En fallstudie om dokumentationens betydelse

i ERP-systemutvecklingsprojekt

Filosofie kandidatuppsats inom informatik

Författare:	Andersson, Andreas
	Bergsten, Thomas
Handledare:	Jörgen Lindh
Framläggningsdatum	2008-01-22
Jönköping	Januari, 2008

A case study of documentation's significance

in ERP system development projects

Bachelor's thesis within informatics

Author: Andersson, Andreas

               Bergsten, Thomas

Tutor: Jörgen Lindh

Jönköping   January, 2008

Kandidatuppsats inom informatik

Titel:	**En fallstudie om dokumentationens betydelse i ERP-systemutvecklingsprojekt**
Författare:	**Andreas Andersson, Thomas Bergsten**
Handledare:	**Jörgen Lindh**
Datum:	**2008-01-22**
Ämnesord	**Dokumentation, ERP-system, Systemutvecklingsprojekt, Integration, SDLC, dokumentationsmetod**

Sammanfattning

För att företag ska kunna behålla sin konkurrenskraft i näringslivet, behöver de hela tiden utvecklas. Men med utveckling kommer en ökad komplexitet som måste hanteras på något sätt. Många företag väljer då att investera i ett affärssystem (s.k. ERP-system), vilket inte är något som görs lättvindigt utan är en ansenlig investering. Detta faktum tillsammans med dagens låga vinstmarginaler i företag gör att det är väsentligt att hålla kostnaderna så låga som möjligt. Mer än hälften av kostnaderna för ett ERP-system sägs vara relaterat till underhålls- och supportverksamhet. En av de faktorer som påverkar denna stora kostnad är dokumentationsaktiviteter, vilken ifall de inte är utförd på rätt sätt kan leda till en ansenlig ökning av ett systems totala kostnad. Det är därför väldigt viktigt att inte förbise dokumentationen som en del av ett ERP-systemutvecklingsprojekt.

Syftet med vår studie var att undersöka existerande dokumentationsrutiner i ERP-systemutvecklingsprojekt hos Volvo AB's dotterbolag Volvo IT, för att sedan presentera förbättringsförslag för framtida projekt i företaget. För att kunna uppfylla syftet med uppsatsen, tog vi fram en forskningsfråga med tre följdfrågor och genom en grundlig analys av det empiriska materialet kommer vi att presentera svaren på dessa frågor som våra slutsatser.

Vi har valt att genomföra en fallstudie med en induktiv ansats, för att införskaffa oss djupare kunskap och därmed kunna generera ny kunskap inom området. Det empiriska materialet analyserades utifrån en given modell där insamlad data delades in i lämpliga kategorier. Kunskapen genererad från vår studie är av följande kunskapsform; kategoriell, förklarande, förutsägande och normativ.

Efter att ha analyserat resultatet av vår fallstudie kan vi presentera ett flertal förbättringsförslag för Volvo IT's användning i framtida projekt. För andra företag, involverade i ERP-systemutvecklingsprojekt, kan dessa fungera som värdefulla förslag att ta i beaktande. Vi kom fram till att den främsta anledningen till bristande dokumentation inte är omedvetenheten om dess betydelse, utan avsaknaden av konkreta bevis på dess positiva effekter på ett systems totala kostnad. Vi anser att det krävs en attitydförändring kring dokumentation som arbetsuppgift för att uppnå dokumentation som är av hög kvalitet och att arbetsuppgiften som sådan behöver få högre prioritet i projekten. Ett annat viktigt förbättringsförslag vi fann var att en kontroll av den producerade dokumentationens kvalité måste utföras, och inte bara en kontroll av att den rätta sortens dokument finns. Vi anser att ett företag inte bör allokera resurser till att skapa en metod eller modell över hur man ska dokumentera förrän vikten och effekten av hög kvalitetsdokumentation är klargjord. Till sist anser vi att systemutvecklingsprocessen bör fokuseras kring den framtida användaren, dels på grund av dagens ökade användning av skräddarsydda system men också för att det minskar risken för en oönskad effekt på projektets omfattning samt att det ökar kvaliteten på den slutgiltiga dokumentationen.

Bachelor's thesis in Informatics

Title:	A case study of documentation's significance in ERP system development projects
Author:	Andreas Andersson, Thomas Bergsten
Tutor:	Jörgen Lindh
Date:	2008-01-22
Subject terms:	Documentation, ERP-system, System development projects, Integration, SDLC, methods of documentation

Abstract

In order to stay competitive in today's changing business world, companies need to manage the increased complexity as they evolve. To be able to handle this complexity, many companies chose to implement an ERP system. Investing in an ERP system is not something that is made in a trice but is a large investment which together with today's low-profit margins in companies makes it essential to keep the costs as low as possible. More than half of the total cost for an ERP investment is said to be related to costs for the system's maintenance and support. Documentation is one of the factors which affect this cost, and if it is not made satisfactorily it is said to result in a considerable increase of the system's total cost. This is why it is important not to overlook the documentation as a part of ERP system development project.

The purpose with this thesis is to investigate the existing documentation routines in ERP system development projects at AB Volvo's subsidiary Volvo IT, in order to find suggestions of improvement for future projects. In order to fulfill the purpose of the study, we formed a research question with three sub-questions and through a deep analysis of the empirical material we presented the answers to these questions as the conclusions of this thesis.

Our study is based on an inductive research approach using a case study to gain deeper and more helpful qualitative knowledge and data. The empirical data was analyzed using the template analysis method where we divided the collected data into appropriate categories. The knowledge created through this study is of exploratory, normative, predictive and categorical nature.

After having analyzed the results from our case study we found several suggestions of improvements for Volvo IT to use in future projects. For other companies involved in ERP system development projects of their own, the conclusions of this thesis will work as valuable issues to take into consideration for upcoming projects. We have concluded that the main reason behind lacking quality of documentation in the development projects is not the unawareness of its importance, but the absence of concrete evidence of high quality documentation's positive effects on a system's total cost. We believe that in order to achieve documentation of high quality, there needs to be a change of attitude to documentation as a work task and the task itself must be higher prioritized in the projects. Another important suggestion is that a control of the conducted documentation's quality must be done, not only a control of the existence of the right kind of documents. We are of the opinion that a company should not allocate resources to create a method of documentation before the importance of high quality documentation is clarified. Ultimately, we believe that the system development process should be focused around the customer, because of the more frequent use of custom made solutions and to help the people in the project to set the right level of abstraction on the documentation.

