



# Qualitative Methods in Software Engineering

*Master's Thesis*  
Sara Persson

*Supervisor*  
Thomas Thelin, LTH

Department of Communication Systems



LUND INSTITUTE  
OF TECHNOLOGY  
Lund University



## **Abstract**

When conducting research there are several methodologies to use and one of them is the qualitative methodology. The qualitative methodology gives guidelines and rules for how to gather qualitative data, how to make a qualitative analysis, and how to organize qualitative data. Qualitative data is data that provides in-depth information and understanding for the setting and the case it was collected from. Qualitative data is described in words, not numbers and is often not generalizable. Within qualitative methodology there are several practical methods of how to perform a research, study, like interviewing and observation and several methods of how to analyze the data. But before collecting the data the researcher has to decide on a strategy, how and from what cases to collect the data and what the purpose of the study is.

This master thesis focuses on how it is possible to use qualitative methods in the software engineering process and contains a theoretical part that guides through the basic theory of qualitative methodology.

After studying what qualitative methods have been used in software engineering, a pure qualitative study of pair programming was conducted. The main purpose of the study was to test how the qualitative methods were usable in software engineering. The study evaluates pair programming in order to find strengths and weaknesses. The results of the study are that there are three strengths of pair programming, the change of partner, programmers getting more insight in the code and the communication between the programmers. Four weaknesses were identified as well, the workspace, communication in roles, adaption of the programmers and tension problem between the programmers.

The conclusion of this master thesis is that some qualitative methods are usable in software engineering and some qualitative methods are not recommended to be used. Because of software engineering being such a technical discipline it is recommended that when qualitative methods are used, the user has some technical experience.



## Contents

<b>1</b>	<b>Background</b>	<b>7</b>
1.1	Problem Description . . . . .	7
1.2	Outlining of the Report . . . . .	8
<b>2</b>	<b>Methodologies in Theory</b>	<b>9</b>
2.1	Varieties of Qualitative Methods . . . . .	9
2.2	Different Approaches of Qualitative Inquiry . . . . .	10
2.2.1	Theoretical Traditions . . . . .	10
2.3	How to Decide Which Method to Use? . . . . .	15
2.4	5 Types of Research . . . . .	15
2.4.1	Basic Research (why research) . . . . .	15
2.4.2	Applied Research (how research) . . . . .	16
2.4.3	Summative and Formative Evaluation Research (evaluations of results) . . . . .	17
2.4.4	Action Research . . . . .	17
2.5	Purposeful Sampling . . . . .	18
2.6	Practical Methods . . . . .	21
2.7	Observation . . . . .	21
2.7.1	Strength . . . . .	22
2.7.2	Risks . . . . .	23
2.7.3	Participant and Non-participant . . . . .	23
2.8	Interviewing . . . . .	24
2.8.1	Approaches in Interviewing . . . . .	24
2.8.2	Types of Questions . . . . .	25
2.8.3	Wording Questions . . . . .	26
2.8.4	Question Formats . . . . .	26
2.8.5	Interview Methods (applications) . . . . .	28
2.8.6	Questionnaires . . . . .	29
2.9	Organizing and Analyzing the Data . . . . .	30
2.9.1	Case Studies . . . . .	31
2.9.2	Early Analysis . . . . .	32
2.9.3	Approaches of Organizing and Reporting . . . . .	34
2.9.4	Practical Methods . . . . .	34
2.9.5	Credibility . . . . .	36
2.9.6	Threats . . . . .	36
2.9.7	Protection . . . . .	38

<b>3</b>	<b>Methodologies Used in Software Engineering</b>	<b>39</b>
3.1	Definition of Software Engineering . . . . .	39
3.2	Qualitative Methods Used in Software Engineering . . . . .	39
3.2.1	Summary of Studies . . . . .	40
3.2.2	A Detailed Description of Three Reports . . . . .	41
3.2.3	Evaluating User Interfaces Using Techniques from Qualitative Data Analysis . . . . .	42
3.2.4	Qualitative Methods in Empirical Studies of Software Engineering . . . . .	43
3.2.5	'Bad Practice' or 'Bad Methods' Are Software Engineering and Ethnographic Discourses Incompatible? . . . . .	45
<b>4</b>	<b>A Qualitative Study of Pair Programming</b>	<b>47</b>
4.1	Extreme Programming and Pair Programming . . . . .	47
4.2	Why Pair Programming? . . . . .	48
4.3	Planning the Study . . . . .	49
4.3.1	Purpose . . . . .	49
4.3.2	Theoretical Tradition . . . . .	49
4.3.3	What Kind of Research . . . . .	50
4.3.4	Sampling . . . . .	51
4.3.5	Practical Method . . . . .	51
4.3.6	Organization and Analysis . . . . .	52
4.4	Performance . . . . .	53
4.4.1	Case Study Description . . . . .	53
4.4.2	ABB . . . . .	53
4.4.3	Introduction Interviews . . . . .	53
4.4.4	Observation . . . . .	54
4.4.5	Member Checking and Follow Up Interviews . . . . .	54
4.5	Results From the Qualitative Study . . . . .	54
4.5.1	Case Study . . . . .	54
4.5.2	Result Possible Strengths . . . . .	55
4.5.3	Result Possible Weaknesses . . . . .	56
4.5.4	Recommendations of Pair Programming Based on the Study . . . . .	57
<b>5</b>	<b>Solutions and Recommendations</b>	<b>59</b>
5.1	Summary . . . . .	59
5.2	Conclusions . . . . .	60
5.3	Future Work . . . . .	63
5.4	My Own Learning Process . . . . .	64
<b>6</b>	<b>References</b>	<b>65</b>

**7 Appendix 69**

7.1 Example of Elicitation Interview . . . . . 69

7.2 Example of Software Programs for Analysis . . . . . 70

7.3 Useful Article that Discusses Useful Points of View . . . . . 70

7.4 Questions for the Introduction Interviews . . . . . 70

7.5 Observation Protocol . . . . . 71

7.6 Questions for the Follow Up Interviews . . . . . 72





# 1 Background

When conducting research there are several methodologies to use. Two of them are qualitative research and quantitative research, and they are different from each other but at the same time tightly bound together. The quantitative methods focus on everything that is measurable, like time and errors, and can be used on a great number of cases, groups or people. The quantitative data can be translated into numbers and statistics and adds a understanding of the information. Qualitative methods, on the other hand, focus on the depth of a small or single number of cases, groups or people. Qualitative data is presented in words, not numbers and are rich in details and descriptions. Then there is the possibility to combine both qualitative and quantitative methods, like collecting the data with qualitative methods and then analyzing the data with quantitative methods. Of course one can also do both a qualitative study and a quantitative study.

## 1.1 Problem Description

Qualitative methods are today widely used in sociology, which is a very different discipline than software engineering. Sociology is science about the community and the social relations between human beings, both in groups and individually. Software engineering is a sub-discipline to computer science, a discipline that is technical and focuses on technical development within computers and computer programs. In other words, a software engineer is someone who uses his or her engineering skill to develop software. The importance in software engineering is the process of engineering, not the writing of code. The engineering process contains different parts like programming concepts, analysis and design methods. Qualitative methodology gives guidelines and rules for how to make an analysis, of how to gather data and how to organize data. Is it possible to take the qualitative methods from sociology and adapt them into software engineering process? Both disciplines use analysis, gathering of data and organization of data. Highly different data, but still there are similarities of what work is done.

The purpose of this master thesis is to examine how qualitative methods can be used in software engineering by answering the following questions. Have any of the qualitative methods been used in software engineering? Can any other qualitative methods be used and how can they be used? What are the limitations of the methods to use?

"Qualitative results often are considered "softer" or "fuzzier" than quantitative results, especially in technical communities like ours. They are more difficult to summarize or simplify. But then, so are the problems we study in software engineering", Seaman C.B. [20].

## **1.2 Outlining of the Report**

First the report describes the basics of qualitative methodology and all its methods, chapter 2.

Then the report presents a short definition of software engineering and what qualitative methods that have been used in this subject, chapter 3.

A qualitative study was made and its conduction, theory and purpose are presented together with the results in chapter 4.

Then follows the most important chapter containing solutions and recommendations for the use of qualitative methods in software engineering, chapter 5.

Finally there are references, chapter 6 and appendix, chapter 7.

## 2 Methodologies in Theory

Qualitative methods produce a wealth of detailed information about a small number of cases. This increases the depth of the understanding of the cases but reduces the ability of generalization. The quality of the data depends a great deal on the user's methodological skill, sensitivity and integrity [15]. To generate useful qualitative data requires discipline, knowledge, training, practice and hard work, but most of all it requires patience.

There are two paths a qualitative study can take, research and evaluation. The difference between research and evaluation is its primary purpose. Therefore one has to know the purpose of the studies before realization. The judging of the qualitative study can vary a lot depending on the study's purpose and what criteria the judging is based on. There is a difference if the study was made with the purpose to generate a new theory or if the study was made with the purpose to evaluate an existing problem. [15].

Evaluation is systematic gathering of information about, e.g, a program's activities and the outcome of the program. A judgement of the program includes among other things its effectiveness and decisions about future programming. To make the decision more complete, the personnel involved can also be evaluated. But when an evaluation is performed systematically and empirically (with careful data collection and thorough analysis) it becomes evaluation research. The purpose of the research is to generate or test a theory and contribute to "knowledge for the sake of knowledge". Qualitative research is powerful as source of grounded theory like theory derived from fieldwork, i.e, the theory that comes to light from the researcher's observations and interviews in the real world rather than in the laboratory [15].

### 2.1 Varieties of Qualitative Methods

Qualitative methods have a great variety of results which vary as much as the researchers and their skills in the methods do. A source of variation is due to how the users collect the data. There are three different orientations: the *past orientation*, that is collecting things that are the results of past living, like artifacts or literature. The *present orientation*, that is observing what is happening now. The *future orientation*, that is eliciting the data, making it happen, as in an interview or a project [4].

The data is categorized into three types, documents, observation and interviews. Documents are written material, e.g, excerpts from documents like publications, letters and diaries, captured in a way that preserves the context. Observation is fieldwork descriptions of activities, actions and conversations. The data is a detailed description including the context in which the observation was made.

Interviews are responses about people's experience, opinion and knowledge [15].

There are mainly two different ways of analyzing this data, cool and warm analysis. A cool analysis is technical, e.g, structural analysis, and a warm analysis is an analysis where empathy is integrated [4].

## 2.2 Different Approaches of Qualitative Inquiry

Qualitative inquiry has some core strategies, i.e, approaches, see Section 2.2.1, that are the heart of qualitative inquiry. These approaches have origin in traditional disciplines and are the theory behind different qualitative research methods, i.e, how, and for what aim a discipline uses qualitative research. One may look upon the approaches as possibilities within qualitative inquiry. Especially in basic research, see Section 2.4.1 and applied research, see Section 2.4.2, when these are based on contribution to theory. Ethnography is the oldest approach of qualitative inquiry and anthropologists' primary method, and auto ethnography is the newest one. Between those two, there are a dozen other approaches [15].

### 2.2.1 Theoretical Traditions

The following part of the text is a review of the approaches, also called theoretical traditions. With each approach follows a question that captures the essence of the approach [15].

#### **Ethnography**

Question: *What is the culture of this group of people?*

Ethnographers study social and society problems in cultures. The primary method to do so is fieldwork via participant observation, see Section 2.7. Their findings are presented from a cultural perspective.

#### **Auto Ethnography**

Question: *How does my own experience of this culture connect with and offer insights about this culture?*

Ethnographers' perspective is an onlooker's view and the auto ethnographers' perspective is an insider's view. The conclusion is that an ethnographer should not study its own culture, i.e, study a culture from inside (not quite the same with participant observation) but an auto ethnographer should. Auto ethnographers study their own culture, from inside the culture, using their experience of the culture, just to get a first perspective on it.

#### **Reality Oriented, Reality Testing**

Question: *What is really going on in the world? What can we establish with some*

*degree of certainty?*

The basic thought is that all explanations and points of view hold the equal merit in search for the truth. This approach uses the language and basic thoughts in science to carry out an inquiry. Reality researchers are concerned about the study's objectivity and credibility.

### **Construction and Constructivism**

Question: *How have people in this setting constructed reality?*

When our human world differs from the physical world, i.e., we all perceive the reality in different ways, it must be studied in different ways. In this approach the researchers try to understand how, objectively, e.g., the participants' perception differs from the staffs' perception. This is done by open-ended interviews, see Section 2.8 and observations, see Section 2.7. Constructivism is about constructing knowledge about reality and not constructing reality itself.

### **Phenomenology**

Question: *What is the meaning, structure and essence of the lived experience of this phenomenon for this person or group of people?*

The aim of this approach is to examine how people make sense of their experience and how they transform it into consciousness, i.e., how people experience a phenomenon, how they describe it, how they judge it and how they talk about it with other people. This is done via in-depth interviews with people who have experience of the phenomenon. The phenomenon could be an emotion, a job, an organization or even a culture.

### **Heuristic Inquiry**

Question: *What is my experience of this phenomenon and the essential experience of others who also experienced this phenomenon intensely?*

Heuristic approach is a form of phenomenon inquiry which focus on the researcher's personal experience and insight. There are two elements of focus in heuristic inquiry. The researcher must have personal experience of the phenomenon and the other leading characters, such as co-researchers, also must have personal experience of the phenomenon. This when the approach is based on their intense experience. The reports of heuristic inquiry contain the researchers' personal insights and reflections. This approach is, unlike construction and constructivism, not objective.

### **Ethnomethodology**

Question: *How do people make sense of their everyday activities like to behave in socially acceptable ways?*

Heuristic inquiry focuses on the intense personal experience and ethnomethodol-

ogy focuses on the ordinary, the everyday routine, i.e., "a group member's methods of making sense to the social world" [15]. This approach determines how, and not why, something is done. It also determines what an outsider must learn to become a functional member of a group (like ethnography, see Section 2.2.1). Via depth-interviews and participant observation, it is studied how people get things done, step by step.

