes the theoretical model that will be tested in the survey. Circles represent constructs, squares represent independent and dependant variables and arrows represent causality. Measures in the model have been operationalised according to question numbers in the survey. Q16, for example, is a measure for collaboration status, which maps to question 16 in the survey. The survey questions can be found in appendix 3. H3 to H10 in the model refer to the hypotheses described in the previous section.

In this model, communication quality directly influences collaboration effectiveness. This means that whenever communication quality increases, the chance of effective collaboration increases.

The importance of trust in alliances has been stressed by many authors. Buckley, for example states that formal organisational arrangements are insufficient guarantee for the continuity of an alliance; without trust in the partner's commitment the chance of success is slight [BUC88]. According to Lorange, trust and commitment are necessary conditions for co-operation [LOR92] and Williamson states that relationships that feature trust will survive a greater stress and will display greater adaptability [WIL85].
The theory of Ellram is another strong indicator that poor communication and trust both have an important effect on the effectiveness of collaboration with OS companies. Both case studies seemed to confirm this. Furthermore, one of the conclusions of the case study indicated a causal relationship between the employee turnover rate of a company on the communication quality.

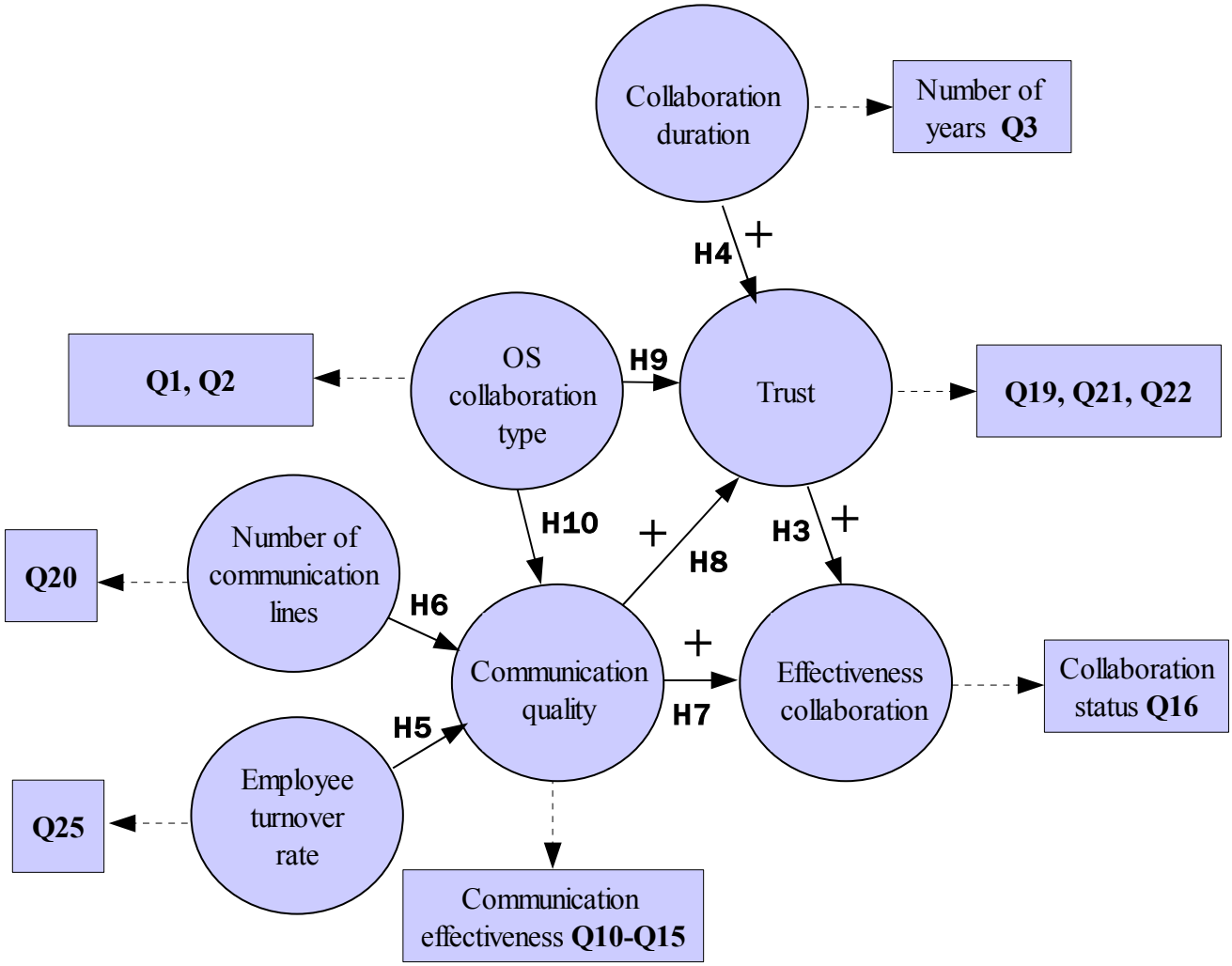

Figure 6.1: Theoretical model under research

6.4 Model alternatives

There is no theoretical evidence of the effects of the type of collaboration on trust and communication quality. By the means of the survey it will be investigated whether the OS collaboration type has a direct effect on trust and whether it is direct as stated in figure 6.1 or indirect via communication quality (see figure 6.2).