Acknowledgements

We want to take the opportunity to thank a number of people who have been important for the fulfillment of the thesis;

Our tutor, for his commitment and guidance;
Jörgen Lindh, *Ph.D. in Science, Associate professor Informatics.*

* * *

Our contact persons at Volvo IT, for the possibility to conduct our empirical study;
Per Bengtson & **Olof Sidén**

Finally we want to thank all respondents who participated in the conducted interviews and the persons who proof-read this thesis.

Andreas Andersson & Thomas Bergsten

Jönköping, January 2008

Table of content

Table of figures

Appendix

1 Introduction

In this chapter the background to the subject is introduced, and a discussion around the problem of the thesis is presented. Furthermore will the stated research questions and its purpose to be clarified. The chapter is concluded with the delimitations of the study, the interested parties, some helpful definitions and a description of the disposition of the thesis.

During our education at Jönköping International Business School, we have encountered many courses within business modeling, business re-engineering, organizational change, and software development. Enterprise Resource Planning (ERP) systems have been a part of our education and one of the authors of this thesis had the opportunity to attain a certificate within SAP R/3 from the University of Arkansas, U.S.

We both have a great interest in ERP systems with its intention to help an organization by creating hopefully faultless connections between departments and program modules, which can lead to more efficient business processes.

For this thesis, we had the opportunity to make contact with Volvo IT (section 3.4.1) which uses SAP as one of their ERP system. A problem concerning documentation was brought to our attention and we decided to focus our thesis on that issue. Although our study is based on Volvo IT's stated problem, we are confident that the result of this thesis will be of great significance for other companies involved in ERP system development projects as well.

The development of information technology has been a frequent part of the literature and in different medias for a long time. There are articles about different opinions regarding ERP systems pros and cons. The majority of the existing literature are dealing with the implementation of ERP systems, how to develop an ERP system and also a little bit about the documentation of system development in general. The existing literature will be used in this thesis in order to write the theoretical framework which will help the reader to comprehend the empirical findings.

1.1 Background

Companies today are part of a global environment which is characterized by rapid change, explosive technical development and tough competition, together with a gigantic flow of information. To improve and sustain the competitive advantage many companies have realized that it is necessary to find new ways of conducting the business as effective as possible, which demands new work methods and a different perspective on the companies' businesses.

The rapidly changing environment, forces organizations to develop, find new solutions to problems and expand. As organizations expand, they also become more complex. To be able to manage the complexity, the implementation of an enterprise resource planning (ERP) system can be a solution. The ERP system is meant to support the organization by combining the different functionalities of the organization, to a process-oriented business where the intention is to create higher value for the customer. The concept of ERP system can be seen as software with the intention to control a company's internal and external information flow or a system designed to manage the data and information requirements of the entire organization (Avraham, 1999). ERP systems are standardized systems, which mean that the systems to a large extent control the design of processes and work methods. Because of the fact that an enterprise system is a standardized system, it cannot easily be custom made for the specific company which purchased it.

An ERP system is characterized by a number of modules. Which modules to be implemented are depended on the company's different demands regarding functions and work methods, and can be considered to be a kind of customization of the system. The modules of an enterprise system cover most of the processes which can be found in an organization and each and every module consists of many different functions (Avraham, 1999).

1.2 Problem discussion

With an implementation of an enterprise system comes many promises, for example to develop the business or making it more effective. Successful or not, the implementation of an ERP-system is a large investment (Kaplan, 2002). With an increase in competition and lower profit margins, it is essential that the cost is kept as low as possible. What is important to keep in mind is that up to 80 percent of the total cost for an ERP system investment is connected to maintenance activities of some sort (Kaplan, 2002). One of the main factors that affect the system's maintainability is the actual quality of the documentation in the system development process. If the documentation throughout the development process is not satisfying, it leads to a substantial increase in maintenance effort and total cost (George, J.F., Batra, D., Valacich, J.S., and Hoffer, J.A., 2004).

Different businesses have different needs. It is therefore essential that the software can handle and adapt to business specific solutions. Earlier research shows that different models and modeling techniques are needed for the implementation process, the distribution process, and the development process (Buck-Emden, 2000). Every development process is unique in one way or another. But it is argued that they are still more similar than they are different, meaning that every process have a similar development cycle with different phases. These phases include many activities to be undertaken, and each of these activities needs to be documented in some way. By the time the development process reaches the final phase which often is related to implementation and operation, the documentation should be considered finalized. Although the documentation should be finalized this is not always the case. Due to different reasons the documentation is sometimes lacking in quality, partly missing and/or not up to date which leads to a more difficult continuous work with the system e.g. higher costs of the system maintenance (George et al, 2004).

1.2.1 Problem specification

The problems around documentation discussed above are something that was brought to our attention by Volvo IT. Volvo IT experience that their existing documentation routines in system development projects are insufficient due to different reasons and in need of an overhaul and possibly some suggestions of improvement.

In order to fulfill the purpose of this thesis and solve the stated problem, we have set the following research questions.

Question 1. *How can the ERP-system development projects documentation be improved?*

To be able to answer the previous research question we first need to find answers to the following questions;

Question 2. *In what way is documentation a part of the ERP-system development projects?*

Question 3. *What does the system development projects documentation look like today and why?*

Question 4. *Which are the problems of the existing documentation routines?*

1.2.2　　　Purpose

We will investigate the existing documentation routines in ERP system development projects at Volvo IT, in order to find suggestions of improvements for documentation in ERP system development projects.

1.3　　　Delimitations

We have chosen to look at the documentation which in different degree takes places during a system development project. In this thesis we will assume that the pre-study and the concept-study of Volvo IT's IS-GDP model (appendix 1.1) are completed. This study will not deal with the documentation taking place during these initial phases, such as the requirement specifications or the business blueprints. We will assume that the documentation produced in these phases are completed.

Although the development processes of different systems are similar, it cannot be seen as a standard process in disregard of the type of system being developed. The different systems differ in size, complexity and in the total cost of implementation. The fact that documentation affects the maintainability cost of a system makes the effects of the documentation more apparent in an ERP system development project than in other smaller development projects due to its substantial total cost. This is the reason why we have chosen to look at the documentation routines in an ERP system development project specifically.