### **Symbolic Interaction**

Question: *What common set of symbols and understanding has emerged to give meaning to people's interactions?*

The aim of this approach is to understand the symbolic (token) world of the people in the study, mostly via group discussions. When people interact they create an understanding, or as it is called shared meanings, and this understanding, or meaning, becomes their reality. In other words, if someone told you he has seen a dog (big, brown), and you also have seen a dog (small, white), you would compare the symbol dog to each other's meaning of what it is and thereby expand your size of the symbol dog. However there are three premises for this approach [15]. Human beings act towards things on the basis of the meanings that the things have for them. The meaning of the things comes from the social interaction one has with one's friends. The meaning of things are modified through interpretative process by the person dealing with the things he or she encounters. Symbolic interaction also contains a sub-approach, semiotics, that answers the question "How do signs like words and symbols carry and convey meaning in particular a context?". This sub-approach studies among other things the relationship between language and human behavior when it is a blend of linguistics and social behavior.

### **Hermeneutics**

Question: *What are the conditions under which a human act took place or a product was produced that make it possible to interpret its meaning?*

To interpret a text in a way that makes sense one has to understand and be able to identify the essence of the writer's meaning, underlying purposes, and to know the context in which it is written considering culture and history, i.e., the thought behind hermeneutics is that one has to know something about the thing being researched, as well as one has to know something about the researcher in order to interpret a qualitative study in a correct way. Hermeneutics can also be used to interpret legends, stories and religious texts.

### **Narrative Analysis**

Question: *What does this narrative or story reveal about the person and the world from which it came?*

Narrative analysis focuses on how to interpret stories and comes from the study-

ing of texts. It expands the idea with texts to also include in-depth interviews. Narrative analysis is like hermeneutic approach but with focus on interpretation and context used in, e.g, family stories, graffiti, life stories and reveals cultural and social patterns via individual experience. What differs in use from hermeneutic approach is that narrative analysis uses in-depth interview as a tool when the language used is different when a person tells a story than the language in a formal study and this allows other interpretations.

### **Ecological Psychology**

Question: *What is the relationship between human behavior and the environment?*

The basic thought that imbues ecological psychology is that the individual and the surroundings are independent of each other. A researcher makes some assumptions of what is important to understand concerning human experience, and then focuses on goal-directed behavior, as a description of the main characteristics like a composition of place, thing and time. One thing to notice about this approach is that qualitative description, based on observation as an onlooker, in the beginning of the study becomes quantitative data when it is analyzed. This because researchers that use this approach codes the data to numbers and then analyze it in a statistical way.

### **System Theory (a System Perspective)**

Question: *How and why does this system, as a whole, function as it does?*

The basic of this approach is that in an embedded system the parts cannot be seen as independent because they affect each other and the total system. So a system can be classified as something that contains different parts that interact together, like a program for alcoholics where group meetings, a coach and self discipline interact or like a program for child abuse where the abuser gets punished and the child is maybe moved to a foster home, receiving therapy and so on. If a change is made in one part, the other parts are affected as well as the system, and the system is often not understandable by viewing one part only but by viewing the whole and the interaction between the parts. A system perspective is important when understanding real-world systems, where each part is viewed as an embedded entity. The system orientation can be helpful in coming up with questions and making sense of qualitative data, and some approaches to system research depend a great deal on qualitative inquiry.

### **Chaos and Complexity Theory (Non-linear Dynamics)**

Question: *What is the underlying order, if any, of disorderly phenomena?*

A disorderly phenomenon is phenomena like the weather, fluids in motion, human beings or even a program. When the concepts of system and complexity often are related this approach is how one brings order, via metaphors, to what is observed,

how it is observed and what is known as a result of the observation. It helps to better define a research problem. Chaos research is much about mathematics and describing non-linear dynamics when complexity is the theory and research about non-linear dynamics.

### **Grounded theory**

Question: *What theory emerges from systematic comparative analysis to explain what has been and is observed?*

Grounded theory focuses on the process of generating theory rather than the theory and its content. Grounded theory is the most famous, and perhaps the most useful approach and can be used on all forms of data, not only qualitative. The findings are grounded in the real world, the empirical world, by different methods.

Grounded theory has been explained, by the metaphor as "generating grounded theory is like lifting a veil that hides what is going on" [15]. This approach is meant to build theory rather than test theory. Its aim is to provide researchers with tools to analyze and handle data. There are three main steps to take when conducting grounded theory. First there is basic description which is followed by conceptual ordering, i.e., organizing data into categories that makes the foundation to the theory being developed. Lastly there is theorizing, that is wording intuitive ideas and concepts into a logical order. Grounded theory can be used in cases like health studies, abusive relationships studies and prison time studies by using symbolic interaction. But there is a difference between symbolic interaction and grounded theory. Grounded theory has its methodological steps and its comparison of data, like comparing research sites. To generate a theory from data means more than making hypotheses based on the raw data, it also means systematical work on the relation to the data. In other words, a theory is more than data, it is an objective explanation of the phenomena. The comparing technique of this approach helps to increase objectivity when everything is systematically conducted and it gives rules for every step of the way through a research. It also helps finding variations in the patterns founded in the data.

Seaman [20] offers a different way to define grounded theory. "Theory generation methods are generally used to extract, from a set of field notes, a statement or proposition that is supported in multiple ways by the data. The statement or proposition is first constructed from some passage in the notes, and then refined, modified, and elaborated upon as other related passages are found and incorporated. The end result is a statement or proposition that insightfully and richly describes a phenomenon. Often these propositions are used as hypotheses to be tested in a future study or in some later stage of the same study. These methods are often referred to as 'grounded theory' methods because the theories, or propositions, are 'grounded' in the data."



### **2.3 How to Decide Which Method to Use?**

Knowing the purpose is the main thing in research. Which method and type of analysis a researcher chooses is mainly depending on the purpose of the study, but things like who the audience are, how much time the study is allowed to take and what questions will be answered by the study also matter. Therefore the first thing to do in a study is to get a clear idea about these things. To make it easier to do that there are some questions to be answered [15]:

- What is the purpose of the study, e.g, what makes one do the study?
- Who are the primary audience?
- What questions will guide the study?
- What data will answer the study questions?
- What resources are available?
- What criteria will be used to judge the quality of the findings?

With the study and its purpose well defined the researcher can decide what type of design, data gathering and analysis are necessary in order to meet the purpose. To make it easier, the research is divided into five types, each one suitable for different purposes [15]. These five types will be further discussed in the following parts, see Section 2.4.

- Basic research, to contribute to fundamental knowledge and theory.
- Applied research, to illuminate a societal concern, i.e, work on human and societal problems.
- Summative evaluation research, to determine program efficiency.
- Formative evaluation research, to improve a program.
- Action research, to solve a specific problem.

## **2.4 5 Types of Research**

### **2.4.1 Basic Research (why research)**

The purpose of basic research is to understand and explain. "Knowledge for the sake of knowledge" [15]. A basic researcher typically works within specific disciplines, i.e, within a specific subject area. The questions the basic researcher studies can be derived from traditions and thesis within the respective disciplines.

- Psychology: Why do individuals behave as they do? How do human beings behave, think, feel and know?
- Biology: What is the nature of life? What are variations in forms of life?
- Geography: What is the nature of variations in the earth's surface and atmosphere?

Basic researchers strive to generate new theories or test existing theories that surround the knowledge of the discipline. Basic qualitative research requires a long and intensive period of fieldwork and the findings are often published in scholarly books, journals and dissertations. Particular attention must be given to the accuracy, validity and integrity of the result.

An example within software engineering is why the users of a computer program do not think it works as it should or, more important, as they expected it to do. The basic researcher then has to look into the developer's basic thoughts and theory to find out if the program is the correct one or if the purpose of the program has been misunderstood.

#### **2.4.2 Applied Research (how research)**

The purpose of applied research is to contribute knowledge that will help people understand the nature of a problem. This allows human beings to take action and to more effectively control their environment [15]. The question the applied researcher studies is derived from problems and concerns experienced by people. The applied researcher is often guided by the results of basic research. This because the applied researcher does studies that tests knowledge to real-world problems. In other words, the applied researcher is trying to understand how to deal with societal problems while the basic researcher is trying to understand and explain the basic nature of them or to understand how to take advantage of them. That is why applied research findings are limited to a specific time, place and condition. Work on environmental studies often involves researchers from different disciplines. Applied qualitative researchers bring their personal insights and experience into their recommendations and are often published in journals that are specialized in applied research of a discipline. The targeted group for applied research is policymakers and managers of invention-oriented organizations, professionals and other researchers.

A typical applied research case within software engineering is how the problems in an computer program are solved, like if the program should be redesigned or if it should just be modified.

### 2.4.3 Summative and Formative Evaluation Research (evaluations of results)

The purpose of evaluation research is to evaluate the results of the solution (that often is based on the applied research). Evaluation researchers study programs, policies, personnel and products. There are two distinguishing characteristics within evaluation research. Summative evaluation that judges overall effectiveness and formative evaluation that aims to improve programs [15].

The purpose of summative evaluation is to get an overall judgement of the effectiveness of a program, policy or product, i.e., an overall judgement of the effectiveness of the solution recommended by the applied researcher. All this is to decide whether the thing being evaluated is or is not effective and whether it should or should not be continued. Summative evaluation cannot be generalized to other situations and seldom relies entirely on qualitative data, but qualitative data adds depth, detail and nuance to quantitative findings [15].

A typical summative evaluation research within software engineering evaluates if the solution suggested, for the failed computer program, was effective enough and if it solved the problem. It might also be an evaluation between two suggested solutions.

The purpose of formative evaluation is to improve a specific program, policy, group or product [15] and the aim is to form, i.e., shape the program, policy, group or product being studied. Formative evaluation often relies on qualitative methods. A typical formative evaluation research within software engineering evaluates if there are some features that should be programmed in order to improve the program and raise the users' understanding of the program.

### 2.4.4 Action Research

Action research is also called action oriented, problem solving research. The purpose of action research is to solve specific problems within a program or community. Action research is basic, applied and evaluation research at the same time, in a small scale and the research is often done by the people in the program. The researchers are studying the people in the program and their problems in order to solve these [15]. The findings of action research are often published through briefings, staff discussions and oral communication. The formality of reporting and the nature of research are quite different from the routine in basic, applied and evaluation research.

A typical action research within software engineering is when the developers hand over the program or parts of the program they develop to other developers for feedback or when they conduct some basic evaluation of the program themselves.

## **2.5 Purposeful Sampling**

There are several methods to choose from regarding how and from what cases the information should be gathered, i.e., sampling methods. When quantitative sampling focuses on large, randomly selected samples, qualitative sampling focuses on depth of single, purposefully selected cases. Quantitative and qualitative sampling also differ in purpose of sample logic and strategy. Qualitative sampling has its strength in its ability to select information-rich cases to study in depth. Information-rich cases are cases that, when studied, provide insight and in-depth understanding of the important issues of research. There are different strategies to in a purposeful way choose cases and the logic for each one of them serves a specific purpose [15].

### **Extreme Case Sampling**

Extreme case sampling handles cases that are extreme or special in some way. It can be extremely successful cases or cases that are extreme failures. The theory is that one learns more from the extreme cases that differ from the rest, than from the ordinary ones. If one program is a great hit, one is a total failure and eight of them are ok, choose the one that is doing great and the one that is a failure as these cases have the most useful information. It is in other words a focus on the end points of a curve with a normal distribution. This strategy is often used in ethnomethodology studies [15].

### **Intensity Sampling**

Intensity sampling is the same logic as in extreme cases sampling but with less stress on the extreme, so while extreme case sampling seeks highly unusual cases, intensity sampling seeks excellent cases. This strategy is often used in heuristic research [15].

### **Maximum Variation (heterogeneity) Sampling**

Maximum variation sampling strategy focuses on variations and differences. Too many differences, heterogeneity, can be a problem in small samples when the cases differ from each other a lot. But in maximum variation sampling it is the strength and aim. All common patterns that appear during the sampling are of interest. These patterns capture the core experience of a setting, as Patton [15] puts it. How can maximum variation be fulfilled? A difference between the sites where every site is unique, e.g., different geographic areas is one way to get maximum variation. The main aim is to find similarities in the samples despite (or maybe thanks to) the differences. The similarities and patterns found are important because they arise from widely different sites. Maximum variation sampling brings two things which are important for qualitative research. It generates high-

quality and detailed descriptions in each case and reveals important patterns that are found in most of the cases. These patterns significance is derived from having emerged out of differences (heterogeneity) [15]. In lack of resources this strategy is recommended when randomly chosen cases make the strategy's result generalizable to some degree.

### **Homogeneous Samples**

Homogeneous samples are the opposite to heterogeneous samples. Its purpose is to describe sub-groups in depth. If, e.g., a program includes children in school the parents may also be studied, and in homogenous samples a sub-group of parents would be chosen, such as single female parents. This strategy gathers people in similar situations with similar background. Focus group interviews, see Section 2.8.5 are based on this strategy [15].

### **Typical Case Sampling**

Typical case sampling handles typical cases and has the aim to describe to an on-looker (or someone outside the group) what is typical for this setting. This strategy is an illustrative strategy. [15].

### **Critical Case Sampling**

Critical case sampling is used on cases that deserve comments like "if it doesn't happen here it won't happen anywhere" or "if they understand it everyone will understand it". It means that in this strategy, cases and sites that are critical are studied. But they have to be critical in the way the comments say, "if it doesn't happen here it won't happen anywhere". If a technical manual is created and the company is not sure if it is understandable, they will test it on a group of non technical people and "if they understand it everyone will understand it". One weakness is that this strategy can be used only in one case or site, but it is this strategy's strength as well. It can be used if there are limited resources, a logical generalization, and it is often used in evaluation research [15].

### **Snowball or Chain Sampling**

Snowball sampling is to first locate critical cases and then ask people for references. This strategy has the same aim as nomination in any other subjects have, to in the end be able to filter out a group of people. If a lot of people are asked to name someone else who knows much about the particular subject, the same name is likely to come up more than once. And if several different sources name the same person, i.e., the same person gets several nominations, this is the person who has valuable information [15].