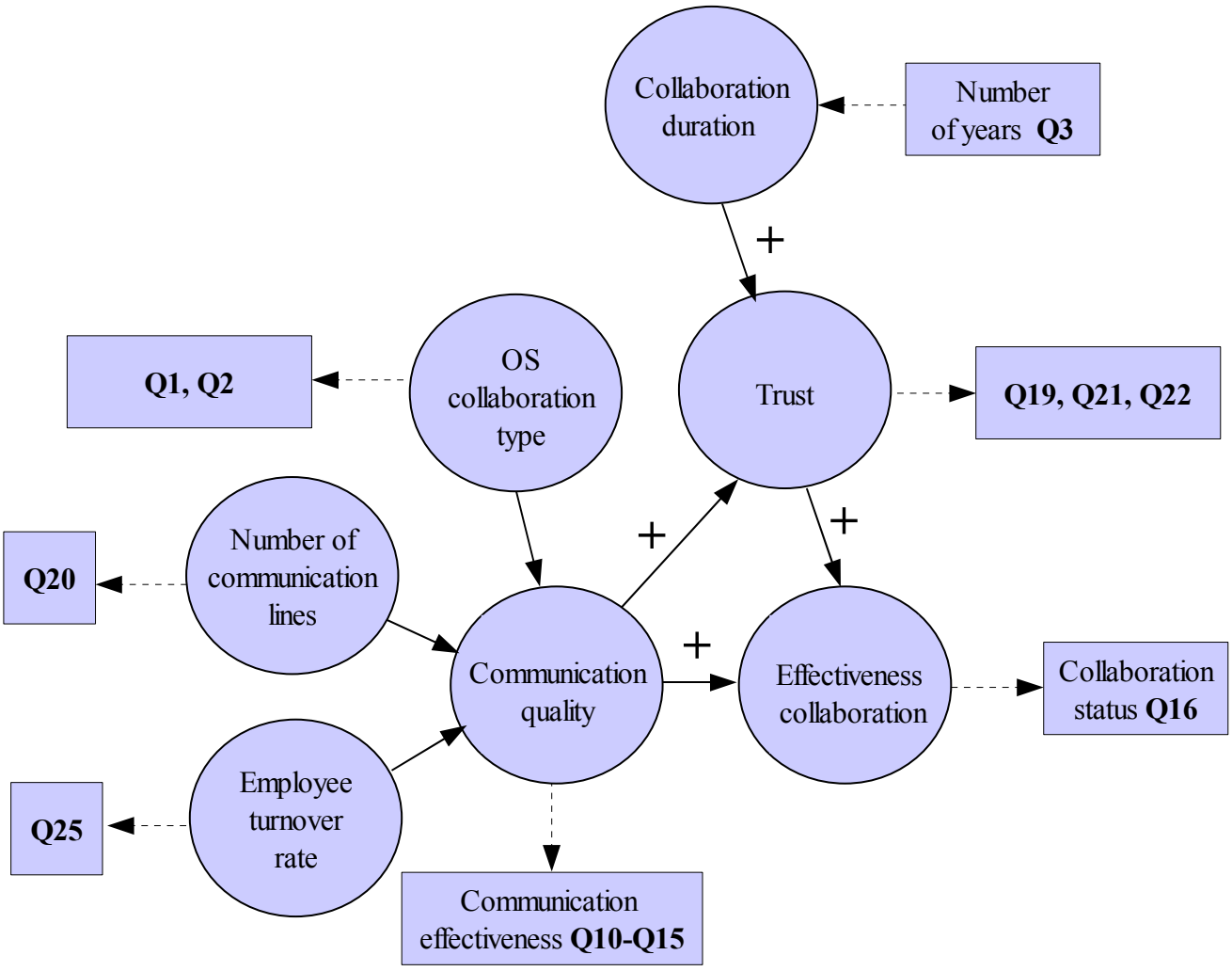

Figure 6.2: Theoretical model, indirect effect of OS collaboration type on trust

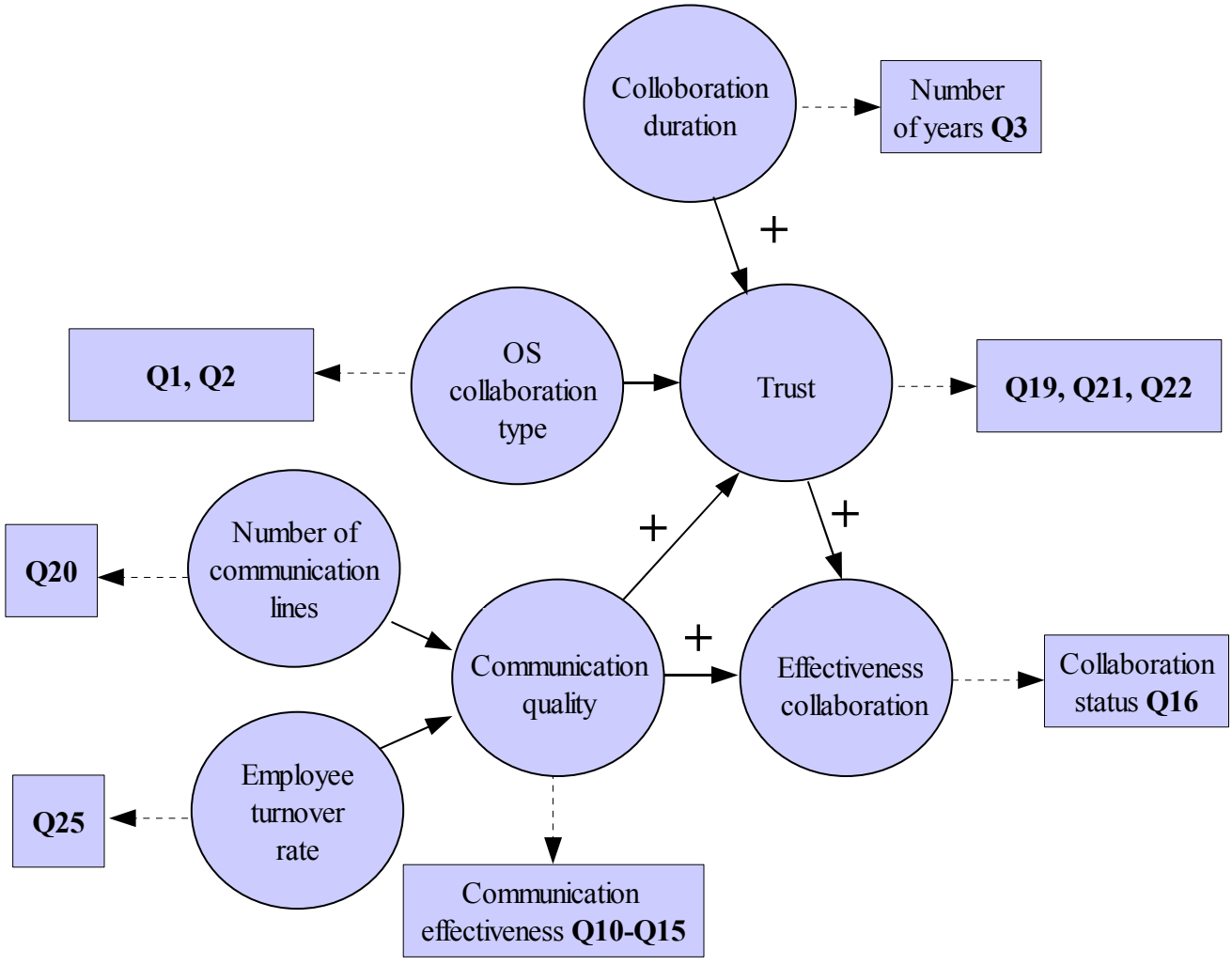

Figure 6.3: Theoretical model, direct effect of OS collaboration type on trust

6.5 Variables

All the variables mentioned in the theoretical model in figure 6.1, 6.2 and 6.3 are covered by questions in the the survey.

As a result of the case studies and theory, the assumption is made that communication quality is the most important factor contributing to collaboration success and failure. This assumption will be tested in the survey according to the research of L. Ellram, which accounts for Q6 to Q9 in the survey. Some factors that have nothing to do with software companies, such as JIT initiatives, have been excluded from Ellrams research. Furthermore, some specific factors have been added in order to be able to identify

specific problems in collaboration with OS companies. These added factors, which originate from the case study and the literature, are summed up in table 6.1.

Some questions in the survey are not related to the theoretical model. These are the questions that are based on the research of Ellram (Q6-Q9) and questions about the goal of the partnership (Q17, Q18), the licensing model used (Q23) and the size of the company (Q24). These questions haven been used for background information and for testing unexpected relations.

Communication quality will be measured by communication effectiveness. Communication effectiveness is measured by the "Communication Satisfaction Questionnaire (CSQ)" by Downs and Hazen [DOW77].

Added success factors	Added failure factors
Good documentation of the OSS	Absence of domain experts at partner
Good division of public and value adding information	Bad documentation of the OSS
Good licensing scheme	Fear of hijacking (take-over of the OS project)
Good quality of the complement of OSS	Low demand for complement of OSS
High demand for complement of OSS	Poor OSS complement quality
	Poor usability of the OSS
	Wrong licensing scheme

Table 6.1.: Added success and failure factors to factors of Ellram in survey.

6.6 Chapter summary

In this chapter the theoretical model for the survey was discussed. The hypotheses that followed from literature research and the case study were discussed first. From the hypotheses followed a graphical representation of the theoretical model with a detailed description. The last part of this chapter describes how the constructs were measured.

Chapter 7. Survey

7.1 Introduction

This chapter starts with the methodology of the survey in section 7.2. Then the analysis of the gathered data and the findings of the survey will be discussed in section 7.3 and section 7.4.

7.2 Methodology

Goal

The survey will be held in order to rate the most important factors that contribute to collaboration failure and success and to verify whether the hypotheses mentioned in section 6.2 can be confirmed or not.

Survey type

The response rate of surveys by telephone are usually higher than by mail or through the Internet. [FOW93] On the other hand, surveys by telephone take longer to conduct and there might be geographical complications. In this research time and geographical constraints have forced me to use an online survey.

The survey was administered online, through an existing free surveying tool. Thesistools.com can be used by students that want to administer surveys online for free. For a small fee an extended version is available that is totally free of advertisements, but the functionality is the same. With the tool it is possible to view results online or to export the respondent data in several formats. Simple online analysis is available, but it is restricted to percentages of respondents that gave a specific answer. The choice to develop the questionnaire specifically for the purpose of this research was rejected when Thesistools.com was found. All the functionality needed is available in this tool, which saved a lot of time.