1.4　　　Interested Parties

There are a number of interested parties of this thesis. We have been granted access to Volvo IT, and the access will be further explained under chapter 3.4.1. Volvo IT has an extra interest in the result of this thesis and should be considered the main interest party because of their direct involvement. There are also other interested parties such as companies using an ERP system, which can use the result of this thesis when developing their current system. Companies which plan to implement an ERP system can take advantage of the study when implementing the system into the organization. Another interested party is companies that work with the development or implementation of ERP systems.

1.5　　　Definitions

In this section a number of terms and abbreviations used in the thesis are defined.

ERP-system	Enterprise resource planning system is a packaged business software system that can enable a company to manage the efficient and effective use of resources by providing an integrated solution for the organization's information processing needs (Nah & Kuang, 2001).
ISO	International Organization for Standardization is an international organization which sets standards for different business areas.
PERT	Program Evaluation and Review Technique (PERT) is a project planning technique where the different tasks are divided into smaller and more manageable parts and where the order in which the tasks needs to be done are visualized (Avison & Fitzgerald, 2003; George et. al, 2004).

RUP	Rational Unified Process is an iterative development process. The method includes guidelines for the all the different phases during a system development project (Strand, 2003).
SAP	A German company which is one of the world's largest business software company and the world's third-largest independent software provider. The original name for SAP was Systeme, Anwendungen, Produkte, which is German for "Systems Applications and Products" (SAP, 2007).
SAP R/3	An ERP-system with a comprehensive set of integrated business applications from the company SAP and R/3 is the latest system version. It replaced the system R/2 which is still used in many organizations (SAP, 2007).
SDLC	Systems Development Life Cycle is a methodology that marks the different phases of information systems development (Avison & Fitzgerald 2003).
UML	Unified Modeling Language is a modeling language constructed to model, document and develop different parts of a software system. It is based on "best engineering practices" (Strand, 2003).
WBS	Work Breakdown Structure is a project planning technique with the same concept as the PERT but where the relation between the parts are visualized (Avison & Fitzgerald, 2003; George et. al, 2004).

1.6   Disposition of Thesis

Chapter 1
Introduction

This chapter is the preface for the thesis where the problem, purpose and research questions will be focused upon.

Chapter 2
Theoretical framework

The reason for this chapter to precede the method part of the thesis is that we could not find any previous theory for how to document in relation to ERP-systems specifically, and by using the inductive approach this is the most appropriate choice. This chapter will give the reader required knowledge of the subjects brought up in the thesis.

Chapter 3
Method

The appropriate selection of method will be discussed in this chapter with the intention to create an understanding for the reader of the thesis regarding the chosen research method and the knowledge which are intended to be created through the conclusion of the thesis.

Chapter 4
Empirical findings

This is the chapter where the result of the empirical study is presented. Only relevant data in regard to the research questions and drawn from the conducted interviews will be included here.

Chapter 5
Analysis

An analysis of the empirical data will be presented in this chapter, where the data is analyzed with the research questions of the thesis in mind.

Chapter 6
Conclusion

The most important conclusions drawn from the thesis work will be presented in this section.

Chapter 7
Discussion

The drawbacks of the thesis and our reflections over the thesis work will be included in this chapter as well as our opinions regarding possible future work within the area.

2   Theoretical Framework

Concepts regarding system development in general such as the system development life cycle and different approaches or methods for developing a system will be included in the theoretical framework. This chapter will also include concepts concerning documentation such as its involvement in system development projects, different types of documentation, the quality of the documentation and the existing standards of documentation. All these terms and concepts are important for the reader to understand in order to comprehend the empirical chapter.

2.1   Systems development concepts

To get a basic understanding of the core concepts of system analysis and design, figure 2-1 shows an overview of these elements which are part of the software engineering process.
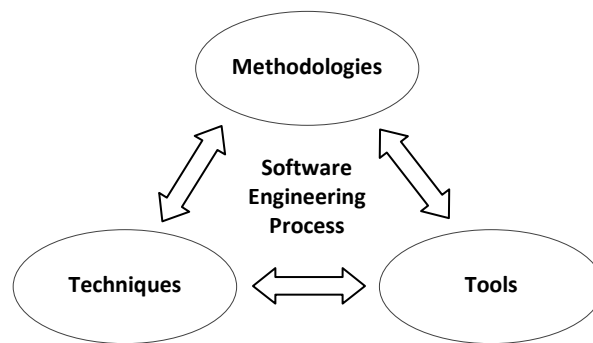

Figure 2-1 The core concepts of system analysis and design.

Methodologies are a sequence of step-by-step approaches that help in the development of the final product.

Techniques are processes that analysts follow to help ensure that their work is well thought out, complete and comprehensible. Techniques include guidance of how the different tasks should be done such as planning, interviews with current and future users, and also mapping important tasks (which will be explained in 2.1.2).

Tools are most often computer programs, for example computer-aided software engineering (CASE) tools such as the Office package (George et al, 2004).

2.1.1   Systems Development Life Cycle (SDLC)

The purpose of the SDLC methodology is to establish procedures, practices, and guidelines governing the initiation, concept development, planning, requirements analysis, design, development, integration and test, implementation and operations, maintenance and disposition of information systems development (IRM, 2003).

It was first in the 1970s and especially during the 1980s when the methodology SDLC became very popular. The whole idea of the SDLC is that you divide your systems development project in three or more manageable phases. Different organizations have different approaches to the SDLC model but the concept is all the same. The basic structure is feasibility study; system investigation; systems analysis; systems design; implementation; and review and maintenance (Avison & Fitzgerald 2003). Having to "pass" through phases in

the development helps identify problems. The majority of errors and problems can be traced back to the user requirements specification (George et al, 2004).

The SDLC contains all these different functions; the organization that intends to develop or upgrade their system and use this system has to decide and devote the necessary recourses to acquire it. It all starts out with a careful study of how the system is used today or how a new system should work. Then a team develops a strategy for how to design the new system, which is then either purchased or developed in-house. Once the system is complete, it is implemented, tested and put to use in the daily work.

Using the SDLC gives a number of different advantages. The SDLC is based on a strong methodology of the development process. Following this process and using the SDLC in the guiding way as it is suppose to, it will help the project to, in a structured way get from one phase to the next. Dividing each phase in to sub-phases and tasks, or in other words creating a Work Breakdown Structure (WBS) to make sure everything is complete and that it is done with quality (George et al, 2004). In section 2.1.2, the WBS is more thoroughly explained.