**Criterion Sampling**

Criterion sampling strategy has the logic review and studies all cases that meet a criterion. The aim is to understand the cases that reveal weaknesses, e.g, day-care queues where everyone who has been in the queue for more than a year is studied. or all students in a class who did not pass a course etc [15].

**Theory-based Sampling**

Theory-based sampling samples things like incidents and time periods. This strategy is used in cases that are representative of a phenomenon, e.g, a subgroup chosen from another group by criterion sampling. One sub-strategy is operational construct sampling. This strategy is a more conceptual version of criterion sampling where real world examples are studied and can be used in cases where the samples cannot be randomly selected. An example is studies on early and late adopters where the cases are purposefully selected due to the fact that the population of adopters may not be known but still meet a criterion. Another sub-strategy is theoretical sampling which is often used in grounded theory as it supports comparative methods of analysis, and comparisons are what grounded theory is based on. The sub-strategy is defined as "sampling on the basis of emerging concepts, with the aim being to explore the dimensional range of varied conditions along which the properties of conceptus vary" [15].

**Confirming and Disconfirming Cases**

Confirming and disconfirming cases strategy is used to confirm or disconfirm hypotheses which exist in the beginning of a fieldwork. Do the hypotheses confirm the already emergent patterns? If they do, they add depth to the study. The hypotheses to confirm or disconfirm emerge from stakeholders and literature rather from the fieldwork [15].

**Stratified Purposeful Sampling**

Stratified purposeful sampling is the sampling divided into layers, like samples within samples. Stratified purposeful samples may be samples over the result of two other sample strategies and the purpose is to capture major variations [15].

**Opportunistic or Emergent Sampling**

Opportunistic or emergent sampling strategy takes advantages of whatever unfolds as it unfolds, or as Bracette F. Williams said about the fieldwork conducted "I'm following where the data take me, where my questions take me". This captures the essence of opportunistic or emergent sampling [15].

**Purposeful Random Sampling**

In purposeful random sampling cases are often, but not always, randomly selected

with the purpose of increasing the result's credibility. Purposeful random sampling is about credibility and not representatives [15].

### **Sampling Politically Important Cases**

Sampling politically important cases is a variation of critical case sampling with a strategy trying to increase the usefulness of information. But as the name tells you there are more complex aspects to it as well, that have to do with a political strategy [15].

### **Convenience Sampling**

Convenience sampling's aim is to do what is fast and convenient, though this is neither purposeful nor strategic. But still it is the most common strategy even if it is the least desirable [15].

## **2.6 Practical Methods**

So far the theoretical aspects have been considered. But how does one put the theory into practice? There are mainly two practical methods to conduct the data gathering, observation and interviewing and these are discussed in the following part of the text. When the data has been gathered it is time to analyze it. There are several ways to do that as well. They will also be discussed in the following parts.

## **2.7 Observation**

When people cannot describe what or how they are thinking or how they are performing a case, observation is recommended. The purpose with data from observation is to describe the setting that was observed, the activities that took place, the people who participated and the meaning of what was observed. The quality of an observational report is judged by the extent to which the report permits the reader to understand the situation described. The definition of fieldwork and purpose of qualitative data can be summarized as follows. Qualitative observation, fieldwork, refers to the circumstance of being in or around an on-going setting for the purpose of making a qualitative analysis of the setting [15].

If the study includes a specific case it might have to be artificially created [4]. An example of such a case is fish mating. It is hard to study the procedure in nature but if an aquarium is built, the study comes in easier as long as the artificial "river" (the aquarium) is built in a way to resemble the natural environment.

Scientific research with observational methods requires discipline, training and preparation. People often think they are trained observers, while in fact they are not. There are some skills which have to be improved, like [15]

- Learning to pay attention, see what there is to see and hear what there is to hear.
- Practice in writing descriptively.
- Acquiring discipline in recording field notes.
- Knowing how to separate detail from trivia.
- Using rigorous methods to validate observation.
- Reporting the strengths and limitations of one's own perspective, which requires both self-knowledge and self-disclosure.

### **2.7.1 Strength**

Is not talking to people and asking questions just as good as observing people? No, the risk is, without observation, that a, e.g, societal program or problem never is understood without experiencing it. The mark is missed and inappropriate questions are asked. Observation has several advantages [15]:

- The researcher is able to understand and capture the context.
- It allows the research to be open and discovery oriented because the researcher does not need to rely on written documents or verbal reports.
- The researcher has the opportunity to see things that might be missed among the people in the setting, i.e, an observer can discover things no one else has ever paid attention to.
- The researcher has a chance to learn things that people would not willingly discuss in an interview.
- The researcher has the opportunity to move beyond the selected perceptions of others when interviews present the understanding of the people being interviewed.
- The researcher is able to take in information and impressions that go beyond what can be recorded in even the most detailed field notes.



### 2.7.2 Risks

Observation does not only has its strengths, it has its problems and risks too especially since it is based on other people [4].

- People try to "look good" to give good impression, i.e, people often try to please you, to give you what they think you want.
- People may suddenly come to see the researcher as a researcher, which can feel like suddenly realizing you have no clothes on.
- People sometimes like to be looked at, and will act to retain that attention.
- People may have prejudice against the researcher, the psychologist, the sociologist, the academician, the college student.
- Feelings (love, hate) towards particular people can "blinker" us towards or away from their perceptions.
- Different cultures have different rules of exclusiveness regarding sex, age, and so on.

### 2.7.3 Participant and Non-participant

Observation is seldom just either participant or non-participant. There are many nuances in between. The researcher's participation varies from a non-participant, an onlooker, to take a full and complete part in the setting, so-called "going native". The researcher may also begin as an onlooker and gradually become a participant as the fieldwork proceeds and of course the other way around, where the researcher begins as a participant and gradually becomes an onlooker. In full participation the researcher participates in the setting and is at the same time taking notes and making interviews. There is no separation between the participation by the researcher and the other participants. If the researcher is an onlooker, the observation process can be separated from interviewing and documentation. The researcher can choose to take notes during or after the current activity. Most important is to keep the noting from being a disturbance for the participants [15].

There are some cases where it is not possible to become a participant in a societal program, it depends on the nature of the program. In a program for children, an adult cannot, of course, participate as a child, but may participate as a volunteer, parent or as staff. The same problem does a researcher come across when a study depends on a specific gender. In programs related to treatments, abuses and any form of addiction a researcher cannot make a full participation without receiving the treatment, being an addict or being abused. A researcher may also experience

problems to do a full participation in programs where the group may object to an outsider or where social and cultural factors can limit the participation. In some studies, a full participation will add an additional burden to the study when the resources are limited. Because of these obstacles, a full participation research is rare. Most common is a combination of being an onlooker and a participant [15].

How long shall a researcher participate? What is the length of time it takes to gather the necessary data? It depends on the study's purpose and aim. In anthropological tradition of fieldwork, i.e., living in the culture being observed, the length of time for the participant observer is six months at a minimum to a year. In evaluation and action research that period of time is much shorter, depending on time and resources available in relation to the information needed. Even two hours of feedback from an onlooker can be helpful. According to Patton [15], fieldwork should last long enough to get the job done, to answer the research questions being asked and fulfil the purpose of the study.

## **2.8 Interviewing**

Interviews are used when things cannot be observed, like feelings and thoughts. The purpose is to get another person's perspective on things. When qualitative interviewing is discussed it mostly means open-ended interviewing. Open-ended interviews are interviews where the interviewed person answers the questions in its own words, unlike multiple-choice questions, i.e., questions where the interviewed person chooses between predefined answers [15].

### **2.8.1 Approaches in Interviewing**

There are three different approaches to open-ended interviews and the difference between these approaches is how much of the interview questions that are determined before the interview [15].

#### **The Informal Conversational Interview**

The informal conversational interview approach, also called unstructured or ethnographic interviewing, is when the questions are spontaneous and not predetermined, "go with the flow", and the persons interviewed do not always realize they are being interviewed [15]. The interviewer can ask questions within the topic for clarification and detail but avoids forcing the interviewee in some direction. One technique for unstructured interviews is reflection where the interviewer uses follow-up probes and rephrasing the interviewee's statement [4]. This gives maximal flexibility and is often used in participant observation. But because of the flexibility it takes time to discover sought patterns in the material [15].

**The General Interview Guide**

The general interview guide approach is when the interviewers have a checklist of issues they want to discuss, but within these issues they conduct an informal interview. Often used in focus groups [15].

**The Standardized Open-ended Interview**

The standardized open-ended interview approach, also called structured interviewing, is where the interviewers have a set of questions, carefully worded before the interview, in order to minimize the variations in the questions when each person interviewed is asked the same questions, in the same order, in the same way. This approach is recommended when different people conduct the interviews. The material generated is easy to analyze [15].

**2.8.2 Types of Questions**

There are six types of questions that always can be asked [15].

**Behavior Questions**

Behavior questions are used to extract a behavior or actions with questions like "If I follow you for a day, what will I see you doing?"

**Value Questions**

These questions are used to find out what people think about some experience or issue with questions like "What do you think about ...?"

**Feeling Questions**

Feeling questions aim to extract people's emotions about their experience and their thoughts with questions like "What do you feel about that?"

**Knowledge Questions**

Knowledge questions are used to find out what people know with questions like "What are the rules for ...?"

**Sensory Questions**

Sensory questions are used to extract people's experience of sense, like what is seen, heard or smelled, with questions like "When you do that, what do you see?"

**Background Questions**

These questions are used to be able to place people in relation to other people with questions like "How old are you?"

### 2.8.3 Wording Questions

There are four criteria when wording questions. A good question is open-ended, neutral, singular and clear [15].

#### **Open-ended**

Open-ended means that people answer in their own words. A good definition of open-ended questions is "A truly open-ended question does not presuppose which dimension of thought that will be salient for the interviewee.", i.e, it permits the interviewee to take whatever direction he or she wants with questions like "How do you feel about ...?" or "What is your opinion of ...?". It is tricky because questions like "How satisfied are you with this program?" is not truly open-ended when the answer is expected to be a degree of how satisfied he or she is [15].

#### **Neutral**

An interviewer has to be neutral to the opinions of the interviewee and should not judge them [15]. There are some recommended question formats that help keeping the neutrality, see Section 2.8.4.

#### **Singular**

Singular questions are questions with one idea per question, i.e, asking for one thing at a time. A question like "How well do you know and like the staff in this program?" makes the interviewee confused if the interviewee does not know the staff that well but yet likes them. Should the answer be "not so much" or "a lot"? The question should be divided into two questions "How well do you know the staff in this program?" and "How well do you like the staff in this program?". A common problem is to ask for advantages and disadvantages in the same question [15].

#### **Clear**

To word a clear question the researcher has to learn the language of the setting, like special terms, the meaning of words and phrases. A technician, for example, has a different language with technical terms than a non-technical person has. There are settings where bad means good and so on [15].

### 2.8.4 Question Formats

There are several question formats when conducting interviews. "why"-questions should be avoided as they may lead to more questions which may lead to more questions and so on. A typical situation is when children ask why about something and then ask why about the answer, or if a child is asked a "why"-question

when he or she thinks the answer is obvious [15]. Yes and no questions should also be used with precaution as the answers are short on information. The following formats are specialized to keep the neutrality in the interview.

### **Illustrative Example Questions**

Asking questions via examples is a way to let the interviewee know that there are several similar situations, both worse and better, than the interviewee's. That makes the interviewee feel more secure, but the examples must be closely related to the question and the interviewee's situation. An example of the type of question is "in this situation some people do this and some do that, what do you do?" [15]. The examples must however be of both a worse and a better situation to avoid pushing the interviewee in some direction.

### **Role-playing Questions**

By providing a context for the questions and asking the interviewee to take a role in the context, it allows the interviewee to answer like someone else and by that getting a different answer, i.e., different information [15].

### **Presupposition Questions**

In questions like "What is the most important experience you have had?" the interviewer assumes that the interviewee has had an important experience, i.e., the interviewer presupposes that a situation has occurred [15]. If the interviewee has not had such a experience, the interviewee is forced to answer a question that he or she cannot relate to.

### **Prefatory Statements**

The interviewer starts the question by making a brief summary of the questions that already have been answered and by that giving the interviewee a moment to gather its thoughts and to focus on next question, i.e., making a map of "this is where we have been and this is where we are going" [15].

### **Follow Up Questions (Probes)**

A probe can be defined as "a follow-up question used to go deeper into the response". What separates regular questions from probes is that probes are seldom written out in the interview, it is more like an encouragement to the interviewee to go deeper in the subject. There are several different probes. Detail-oriented probes that ask when, why, where, what and how. Elaboration probes that are an encouragement by a nod or a "uh-huh" from the interviewer. Clarification probes that ask questions like "tell me more about ..." or "what do you mean by ..." and contrast probes that give the interviewee something to push off against like "How does x compare to y?", i.e., asks the interviewee to compare something to get more

information [15].

### **2.8.5 Interview Methods (applications)**

Interviewer can take several different approaches when conducting an interview, i.e, several practical methods.

#### **Elicitation Interview**

This method is based on an interview with the purpose of getting data about his or her image of reality, e.g, how he or she categorizes animals [4], see Section 7.1. The interviewer (researcher) just asks a few open questions like (x stand for the object) "What is that?" (x), "What kind of x is that?" (moving up), "What kinds of x are there?" (moving down) and "Do any of these belong together?" (clustering).

#### **Interviewing as Dialog**

Interviewing as a dialog is when the interviewee is looked upon as a coresearcher and the questions are in the form of "I do this, do you do it too?" or "I think this is frightening, do you think so too?". This makes the interviewee feel secure and he or she does not think of it as a interview, more like a discussion [4].