Population / sample

The whole population of OS companies is small. There is an extensive amount of OSS available, but only a tiny part of it is backed by a company. During research of the population no official lists of OS companies was found. The sample of OS companies was gathered from the Internet by combining unofficial lists, presentations, websites and by reviewing partners of existing OS companies. The companies were filtered using the definition of OS business. The collected sample consisted of 107 companies.

Non response

Respondents that only filled in part of the survey have been skipped from the results. From the 47 that filled in the survey, only 27 were usable, which comes down to a response rate of about 26%, which is not very high. This has several reasons. First of all it is not sure whether all the sent emails reached their destination. It could be possible that the emails were delivered in spam boxes or were never read at all. One of the companies replied to the survey request with a statement that they didn't want to participate, because they didn't want to provide information about their partners what so ever. There

were also some companies that responded that they were really busy and that they couldn't insure that they would fill in the survey. One company simply refused.
Many of the companies have email addresses listed on their websites that bounced. These were the main reasons a lot of companies didn't respond to the survey invitation.

Contact strategy

In order to try to increase the response rate an iPod Shuffle was rewarded to one of the respondents who left their email address.

The sample has been contacted multiple times and with different approaches. In the first email the sample was informed about the research and invited to fill in the survey. The importance of the research for the OS companies was discussed, motivating the sample to fill in the survey. In a second contact round, the non-respondents in the sample were reminded of the research and again invited to fill in the survey. It was explained again that the research is important for themselves as it is for the research. If this didn't result in a response, a different approach was used. The companies that didn't respond were researched more thoroughly to obtain email addresses from people in the company directly. Through this approach people were also reminded to fill in the survey. If the response rate was too low after the second contact round, a third contact round was initialised by phone. Not all phone numbers from the sample were known, so not all the members from the sample were contacted during this contact round.

7.3 Analysis

Analysis method

The gathered data was analysed with SPSS. First the correlation matrix was constructed to see whether there were any unexpected correlations. In the next step all the hypotheses were tested one by one. The causal relations mentioned in the hypotheses were first tested by correlation. Secondly they were tested on partial correlation to see whether control variables influence the relation and if that was the case, how.

7.4 Findings

All the respondents were OS companies, but not their partners. From the 104 OS companies contacted, 47 have filled in the survey, but a lot didn't finish it. Many of the respondents quit after filling in only a few questions and some stopped before a large question. There were four large questions in the survey. Although it had been specifically mentioned how long the survey would take and that after the large questions there were only some small questions left, this probably scared off many respondents. After filtering out the respondents that filled in only a few questions, only 27 respondents were left. Not one partner has filled in the survey, which takes away the possibility for cross referencing some of the results. 59,4% of the partnerships has been legally documented. There were only two types of partnership mentioned: partnerships with system integrators and with complementers The findings of the survey will be discussed according to the hypotheses mentioned in section 6.2.

Hypothesis 1

Support was found that "*Poor communication"* is perceived as *one* of the most important factors in failing OS partnerships, but not the most important. It was the fourth highest rated factor (see table 7.2), is was the most mentioned (see table 7.1) among *"Lack of*

shared goals" and *"Absence of domain experts at partner"*. The highest ranked factor is *"Lack of trust",* which is the number 2 most mentioned factor in the top 5, so it's fair to say that *"Lack of trust"* is perceived as a more important factor in failing OS partnerships than *"Poor communication"*. Factor that are among the most mentioned top 5, but are ranked much lower are *"Lack of strategic direction for the relationship"* and *"Lack of shared goals".* Note that these two factors are much alike.

Factors believed to be most important to the failure of OS partnerships		
	% in top 5	Rank
Poor communication	52,6%	4
Lack of shared goals	52,6%	7
Absence of domain experts at partner	52,6%	3
Lack of trust	36,8%	1
Lack of strategic direction for the relationship	31,6%	10
Agreement not supportive of a partnering philosophy	31,6%	2

Table 7.1.: Factors believed to be most important to the failure of OS partnerships.

Average rating of factors contributing to OS partnerships failure			
	Rating[1]	SD	Rank
Lack of trust	5,87	1,23	1
Agreement not supportive of a partnering philosophy	5,76	0,90	2
Absence of domain experts at partner	5,71	1,02	3
Poor communication	5,70	1,08	4
Low demand for complement of OSS	5,17	1,27	5
Poor OSS complement quality	5,09	1,18	6
Lack of shared goals	5,04	1,12	7
Poor up-front planning	5,00	1,18	8
Changes in the market	5,00	0,96	8
Poor usability of the OSS	4,96	1,23	9
Lack of top management support to the partnership	4,91	1,18	10
Lack of strategic direction for the relationship	4,91	0,97	10
Lack of partner firm's top management support	4,91	1,18	10
Lack of benefit/risk sharing	4,79	1,29	11
Wrong licensing scheme	4,74	1,57	12
Lack of total quality commitment by supplier	4,65	1,37	13
Bad documentation of the OSS	4,63	1,15	14
Lack of distinctive supplier value-added benefit	4,63	1,73	14
Corporate culture differences	4,58	1,55	15
Top management differences	4,48	1,25	16
Too many suppliers for customer to deal with effectively	4,39	1,09	17
Ineffective mechanism for conflict resolution	4,13	1,42	18
Lack of central coordination of purchasing	3,63	1,58	19
Fear of hijacking (take-over of the OS project)	3,21	1,55	20
Distance barriers	2,87	1,7	21
[1] *mean ratings: 1=Very unimportant; 4=Neutral; 7=Very important n=27*			

Table 7.2.: Average rating of factors contributing to OS partnerships failure.

Hypothesis 2

"Two way communication" is the only factor rated higher than 6. It is the highest ranked factor contributing to partnership success with the lowest standard deviation (see table 7.3). This is strong support that *"Two way communication"* is the most important factor

contributing to OS partnership success, even though it is the second most mentioned factor in the top 5. The most mentioned factor in the top 5 is *"Shared goals",* which is ranked number 7. The other most mentioned factors in the top 5, *"Supplier adds distinctive value"* and *"Top management support"* are ranked number 8, which is also low. There is no solid explanation for these low ranked factors being mentioned the most in respondents top 5.

Average rating of factors contributing to OS partnerships success			
	Rating[1]	SD	Rank
Two-way information sharing	6,16	0,61	1
Good quality of the complement of OSS	5,93	0,9	2
Personal relationships	5,77	1,12	3
Sharing examples of success with others	5,73	0,76	4
Presence of domain experts at partner	5,73	0.94	4
High demand for complement of OSS	5,70	1,3	5
Good licensing scheme	5,56	1,2	6
Shared goals	5,54	1,01	7
Top management support	5,44	1,2	8
Supplier adds distinctive value	5,44	1,3	8
Training of buyers (sales personnel) in partnering philosophies/methods	5,36	1,29	9
Early communication to suppliers of specification changes, new products	5,33	1,12	10
Compatible corporate culture	5,31	1,59	11
Flexibility in agreement	5,22	1,34	12
Ongoing relationships between top levels of buying and supplying firms	5,19	1,18	13
Good documentation of the OSS	5,07	1,25	14
Good division of public and value adding information	4,93	1,33	15
Rewards/recognition for progress	4,77	1,12	16
Multiple relationships/points of contact between buying and supplying firms	4,67	1,25	17
Total quality management initiative	4,36	1,55	18
Site visits to supplier	4,24	1,8	19
Establishing a task force	3,52	1,71	20

[1] *mean ratings: 1=Very unimportant; 4=Neutral; 7=Very important n=27*

Table 7.3.: Average rating of factors contributing to OS partnerships success.

Factors believed to be most important to the success of OS partnerships		
	% in top 5	Rank
Shared goals	68,4%	7
Two way information sharing	52,6%	1
Presence of domain experts at partner	47,4%	4
Top management support	42,1%	8
Supplier adds distinctive value	36,8%	8

Table 7.4.: Factors believed to be most important to the success of OS partnerships.