Presenting the SDLC in this way might give the impression that the model is flawless. The truth is that there is no flawless model for developing a software system. This model, called the waterfall model or the SDLC method is basic concepts for how the development should take place. All organizations and companies have their own design of these models and methods. They configure the model to suit their unique business and development needs.

There are a number of mentioned weaknesses with these models, some of them are:

- Failure to meet management needs
- Instability, inflexibility
- Documentation problems
- Incomplete systems
- Lack of control

To avoid the different potential problems and weaknesses with the model, the customization is necessary (Avison & Fitzgerald, 2003). This can be made with two different approaches; the "structured analysis and design" approach or the "object oriented system analysis and design" (OOSAD) approach (George et al, 2004).

2.1.1.1 Structured Analysis and Design

The following model (figure 2-2) is called the traditional life cycle for information systems development, also known as the waterfall model. The structured way of using a SDLC for systems development is an approach characterized by steps. You finish one phase and then move on to the next, with minor concern of controlling what have been done. The difficulty of returning to an earlier phase once it is complete has been compared to the difficulty of swimming up a waterfall (Bennet, McRob & Farmer, 2002).
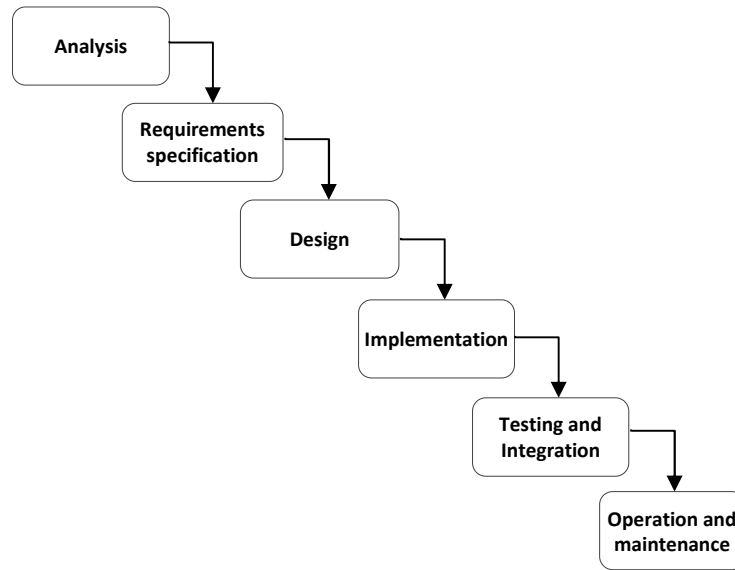
Figure 2-2 The traditional life cycle for IS development.

The lack of iteration and inter-phase communication might lead to that the final project does not turn out as the customer wanted it to (Avison & Fitzgerald, 2003). Due to the lack of communication with the customer during the development process it might be a costly experience. For smaller and less complex systems this might be a good approach. While in a bigger project this is associated with high risks (George et al, 2004). Most structured SDLC models were developed in the early 1970s and 1980s and were based on the waterfall model (Bennet et.al, 2002). When larger projects and more complex systems are developed it requires a higher degree of control and customer integration. To gain more control and be able to go back to the previous step in the cycle, the developers have started to move more towards the OOSAD approach which will be explained in the next section.

2.1.1.2   Object Oriented Systems Analysis and Design (OOSAD)

The differences between the model below (figure 2-3) and the traditional life cycle model (figure 2-2), are apparent.

Figure 2-3 An OOSAD approach to the traditional IS life cycle.

The basic concepts of the two models are still the same, but what differs is that the OOSAD model, or methodology as we may call it, is dependent on the inter-phase communication. After finishing each phase there is an evaluation of the performed work. This analysis integrates the previous phase, the current phase, the scope of the project, and the customer of the project. In the end of each phase more details are added for the next phase, which might not have been seen earlier in the project (Bennet et.al, 2002).
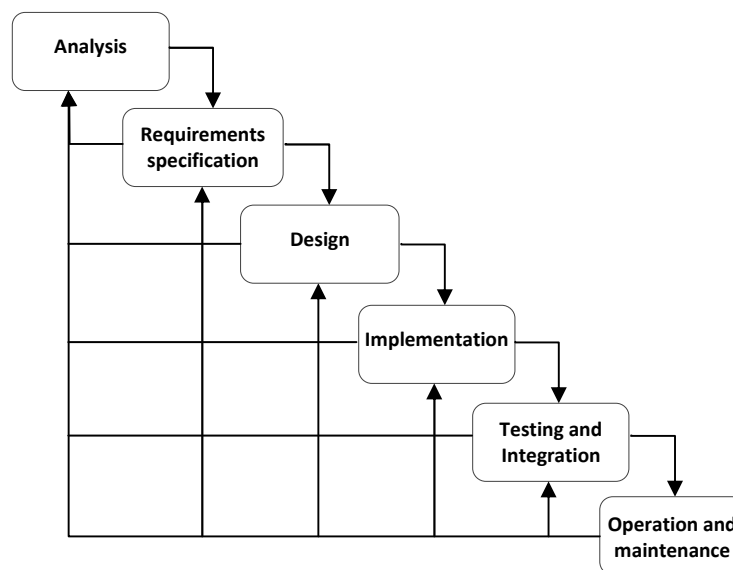
2.1.2 Project planning techniques

There are two different terms for this technique, one is called Program Evaluation and Review Technique (PERT) chart and the other one is called Work Breakdown Structure (WBS). These two correlates with each other and the concept is the same. The PERT chart shows the path in which order the tasks needs to be done, while the WBS shows a tree structure of how the tasks are related.

There are several tasks in each phase during the life cycle, and all these have different relations to each other (Avison & Fitzgerald 2003). Task 1, must be done before task 2 and task 2 and 3 must be done before starting on task 4 etc.

All these rules and relations are set up in the project planning phase but carried out and visualized in the WBS and PERT charts. After each phase all tasks need to be finished in order to continue with the next phase. There are exceptions for those tasks which are not connected to the critical path. The critical path in the PERT chart is the "shortest" way to the finished project, and there can be side tracks with tasks that have "slack". Slack means that you have a task which can start x amount of days after another task but still do not affect the time of the project (Avison & Fitzgerald, 2003; George et. al, 2004).

2.2 Documentation overview

"Documentation is the primary source of information about the system and the current state of development" (Green, 1996, p.122).