#### **Focus Groups**

Focus groups can be explained as "a small group of people on a specific topic" [15]. It is a form of group interview with six to ten persons, but it is an interview and should not be used as a problem solving method. In focus groups, people's decisions grow in discussion with other people, i.e, social behavior affects a person's decisions and values. In a study there may be several focus groups to get more information or to confirm important values. Focus groups are often used in action research [15].

Focus groups are excellent to use in the beginning of a study to narrow the topics, i.e, to find out what topic is the most important one or as consultants do, use focus groups to get to know the company and the employees' values [4].

As a method focus groups has its advantages and disadvantages. Its strength is that the cost for the interview, calculated in hours, is cheap since the data contains between six and ten opinions instead of one. The interaction between the participants raises the quality of the data and most of the participants find focus group to be a pleasant experience. Focus groups also help to identify and weed out extreme opinions and inconsistency. Its weakness is that there must be a limited number of questions due to the time factor, and each participant can only have a limited time per opinion. Therefore focus groups demand much of the interviewer, like being able to divide the time fairly between the participants so that no participant takes over the group. Focus groups work poorly on personal topics and confidentiality

is hard to assure. To get the maximum information out of a group, the participants should have the same background but they should not know each other. One thing that also can affect the outcome of the focus group is if it is held outside the natural setting [15].

### **Group Interviews**

Interviewing a group of people, three, four or ten, with an informal approach is called group interviews. They often take place at dinner tables, in a break or in any relaxed, informal context. Group interviews can be used before the individual interviews to scout the subject [15].

### **Protocol Analysis**

Protocol analysis is also called think-aloud protocol and aims to explain the cognitive process in a person's head when performing a task. This can be done like the person telling his or her train of thoughts when performing the task, both after and during the performance. But it takes skill to be able to perform something and at the same time a person talks out loud [15].

### **2.8.6 Questionnaires**

Instead of doing oral interviews, questionnaires can be used. When a personal interview cannot be done questionnaires can be used. It takes less of the researcher's time to conduct since questionnaires do not require a personal meeting with every interviewee. It is even more important with the wording of the questions when questionnaires are used though the respondent cannot get the question explained if it is not understandable. One risk then is that the respondent interprets the question in its own way in order to be able to understand it more easily [3] and one of the following actions may be taken. The respondent may:

- Distort the question.
- Soften or weaken question elements.
- Omit the difficulty element.
- Qualify some part of the question.
- Convert the question.
- Only consider certain aspects of the matter.
- Give a conditional reply.

There are mainly two pitfalls when wording questions. The first one is if the researcher is "getting carried away in the excitement of writing questions before research goals and research have been formulated and fully understood" [22] and the second one is that the researcher is "overusing open questions arguing that closed questions may fail to provide all possible answers or that respondents will be influenced by the predetermined choices" [21].

When using not open-ended questions in questionnaires the issue of how and in what order the answers should be presented emerges.

### **In What Order**

There are no exact rules on which order the answers should be in, but a theory is the "satisficing principle" [10], which states that people often seek solutions that are acceptable in order to minimize psychological costs. Two types of response-order effects (the effect of in what order the answers are presented) have been documented, primary effects and recency effects [10]. The primacy effect is achieved when the answer placed at the beginning of list (of answers) is more likely to be selected and recency effect is achieved when the answer placed at the end of list is more likely to be selected.

### **How**

When presenting the answers a scale can be used which is usually ranged from two to ten points depending on what nuances that is sought in the answer, even if the most common and recommended one has four or five points. An odd numbered scale gives the respondent the opportunity to be neutral whereas an even numbered scale does not give this possibility and therefore forces the respondent to make an opinion. When forced, the respondent is more likely to choose a positive opinion [16]. After choosing scale, one has to choose how to label it. There are two alternatives, labeling just the end point or every point. Labeling every point allows the researcher to express the results in terms of percent.

## **2.9 Organizing and Analyzing the Data**

Qualitative methods generate a whole lot of data and the first thing to do is to make an inventory of what data one has. Is the data complete? Of what quality is the data? When making sense of a massive amount of data there are no rules to be of help, although there are guidelines to follow. This means that the researcher's sense is the most important thing when analyzing and organizing the data. It is a researcher's duty to "fairly represent the data and communicate what the data reveals given the purpose of the study" [15]. This means that the human factor is both the strength and at the same time the weakness of qualitative analysis. But there are several computer programs that offer assistance in conducting the



analysis, see Section 7.2 [15].

When analyzing there are three main parts, so called "flows of activity", data reduction, data display and conclusion drawing and verification, that flows through the study. All simultaneously [11].

### **Data Reduction**

Data reduction is the process of selecting, simplifying and transforming data. The data reduction begins in an early stage of the study, already when the researcher chooses what questions that will guide the study or what cases to sample from. Data reduction is a kind of analysis that sharpens, focuses and organizes the data.

### **Data Display**

The thought of data display is founded on the every day and ordinary meaning of display. If data is displayed it is easier to understand. Data display in analysis is matrices, graphs and charts. Defining a display means to decide on, e.g, rows and columns or to decide on what data goes where. The display has the purpose to assemble information to a compact form, easy to survey.

### **Conclusion Drawing and Verification**

The final conclusion are not made until the end of the study but already in the beginning the researcher has to make decisions that affects the conclusions in the ending. Like what is the meaning of things. A skilled researcher keeps the meaning in mind through the study but does not let it interfere with his or hers ability to be open-minded and skeptic. The conclusions are verified both in the findings and in the researcher's mind.

#### **2.9.1 Case Studies**

Cases are units of analysis and a case analysis is organizing the data, generated from the case, for in-depth study and comparison. The word case study means both the noun and the verb.

Before analyzing data and finding patterns one has to describe the context of the setting in which the study was made, i.e, do case descriptions of each person or program being a part of the observations or interviews [15]. A case study takes the audience into the case situation and experience, e.g, a person's or a group's life, and each case study in a report stands alone. Not until later, after presenting the case studies, a comparison of each case is done. If interviews were done, a cross-case analysis, instead of the case descriptions, may be done. This means grouping the answers in topics before describing the cases, like a study of teenagers where the teenagers are clustered after characteristics first, as if they have ever been in jail or if they have had a single parent, before doing a case description, see

Section 2.9.4.

### 2.9.2 Early Analysis

In order to not get overwhelmed by the collected data it is wise to start the analysis in an early stage. There are eight methods suitable for early analysis. They are suitable to begin with during the data collection when they help the researcher to make sense of the massive amounts of data that is collected. The methods and what they means are presented as follows [11].

#### Contract Summary Sheet

Contract summary sheet can be used when the fieldwork, like observation or interviewing, has started. A contract summary sheet is a sheet with summarized questions about the fieldwork in one case. For each case a sheet is done. By identifying what the main issues and questions are, and then note it on the sheet, it is possible to summarize the main points.

#### Codes and Coding

If the researcher does not know what data that is more important in the information, he or she thinks that everything is equally important. This makes the organization of the data harder to do in a correct way. By attaching codes to the data, like tags or labels, it is easier to know what to focus on. For example, in a study there are ordinary personnel and there are the administrators. By the fieldwork their motivation for an event is researched. Suitable code then could be to label the personnel's motivation as MOT and the administrators' as ADM-MOT. These codes makes it easier for the researcher to get a view of what data matters more and what data that matters less.

#### Pattern Coding

Like in the method *codes and coding* this methods aims at making it easier for the researcher to overview the data, but pattern coding is easier to do. During the fieldwork the researcher finds threads that tie bits of the data together. Patterns coding hold four functions. First function is that it reduces the amount of data into a small number of units. The second function is allows the researcher to begin the analysis during the data collection. This makes further fieldwork more focused. Third function is that it makes it easier for the researcher to understand local interactions. And lastly, it lays a foundation for cross-case analysis when this is to be used.

#### Memoing

When conducting the fieldwork, whether it is observation or interviewing, there

are so much information that it is hard to make a deeper sense of what has happened. To easier analyze and organize the data, the methods mentioned above is recommend. Memoing helps the researcher to take the step from fieldwork to the actual coding of the data or to write *contract summary sheets*. Memoing captures the analysis thoughts that briefly enters the researcher's mind. A memo is theory, a write-up of ideas about codes and their relationship. Memoing helps refining and expanding the codes and to develop key categories and their relationship.

### **Case Analysis Meeting**

When sampling from many cases it is hard for the researcher to separate what data comes from which case and what analysis belongs to what case. With the purpose that the researcher shall be able to quickly understand what happened in each case a case analysis meeting is held. A case analysis meeting is a meeting with colleagues or co-researchers where each case is discussed. A set of questions guides the meeting where one purpose is to find the answer to these questions. Examples of questions are "What is strange or unexpected about this case?", "What people are in key roles?" and "What will probably happen in the next days in the case?"

### **Interim Case Summary**

There are four scenarios a researcher fear. First one is that the data is not good enough, the second is that their biases have interfered with the data, the third that their conclusions are too trivial and lastly that the data is not possible to analyze. Interim case summary help to identify if any of these fears have happened and is a summary of what the researcher knows about each case. It is a review of the findings and a plan for the next step in data collection. An interim case summary is an opportunity for the researcher to make a data account sheet. The data account sheet gathers each research question by either, e.g, the interviewees or by a criteria or class.

### **Vignettes**

Not all data is equal and it is important to formulate the core issue in a case (the researcher's theory of what has happened). A vignette is a description of events that do this. The description is focused and the events are a series of events that are typical for the case. It has a structure that is storytelling. The main characters in the vignettes are few and can be derived to a specific time and place.

### **Prestructured Case**

Before the data collection begins, but after the researcher has done a focused set of research questions and a sampling plan, prestructured cases are written. A prestructured case is a outline of the case that includes a *data display*. During the fieldwork, after each observation or interview, the *data display* is filled out. The

outline can be looked at as a "shell" for the data. During the fieldwork the researcher knows what the case looks like and fills the "shell" with data. But there is a weakness with prestructured cases. Coding of a prestructured case is done from the raw notes and not from the notes that have been worked on, as write-ups in *memoing*. This makes this method unsuitable for cases where the researcher is not familiar with the setting.

### 2.9.3 Approaches of Organizing and Reporting

There are three main approaches, or theoretical alignments, of how to organize and report the data and they contain several sub-approaches [15].

#### Storytelling Approaches

The chronology and history approach describes what happened chronologically while the flashback approach starts at the end and works backward to describe how the ending emerged.

#### Case Study Approaches

In the people approach, case studies of people or groups are in focus whereas in the critical incidents approach the focus is on critical events that often are presented in order of importance rather than in sequence. The various setting approach focuses on different settings, often by doing case descriptions before doing cross-case analysis.

#### Analytical Framework Approaches

The process approach organizes the data to describe important processes and the issue approach organizes the analysis to illuminate key issues, like how participant in a societal program change by the program. A common approach is question approach which organizes the responses from an interview into questions. Where sensitive concepts, as leadership versus followership, the sensitizing concept approach is used in order to organize those.

### 2.9.4 Practical Methods

There are some methods to practically analyze and organize the data. Most qualitative data analysis methods aim at generating theory, as constant comparison method and cross-case analysis, but some methods aim at validating theory, as triangulation, negative case analysis and member checking.

#### Constant Comparison Method

Constant comparison method is a theory generation method which begins with

coding the field notes and in the end it has generated an explanation of a phenomenon. It can be used on any set of field notes, whether they all come from the same case or setting, or whether they are the data collected from a number of settings. Constant comparison method can be defined as attaching codes or labels to the parts of the data that is especially important to the study. Then these parts are grouped into patterns based on the code or label. Each group is analyzed to find patterns. The next step is the writing of a field memo that articulates a theory or an observation derived from the coded data [20].

### **Cross-Case Analysis**

When the data can be divided into cases, cross-case analysis is appropriate. This means a set of strategies that allows the researcher to easier refine the results. The cases can be divided into groups based on some attribute, e.g, number of people involved, and then examined to see what similarities hold within each group, and what differences exist between the two groups. Another strategy is to compare pairs of cases to determine variations and similarities. A third strategy is to divide the data based on data source, e.g, interviews or observations [20].

### **Triangulation**

Triangulation method is used to confirm the validity of conclusions and is not only useable in qualitative studies. The basic idea is to gather different types of evidence to support the theory. The evidence might come from different sources, be collected using different methods, be analyzed using different methods, have different forms, like interviews, observations or documents, or come from a different study altogether [20]. There are mainly 4 types of triangulation that aims to strengthen a study by combining methods [15].

Data triangulation which is using different data sources in the study. Investigator triangulation which is using different researchers. Theory triangulation which is using many different perspectives to interpret the data, and lastly methodology triangulation which is using many different methods to study a problem.

Triangulation can be done by combining both interviews and observation, i.e, by using several different methods, and that makes triangulation expensive. In practice it means to, e.g, compare observation results with the results from the interviews or to compare the results from the interviews with written documents [15].

### **Negative Case Analysis**

Negative case analysis is another method to confirm hypotheses. When performed the process involves a search for evidence that is in conflict with the generated theory and rechecking the new theory against existing and newly collected data. The search for evidence that is in conflict with the theory can include purposely select-

ing new cases for study that increase representativeness as well as seeking new sources and types of data to help triangulate the findings [20]. The appropriate time and effort to put on searching for contradictions are to "assiduous search ... until no further negative cases are found" [15].

### **Member Checking**

Another method to confirm the findings is member checking and this method is well suited for the studies of software engineering when its aim is to get feedback on the findings from the subject who provided the data in the first place. For example, when the researcher has analyzed the results from an interview, he or she returns to the interviewee to get the data confirmed by the interviewee.

### **2.9.5 Credibility**

When conducting a study, the researcher has to be aware of and think about the credibility of the results. The credibility of the results mainly depends on the three different elements, rigorous methods, the credibility of the researcher and philosophical beliefs in the value of qualitative research. To increase the credibility, *triangulation* or *negative case analysis* could be used.

### **Rigorous Methods**

Rigorous methods means using strict and detailed methods in the fieldwork and doing the analysis of the data with the credibility in mind.

### **Credibility of the Researcher**

The credibility of the researcher depends a lot on the researcher's training, experience and status. If you are a more skilled and trained researcher it is easier to achieve high credibility.