Correlation matrix

The Pearson correlation coefficient indicates the strength and direction of a linear relationship between two variables. It is an indication, which means that causality between the two nominal variables can not be concluded, when correlation exists. In the correlation matrix in table 7.5 the Pearson correlation between all the variables from the theoretical model in figure 6.1 (except collaboration type) is shown. Significant correlation are shown in bold and with added *. The 1-tailed version was chosen, because the direction of the association is known in advance (see theoretical model in chapter 6). Table 7.5 displays Pearson correlation coefficients, significance values, and the number of cases with non-missing values (N).

		Duration	Collaboration effectiveness	Trust	# of communication lines	Employee turnover rate	Communication quality
Duration	Pearson Correlation	1,000	**-,354***	-,136	,124	,475**	,166
	Sig. (1-tailed)	,	,035	,250	,269	,006	,203
	N	27	27	27	27	27	27
Collaboration effectiveness	Pearson Correlation	**-,354***	1,000	**-,330***	,229	**-,365***	**-,537****
	Sig. (1-tailed)	,035	,	,046	,126	,031	,002
	N	27	27	27	27	27	27
Trust	Pearson Correlation	-,136	**-,330***	1,000	-,239	,164	**,489****
	Sig. (1-tailed)	,250	,046	,	,115	,207	,005
	N	27	27	27	27	27	27
# of communication lines	Pearson Correlation	,124	,229	-,239	1,000	**-,341***	-,302
	Sig. (1-tailed)	,269	,126	,115	,	,041	,063
	N	27	27	27	27	27	27
Employee turnover rate	Pearson Correlation	**,475****	**-,365***	,164	**-,341***	1,000	,194
	Sig. (1-tailed)	,006	,031	,207	,041	,	,167
	N	27	27	27	27	27	27
Communication quality	Pearson Correlation	,166	**-,537****	**,489****	-,302	,194	1,000
	Sig. (1-tailed)	,203	,002	,005	,063	,167	,
	N	27	27	27	27	27	27

* Correlation is significant at the 0.05 level (1-tailed).
** Correlation is significant at the 0.01 level (1-tailed).

Table 7.5.: Correlation matrix

Quite an important note should be made about the *"Collaboration effectiveness"* column. All the correlation numbers mentioned in this column should be the absolute number of the mentioned number. This because of the fact that in the survey the question related to this measure has been posed exactly the wrong way around ("*Very successful*" was 1, instead of 5). There are three sets of variables that show significant correlation at the 0.01 level. These are:

1. "*Trust*" and "*Communication quality*"
2. "*Communication quality*" and *"Collaboration effectiveness"*
3. *"Employee turnover rate"* and *"Collaboration duration"*

The first two were expected and will be explained further in their respective sections. The correlation in the third set of variables, between *"Employee turnover rate"* and *"Collaboration duration",* was totally unexpected and there is no direct logical explanation for it. It might be that the longer the partnerships exist, the longer the OS company itself exists, the more employees get bored and leave. This is merely a guess and by no means

supported by the data. Obviously the survey has not been filled in by employees that have left the OS companies.

There are four sets of variables that shows significant correlation at the 0.05 level. These are:

1. Collaboration duration - Collaboration effectiveness
2. Employee turnover rate - Collaboration effectiveness
3. Employee turnover rate - Number of communication lines
4. Collaboration effectiveness - Trust

Hypothesis 3

Correlation between *"Trust"* and *"Collaboration effectiveness"* is |-0,330|, which is significant (p<0,05). Partial correlation between these two variables when *"Communication quality"* is maintained constant, is |-0,0917| (see table 7.6)

Partial correlation coefficients. Controlling for *"Communication quality"*		
	Trust	Collaboration effectiveness
Trust	1,0000 (0) P= ,	-,0917 (24) P= ,656
Collaboration effectiveness	-,0917 (24) P= ,656	1,0000 (0) P= ,

(Coefficient / (D.F.) / 1-tailed Significance)

Table 7.6.: Partial correlation (hypothesis 3)

Hypothesis 4

Correlation between *"Collaboration duration"* and *"Trust"* is -0,136, which is not significant (p>0,25). Partial correlation between these two variables when *"Communication quality"* is maintained constant, is -0,2524, which is not significant (p>0,2) (see table 7.7).

Partial correlation coefficients. *Controlling for "Communication quality"*		
	Trust	Collaboration duration
Trust	1,0000 (0) P= ,	-,2524 (24) P= ,214
Collaboration duration	-,2524 (24) P= ,214	1,0000 (0) P= ,

(Coefficient / (D.F.) / 1-tailed Significance)

Table 7.7.: Partial correlation (hypothesis 4)

Hypothesis 5

Correlation between the *"Employee turnover rate"* and *"Communication quality"* is 0,194, which is not significant (p>0,1). Partial correlation between these two variables when *"Number of communication lines"* is maintained constant, is 0,1010, which is not significant (p>0,6) (see table 7.8).

Partial correlation coefficients. *Controlling for "Number of communication lines"*		
	Employee turnover rate	Communication quality
Employee turnover rate	1,0000 (0) P= ,	,1010 (24) P=,623
Communication quality	,1010 (24) P=,623	1,0000 (0) P= ,

(Coefficient / (D.F.) / 1-tailed Significance)

Table 7.8.: Partial correlation (hypothesis 5)

Hypothesis 6

Correlation between the *"Number of communication lines"* and *"Communication quality"* is -0,302, which is not significant (p>0,05). Partial correlation between these two variables when *"Employee turnover rate"* is maintained constant, is -0,2565, which is not significant (p>0,2) (see table 7.9).

Partial correlation coefficients. *Controlling for "Employee turnover rate"*		
	Number of communication lines	Communication quality
Number of communication lines	1,0000 (0) P= ,	-,2565 (24) P=,206
Communication quality	-,2565 (24) P=,206	1,0000 (0) P= ,

(Coefficient / (D.F.) / 1-tailed Significance)

Table 7.9.: Partial correlation (hypothesis 6)

Hypothesis 7

Correlation between *"Communication quality"* and *"Collaboration effectiveness"* is significant (p<0,01). Partial correlation between these two variables when *"Trust"* is maintained constant, is |-0,4561| (see table 7.10). This is smaller than the original correlation of |-0,537|. By partial intervening explanation this supports the main theoretical model in figure 6.1.

Partial correlation coefficients. Controlling for *"Trust"*		
	Communication quality	Collaboration effectiveness
Communication quality	1,0000 (0) P= ,	-,4561 (24) P= ,019
Collaboration effectiveness	-,4561 (24) P= ,019	1,0000 (0) P= ,

(Coefficient / (D.F.) / 1-tailed Significance)

Table 7.10.: Partial correlation (hypothesis 7)

Hypothesis 8

Correlation between *"Communication quality"* and *"Trust"* is 0,489, which is significant (p<0,01). Partial correlation between these two variables when *"Collaboration duration"* is maintained constant, is 0,5240, which is significant (p<0,01) (see table 7.11).

Partial correlation coefficients. *Controlling for "Collaboration duration"*		
	Trust	Communication quality
Trust	1,0000 (0) P= ,	,5240 (24) P=,006
Communication quality	,5240 (24) P=,006	1,0000 (0) P= ,

(Coefficient / (D.F.) / 1-tailed Significance)

Table 7.11.: Partial correlation (hypothesis 8)

Hypothesis 9 & 10

These two hypotheses can not be tested with current data. Not all the partnership types were present in the survey and there was an option called "others" in the survey. This was an inconsistency between the survey and the theoretical model, which could have been prevented if it weren't for the time constraint surrounding this research.

Correlation outside the hypotheses

There were correlations outside of the investigated hypotheses. As stated before, the correlation between *"Employee turnover rate"* and *"Collaboration duration",* was totally unexpected and can not be explained rationally. Partial collaboration between the two variables, controlling all the others, is 0,5260, which is significant (p<0,05).

The correlation between the *"Employee turnover rate"* and *"Collaboration effectiveness"* is |-0,365|, which is significant (p<0,05). Partial collaboration between the two variables, controlling all the others, is 0,1158 (p>0,25), which is not significant. This means there is no support for a direct causal relationship between the two variables.