To document the work in progress is an important part of a system development project. The documentation should work as the most important communication tool between the different groups of the project (Green, 1996). The documentation should be seen as an ongoing process throughout the entire project, to avoid it being incomplete if it is conducted afterwards. Green (1996) argues that by overlooking documentation during the project process and doing most of the work afterwards, anything that can go wrong in the project probably will. Another reason for documentation in the first place is that the documentation offers a history view over the project and can be of assistance when needed to motivate why something was made as it was (Eriksson, 2007). When a project is finalized, the result should be easily reviewed as documentation of some kind (Green, 1996). Standardized systems often are delivered with a lot of documentation as a complement, but usually it is not sufficient and if it is not reflecting the actual reality and complements need to be added. Often these standardized systems, such as ERP systems, require some kind of adjustment to fit the specific organization. It is common that these changes are made; both to the system itself and to its environment and all of these changes must be documented. Compared to a non-standardized system however, the amount of documentation needed is less. Another important aspect is that the documentation as far as possible must be adapted to both its future users, and for the purpose which it is intended to be used (Green, 1996).

When dealing with large projects, a large number of different documents and forms to support and record the activities taking place within the project are produced by the differ-

ent members of the development team. The documentation can according to Green (1996) include management and quality assurance materials, various specifications, which primarily are the product of the analysis and design phase of the project, source code, technical guidelines and user manuals etc. All of these different kinds of documentation material which are related to system development projects can be divided into two types of documentation, system documentation and user documentation (Dennis & Wixom, 2003).

2.2.1 System documentation

System documentation contain information about the system's usage and functionality such as source code and should be used for support, maintenance and when new changes are made to the system. System documentation is the by-product of the system analysis and design phases and is created as the system development process proceeds (Dennis & Wixom, 2003). This type of documentation should be regularly updated to avoid the system to be depended on a few key persons. If the systems documentation is not held updated and some key persons for different reasons leave the project or the organization, problems may occur. The existing system documentation makes it possible for the work to continue even if these key persons have left the project or if the system development project is interrupted in some way (Burch, 1992). Accessibility is another important issue to consider when writing system documentation. According to Berg (1987), it is essential that there is a clear understanding regarding the routines for system documentation, of who have been granted access to the documentation for making changes or adding information etc.

2.2.2 User documentation

The aim with user documentation is to help the user operate the developed system. This kind of documentation can be user's manuals, training material, online help functions etc. It is common that user documentation is left until the very end of the project which can be a risky strategy (Dennis & Wixom, 2003). Creating user documentation can be very time-consuming, but there is still a big difference between creating good documentation compared to the creation of lower-quality documentation. No matter what quality the user documentation has, the time for creating it should always be a part of the project planning. The most common approach to user documentation is to start creating it after the specifications are complete. This decreases the chances of needed changes in the documentation because of changes in the system software. By starting this early in the system development process, there is time for testing and reviewing of the documentation before the acceptance tests starts.

Paper-based user documentation is important but online user documentation is becoming a more common way of presenting the documentation. The downside with online documentation is that it requires some computer-skills from the user compared to the paper-based documentation. But in the end, online documentation has many advantages where the most important ones are the possible help search index, it is fairly easy to update and the fact that it is much cheaper to distribute than the paper-based documentation (Dennis & Wixom, 2003).

2.2.3 Documentation standards

Organizations' awareness of the system development processes' iterative nature (see figure 2-3) and the importance of dealing with changes of the requirements under the development process have increased. This has led to that the Rational Unified Process or RUP and the Unified Modeling Language (UML) during the last couple of years has become a more

common phenomenon. RUP is a method for software development which can be easily adapted to specific projects, small or large. RUP is a production of Rational Software and is an iterative development process. The method includes guidelines for all the different phases during a system development project and should answer questions like "who, how, when and what?" Documentation is a central part of RUP which includes clear guidelines for the division of roles, specifically among the project members involved in documentation. The method also includes standard templates and checklists for all types of documentation (Strand, 2003).

UML is a modeling language constructed to model, specify, illustrate, document and develop the different parts that a complicated software system consists of. UML represents a collection of "best engineering practices" e.g. methods for modeling the construction of large and complex systems. UML does not only support the system architecture but also the development process itself which means that UML not only is supporting the system architects but also can be useful for the project leader and other members of the project. As mentioned before, a number of documents are produced during a system development project. These documents can be everything from test documentation and design specifications to user manuals. Depending on the culture in the project, these documents can be handled differently and UML's goal is to facilitate the information spread and simplify the communication between many of the project's stakeholders. UML and RUP make a good fit according to Strand (2003). The point with UML is to create models, simplified images of the desired future system, and these models are often used within RUP (Strand, 2003).

The system development method RUP does not contain the only documentation standard out there. There are others which have been produced by international organizations such as the International Organization for Standardization (ISO). In ISO 12207 there are standards specifically for documentation during system development. The definition of the purpose of documentation is included and is stated as to collect information of the product, which are intended to be developed, to both the customer and the system developer; to support and making the product applicable and possible to maintain; to define the process; to put across the communication between all involved actors and to offer a historic view to enable traceability.

2.2.3.1 Documentation quality

"The importance of good documentation cannot be overemphasized" (Green, 1996, p.122).

Sommerville (1992) claims that if the lacking quality of the documentation leads to that it is not being used, the time and cost put into writing it becomes useless. The quality of the documentation is according to Sommerville (1992) as important as the system's quality itself. To produce documentation of high quality, a commitment from both the people meant to produce the documentation and from the organization as a whole is required. It is required that there is some kind of documentation standard (see 2.2.3) which is followed and that the organization has a quality insurance process which works. Standardized documents are often much easier to read and understand and Berg (1987) argues that the agreement of which standard to use during the system development project should be stated in the planning phase of the project.

According to Mathiassen et. al. (1998) good documentation is defined as having clarity and elegance. Documentation that is clear is easy to read and to understand. To make the documentation more comprehensible, a standardized look, a consequent use of technical terms and a complex index are all requirements that need to be fulfilled. Elegant documentation refers to the degree in which the documentation contains information which is rele-

vant for the user of the documentation. Mathiassen et al. argues that writing summarized documentation with only relevant information is the only way to achieve documentation of high quality. In the IS0 12207, standards for the characteristics of high quality documentation are included and according to those, documentation should be clear, complete, verifiable, consistent, modifiable, detectable and presentable.