### **Philosophical Beliefs in the Value of Qualitative Research**

Philosophical beliefs in the value of qualitative research means that the research should be conducted with the perspective of qualitative methods, inductive analysis, purposeful sampling and holistic thinking.

### **2.9.6 Threats**

When conducting a study there can be different threats to the study that needs to be avoided or the credibility will be affected. By learning what the main threats are, it is possible to learn how to avoid them or at least how to handle them. There are mainly three important threats and there are six different strategies (guidelines) to

avoid these threats [1].

### **Three Main Threats**

First, the researcher's presence might have an impact in fieldwork that causes the researched people to counteract with the goal of fieldwork (reactivity). Second, the researchers' biases may affect, e.g., what questions and in what way the questions are asked or what is noticed during an observation. Lastly, the respondent's biases may interfere with what issues that are focused on.

### **Strategy Prolonged Engagement**

Prolonged engagement means that the interviews are longer and follow-up interviews, with the same respondent several times, are conducted. It means spending longer time observing before trusting what is observed or trusting the relationship with the people observed. This may help to reduce and overcome biases and reactivity.

### **Strategy Triangulation**

By using *triangulation*, checking different sources and performing multiple analysis, reactivity and biases can be avoided, see Section 2.9.4.

### **Strategy Support**

Support protects against the researcher's biases. Support means in this strategy, e.g., getting questions. By getting questioned by peer researchers, biases are identified.

### **Strategy Member checking**

Another way to avoid biases and reactivity is *member checking*, i.e., getting feedback on the findings from the subject who provided the data in the first place, see Section 2.9.4.

### **Strategy Negative Case Analysis**

*Negative case analysis* means a search for evidence that is in conflict with the generated theory, i.e., a thorough search for cases that do not fit the researcher's interpretations, see Section 2.9.4.

### **Strategy Trail of Raw Data**

Leaving a trail of raw data, like observation protocol and interview notes that is reviewed, makes it possible for other researchers to judge if the findings are verified and reproducible.

**2.9.7 Protection**

To protect the data is essential. It should be treated like valuables, i.e, they should be kept safe. There should also be a number of copies of the raw data, as this is the data that never can be reconstructed since it contains, among other things, the original wording of people's answers and are very valuable when conducting the analysis [15].



### 3 Methodologies Used in Software Engineering

What is software engineering? What qualitative methodology has been used in software engineering? The following parts answer these questions.

#### 3.1 Definition of Software Engineering

To be able to discuss the methodology used in software engineering we first have to define the concept of software engineering and its relationship to other disciplines. There is no unambiguous definition of what software engineering is but one way to put it is "Software engineering is that part of computing which especially concerns itself with a systems view of software" [24]. One definition of computer science and software methodology is stated in [6]. It provides a definition of computer science that states computer science as a discipline with nine sub-fields where one of those is software methodology and engineering. Another way to put it is "a software engineer is one who applies a disciplined engineering approach to software development" when "understanding engineering is more important than understanding coding" [2].

Software engineers must have established purposes and goals when developing software. Purposes and goals are some of the key ingredients separating programmers from software engineers. In general, the six engineering goals or purposes that must be met to ensure the successful development of a computer software project are timeliness, efficiency, reliability, simplicity, modifiability and cost-effectiveness [5] where simplicity and modifiability are attributed to the user and his ability to use the software [2]. Serious programmers are not software engineers if they are not aware of real-world factors.

"Since software engineering's founding 1969, software engineering as a discipline has focused on the effort to re-design practice in accordance with the model of other engineering disciplines. Software engineering as a research discipline has focused on methods for product development, it has been successful in developing programming concepts, analysis and design methods, and process models widely used today" [18].

#### 3.2 Qualitative Methods Used in Software Engineering

In order to find out what qualitative methods have been used in software engineering, a search for such studies was done. There is little work done with qualitative methods, especially pure qualitative studies, in software engineering. Some of the qualitative methods have been used in software engineering, but often in combination with quantitative methods.

A summary of qualitative studies reports and reports on evaluation of studies is presented, see Section 3.2.1. Then follows a more in detail description of three different reports chosen for their widely different use of qualitative methods, see Sections 3.2.3, 3.2.4 and 3.2.5. In section 7.3 there is a reference to an article where five researchers discuss their points of view on qualitative methods in software engineering.

### **3.2.1 Summary of Studies**

The studies that are presented in the sections as follows are studies chosen via ELIN@Lund, i.e, Electronic Library Information Navigator. Via ELIN there is a possibility to search over 200 databases and approximately 11600 journals and over 200 other resources like dictionaries, encyclopedias, e-books and collections. ELIN is found on the web at <http://www.lub.lu.se>.

The search words used to find the studies are software engineering, qualitative, study in different orders and combinations. Besides this, references from the research papers found by ELIN are followed to other studies.

#### **Evaluating User Interfaces Using Techniques from Qualitative Data Analysis [9]**

This study is not entirely based on qualitative methods but quantitative as well, but the focus is on the qualitative methodology especially when analyzing and organizing the data. Its aim is to evaluate two user interfaces, to determine which of the two the users prefer, via observation and follow-up interviews.

#### **'Bad Practice' or 'Bad Methods' Are Software Engineering and Ethnographic Discourses Incompatible? [18]**

In this report ethnography and its contribution to software engineering is discussed and is revealing the different research attitudes of ethnographers and software engineers. It is mainly about different approaches to ethnography, the problems with, and how to use, these approaches.

#### **Qualitative Methods in Empirical Studies of Software Engineering [20]**

This report focuses on showing how qualitative methods can be adapted into the designs of empirical studies in software engineering by referring to two different studies, see study [19] and study [14], section 3.2.1.

#### **Communication and Organization: an Empirical Study of Discussion in Inspection Meetings [19]**

This is an empirical study that investigates the issue of communication among members of a software development organization. The methods used in this study

are both quantitative and qualitative. The data collection was qualitative, and the data analysis used methods of both types. The qualitative methods used are a prior ethnography, participant observation and semi-structured interviews.

**The Package-Based Development Process in the Flight Dynamics Division [14]**

In this study the purpose was to document the process that a software project team was following to produce software systems constructed from COTS components, i.e, Commercial Of The Shelf. The method used is interviews, open-ended and semi-structured. A different approach to cross-case analysis was used where each development project that was studied was treated as a separate case.

**COTS-based Software Development: Processes and Open Issues [12]**

This study is an investigation of the COTS-based software development, with an emphasis on the processes used. More than 30 COTS packages were used by 15 groups during this study. The method used is structured interview, and the study led to a new proposed process for COTS-based projects.

**The Time Famine: Toward a Sociology of Work Time [17]**

This study is not an absolute study within software engineering but it still shows a certain area in which qualitative methods may be used. The study is a qualitative study of how people use their time at work and includes a nine-month field study of the work practices of a software engineering team. The methods used are participant observation, interviews and shadowing (shadowing group members provides an in-depth understanding). The findings are analyzed sequentially and when applied with a sociology of work time perspective they provide new insights.

**A Qualitative Empirical Study of Case Tool Support to Method Learning [7]**

This is a study that investigates when it is appropriate to introduce CASE tools, i.e, Computer-Aided Software Engineering tools, in the process of learning a software engineering method. It is an empirical study and it uses questionnaires as a supplement, with both qualitative and quantitative views.

### **3.2.2 A Detailed Description of Three Reports**

A more detailed description of three different reports follows. They are chosen for their widely different use of qualitative methods, see Sections 3.2.3, 3.2.4 and 3.2.5.

### 3.2.3 Evaluating User Interfaces Using Techniques from Qualitative Data Analysis

In this report an approach to handling qualitative, data for the purpose of getting an overall picture of an interface evaluation, is described. The aim is to evaluate two user interfaces to determine which of the two the users prefers. Observation and follow-up interviews with the users have been performed and have provided a great deal of information but there are problems handling the unstructured data. The authors of the study have suggested that qualitative methods can be used to handle the information. To speed up the analysis process they used a computer program, NUD.IST, see Section 7.2, to assist the analysis of the qualitative data.

This study is not entirely based on qualitative methods but quantitative as well, but the focus is on the qualitative methodology especially when analyzing and organizing the data. By using a qualitative analysis it is possible to categorize the data in a way that allows retrieving (recovering) the data. Also quantitative analysis like frequency may be used to identify patterns that occur in the data.

The users of the interfaces were students and they only used them once a year. This lead to the conclusion that the software must be easy to use, although the students have experience of both Macintosh and PC interfaces. This is an important conclusion, i.e, it is important to identify the main group.

The practical methods in the study were observation and follow-up interviews. The users worked their way through seven case studies or scenarios in the same order for each interface. The reason that scenarios could be used was that the typical tasks were known from earlier analyses.

The interviews were done by structured interviewing and the questions were open-ended, neutral and clear, but the singularity in question number three was not fulfilled. Some of the questions, like number one, two, five and six, were also on the format presupposition questions. These were the questions about the two interfaces:

1. What are the best aspects of the system for the user?
2. What are the worst aspects of the system for the user?
3. Are there any parts of the system which you found confusing or difficult to fully understand?
4. Were there any aspects of the system which you found particularly irritating although they did not cause major problems?
5. What were the most common mistakes you made when using the system?
6. What changes would you make to the system to make it better from the user's point of view?

7. Is there anything else about the system you would like to add?

The students participating in the study were divided into two groups, one with six students and one with seven students, testing the interfaces and then being interviewed. The interviews were taped for later analysis. The notes taken by the observer were also transcribed so they could be used in the evaluation.

The analysis of the results of the interviews should answer the following questions:

- Which interface was preferred?
- What reasons were given for the preference? Did usability problems confound the evaluation?
- Was there any effect of the order in which the students used the interfaces?

The expected result was that, since the students had experience of computers, they should have no problems using any of the systems. And since they signed up for the study by filling out forms there should be no problems with these interfaces based on form filling.

The data was analyzed and categorized by the interview and was divided up into sections. Each interview was then entered into NUD.IST as separate documents named S1 for the first student, S2 for the second and so on.

To easier be able to understand the data, it was categorized into an index structured as a tree with a root and nodes named Ease of use, Speed, Preference, Problem areas, Interface type and Operation choice. Once this information had been collected, it was possible to answer the question "Which interface was preferred?"

To make sense of the patterns that had emerged, special terms were chosen. The terms describing ease of use was "hard", "difficult", "easy", and "easier". To describe the speed of the interfaces words such as "quick", "quickly", "fast", "faster" and "slow" were chosen (notice it is a four and a five point scale that is used).

### **3.2.4 Qualitative Methods in Empirical Studies of Software Engineering**

In this report [20] several qualitative methods for data collection and analysis are presented, like participant observation, interviewing and coding. They are described in terms of how they might be used in empirical studies of software engineering, in particular how they might be combined with quantitative methods, e.g. coding. It also focuses on showing how qualitative methods can be used in the designs of empirical studies in software engineering and refers to two studies that have been made.

Human behavior is one phenomenon that is complex enough to really need qualitative methods to study it. In software engineering there is a blend of technology and human behavior and naturally, according to the author Seaman [20], it requires a blend of quantitative and qualitative methods to study it.

A study based largely on observation data is a study of code inspection meetings, see study "Communication and Organization: an Empirical Study of Discussion in Inspection Meetings" above. Most of the data for this study was collected during direct observation of 23 inspections of C++ classes. The purpose of the study was to investigate the relationship between the amount of effort developers spend in technical communication and the organizational relationships between them. Data about organizational relationships was collected during interviews with inspection participants. The observations were done as follows. The observer noted the time the discussion started and ended, the participants in that discussion and the topic. The observer also noted a code that interprets what kind of discussion it was and the atmosphere. In order to evaluate the validity and consistency of data collected during the observations, rater agreement exercises were often conducted, this to ensure that the data was recorded in a form that is understandable not only to the observer. The results of the rater agreement exercise indicated that the data would have been more exact if more observers had been used for all observations, or if the meetings had been recorded. This is a very important conclusion to bear in mind when observation is conducted.

The other study referred to is a study of COTS integration, see study "The Package-Based Development Process in the Flight Dynamics Division" above. Interviews with developers were done on projects that involved COTS integration where many interviewees were interviewed multiple times. These interviews were open-ended and semi-structured. The purpose of the open-ended questions was to find information not anticipated by the interviewer. An example of a question is "What are the disadvantages of Package-Based Development (i.e, COTS integration) in comparison with traditional development?".

In the study, much of the qualitative data was coded into quantitative variables to be used in statistical analyzes. In the report, coding is discussed as a way to combine qualitative and quantitative data. The danger with this is rich information loss. The following excerpt is an example of that. Tom, Shirley, and Fred were the participants in the meeting. Now consider the following quantitative data, which was generated by coding the above qualitative data: num-participants = 3. As a consequence of coding, the participants' names for example is lost. Qualitative data often carries more content than quantified. This leads to the conclusion by coding qualitative information into quantitative data, statistical analyzes can be performed but the coding must be done carefully. The coding does not add objectivity or accuracy and is difficult to perform in a proper way where there is no more information loss than necessary.

When combining both qualitative and quantitative methods there is some variety of design, that can be categorized as follows.

**Blocked subject project study**

In this approach several subjects work in one project and the study contains several projects. It reduces biases but increases the cost due to the high number of subjects.

**Replicated project study**

In this approach several subjects work in one project, that is constant, to minimize the differences between the subjects.

**Multiproject variation**

In this approach the performance of the subject is studied before a treatment is introduced, e.g, training in a new technique.

**Single project study**

In this approach an in-depth study of a project, or an instance of a project, is done. Some of the project attributes are studied and then compared to a typical value.

Quantitative and qualitative methods of data collection and analysis can be combined in any of these types of study design. In the report there are some examples of such studies. An example is a study that evaluates a new software engineering technique and does this via blocked subject project design. This, so the technique can be tried on a variety of different applications that vary in different ways, i.e, the aim of blocked subject-project design is that several subjects work in one project and the study contains several projects, all to reduce biases. The result might be that the new technique was effective on some applications but not on others.