The correlation between *"Employee turnover rate"* and the *"Number of communication lines"* is -0,341, which is significant (p<0,05). Partial collaboration between the two variables, controlling all the others, is -0,4265 (p<0,05), which is significant. This means there is support for a direct causal relationship between the two variables. It was actually already mentioned in the case study that communication channels aren't always renewed, when people change jobs. This could be an explanation for the effect.

The correlation between *"Collaboration effectiveness"* and *"Collaboration duration"* , which is significant (p<0,05). Partial collaboration between the two variables, controlling all the others, is |-0,2530|, which is not significant. This means there is no support for a direct causal relationship between the two variables.

7.5 Conclusions from correlation numbers

The original correlation between the variables mentioned in hypotheses 3 to 8 will be compared to their partial counterparts in order to make inferences about their causality. The original correlations and the partial correlations are summarized together with their differences in table .

Original and Partial correlations per hypothesis					
Hypothesis	*Variable i*	*Variable j*	*Original correlation* r_{ij}	*Partial correlation* $r_{ij\|k}$	*Difference*
H3	Trust	Collaboration effectiveness	|-0,330|	|-0,0917|	$r_{ij\|k} \approx 0$
H4	Collaboration duration	Trust	-0,136	-0,2524	$r_{ij} < r_{ij\|k}$
H5	Employee turnover rate	Communication quality	0,194	0,1010	$r_{ij} > r_{ij\|k}$
H6	Number of communication lines	Communication quality	-0,302	-0,2565	$r_{ij} < r_{ij\|k}$
H7	Communication quality	Collaboration effectiveness	|-0,537|	|-0,4561|	$r_{ij} > r_{ij\|k}$
H8	Communication quality	Trust	0,489	0,5240	$r_{ij} < r_{ij\|k}$

Table 7.12.: Original and partial correlation per hypothesis

Partial correlation between "*Trust*" on *"Collaboration effectiveness"* approaches 0 and is no longer significant, which means that there is no direct effect between the two variables, but the control variable might influence both variables. In the case of H3 the control variable is *"Communication quality"*. It can be concluded that there is no support for H3.

The partial correlation between *"Collaboration duration"* and *"Trust"* is higher than the original correlation, but still not high enough to be significant. This means that there is not enough support for H4.

The original correlation between *"Employee turnover rate"* and *"Communication quality"* is higher than the partial correlation, but they are both far not high enough to be significant. Therefore there is no support for H5

The original correlation between *"Number of communication lines"* and *"Communication quality"* is higher than the partial correlation, but they are both far not high enough to be significant. Therefore there is no support for H6

Both the original correlation and the partial correlation between *"Communication quality"* and *"Collaboration effectiveness"* are significant. The control variable *"Trust"* has almost no effect on the correlation between both variables. Therefore there is support for the causal relationship between *"Communication quality"* and *"Collaboration effectiveness"* (H7).

Both the original correlation and the partial correlation between *"Communication quality"* and *"Trust"* are significant. The control variable *"Collaboration duration"* has no effect on the correlation between both variables. Therefore there is support for the causal relationship between *"Communication quality"* and *"Trust"* (H8).

7.6 Chapter summary

In this chapter the second part of this research, which consisted out of a survey, has been described. First the survey's methodology has been discussed, after which the data analysis took place. Data analysis was structurally done in the order of the list of hypotheses. The first step in the data analysis was the construction of a correlation matrix. Then all the individual causal relationships were analyzed with Partial correlation.

Chapter 8. Conclusions and recommendations

8.1 Introduction

In this chapter the conclusions from this research are summarized. As a result of these conclusions, some recommendations are formed and discussed in section 8.4. A revised model can be found in section 8.3. In Section 8.5 possible future research is discussed.

8.2 Summarized conclusions

Support has been found for the causal relationship between communication quality and collaboration effectiveness and between communication quality and trust. All the other proposed causal relationships were insufficiently supported by the data and therefore left out the revised model in figure 8.1.

Next to the analyzed hypothesis, there was also support found for the causal relationship between the employee turnover rate and the number of communication lines. Partial collaboration between these two variables, controlling all the others, is -0,4265 ($p<0,05$), which is significant. This can be explained by the fact that communication channels aren't renewed when people change jobs.

8.3 Revised model

On the basis of the conclusions in section 7.5, a revised and model can be constructed from the hypotheses that are supported. This model is shown in figure 8.1. The OS collaboration type is still in this model because there was no opportunity to test the possible causal relationship between the two variables in this research because of an inconsistency issue between the survey questions and the theoretical model. These relationships are open for future research and are therefore coloured lighter in this model.

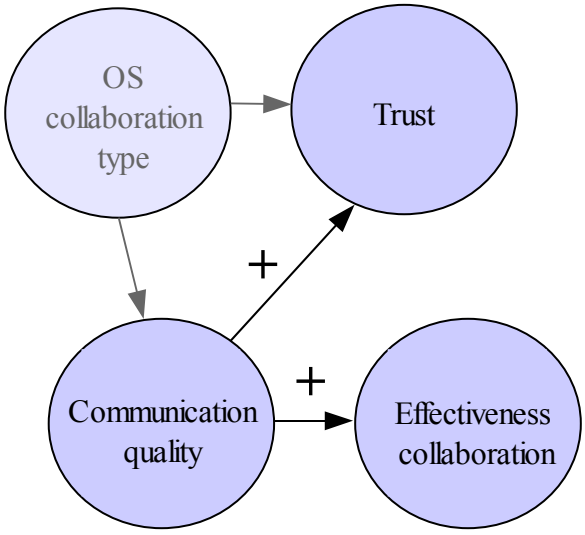

Figure 8.1: Revised model

8.4 Recommendations

In this research problems and opportunities in OS business partnerships have been analyzed from different angles. Different problems and opportunities arose, but one problem comes up every time: communication quality. It should be very clear to OS companies that communication quality works two ways. When it is too low, partnerships will suffer or cease to exist at all, and when it is high it will help the partnership to success. Therefore it is absolutely vital for OS companies to focus their attention to communication quality by implementing communication plans and by training employees. Higher communication quality will lead to higher collaboration effectiveness and higher trust. Higher trust on its turn might affect the flexibility in future agreements.

8.5 Future research

In this research communication quality is stated to be the most important factor in OS partnerships. The detailed facts about communication should be researched further to be able to take action. What factors influence communication quality in OS business and how should these factors be handled?

Because of the inconsistency issue between the survey and the theoretical model, the effect of the OS collaboration type on trust and the effect of the OS collaboration type on the communication quality are still to be investigated. Furthermore it would be interesting to know whether the used licensing model has an effect on trust.

As mentioned earlier, the effectiveness of OS collaboration can be improved in other ways than discussed in this thesis. It would be interesting to see which collaborative tools, such as Launchpad and Mantis, are mostly used in OS business and their effect on the effectiveness of the collaboration.

Although research has shown that some OSS products are more robust than their proprietary counterpart, little quantitative research has been conducted on the effect of OSS development on the quality of software according to ISO9126.

The model that results from the quantitative part of this research seems useful for proprietary software companies too. Future research in this respect might entail the differences between proprietary and OS companies.

During preliminary analysis of the data it became apparent that the agreements between OS companies and their partners are not always supportive of a partnering philosophy. It is still unknown why this is. What kind of partnering agreements are made and why don't these fit the expectations of one of the partners? The higher the amount of freedom in these agreements, the more flexible partnerships are [DOU97]. Flexibility in its turn might have a positive effect on collaboration effectiveness.

8.6 Chapter summary

In this chapter the results of the case study and the survey are summarized. Recommendations are given and a revised model is shown. In the last part of this chapter, possible future research was discussed.