2.2.4 Documentation review and control

Documentation review is about reading a document in the search for errors and to improve the documentation quality. The review can be divided into two parts, where the first part is about finding errors and the second part is about finding certain parts of the documents which are in need of improvement to minimize the possible misunderstanding between developer and tester of the system. According to Eriksson (2004) and Sommerville (1992), it is important that the documentation is reviewed and controlled. Sommerville (1992) claims that the documentation should follow the same path as the system does which is being tested and approved before being delivered to the customer. The cost for conducting this kind of documentation control is pretty low according to Sommerville (1992) and Eriksson (2004). Eriksson (2004) even claims that a carefully conducted documentation review is as effective for finding errors as to ordinary test methods which are generally thought of as a method which leads to that a lot of errors are identified. Documentation review has many possible outcomes according to Eriksson (2004) which mentions an improvement of the team work as one of them. He argues that by letting a number of people from different functions in the system development project be a part of the documentation review , the common knowledge of the system is improved and the risk for misunderstandings between the different functions are decreased. He also mentions that the documentation reviews is a good way of involving new project team members and increase their understanding of the system. The actual process is described by Eriksson (2004) as an ongoing process which follows the different steps or phases in the so called waterfall model, which is commonly used in system development projects. The people involved in the documentation review process are entirely depended on what kind of documentation being controlled.

3 Method

The following chapter outlines the method of the subject with the purpose of describing the knowledge analysis based upon the research questions. Based on that analysis we aim to present a proper research approach. Throughout the chapter there is an argumentation around the chosen research method, and the methods chosen for the data collection are introduced. In this thesis a case study will be conducted, which will be explained in this chapter. Finally there is a discussion around the quality assessment of the thesis to clarify the factors which contribute and strengthen the result of the thesis.

3.1 Knowledge analysis

The most important aspect in relation to thesis work is to generate knowledge. To analyze which knowledge is to be developed is according to Goldkuhl (1998) important, mainly because the choice and implementation of the method of the thesis is facilitated by the generated knowledge. Different types of knowledge can be generated by different research studies. The purpose of the study gives a direction of what knowledge to be generated, and for the fulfillment of the purpose, a number of research questions need to be recognized. Through the research questions combined with the intention of the purpose, the intended knowledge of this thesis can be established. After all, the generated knowledge of the thesis should be able to answer the stated research questions.

According to Goldkuhl (1998) there are two ways of conducting an empirical research, exploratory and descriptive. An exploratory study is used when deeper knowledge of a specific field is needed. This kind of study is normally used when there is little information about the subject which is studied or if the previous research is inadequate (Goldkuhl, 1998). We have not found any literature within the subject which provides any specific guidelines for documenting during the system development process within ERP systems. This is the reason why we believe the exploratory study is appropriate for this thesis. To be able to fulfill the purpose of this thesis and because of the fact that our thesis is based upon a single company, deeper knowledge in the field is necessary.

By fulfilling the purpose of this thesis, knowledge of a normative nature is created. Creating that kind of knowledge is a practical approach and should give advice, an outline and propositions for how something should be carried out. Because of the fact that this thesis deals with changes in the documentation routines in the future, the knowledge should also be considered, what Goldkuhl (1998) named, predictive knowledge.

Categorical knowledge is a fundamental form of knowledge which all other forms of knowledge is depended upon. When the reality is studied it must be conceptualized, which is done by explaining and defining concepts (Goldkuhl, 1998). This is an essential issue when creating conditions for further knowledge development. The study in this thesis will be conducted in a manner in which the existing documentation routines of the company are presented. Examples of how to improve the routines are also given, something that will benefit other interested parties in similar situations. Concepts regarding system development projects and ERP systems are of significant importance and will be defined and explained. It is concepts which are important for the reader to understand and will therefore be conceptualized. To sum up, the knowledge which is to be generated through this thesis is of exploratory, normative, predictive and categorical nature.

3.2   Research approach

When deciding upon the research approach for the thesis, there is a need to describe the different approaches to a research or empirical study to make clear why a certain method of approaches is chosen. When discussing different approaches to research studies, the terms deduction and induction are usually mentioned. Deduction and induction are two ways of conducting a research study or two ways of reasoning when it comes to research, e.g. a deductive or inductive research approach.

The deductive approach originates from what is general to more of a specific context. The intention with this approach is to test already existing theories against the reality through an empirical study. When a particular theory is chosen which concerns a certain phenomenon, it is narrowed down into a more specific hypothesis that is easier to test. Then observations or empirical data are collected in order to address the chosen hypothesis. This ultimately leads us to be able to test the hypothesis with specific data and the original theory may be confirmed or not, and a conclusion about the phenomenon is made. The inductive approach or reasoning works the other way. It moves from specific observations or empirical data to a broader generalization and new theories. This approach starts out with specific empirical data collected from the reality, and from that a hypothesis is created, which is explored. From these particular facts from the reality, drawn from the empirical data and tested trough the hypothesis, new conclusions or theories are developed (Holme & Solvang, 1997; Lundahl & Skärvad, 1999; Trochim, 2001). The two approaches can therefore be seen as a cycle where the output of one approach servers as an input to the other (see figure 3-1).
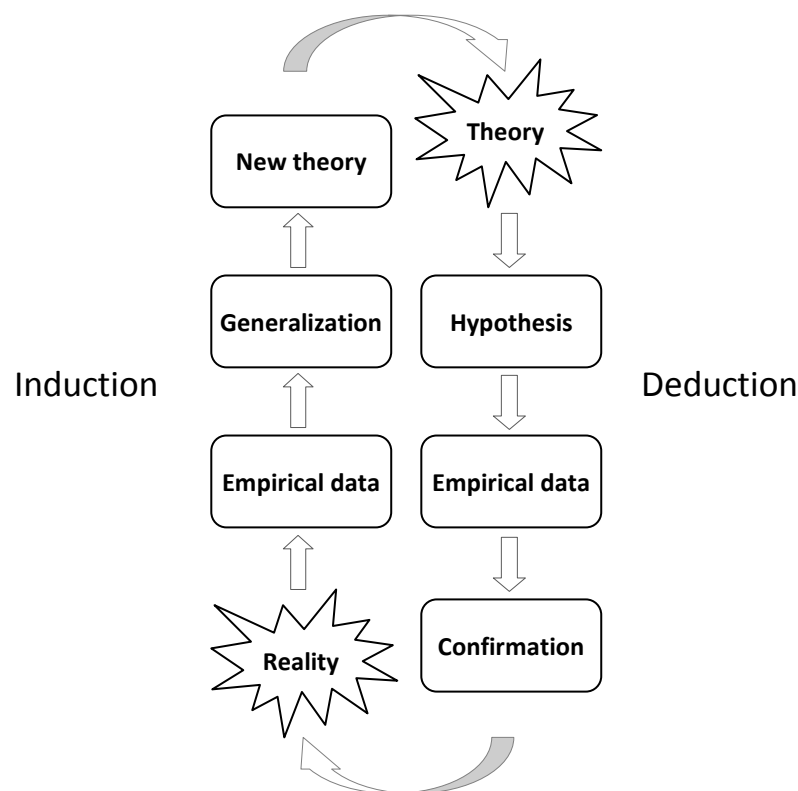

Figure 3-1 The inductive and deductive approach as a cycle (Seigerroth, 2007).