**3.2.5 'Bad Practice' or 'Bad Methods' Are Software Engineering and Ethnographic Discourses Incompatible?**

In this report [18] ethnography and its contribution to software engineering is discussed and promoted by revealing the different research attitudes of ethnographers and software engineers. It discusses major topics that are important to know when conducting an ethnographical study. It is also discussed how to combine ethnographic studies with a software method. It seeks answers to questions like how useful ethnographic studies are for software engineering.

Qualitative methods make the researcher unable to ignore the social aspects and according to the author, Rönkkö et al. [18], ethnography is one qualitative

methodology and has gained some recognition in software engineering. The results of the qualitative methods are produced in the form of words, not numbers.

The report presents three different approaches to relate to ethnography. The first uses ethnographical fieldwork to collect qualitative data in an initial research phase and then transforms it to qualitative data in the analysis phase. This does not qualify as an ethnographic study in its original sense. The second takes its inspiration from the same roots from which ethnography originated. The third approach relates ethnography in its original understanding to software engineering. There are several references to conducted studies, among them the studies mentioned above, see Section 3.2.4, but it does not, according to the author, Rönkkö et al.[18], qualify as ethnography in its original sense when qualitative data is transformed into quantitative data in the analysis. Ethnographical studies provide an inside view but because of the influence of ethnography in software engineering many software engineers think that qualitative data is better when transformed into quantitative data. But the information loss is extensive and the rich descriptions are destroyed by the transformation. Because of that, the author, Rönkkö et al.[18], offers another definition of what ethnography really is from a point of view where ethnography turns to the hermeneutic disciplines instead of the natural sciences for guidance, and where fieldwork experience is more important than the fieldwork findings in themselves.

Several small sections of larger studies that have software development as a common theme are presented. One of these studies presents the problem of merging software engineers work methods with ethnographical study methods. It also reveals problems within software engineering, like when software engineers planning the code in a low level to solve the problems in a high level, how and when to name variables, and if requirements or architecture should be developed first and so on.



## 4 A Qualitative Study of Pair Programming

How can a qualitative study be conducted within software engineering? After learning about qualitative theory and reading the studies and articles it is suitable to make a study to try out the qualitative theory and to come to a conclusion how to use the qualitative methods in software engineering. Because of that, the study will be based entirely on qualitative methods. The study will be conducted on pair programming which is a part of extreme programming [8]. When evaluating pair programming with the main purpose of the pair programming itself, the study might not be based on qualitative methods alone but on quantitative as well. In the sections that follow pair programming is explained and then the planning of the study is described.

### 4.1 Extreme Programming and Pair Programming

Extreme programming is a defined discipline of software engineering. It focuses, among other things, on communication, automated testing and pair programming. Extreme programming is a methodology that is flexible and meant to be used in projects with two to ten programmers. Extreme programming focuses on problems of the development process and derives solutions that dictate a set of activities [8].

There are certain values such as communication (via unit testing, pair programming and task estimation), simplicity and feedback. These values are distilled into concrete principles of extreme programming and then four basic activities, coding, testing, listening and design, are developed [8].

The 14 principles/practices of extreme programming are as follows [25]:

- Metaphor, this means that each application should be able to, via a simple metaphor, explain how the system works.
- Unit testing, test cases are written before the code is produced and are automated tested. The part of the code must pass all tests written before it is integrated into the rest of the code.
- Collective code ownership, means that no programmer owns the code, all programmers may change the code without the writer's permission.
- Functional test, the test cases are based on the customer's scenarios.
- Simple design, the design should be as simple as possible.
- Pair programming, two programmers work side-by-side at one computer.

- Refactoring, means the process of improving the structure of the code.
- Coding standards, an agreement on a code standard.
- Small releases, that means short time period between releases, like every 3-4 week.
- Open workspace, to enhance the communication all workspaces are common areas.
- Continuous integration, means that the coding assignments are divided into one day work.
- 40-hour week, no programmer should work more than 40 hours a week.
- On-site customer, the customer should always be available to the programmers.
- Planning game, each work period, like a month, is scheduled by giving every task a time estimation and a priority.

Pair programming is a technique which aims at making software development more efficient. With pair programming, all tasks must be performed by pairs of programmers using only one display, keyboard, and mouse, [13].

In [13], the advantages and disadvantages are discussed from an economical point of view, with a quantitative evaluation of pair programming. In [8], the basics of extreme programming are further analyzed.

## 4.2 Why Pair Programming?

Extreme programming is very much software engineering and is thereby a suitable issue for the study. Extreme programming contains many practices but one practice that is particularly interesting is pair programming since it has the complexity of two people working together, at the same computer. And that pair programming is a relatively new phenomenon (1996), [25], makes it even more interesting to study. With quantitative methods one can measure the number of faults, hours spent on programming and do some quality measurement to get an opinion of how effective pair programming is. But are there other things that may affect the efficiency of pair programming that can be discovered with qualitative methods? Like misunderstandings between the programmers? Does the main perception between the programmers differ? Does pair programming cause tensions between the programmers and if so, does it affect their job?

### 4.3 Planning the Study

In order to plan a study one has to clarify things like purpose, what methods shall be used and the basic thought behind the study. The following sections discuss these issues.

#### 4.3.1 Purpose

The purpose is to find both the strengths and weaknesses in pair programming, especially in the teamwork between everybody involved.

#### 4.3.2 Theoretical Tradition

What is the strategy and theoretical perspective of the study, i.e., what is the approach of the study? The time perspective does not allow a full participation research so all forms of *ethnographical* (*ethnography and auto ethnography*) studies are out of the question as these aim at studying a culture both from an outside and from an inside perspective and requires full participant observation which takes time. *Phenomenology* requires that the researcher has (enough) own experience of the phenomenon, this is not the case in this study, and when it is not clear what to look for and what weaknesses or strengths to expect this approach is not suited. It is the same with heuristic inquiry. *Narrative analysis* is about interpreting stories and is of course not the approach fitted for the study. *Ecological psychology* focuses on the relationship between human behavior and the environment. Qualitative description, based on observation as an onlooker, in the beginning of the study becomes quantitative data when it is analyzed. That is not consistent with the aim since the study should be based entirely on qualitative methods.

*Construction and constructivism* could be of current interest as it aims at constructing knowledge about reality, and the people involved in pair programming perceive the reality in different ways but it is not exactly what the study aims at. *Grounded theory* seems suitable when the data collected in the study is well grounded in the empirical world but grounded theory focuses on the process of generating theory rather than the theory and its content. It does not fit the purposes to extract weaknesses in pair programming that well. *System theory*, where the system is the pair programming method as a whole, could be of assistance when trying to understand the method when it answers the question "how and why does this system, as a whole, function as it does?". From a technical perspective it could be an advantage to look at pair programming, and its different elements, as a system.

If there was time, the most preferred approach would be both *grounded theory* and *system theory*. Since the researcher is inexperienced and the study has to be

conducted within a limited time period it is not possible even though they both are well suited for the purpose. Under these conditions *system theory* will be the strategy used in the study, i.e, to look upon pair programming and its different parts as a system where each pair programmer is a part. If one part is changed, the whole system is affected.

#### 4.3.3 What Kind of Research

To determine what kind of research the study is, the following questions have to be answered.

What is the purpose of the study, e.g, what makes one do the study?

Who are the primary audience?

What questions will guide the study?

What data will answer the study questions?

What resources are available?

What criteria will be used to judge the quality of the findings?

Purpose has already been established, to find strengths and weaknesses in pair programming and that is the questions the study will be guided by. The search for the data and how it is handled and judged is presented in the sections below. The resources available are pair programmers in a company but resources also means what time perspective, what aids and what literature that are available. The time is scarce, like a couple of days at the company and a couple of days to organize and analyze the data. This eliminates some practical methods like "going nature".

So, what kind of research will it be? *Evaluation research* is the kind of research that this study will answer to. *Evaluation research* aims at evaluating the results of the solution and that is what the study is about, evaluating the software engineering method pair programming. Is it *summative* or *formative evaluation research*? The answer is both, *summative evaluation* judges overall effectiveness and *formative evaluation* aims to improve programs. The purpose of the study was to extract strengths and weaknesses in pair programming and that makes it *summative*. But it is closely related to *formative* because when the weaknesses are identified some kind of solution of how to handle the weaknesses can be suggested, although the focus is on the *summative evaluation research*.

Why not use the other options? It is not "why"-questions that will be answered which make *basic research* out of the question, even if the aim is to "understand and explain". It is not *applied research* either because the study does not answer any "how"-questions nor helps to understand the nature of a problem as it is not sure there is a problem in the first place. *Action research* is also not the case as action research aims to solve problems within a program by persons in the program.

#### 4.3.4 Sampling

Now it is time to choose how and from what cases the information should be gathered, i.e, sampling methods. Information-rich cases are sought combined with cases that answer the purpose.

The sampling method that will be used is *convenient sampling* because of the narrow time limit. There has been no choosing between cases. The case from which the data is gathered is the case available at the time of the study. *Opportunistic sampling* takes advantage of whatever unfolds as it unfolds and could be used in the study as a complementary method as it leaves the door open for unexpected phenomena.

But what sampling method would be chosen if there was an opportunity to choose programming pairs?

*Extreme case sampling* where cases are chosen at the end point of the spectrum could be suited for the study when the time limit is narrow. If a programmer pair that is used to, and have worked with, pair programming for a long time is chosen and observed and then a beginner pair is chosen and observed, one can make out some difficulties in the beginning of pair programming. But the purpose was to find overall strengths and weaknesses of the programming method. So *extreme case sampling*, alternatively *intensity sampling*, could only be used as complementary sampling methods.

*Maximum variation sampling* is a better alternative. This sampling method is recommended when the resources are scarce, and focuses on difference between the sites where every site is unique. This method's aim is to find similarities in the samples despite the differences where the findings are important because they arise from widely different sites. The opposite of *maximum variation sampling*, *homogeneous sampling*, is by that discussion out of the question.

*Critical case sampling* is not well suited for the study either as it is not clear what weaknesses to expect. In *criterion sampling* the criterion would have to be defined, e.g, one beginner pair and one experienced pair, and is an alternative sampling method to consider. The same goes for *theory-based sampling*. Since *snowball sampling* asks people for references and nominations which takes too much time, it is not recommended to be used in this study. *Confirming and dis-confirming sampling* cases demands that one has a hypothesis of what to expect and so is not the case. In *purposeful random sampling* the cases are selected randomly to increase the credibility but the study focuses more on representatives.

#### 4.3.5 Practical Method

The study is divided into three parts, first some introduction *interviews* are conducted, then the main part *observation*, and last some follow-up *interviews* to

confirm and clarify the data.

### **Introduction Interviews**

The aim of introduction interviews is to get background facts about the programmers, as gender, age and education. The interviews were conducted with the *standardized open-ended interview approach*, i.e, the interviewers have a set of questions, carefully worded before the interview, and are of background question type. For the questions see Section 7.4.

### **Observation**

The main method in the study is observation. It is meant to be a *non-participant observation*. Since the observer is inexperienced it will be hard to keep the observation from interfering with the programmers work and with this in mind, together with the time limit, it will be a session of three to four hours of observation conducted on one programmer pair. Each programmer in a pair is looked upon as a part of the system pair programming, where the parts should work together using one computer. Observing how the work between the part goes is a key issue.

### **Follow-Up Interviews**

The aim of the follow-up interviews is to get more information about cases that still are fuzzy and to confirm data. The interviews were conducted with the *general interview guide approach*, i.e, the interviewer has a checklist of issues to discuss but within these issues an informal interview is conducted. The format of the questions is something like *follow-up probes* but done after the observation as a way to confirm and clarify the data. The interview method will be a kind of *elicitation interview* with the aim to clarify their image of reality but it is not a genuine *elicitation interview* as part of it may interfere with the *interviewing as dialog* method when the aim is to confirm data. For the questions see Section 7.6.

#### **4.3.6 Organization and Analysis**

To organize and analyze the data properly it has to be decided to whom it is going to be presented. In this case it is within a discipline to both students and companies. After deciding that, there are two choices, *case study* or a *cross-case analysis*. Since one of the methods is interviewing, a *cross-case analysis* of the answers can be done. This means grouping the answers in topics before describing the cases. Then a *case study* can be done on the *cross-case analysis* and the observation results.

What approach will be used to organize the results? None of the *storytelling approaches* is suited, as the results are not depending on any chronological order, and thereby do not add much structure in the organization. To find strengths and

weaknesses in pair programming, especially in the teamwork between everybody involved in the pair programming a *case study approach* will be used, more specific *people approach* where case studies of people or groups are in focus. One of the *analytical framework approaches*, *process approach*, will also be used as this approach organizes the data to describe important processes.

The practical method that will be used to confirm and analyze the data besides *cross-case analysis* is *member checking* when this is a recommended method in software engineering.

## 4.4 Performance

How was the study practically performed, based on the planning above? First a request of permission to study a programmer pair was sent to companies that use pair programming. From the answers a company was chosen, based on the accessibility and time perspective. A meeting with either the director of the development department or the pair programmers was scheduled. The company where the study finally was conducted is ABB in Malmö.

### 4.4.1 Case Study Description

A case study should take the reader into the context and be rich and detailed. First the company is described, then the characters and finally the observation procedure.

### 4.4.2 ABB

ABB is a worldwide company with offices in Asia, Europe, Africa and America. ABB focuses on two core businesses, Power Technologies and Automation Technologies. ABB Power Technologies serves electric, gas and water utilities and ABB Automation Technologies delivers products and services that is solutions for control, motion, protection, and plant integration. For more information see [www.abb.se](http://www.abb.se).