References

[AFU00] Afuah, A., (2000), How much do you co-opetitors' capabilities matter in the face of technological change? (Strategic management journal 21 (3), 397-404)

[ALF07] Alfresco website, (2007), (27/09), http://www.alfresco.com/,

[AMI01] Amit, R., Zott, C., (2001), Value creation in e-business. (Strategic management journal 22 (6-7), 493-520

[ATS07] Atos Origin company website, 2007 (31/07), http://www.atosorigin.com

[BEH99] Behlendorf, B., (1999), Open source as a business strategy, Sebastopol, CA: O'Reilly, pages 149-170

[BEN00] Bengtsson, M., Kock, S.,(2000), Coopetition in business networks – to cooperate and compete simultanously, Industrial marketing management,

[BLE92] Bleeke, J., Ernst, D., (1992), The way to win in cross-border alliances, McKinsey quarterly 1, 113-133

[BOW00] Bowman, C., Ambrosini, V., (2000), Value creation versus value capture: towards a coherent definition of value in strategy., British Journal of Management vol 11, 1-15

[BRA89] Bradach, J. L. and R. G. Eccles, (1989), Price, Authority, and Trust: From Ideal Types to Plural Forms, Annual Review of Sociology 15, 97–118

[BUC88] Buckley, P.J., Casson, M., (1988), A theory of co-operation in international business, Cooperative strategies in international business, Lexington books, Lexington.

[BUL07] Bull company website, 2007 (31/07), http://www.bull.com

[BUT94] Butler, J. K., R. S. Cantrell, (1994), Communication Factors and Trust: An Exploratory Study, Psychological Reports 74, 33–34.

[CHE02] Chesbrough, H., Rosenbloom, R., (2002), The role of the business model in capturing value from innovation: evidence from Xerox corporation's technology spin-off companies. Industrial and Corporate change, 11 (3), 529-555

[CHU04] Chua, B.B. & Dyson, L.E., (2004). Applying the ISO9126 model to the evaluation of an elearning system. In R. Atkinson, C. McBeath, D. Jonas-Dwyer & R. Phillips (Eds), *Beyond the comfort zone: Proceedings of the 21st ASCILITE Conference* (pp. 184-190). Perth, 5-8 December.

[CUR88] Curtis, B., Krasner, H., Iscoe, N.,(1995), A Field Study of the Software Design Process for Large Systems, Communications of the ACM 31, 11, 1988, 1268-1287

[DAV85] Davis jr., F.D., (1985), A technology acceptance model for empirically testing new end-user information systems: theory and results., Sloan school of management.

[DOU97] Douma, M.U., (1997), Strategic alliances, fit or failure, Drukkerij Elinkwijk BV,

Utrecht, The Netherlands

[DOW77] Downs, C.W., Haze, M., (1977), A factor analytic study of communication satisfaction, Journal of Business Communication, 14, 63-73

[ELL95] Ellram, L.M., (1995), Partnering Pitfalls and Success Factors, International Journal of Purchasing and Materials Management, Spring 1995

[FEL02] Feller, J., & Fitzgerald, B. (2002) Understanding Open Source Software Development, London: Addison-Wesley

[FIT04] Fitzgerald, B., (2004), A critical look at Open Source, IEEE...

[FLO02] Berlescon research, (2002), FLOSS Final Report – Part 3, Free/Libre Open Source Software: Survey and Study

[FOW93] Fowler, F.J., (1993) Survey research methods, 2nd edition, Sage publications, Inc., Newbury Park, CA

[FUT07] Futureware website, (2007), (14/09), http://www.futureware.biz

[FSF07] Free Software Foundation website, (2007), (03/08),http://www.fsf.org/,

[GNU07] GNU's not Unix website, (2007), http://www.gnu.org/licenses/

[GRI99] Grinter, R.E., Hersleb, J.D., Perry, D.E.,(1999), Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work, Phoenix, Arizona, USA, 306-315

[HAM00] Hamel, G., (2000), Leading the revolution, Harvard business school press, Boston

[HEC99] Hecker, F., (1999), Setting up shop: The business of open-source software. IEEE Software 16, pages 45-51

[HER99] Hersleb, J.D., Grinter, R.E., (1999), Splitting the organization and Integrating the Code: Conway's Law Revisited, 21st International Conference on Software Engineering, Los Angeles, CA, 85-95

[IAN04] Iansiti, M., Levien, R., (2004), The keystone advantage: what the new dynamics of business ecosystems mean for strategy, innovation and sustainability., Harvard business school press, Boston

[INK98] Inkpen, A.C., (1998), 'Learning and knowledge acquisition through international strategic alliances', Academy of Management Executive, Vol. 12, No. 4, pp.69–80

[KIL88] Killing, J.P., (1988), Understanding alliances: the role of task and organizational complexity, Cooperative strategies in international business, Lexington books, Lexington

[KRA95] Kraut, R.E., Streeter, L.A., (1995), Coordination in Software Development, Communications of the ACM 38, 3, 69-81

[KRI03] Krishnamurthy, S., (2003), An analysis of Open Source Business Models, Making sense of the bazaar: perspectives on Open Source and Free Software, MIT Press, Forthcoming

[LAK03] Lakhani, K., Von Hippel, E., (2003), "How Open Source Software Works: Free User-to-User Assistance", Forthcoming in *Research Policy*.

[LEP07] Lepak, D., et al., (2007), Value capture and value capture: a multilevel perspective, Academy of Management Review, Vol. 32 No.1, pp. 180-94

[LER02] Lerner, J., Tirole, J., (2002), The Scope of Open Source licensing, Journal of Law, Economics, and Organization, Volume 21, Number 1, April 2005 , pp. 20-56(37), Oxford University Press

[LON02] DeLone, W., McLean, E., "Information Systems Success Revisited," hicss, p. 238, 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 8, 2002

[LOR92] Lorange, P., Roos, J., Bronn, P.S., (1992), Building successful strategic alliances. Long range planning.

[LUI93] Luimes, W., Spitholt, M.G.M., (1993), Verzelfstandiging van concernonderdelen, een onderzoek naar de meetbaarheid van cultural fit, human fit en synergie, Universiteit Twente

[MAR02] Margretta, J., (2002), Why business models matter, Harvard Business review 80(5), 86-92

[MAR07] Martin, K., Hoffman, B., (2007), An Open Source approach to developing software in a small organisation, IEEE Software 2007 Jan-Feb, 24(1), 54-57

[NIC03] Nichols, David M. and Michael B. Twidale (2003), "The Usability of Open Source Software", *First Monday*, 8(1), Available at: <http://firstmonday.org/issues/issue8_1/nichols/index.html>.

[OSC07] The open source car project web site, (2007) (03/08), http://www.theoscarproject.org/

[PAL02] Pal, N., Madanmohan, T., (2002), Competing on Open Source: Strategies and Practise, http://opensource.mit.edu/papers/madanmohan.pdf

[PEN07] Pentaho website, 2007, (04/09), http://www.pentaho.org,

[PEP06] Peppard, J., Rylander, A., (2006), From value chain to value network: insights for mobile operators, European Management Journal 24, 2-3, April-June 2006, pp. 128-141

[PER94] Perry, D.E., Staudenmayer, N.A., Votta, L.G., (1994), People, Organizations, and Process Improvement, IEEE Software 11, 4, 36-45

[PHI97] Phillips, R. A., (1997), A Stakeholder Theory and A Principle of Fairness, Business Ethics Journal 7(1), 51–66.

[POW96] Powell, W. W., (1996), Trust-Based Forms of Governance', in T. R. Tyler, and R. M. Kramer (eds.), Trust in Organizations: Frontiers of Theory and Research, Sage Publications, Thousand Oaks, CA, pp. 51–67.

[RAY99] Raymond, E.S., (1999), The cathedral and the bazaar: Mustings on Linux and open source by an accidental revolution, Sebastopol, CA: O'Reilly.

[TRL05] Annual report, Trolltech, 2005, http://www.trolltech.com/investors/financereports

[RED05] Annual report, Red Hat, 2005, http://investors.redhat.com/

[SAM03] Samoladas, I., Stamelos, I., (2003), Assessing Free/Open Source Software Quality, http://opensource.mit.edu/papers/samoladasstamelos.pdf

[SHA07] Shaikh, S.A., Cerone, A., (2007), Towards a quality model for Open Source Software (OSS), Elsevier Science B.V.