A literature study was performed before the empirical study, a procedure which according to Holme and Solvang (1997) is distinguished as a deductive approach. However, the research approach of this thesis should still be considered to consist of inductive reasoning taking into consideration the focus on the empiric findings. It is sometimes difficult to exactly state the approach of the research. According to Trochim (2001) most research involves both inductive and deductive reasoning at some time during a research study. Due to the purpose of this thesis the inductive approach is the most natural choice. Empirical data will be gathered about the specific company's own existing documentation routines for system development. Together with the theoretical framework, we will answer our research questions and new conclusions about conducting documentation will be drawn. The main reason why the deductive approach is not chosen for this thesis is the fact that existing literature and research of the specific subject are lacking, and therefore it would be irrational to choose such approach.

3.2.1 Quantitative Study

Quantitative study is the more structured way of conducting a study, and the result from this kind of study is structured data, numbers and statistics. As the name suggests this is a way of doing a mass research and get a lot of answers or results which you can put together and run statistics on. The down side of this sort of study is that you need many objects to study, and also have something to compare the study with (Holme & Solvang, 1997). In our case we have come to the conclusion that our study is somewhat more abstract and that a quantitative study would not be appropriate because we are not in need of any sort of statistical data at this time.

3.2.2 Qualitative Study

As Holme & Solvang (1997) states, qualitative study is given by reasoning, hypothesis testing and discussion. All this to get a deeper understanding of the problem at hand, and in this study we will follow this type of research. We found that the qualitative approach of conducting the empiric study would fit this study better than the quantitative. The reason is that we need a deeper understanding of the systems development phase and its relation with documentation methods and by interviewing primary sources we hope to attain that and at the same time retrieve more useful information than we would by only reading books or articles (Holme & Solvang, 1997). The qualitative approach is about characterizing the data rather than quantifying it, this is the approach we seek to use through out our research study.

3.3 Data Collection

In this section of the thesis we will cover some of the different data collection methods. According to Holme and Solvang (1997) there are four important things to keep in mind when gathering information; observation, source, interpretation, and usefulness which will be referred to as steps in this section. It is important to consider where the information can be found and how to choose from all the different sources. In the observation step we decided from which of all the different sources we wanted to gather information e.g. library, newspapers, articles, the internet, and interviewing. We read as many books, articles as we could find about the subject. We will use secondary sources which are the articles and books that we have read, but also primary sources such as conducted face-to-face interviews. It is important to look into the background of the persons that you interview so you know that they are reliable sources, and that they have expertise knowledge within the subject of matter. After reading books, articles or interviewing it is very important that you

stop and analyze the information retrieved. A critical perspective is needed and you have to ask yourself if the information is useful, accurate and reliable, and how to obtain generalizability.

3.3.1 Literature Study

Ejvegård (1996) states that it is important to precede the literature study with a literature search. Our literature search was based on articles, library databases and different search engines on the internet. The terms that were used in the search were similar to the following;

- ERP system
- Documentation
- Effects of documentation
- Documentation costs
- Documentation during development
- System development
- Software engineering
- SDLC
- System development projects
- System development process
- System development costs
- Maintenance
- Maintenance costs

During this literature search we were careful to take notes regarding interesting books, web-pages, articles, companies, and interesting people which were brought to our attention by the use of the previous stated terms. After the literature search, we started the literature study by reading the collected material to get an understanding of the problem area, and by using the books indexes and reading summaries we saved a lot of time not having to read whole books. This study was also conducted in order to collect data for the frame of reference.

3.3.2 Interviews

We will conduct face-to-face interviews in this study, and when using interviews in a study it is according to Lundahl and Skärvad (1997) important that we decide whether to choose the standardized or the non-standardized interview type. The standardized interviews always have pre-defined questions and also in what order the questions are going to be asked and answered. It is important that you use the same order and the same questions in all interviews. That is what makes it a standardized interview and it is always structured. The non-standardized interview is much more flexible in the sense that you are allowed to change the questions in between the different interviews and you may change the order of the questions. But also in this type of interview you can and should ask follow up questions. The non-standardized interview will give more complex results and will take longer type to analyze, but it will also give you as an interviewer more knowledge and more understanding. In this study we will use the non-standardized semi-structured interview type. We want to be able to ask follow-up questions about things that we come up with as the interview goes along. And we also think this is a somewhat complex subject for us so we need to ask simple questions just to understand the person we are talking to. We will bring an interview guide with us to the interview which will be our set of base questions. This interview guide will be explained in the following section. The structured interview can or might

be fixed questions with fixed answer alternatives, while the semi-structured interview is some fixed questions but also leaves room for questions to be brought up during the interview. The unstructured interview is most often an interview where there are no pre-set questions and the interview is very flexible. When conducting several interviews there might be a slight advantage to use either the structured or the semi structured approach due to the amount of time that will be saved in comparison to analyze the unstructured interview, which is very complex and take very long time (Lundahl & Skärvad, 1997; Holme & Solvang, 1997).

3.3.2.1 Interview guide

We chose to do semi structured interviews; therefore our interview guide (appendix 2) should be seen more as a guideline than the exact template for the interviews. We put together as many questions as possible affiliated with the subject of documentation routines, documentation models and methods, and documentation tools. We have divided the interview in sections to make it more comprehensible with its many questions in mind. We start the interviews with an introduction section where we ask questions to get some background information of the respondent and also make the respondent feel comfortable with answering our questions.

The sections which the interviewguide consist of are mainly based upon the initial discussion with the contact persons at Volvo IT mentioned in section 3.4.1.