### 4.4.3 Introduction Interviews

The study started with a meeting with the programming pair, Christina and Ingvar, where they were asked the background questions. They are both Master of Engineering and have been working within software engineering for several years. They got their education within extreme programming and pair programming 2.5 years ago via a "coach", i.e, a consultant, that held a couple of lectures. Since then, they have been using extreme programming. The introduction interviews

were followed by a discussion. This as an extension of the answers to the question "What is your opinion of pair programming?" where Christina and Ingvar expressed several opinions about pair programming. Some issues were change of programming partner, how they thought they had more insight in projects and what would happen if they did not get along.

#### 4.4.4 Observation

The observation was conducted as a *non-participant observation* as planned. It was chosen not to ask questions during the observation in order not to interfere with the observation. Any questions raised during the observation were asked during the follow-up interviews. The work to be done by Christina and Ingvar was correcting code based on the outcome of test cases. In the observation notes a time stamp (hours.minutes) is used. When the study started the time was set to 00.00 and at different occurrences the time was noted as how long time since the study started, for the protocol see Section 7.5. The programmers Christina and Ingvar are referred to as C and Ing respectively like in the excerpt from the observation protocol as follows:

time 01.11: C and Ing have now finished one of the assignments and start on another part. C is still the one at the computer and she is making suggestions which Ing confirms. Now there are some code to remember so C is reading out loud while Ing is taking notes.

#### 4.4.5 Member Checking and Follow Up Interviews

The *member checking* aims at getting feedback on the findings from the subject who provided the data in the first place and is performed as Christina and Ingvar read the findings of the study and comment if they have objections. They had no objection of the data. According to their opinion, the data was correct. The follow-up interview was done at the same time as the member checking, see Section 7.6 for the questions and answers.

### 4.5 Results From the Qualitative Study

How did the qualitative study work out and what was discovered?

#### 4.5.1 Case Study

Christina is a female and she graduated as a Master of Engineering 1994. She got her training within extreme programming 2.5 years ago and has since then used it



in her work. During these 2.5 years she has mainly pair programmed and prefers this instead of single person programming.

Ingvar is a male and he graduated as a Master of Engineering 1980. He got his training within extreme programming 2.5 years ago and has since then used it in his work. During these 2.5 years he has mainly pair programmed and prefers a combination of pair programming and single person programming.

The study took place at ABB in Malmö, in Ingvar's office. The office has an ordinary desk with two computers, one work computer and one test computer, and two chairs.

#### 4.5.2 Result Possible Strengths

The results and recommendations are based on both the answers from the introduction interviews and on what was observed during the observation. The *system theory*, where each programmer is considered a part of the system, was also kept in during the analysis when focus is on the communication between the parts. If one part is changed, the whole system is affected. Some of the strengths could be weaknesses as well, depending on how they are handled. The communication is a strength when it is working properly but could, if it fails, turn into a weakness.

##### **Change of Partner**

In projects of a larger scale, the programmers had experience of changing partner every morning. This was experienced as a positive change. But it requires a straightforward dialog between the programmers and it is an advantage if the programmers know each other from the past.

##### **More Insight**

Based on what opinions the pair programmers expressed after the interviews the concept extreme programming, and indirectly pair programming, allows them to get a better and more detailed insight in the project. This affects their programming in a positive way when they feel they have better control and a better overview of the code. This is experienced positively by the programmers. If it produces better code is not an issue in this study, but the fact that they think they produce better code and have a more positive attitude to the coding is.

##### **Communication**

When the communication between the programmers works as it should it is definitely a strength - Two brains collaborating instead of one. But the communication is a delicate factor that is the most powerful. But at the same time the most vulnerable one when each programmer is a part of the system pair programming. This conclusion is mainly based on the results from the introduction interviews and the

discussion that followed. During the observation the communication was fine but, observed closely, some hints for it being a pitfall were found. See communication and roles section below.

#### **4.5.3 Result Possible Weaknesses**

All the strengths above can turn into weaknesses if they are mistreated or made use of in the wrong way or with the wrong purpose.

##### **Workplace**

To be able to be two persons on one computer the space in front of the computer has to hold two chairs and at the same time, both programmers must be able to have a clear vision the screen. This demands a well planned and ergonomic workspace. Is it possible to have two screens connected to the same computer so the programmers have one screen each? Does this affect the pair programming? Or perhaps a larger screen, big as a normal size tv would be better? Then the programmers do not have to twist their necks to be able to have a clear view of the screen. The workspace at ABB is poor, and forces the programmers to sit in a non-ergonomic position. This is however a one time problem, once it is solved it will not come back.

##### **Communication and Roles**

The programmers take on roles when they are coding. Which role is depending on which one of them managing the computer. When they change the manager of the computer the current role could be hard to let go. In the observation there was a tendency to hang on to their old role instead of adapting the new one like in the following excerpt: "time 00.43: The programming continues. C is taking over at the computer while Ing is sitting at her side. C is now starting to think her train of thoughts out loud and Ing is still doing so." Communication is the most important quality of pair programming to maintain, preferably before it becomes a problem.

##### **Adaption**

When working in pair programming the individual programmer has to adapt to the surroundings in a way that is not necessary when working in single programming. Pair programming sets a higher requirement on the individual programmer by the teamwork. If programmers fail to adapt to each other the result of the pair programming worsen. Based on the facts from the interviews, pair programming requires more from the individual programmer in terms of adaption.

##### **Tension and Feelings**

What would happen if the programmers in a programming pair disagree and there

becomes a tensed situation? What happens if the programmers become enemies? It is bound to affect their work in a way that it might not do when single programming is used. One of the solutions is to change partner.

#### **4.5.4 Recommendations of Pair Programming Based on the Study**

There are two possible weaknesses that I would recommend companies to commit to before they become weaknesses. The first is the workspace. Make sure the programmers have enough space to be able to work in an ergonomic way. The second is to maintain the communication between the programmers. Make sure that they get along and that they get to know each other properly.



## 5 Solutions and Recommendations

This section contains a summary of what has been done, the conclusions and future work. At the end the learning process is discussed.

### 5.1 Summary

In this master thesis the use for qualitative methods within software engineering is discussed. The focus is on how qualitative methods are usable in software engineering processes or methods, and not on how they are usable in a software engineering product. The report describes the basic theory and different strategies for the thoughts behind a study, different strategies for how and from what cases to sample the data and different practical methods to conduct a study. There are a discussion of the risks with *observation* as well as the strengths. These risks and strengths are important to keep in mind when conducting a study.

The importance of having a well defined purpose is discussed and there are six questions presented to help with definition of the purpose. Questions like "what is the purpose of the study, e.g, what makes one do the study?", "who are the primary audience?" and "what resources are available?".

Also presented is qualitative theory of how to organize and analyze the data and the importance of protecting the original data. There are several threats to a study, like the researcher's biases, that are important to identify as soon as possible and the strategies to do so are presented. The credibility of a study is important, and one way to ensure higher credibility is to choose the right methods. Another way is to use *triangulation* or *negative case analysis*.

This master thesis also has a briefing about studies made within software engineering where qualitative methods were used, which were followed by a more in-depth presentation of three of the studies in order to illustrate the widely different uses of qualitative methods.

A study on pair programming was done in order to use qualitative methods in software engineering. Strengths and weaknesses were identified within *pair programming* by using, among other things, *system theory*, *evaluation research*, *observation* and different *interview* approaches.

The theory behind the study and the results from the study are presented and discussed. The sections that follow discusses conclusions of this master thesis and what work that would be interesting to do in the future. The last section is about the learning process of this master thesis.

## 5.2 Conclusions

When using qualitative methods within software engineering it is important to adapt the methods to the technical discipline that software engineering is. For example, in the study of pair programming, the strategy *system theory* was adapted into a technical strategy. Pair programming was looked up on as a whole system where the components were pair programmers.

Software engineering is a technical subject. The study and its purpose would be better chosen and aimed if the researcher has some technical knowledge. This leads to the belief that when conducting a study based on ethnography or auto ethnography, an auto ethnography study it is to prefer. Then the researchers are able to use his or hers own technical experience and knowledge of software engineering.

The opposite of this strategy is strategies that focus on a person's methods and thoughts. Like how a person makes sense to the social world. These strategies are not based on a phenomenon or an specific activity but focuses on how a person handles situations. An example of such strategy is ethnomethodology.

One strategy that seems well suited to use in software engineering is grounded theory. Grounded theory focuses on the process of generating theory rather than the theory itself and provides guidelines of how to generate theory. Since the data is grounded in the real world there has to be a phenomenon involved to ground the theory with. Starting with the phenomenon, grounded theory could be of assistance, with adaption to software engineering, when making sense of the data, i.e, building the theory.

The strategies are summarized in the following table. The aim, i.e, the question that captures the essence of the approach can be read in column two and the recommendation for the strategy in relation to software engineering in column three.

Strategy	Aim of the Strategy	Recommendations when used in Software Engineering
Ethnography	What is the culture of this group of people?	Not to prefer when software engineering is such a technical discipline that it would be hard for a non-technical researcher to conduct a participant observation.
Auto Ethnography	How does my own experience of this culture connect with and offer insights about this culture?	To prefer over ethnography, suitable to use as the researcher has the necessary technical knowledge.
Reality Oriented, Reality Testing	What is really going on in the world? What can we establish with some degree of certainty?	Suitably, but only in small scale studies as computer science and software engineering develop at high speed, so the basic thoughts today may not be the basic thoughts of tomorrow. Could easily be adjusted to fit the technical perspective.
Phenomenology	What is the meaning, structure and essence of the lived experience of this phenomenon for this person or group of people?	Suitably, because it focuses on how people experience a phenomena and not on the phenomena itself. Adds nothing extra specific for software engineering.

<i>Strategy</i>	<i>Aim of the Strategy</i>	<i>Recommendations when used in Software Engineering</i>
Construction and Constructivism	How have people in this setting constructed reality?	Suitably when the study focuses on high credibility but adds nothing extra specific for software engineering.
Heuristic Inquiry	What is my experience of this phenomenon and the essential experience of others who also experienced this phenomenon intensely?	Theoretically suitable but difficult when all the researchers involved must have experienced the researched phenomena.
Ethnomethodology	How do people make sense of their everyday activities like to behave in socially acceptable ways?	Suitably in software engineering but adds nothing extra specific for software engineering.
Symbolic Interaction	What common set of symbols and understanding has emerged to give meaning to people's interactions?	Theoretically suitable but difficult when the researcher must be familiar with the technical language used in software engineering.
Hermeneutics	What are the conditions under which a human act took place or a product was produced that make it possible to interpret its meaning?	Not recommended. Focuses on interpretation of written material, e.g., interpreting a study, and not the process of engineering.
Narrative Analysis	What does this narrative or story reveal about the person and the world from which it came?	Not recommended. Focuses on interpretation of written material and stories, and not the process of engineering.
Ecological Psychology	What is the relationship between human behavior and the environment?	Theoretically suitable but focuses on the people and their environment. Adds nothing extra specific for software engineering.
System Theory	How and why does this system, as a whole, function as it does?	Suitably to use, software engineering can be divided into systems with many components like in the study of pair programming.
Chaos and Complexity Theory	What is the underlying order, if any, of disorderly phenomena?	Not suitable when software engineering not is supposed to contain any disorderly phenomena.
Grounded Theory	What theory emerges from systematic comparative analysis to explain what has been and is observed?	Suitably when making sense of the data, i.e., building the theory.

Basic and applied research focus on the questions how and why and would therefore be useful in software engineering. Especially applied research, that answers "how" questions can be used in studies that aim at explaining how a method work. On the other hand evaluation research is also useful as this type of research could be used to evaluate the work process and the computer programs within software engineering. For a small company action research is more suited because it is not as formal as the evaluation research.

In the subject sampling within the software engineering discipline it is hard to make any recommendations as the cases vary a lot and the development process never is the same. There are however one sampling strategy that should be avoided, convenience sampling, which focuses on sampling what is convenient and not necessary what is purposeful. The sampling strategies are summarized in the following table. What the sampling method focuses on can be read in column two and examples of when to use each method in relation to software engineering can be read in column three.

Strategy	Focus of the Strategy	Examples of when to be used in Software Engineering
Extreme Case Sampling	Handles cases that are extremely successful cases or cases that are an extreme failure.	Can be use when evaluating how well a programming method works and when to explain why it works as it do. Often used in ethnomethodology studies.
Intensity Sampling	Same logic as in extreme cases sampling but with less stress on the extreme, so while extreme case sampling seeks highly unusual cases, intensity sampling seeks excellent cases.	Can be use when evaluating how well a programming method works and when to explain why it works as it do. Often used in heuristic research.
Maximum Variation Sampling	Focuses on variations and differences. Has its strength in the variation of sampling sites.	Can be use when evaluating the same programming method on several different companies or in several different countries. Recommended in lack of resources.
Homogeneous Samples	Describes sub-groups in depth. Gathers people in similar situations with similar background.	Suitable when researching the programming technique of programmers that been working as software engineers for more than a certain number of years in a company. Focus groups are based on this strategy.
Typical Case Sampling	Describing to an onlooker what is typical for this setting.	Not that usable in software engineering. Could be used as a "pre-study" if a researcher is about to do a study within software engineering but has no technical education.
Critical Case Sampling	Cases and sites that are critical are studied.	If a technical manual is created and the company is not sure if it is understandable, they will test it on a group of non technical people and "if they understand it everyone will understand it".
Snowball or Chain Sampling	First locate critical cases and then ask people for references.	Can be used when looking for a way to solve how to design specific type of program. Ask around what the best way to do so is and then trust the solution that was suggested most times.
Criterion Sampling	Review and studies all cases that meet a criterion.	Suitable when, e.g, evaluating how well a computer program for people in a targeted age works. Base the evaluation on the people's opinions.
Theory-based Sampling	Samples things like incidents and time periods.	Suitable when, e.g, evaluating how well a computer program works for people in a targeted age with no computer skills that lives in a large city. Base the evaluation on the people's opinions.
Confirming and Disconfirming Cases	Used to confirm or disconfirm hypotheses which exist in the beginning of a fieldwork.	Can be used to confirm or disconfirm the believes that to much time is put into a certain phase in the development.
Stratified Purposeful Sampling	The is sampling divided into layers, like samples within samples.	Suitable to use to find variations in evaluations from different companies made on the same programming method.
Opportunistic or Emergent Sampling	Takes advantages of whatever unfolds as it unfolds.	Can be used when a company evaluates a new unknown technique that they do not know much about.
Purposeful Random Sampling	often randomly selected with the purpose of increasing the result's credibility.	Useful when evaluating a test period of a new programming method in order to convince the company managers to allow permanent use of it.
Sampling Politically Important Cases	variation of critical case sampling with a strategy trying to increase the usefulness of information with a political point.	Not that usable in software engineering.
Convenience Sampling	To do what is fast and convenient.	Not recommended. Can be used in studies where there is no possibility to choose cases.