[SHA04] Shafer, S.M. et al, (2005), The power of business models, Business horizons, 48, issue 3, pp 199-207

[SCH93] Schein, E. H., (1993), On Dialogue, Culture, and Organizational Learning, Organizational Dynamics 22(2), 40–51

[SPR00] Sproull, L., Moon, J., (2000), Essence of distributed work: The case of the Linux kernel. First monday, (20-04-2007) firstmonday.org/issues/issue5_11/moon/index.html

[UNI07] Unisys company website, 2007 (31/07), http://www.unisys.com

[VER00] Verschuren, P., Doorewaard, H.,(2005), Het ontwerpen van een onderzoek, Utrecht, The Netherlands: Lemma BV, 3rd edition, 7th circulation, ISBN: 90-5189-886-X

[WEI94] Weimer, J., van Riemsdijk, M.J., (1994), A new magical lamp to rub?;the multiple constituency approach, a potential useful framework for research on the organisational effectiveness construct, paper presented at WESWA congress, University of Twente

[WIL85] Williamson, O.E., (1985), The economic institutions of capitalism, The free press, New York

Appendices

Appendix 1: Average rating of factors contributing to partnerships that have not worked out or were dissolved [ELL95].

Average rating of factors contributing to partnerships that have not worked out or were dissolved				
	Buyers' response		Suppliers' response	
	Rating[1]	Rank	Rating[1]	Rank
Poor communication	5,76	1	5,89	1
Lack of top management support to the partnership	5,31	2	4,44*	10
Lack of trust	5,19	3	5,59	4
Lack of total quality commitment by supplier	5,09	4	3,18***	18
Poor up-front planning	4,89	5	5,25	5
Lack of distinctive supplier value-added benefit	4,81	6	4,17*	13
Lack of strategic direction for the relationship	4,75	7	5,60**	3
Lack of shared goals	4,72	8	5,61***	2
Ineffective mechanism for conflict resolution	4,59	9	5,07	7
Lack of benefit/risk sharing	4,52	10	5,21*	6
Agreement not supportive of a partnering philosophy	4,23	11	4,84	9
Lack of partner firm's top management support	3,89	12	4,94**	8
Changes in the market	3,70	13	3,64	16
Too many suppliers for customer to deal with effectively	3,57	14	3,95	15
Corporate culture differences	3,56	15	3,48	17
Top management differences	3,52	16	3,98	14
Lack of central coordination of purchasing	3,30	17	4,19**	12
Low status of customer's purchasing function	2,68	18	4,32***	11
Distance barriers	2,35	19	2,70	19

[1] mean ratings: 1=Very unimportant; 4=Neutral; 7=Very important
*Note: Asterisks represent a statistically significant difference between buyer's and supplier's perceptions at the following levels: * p < .05; ** p < .01; *** p < .001*

Appendix 2: Average rating of factors important in establishing and maintaining partnerships, in general [ELL95].

Average rating of factors important in establishing and maintaining partnerships, in general				
	Buyers' response		Suppliers' response	
	Rating[1]	Rank	Rating[1]	Rank
Two-way information sharing	6,33	1	6,17	2
Top management support	6,17	2	6,19	1
Shared goals	5,96	3	5,83	6
Early communication to suppliers of specification changes, new products	5,94	4	5,65	11
Supplier adds distinctive value	5,88	5	5,88	5
Flexibility in agreement	5,82	6	5,69	10
Total quality management initiative	5,79	7	5,88	5
Training of buyers (sales personnel) in partnering philosophies/methods	5,54	8	5,13	15
Site visits to supplier	5,45	9	5,47	12
Multiple relationships/points of contact between buying and supplying firms	5,42	10	5,79*	8
Sharing examples of success with others	5,32	11	4,87	16
Ongoing relationships between top levels of buying and supplying firms	5,31	12	5,82*	7
Rewards/recognition for progress	5,10	13	5,24	13
Personal relationships	5,08	14	5,77*	9
Compatible corporate culture	5,00	15	5,13	15
Establishing a task force	4,63	16	4,69	17
JIT initiatives	4,59	17	5,88**	5

[1] *mean ratings: 1=Very unimportant; 4=Neutral; 7=Very important*
*Note: Asterisks represent a statistically significant difference between buyer's and supplier's perceptions at the following levels: * p < .05; ** p < .0001*

Appendix 3: Interview protocol

These are the interview questions for the case studies. The differences between open source partnerships and other partnerships will be discussed after most questions.

Management

1. What roles in open source partnerships do you play and how does this fit your main company strategy?
2. How does communication with open source companies take place and what role does technology play herein? What are the main problems in communicating with open source companies and how can these be solved or prevent them from happening?
3. Do you trust your open source partner companies? Do you think these companies trust your company too? What makes this apparent and how are you influencing this?
4. How well is top management support for the partnerships and how does this show?
5. How are risks and benefits shared among the partners and how does the open source license influence this distribution?
6. How do you control the quality of the work of partners?
7. How are conflicts resolved in partnerships?
8. What are the primary goals for entering in partnerships with open source companies? To what degree do shared goals influence the effectiveness of collaboration? What are the most common shared goals?
9. What are the criteria for selecting open source partners and why those? And what are the criteria for the open source software?
10. How do you cope with distance barriers between your organization and partners?
11. When are partnerships considered successful? When unsuccessful? What measurements do you use to measure the effectiveness of collaboration?
12. How are large amounts of partnerships dealt with? What are the main problems in dealing with large amounts of open source partnerships?
13. How are the strategic directions for relationships determined? What problems might be expected during the determination?
14. How do changes in the market affect partnerships?
15. Because of the voluntary nature of open source development, planning can be a major issue. How do you cope with this?
16. How do you cope with differences in corporate culture and in top management?
17. What kind of agreements are made with partners?
18. How do you assure yourself of the distinctive value-added benefits of open source partners?
19. There are many tools that support collaboration, such as usenet, Internet Relay Chat (IRC), Concurrent Version System (CVS) and wiki's. What kind of technologies are mainly used to support collaboration in open source software development and in what way? Are these technologies centralized or distributed and how are they coordinated? What are the main problems in using technology to support collaboration?
20. What are the top five critical success factors for collaboration with open source companies according to your organization? What are the top five main pitfalls?

Appendix 4: Survey questionnaire

1. What type of company are you? (OS, System integrator, OEM/complementer)
2. What type of company is your partner?
3. How many years has the partnership lasted (up till now)?
4. How many years do you expect the partnership will last from now?
5. Has the partnership been legally documented? (yes, no)
6. Please rate the following factors contributing to partnership success. (Very unimportant to very important, 7 point Likert scale)
    1. Compatible corporate culture
    2. Early communication to suppliers of specification changes, new products
    3. Establishing a task force
    4. Flexibility in agreement
    5. Good documentation of the OSS
    6. Good division of public and value adding information
    7. Good licensing scheme
    8. Good quality of the complement of OSS
    9. High demand for complement of OSS
    10. Multiple relationships/points of contact between buying and supplying firms
    11. Ongoing relationships between top levels of buying and supplying firms
    12. Personal relationships
    13. Presence of domain experts at partner
    14. Rewards/recognition for progress
    15. Shared goals
    16. Sharing examples of success with others
    17. Site visits to supplier
    18. Supplier adds distinctive value
    19. Top management support
    20. Total quality management initiative
    21. Training of buyers (sales personnel) in partnering philosophies/methods
    22. Two-way information sharing

7. Please rate the following factors contributing to partnership failure. (Very unimportant to very important, 7 point Likert scale)
    1. Absence of domain experts at partner
    2. Agreement not supportive of a partnering philosophy
    3. Bad documentation of the OSS
    4. Changes in the market
    5. Corporate culture differences
    6. Distance barriers
    7. Fear of hijacking (take-over of the OS project)
    8. Ineffective mechanism for conflict resolution
    9. Lack of benefit/risk sharing
    10. Lack of central coordination of purchasing
    11. Lack of distinctive supplier value-added benefit
    12. Lack of partner firm's top management support
    13. Lack of shared goals
    14. Lack of strategic direction for the relationship
    15. Lack of top management support to the partnership