To fully be able to use any practical research method within software engineering it is necessary for the researcher to know about software engineering. If the researchers do not have knowledge about programming it is hard to observe and interview software engineers. My recommendation to future researchers is to acquire knowledge about computers and coding before the study starts. Software engineers talk a very technical language that is hard to understand without knowing the common terms.

Qualitative methods produces in-depth and rich information but the analysis of qualitative data is fuzzy and complex. Could qualitative methods be used to gather the data and another type of methods be used to analyze it? Is the information loss worth a simpler and less fuzzy analysis method? Is there enough detailed information left after a non qualitative analysis? Because of the technical threshold in software engineering it would be recommended to do so.

### 5.3 Future Work

A pure qualitative study has been made using the core strategy system theory, but a long term study of software engineering using the strategy of *auto ethnography* would be interesting to do. Auto ethnography answers questions like "how does my own experience of this culture connect with and offer insights about this culture?". An auto ethnographer's perspective is an insider's view, i.e, he or she study his or hers own culture, from inside the culture, using his or hers experience of the culture. The purpose that would guide such a study is that it would be of interest to see if such a study could provide further in-depth information on software engineering. For such a study *applied research* would be used. The purpose of applied research is to contribute knowledge that will help people understand the nature of a problem. What cases and strategy would an auto ethnography choose to sample? *Typical case sampling* handles typical cases and has the aim to describe to an on-looker what is typical for this setting. Combined with *extreme case sampling*, that focuses on cases that are extreme in some way, this would be the sampling methods of choice. The practical methods that would be used are both *observation* and *interviews*. Would it be participant observation or not? Participant observation is the primary method of any ethnographical study and is suitable for this study. The participant observation would go on for a fairly long time, it would almost be like "going nature". The interviews are used as follow-up interviews and to clarify the findings. It would also be of interest to conduct a study where both qualitative and quantitative methods were used. This to compare if a pure qualitative study or a mixed study is to be recommended.

## 5.4 My Own Learning Process

In this master thesis it has been quite a long learning period, approximately half the time spent on the master thesis, where I learned about qualitative methods and the different strategies. Then a study was made. I had never conducted a study before and I learned a lot from it. It is hard to observe and interview. It takes a lot of skills that I noticed I do not fully possess. But at the same time I noticed that I had an advantage from my technical education within software engineering. At the end of this master thesis I have learnt so much and have got so much insight and knowledge in qualitative methods that it is tempting to continue within this subject. There are so many strategies I want to apply in software engineering, so many studies that can and needs to be done. What surprised me the most during this master thesis? Mainly two things. The first thing was that I found very little work done on the use of qualitative methods in software engineering. There are very much information on qualitative methods, and not so much on how it is used in software engineering. Secondly, when I started this master thesis I knew nothing about qualitative methodology. I had heard about observation and interviewing, these methods have been used in courses to elicit requirements for products, but I never thought qualitative methods could be so complex and with so many variations. I thought interviewing was just asking some questions and that observation was just to observe other people.

## 6 References

### References

- [1] Allen, R., "Standards for Rigor in Qualitative Inquiry", *Research on Social Work Practice*, Vol. 10, No. 2, pp. 173-179, 2000.
- [2] Baker, L.G., "Aggressive and Enthusiastic Software Engineering", *Program Manager*, Vol. 23, No. 4, pp. 24-27, 1994.
- [3] Belson, W.A., *The Design and Understanding of Survey Questions*, Gower, Aldershot England, 1981.
- [4] Boeree, C.G., *Qualitative Methods Workbook* Retrieved June, 2004 <http://www.ship.edu/cgboeree/qualmeth.html>.
- [5] Booch, Grady, *Software Engineering with Ada.*, The Benjamin-Cummings Publishing Company, USA, 1987.
- [6] Denning, P.J., Comer, D.E., Gries, D., Mulder, M.C., Tucker, A., Turner, A.J., Young, P.R., "Computing as a Discipline", *Computer*, Vol. 22, No. 2, pp. 63-70, 1989.
- [7] Eriksen L.B., Stage J., "A Qualitative Empirical Study of Case Tool Support to Method Learning", *Information and Software Technology*, Vol. 40, No. 5-6, pp. 339-345, 1998.
- [8] Juric R., "Extreme Programming and its Development Practices", *Information Technology Interfaces*, Proceedings of the 22nd International Conference on, pp. 97-104, 2000.
- [9] Kemp, E.A., Ots, S., "Evaluating User Interfaces Using Techniques from Qualitative Data Analysis", *Software Engineering, Education & Practice*, Proceedings. 1998 International Conference, pp. 222-229, 1998.
- [10] Krosnick, J.A. & Alwin, D.F., "An Evaluation of a Cognitive Theory of Response-order Effects in Survey Measurement", *Public Opinion Quarterly* Vol. 51, No. 2, pp. 201-219, 1987.
- [11] Miles, M.B., Huberman, A.M., *Qualitative Data Analysis*, Sage publications, USA, 1994.
- [12] Morisio M., Seaman C.B., Basili V.R., Parra A.T., "COTS-based Software Development: Processes and Open Issues", *The Journal of Systems and Software*, Vol. 61, No. 3, pp. 189-199, 2002.

- [13] Padberg F., Muller, M.M., "Analyzing the Cost and Benefit of Pair Programming", *Software Metrics Symposium*, 2003. Proceedings. Ninth International, pp. 166-177, 2003.
- [14] Parra A., Seaman C., Basili V.R., Kraft S., Condon S., Burke S., Yakimovich D., "The Package-Based Development Process in the Flight Dynamics Division", *Proc. 22nd Software Eng.*, pp. 21-56, 1997.
- [15] Patton, M.Q., *Qualitative Research & Evaluation Methods*, Sage publications, USA, 2002.
- [16] NCS Pearson, "Research notes: Choosing the right scale", Retrieved August 30, 2001 <http://www.ncs.com/ncscorp/research/95-3.htm>.
- [17] Perlow L.A., "The Time Famine: Toward a Sociology of Work Time", *Administrative Science Quarterly*, Vol. 44, No. 1, pp. 57-81, 1999.
- [18] Rönkkö, K., Lindeberg, O., Dittrich, Y., "'Bad Practice' or 'Bad Methods' Are Software Engineering and Ethnographic Discourses Incompatible?", *Empirical Software Engineering*, Proceedings. 2002 International Symposium, pp. 204-210, 2002.
- [19] Seaman C.B., Basili V.R., "Communication and Organization: an Empirical Study of Discussion in Inspection Meetings", *Software Engineering*, IEEE Transactions on, Vol. 24, No. 7, pp. 559-572, 1998.
- [20] Seaman C.B., "Qualitative Methods in Empirical Studies of Software Engineering", *IEEE Transactions on software engineering*, Vol. 25, No. 4, pp. 557-572, 1999.
- [21] Shuman, H., Presser, S., *Questions & Answers in Attitude Surveys - Experiments on Question Form, Wording and Context*, Sage publications, USA, 1996.
- [22] Sudman, S., Bradburn, N.M., *Asking Questions - A Practical Guide to Questionnaire Design*, Jossey-Bass, USA, 1982.
- [23] Walker R.J., Briand L.C., Notkin D., Seaman C.B., Tichy W.F., "Panel: Empirical Validation-what, why, when, and how", *Software Engineering*, Proceedings. 25th International Conference on, pp. 721-722, 2003.
- [24] Werth, J., Werth L., "Directions in Software Engineering Education", *Software Engineering*, Proceedings of the 13th International Conference on 1991, pp. 353-357, 1991.

- [25] Williams L., Upchurch R., "Extreme Programming for Software Engineering Education?", *Frontiers in Education Conference 2001*. 31st Annual, No. 1, pp. 12-17, 2001.



## 7 Appendix

### 7.1 Example of Elicitation Interview

An example of an elicitation interview [4] (R = researcher, I = informant).

R: Tell me, would you, what is that (pointing to an animal)?

I: That is a cat.

R: What kind of animal is a cat?

I: It is a meat-eater.

R: What other kinds of meat-eaters are there?

I: Oh, there's dogs, wolves, bears, rats, a few others.

R: Do some of these meat-eaters belong together?

I: Yes: Dogs and wolves belong together. So do rats and similar animals. Bears are different.

R: Is there something you call both dogs and wolves?

I: No.

R: How about rats and the similar animals, is there something you call that group of animals?

I: Yes, we call them vermin.

R: I see. What kind of animals are meat-eaters?

I: I don't know what you mean. They are just animals.

R: Are there other kinds of animals?

I: Sure. There are vegetable-eaters.

R: What kinds of vegetable-eaters are there?

I: Well, there's the little ones and the hoofed ones.

R: Can you give me some examples of the little ones?

I: Yes: There are rabbits, squirrels, chipmunks, woodchucks, and that's all I can think of.

R: Do any of these belong together?

I: Yes, the squirrels and the chipmunks do, they are tree-dwellers.

R: And the hoofed animals, what are some examples of these?

I: Deer, Cattle, Horses, Moose, Elk...

R: Any of these belong together?

I: Yes, the deer, moose, and elk belong together.

R: Is there a name for them?

I: They are the antlered animals.

R: Tell me, what kind of animal is that (pointing to a bird)?

I: Ah, that's not an animal. That's a spirit.

## 7.2 Example of Software Programs for Analysis

Some web addresses to where one may find more information about computer programs that are of assistance when analyzing qualitative data.

[www.cdc.gov/hiv/software/answr.htm](http://www.cdc.gov/hiv/software/answr.htm)

[www.atlasti.de](http://www.atlasti.de)

[www.scolari.com](http://www.scolari.com)

[www.qualisresearch.com](http://www.qualisresearch.com)

[www.researchware.com](http://www.researchware.com)

[www.spss.com](http://www.spss.com)

"The program NUD.IST (Non-numerical Unstructured Data. Indexing Searching and Theorizing) is widely used to assist with the analysis of qualitative data. Several features of NUD.IST can be used to help with the analysis of interview data collected for the purpose of a holistic evaluation" [9].

## 7.3 Useful Article that Discusses Useful Points of View

### **Panel: Empirical Validation - What, Why, When, and How [23]**

In this article the authors discuss their points of view on qualitative methods in software engineering. Since they all have different experiences and background they differ a bit in their opinions but they all have an idea of how and for what it is useful.

## 7.4 Questions for the Introduction Interviews

The aim with introduction questions is to get some background information to easier be able to describe the settings. Like the programmers' age, education and experience of pair programming. The following questions are the ones to be asked.

What is your age? (Then note the gender as well)

What is your education?

What is your experience of pair programming?

What is your training in pair programming?

What is your opinion of pair programming?

The three latest questions are in the format of presupposition questions and can, as an alternative, be divided into two questions each like: Do you have any experience of pair programming? If the answer is yes, then ask the question What is your experience of pair programming? The same with the question of education and opinion of pair programming. It is not an option to ask the question "Do you



have any experience of pair programming and what is the experience?" when this question is not in a singular format.

Finally the question "Do you prefer pair programming or ordinary single person programming?" was asked to round up the interview.

## 7.5 Observation Protocol

The two programmers involved in the study, Christina and Ingvar, will be referred to as C and Ing respectively.

Date: 2004-09-22

time 00.00: C and Ing gather and the workday begins.

time 00.07: Planning game, a short meeting with the project leader, takes place in order to get the assignments of the day.

time 00.09: C and Ing decide on room to be in, and Ing's room is chosen. The pair programming begins. At this moment Ing manages the computer and C is the programmer sitting at his side. In this composition Ing is talking his train of thoughts out loud while C is questioning, giving response to his thoughts and propositions of what to do.

time 00.19: C calls someone to get clarity of the naming of some of the code. Meanwhile Ing is checking the test computer.

time 00.22: Ing continues to think out loud and C continues to give response.

time 00.32: Time for coffee break

time 00.43: The programming continues. C is taking over at the computer while Ing is sitting at her side. C is now starting to think her train of thoughts out loud and Ing is still doing so.

time 00.50: The communication between Ing and C is now gone to a mutual communication, i.e, Ing is not thinking out loud as much as he is giving response to C's thoughts and propositions of what to do. The communication has stabilized and they have found their converted roles. The communication is light and playful.

time 01.11: C and Ing have now finished one of the assignments and start on another part. C is still the one at the computer and she is making suggestions which Ing confirms. Now there are some code to remember so C is reading out

loud while Ing is taking notes.

time 01.30: So far, they have been correcting code and some refactoring has been done but now it is time to write some tests. Ing takes over the computer.

time 01.45: The code that they are writing now is introducing changes in the naming and they are finishing the changes. At this time C is the one who remembers where the changes are made and is telling Ing where it has to be done.

time 01.53: The project leader comes to the room to ask how it is going, he stays for a minute and then C and Ing continue as before.

time 02.15: C and Ing are now finished with the coding assignments for today and thereby the study ends.

## 7.6 Questions for the Follow Up Interviews

The member checking and follow-up interview were conducted at the same time. The additional questions asked were:

Do you have a favorite position when pair programming, i.e, is there one of you who likes it better to manage the computer and the other one better likes to sit beside?

What determines who does what, i.e, does one of you has his or her strength in certain areas or do you change the manager of the computer randomly?

The answers are:

We have no favorite position, we like to take turn at the computer and change positions on regular basis. There is nothing particular that decides who is doing what.