16. Lack of total quality commitment by supplier
17. Lack of trust
18. Low demand for complement of OSS
19. Poor communication
20. Poor OSS complement quality
21. Poor up-front planning
22. Poor usability of the OSS
23. Too many suppliers for customer to deal with effectively
24. Top management differences
25. Wrong licensing scheme

8. Select the 5 most important items contributing to partnership success. Divide the numbers 1 to 5 to select the top 5. Please read all the factors before rating. (Use the factors from question 6)
9. Select the 5 most important items contributing to partnership failure. Divide the numbers 1 to 5 to select the top 5. Please read all the factors before rating. (Use the factors from question 7)
10. How satisfied are you with your job? (Very dissatisfied to very satisfied, 7 point Likert scale)
11. In the past six months, what has happened to your level of satisfaction? (Gone up, stayed the same, gone down)
12. If the communication associated with your job could be changed in any way to make you more satisfied, please indicate how:
13. Listed below are several kinds of information often associated with a person's job. Please indicate how satisfied you are with the amount and/or quality of each kind of information.
    1. Information about my progress in my job
    2. Personal news
    3. Information about organizational policies and goals
    4. Information about how my job compares with others
    5. Information about how I am being judged
    6. Recognition of your efforts
    7. Information about departmental policies and goals
    8. Information about the requirements of my job
    9. Information about government action affecting my organization
    10. Information about changes in my organization
    11. Reports on how problems in my job are being handled
    12. Information about benefits and pay
    13. Information about our organization's financial standing
    14. Information about accomplishments and/or failures of the organization

14. Please indicate how satisfied you are with the following: (Very dissatisfied to very satisfied, 7 point Likert scale)
    1. Extent to which my superiors know and understand the problems faced by subordinates
    2. Extent to which the organization's communication motivates and stimulates an enthusiasm for meeting its goals
    3. Extent to which my supervisor listens and pays attention to me
    4. Extent to which the people in my organization have great ability as communicators

5. Extent to which my supervisor offers guidance for solving job related problems
6. Extent to which the organization's communication makes me identify with it or feel a vital part of it
7. Extent to which the communications are interesting and helpful
8. Extent to which my supervisor trusts me
9. Extent to which I receive in time the information I need to do my job
10. Extent to which conflicts are handled appropriately through proper communication channels
11. Extent to which the grapevine is active in our organization
12. Extent to which my supervisor is open to ideas
13. Extent to which horizontal communication with other organizational members is accurate and free flowing
14. Extent to which communication practices are adaptable to emergencies
15. Extent to which my work group is compatible
16. Extent to which our meetings are well organized
17. Extent to which the amount of supervision given me is about right
18. Extent to which written directives and reports are clear and concise
19. Extent to which the attitudes towards communication in the organization are basically healthy
20. Extent to which informal communication is active and accurate
21. Extent to which the amount of communication in the organization is about right

Answer the following question (15) only if you are a manager or supervisor:

15. Please indicate how satisfied you are with the following: (Very dissatisfied to very satisfied, 7 point Likert scale)
    1. Extent to which my subordinates are responsive to downward directive communication
    2. Extent to which my subordinates anticipate my needs for information
    3. Extent to which I do not have a communication overload
    4. Extent to which my subordinates are receptive to evaluation, suggestions and critisisms
    5. Extent to which my subordinates feel responsible for initiating accurate upward communication

16. What is the currents status of the partnership? (Very successful, successful, unsuccessful, very unsuccessful, ended)
17. What was the initial goal of the partnership?
18. Has this goal changed over time? (yes, no)
19. Combined questionnaire
    1. I trust my partner (Disagree to Agree, 7 point Likert scale)
    2. My partner trusts me (Disagree to Agree, 7 point Likert scale)

20. How many communication lines (for example sales<->sales, support<->support) exist between your company and your partners'?

Answer the following questions (21,22 and 23) only if you are working for an OS company.

21. Are you afraid for source code hijacking? (Yes, by my partner, Yes, by others, No)
22. Do you try to protect your company's intellectual property by not revealing how (part of) your software works? (Yes, No)

23. Under what license is your software available? (GPL, BSD style license, Dual license (an open source one and a proprietary one), Other license)
24. How many full time employees in your organization? (<50, 50-100, 100-500, >500)
25. What's your company's employee turnover rate? (Total number of employees leaving in the past 12 months divided by average number of employees times 100)

Appendix 5: ISO 9126 Software Quality Model

Characteristic	Sub-characteristic	Explanation
Functionality	Suitability	Can software perform the tasks required?
	Accurateness	Is the result as expected?
	Interoperability	Can the system interact with another system?
	Security	Does the software prevent unauthorised access?
Reliability	Maturity	Have most of the faults in the software been eliminated over time?
	Fault tolerance	Is the software capable of handling errors?
	Recoverability	Can the software resume working and restore lost data after failure?
Usability	Understandability	Does the user comprehend how to use the system easily?
	Learnability	Can the user learn to use the system easily?
	Operability	Can the user use the system without much effort?
	Attractiveness	Does the interface look good?
Efficiency	Time Behaviour	How quickly does the system respond?
	Resource Utilisation	Does the system utilise resources efficiently?
Maintainability	Analysability	Can faults be easily diagnosed?
	Changeability	Can the software be easily modified?
	Stability	Can the software continue functioning if changes are made?
	Testability	Can the software be tested easily?
Portability	Adaptability	Can the software be moved to other environments?
	Installability	Can the software be installed easily?
	Conformance	Does the software comply with portability standards?
	Replaceability	Can the software easily replace other software?
All characteristics	Compliance	Does the software comply with laws or regulations?

Table 8.1.: ISO 9126 Software Quality Model [CHU04]

Appendix 6: Alfresco partner benefits

Alfresco partner benefits	Bronze	Gold	Platinum
Development Benefits			
Alfresco Insider RSS Feed	Y	Y	Y
Enterprise Edition for internal education, testing, demos. Number of supported deployments.	1	3	Unlimited
Discussion and Alignment of Roadmap Plans		Y	Y
Discount on Alfresco Training		20%	25%
Discount on Alfresco Certification		20%	Free
Partner Advisory Board		Y	Y
Partner Forums	Y	Y	Y
Access to email Technical Support - Calls	10	Unlimited	Unlimited
Access to phone Technical Support		Y	Y
Go-To-Market Benefits			
Listing in Alfresco Website Catalog	Y	Y	Y
Preferred Partner Listing		Y	Y
Use of Alfresco Logo	Y	Y	Y
Access to Marketing Material	Y	Y	Y
Press Release	Supported	Joint	Joint
Sales Benefits			
Access to Alfresco Sales teams		Y	Y
Joint Alfresco/Partner Sales Campaigns		Y	Y
Access to Hosted Alfresco Network for Demonstrations and Evaluations		Y	Y
Joint Collateral		Y	Y
Joint Webinar		Y	Y
Events - Co-Exhibiting & Sponsorship Opportunities		Y	Y
Partner Sales and Marketing Training		$	Y
Resell royalty on 1st line support		15%	20%
Resell royalty on 1st & 2nd line support			30%

Table 8.2.: Alfresco partner benefits [ALF07]

Appendix 7: List of abbreviations

API	Application Programming Interface
CRE	Customer Relationship Executive
ECM	Enterprise Content Management
FSF	Free Software Foundation
GNU	GNU's Not Unix
GPL	General Public License
JIT	Just-In-Time
IT	Information Technology
LAMP	Linux, Apache, MySQL, Perl/PHP stack
OEM	Original Equipment Manufacturer
OS	Open source
OSI	Open Source Initiative
OSS	Open source software
PHP	PHP: Hypertext Preprocessor
R&D	Research and development
TAM	Technology Acceptance Model

Table 8.3.: Abbreviations used in thesis