The first section is called documentation characteristics and there we hope to get the respondents point of view on what is important and can be done to improve the documentation.

The second section is called globalization, in which hope to get some information about the linguistic and cultural importance of the documentation.

The third section we call documentation routines, here we go in to the actual "as is" in the company. We ask the respondent to tell us how the documentation is handled in the development process. We also try to find out if the company is using any form of standard development model or method in which they implement the documentation routines.

The fourth and last section is called key users, in which we aim to get some additional information about how the company integrates key users into the development process.

In the end, we are confident that the response to our interview guide will make it possible to answer the research question and by doing so, fulfill our purpose with this thesis.

3.3.2.2 Transcribing the qualitative data

Most experienced researchers recommend using a sound recorder when conducting qualitative and planned interviews. There are both advantages and disadvantages with interview recording. Starting with the many advantages; by using some kind of recording media we hope to be able to give the interview object the full focus it deserves. This will enable us not only to attain the verbal answer but also to study the respondent's body language when answering the questions. According to Repstad (1993), the verbal answer of a respondent is not always trustworthy and can be interpreted differently when taking the body language of the respondent into consideration. We cannot rely only on the recording to be sufficient protocol from the interview; we still have to take notes on e.g. body language, and interpretations of the answers. When analyzing and drawing conclusions from the interview it is important to use the exact words from the respondent which can be complicated if a sound recorder is not used, because of the difficulties taking notes during a long interview.

Disadvantages by recording interviews is according to Repstad (1993) that some people do not feel comfortable having everything they say recorded and this can make them insecure which in turn can result in answers that are not the complete truth or the whole truth. When conducting the interviews for this thesis we will use a recording device to make sure we get all the information. Since we are two people conducting the interview we will have a better chance of remembering and get all of the information noted.

The material drawn from the interviews will be presented in its whole (appendix 2.1-2.4), but because of the amount of data collected we decided to present the material as a summary in the empirical findings chapter. The summary consists of the empirical data which we believe are the most relevant in regard to the research questions of this thesis (1.2.1). Another reason for the empirical data to be presented as a summary is because of the fact that the respondents (3.4.2) requested to be anonymous. We believe that by keeping the respondents anonymous, more truthful and fruitful answers will be derived from the interviews, which can be a problem when using a sound recorder as mentioned earlier in this section. According to our opinion, it is not relevant to present each respondent for itself because of their anonymity, and the usage of quotes would not be possible.

3.4 Case Study

A case study is an empirical study that analyzes a present phenomenon in its own environment. The case study is more useful when the line between the phenomenon and the environment is unclear. The study is based on several different sources where data collections and analysis is guided by theoretical assumptions (Yin, 1994).

According to Yin (1994), a case study, is especially applicable when the purpose is to get a deeper understanding of the problem within a limited amount of time. Case studies are based on qualitative studies, which are often inductive and means that they focus on process, understanding, and interpretation rather than the deductive and experimental approaches (Merriam 1994). Using this method we hope to be able to emphasis illustrations and explanation within a limited context.

Yin (1993) has stated a set of example categories where he thinks a case study is more applicable. The stated problem of our thesis can be connected to many of the different categories that he stated in his research e.g. explorative, describing, understanding, valuating, qualitative, and studies to test already existing theories. Using a case study can and will help us capture more complex relationships which other methods might not. It will also help us get a deeper understanding and a more complete coverage of the subject.

ERP systems involve large and complex system development projects which makes it hard to comprehend. Therefore we choose to do a case study in order to get a deeper understanding of the system development projects, something that is essential for us to be able to fulfill the purpose.

3.4.1 Presentation of the company

The case study of this thesis will be conducted on Volvo Information Technology (Volvo IT) in Gothenburg, Sweden. Volvo IT is a fully-owned subsidiary of AB Volvo and provides solutions for many areas of the industrial process, and offers knowledge in the field of product lifecycle management, ERP system solutions and various IT operations. Volvo IT has many different clients for example the Volvo Group, Volvo Car Corporation which is owned by Ford, and other industrial companies.

Volvo IT was brought to one of the author's attention and contact was made. A first meeting was scheduled with Volvo IT during which an introduction to the problem were held. In fact, several problems in the organization were presented during this first visit and a discussion took place between the authors and the contact person in Volvo IT, to find an appropriate subject for the thesis work.

The subject we decided to focus upon was regarding the documentation during their ERP system development projects (1.2.2). This was a subject we believed was interesting and most appropriate in relation to our academic background.

3.4.2   Selection of respondents

The people at Volvo IT who are participating in the case study as respondents were chosen by our contact persons at Volvo IT. They were chosen of all of the many available respondents at the company for their specific experience of the subject in question. The respondents will be presented with a short description of their present role in the company and their previous experience in the area. We will not present the respondent with their names but will instead be referred to as a letter, with the beginning on the letter A. The letter itself does not have any connection to which order the interviews were made. The reason for us not presenting their names is that the respondents requested to be anonymous and so will the quotes in the summary of the empirical data (4.1). The respondents participating in the case study are the following;

- ***Respondent A*** - has worked a total of three years as a senior project manager within Global SAP. The respondent started his career as a programmer, and therefore can be regarded to have deep knowledge of system documentation.

- ***Respondent B*** - currently works in a business unit called user documentation and training within Global SAP. As the name of the unit infers, the respondent has knowledge about user type of documentation.

- ***Respondent C*** - is working as a global manager in the business unit architecture and design within Global SAP.

- ***Respondent D*** - works as a program manager at Volvo IT. The respondent has been working within the area of IT since the 1980's and has experience both from the hands-on work with development and with different projects. The respondent has been working as a developer in IT-projects and has also managed projects of various sizes. The respondents responsibility in the projects at Volvo IT concerns the projects deliverance in the right time, cost and with a sufficient scope.

- ***Respondent E*** - has worked at Volvo IT since 1999 and is positioned under Global SAP in the business unit architecture and integration center. The respondent is a so-called enterprise architect and has been so for approximately two years. Before that the respondent worked with access control and identity management. The respondent spends half of the time working as a maintenance manager for an administrative system called solution manager. Within this system documentation is handled and stored for implementation projects or system development projects. The other half of the time the respondent is working with other issues related to the system solution manager.

3.5   Analyzing the empirical findings

Because of the fact that an inductive approach is used in this thesis (3.2), an inductive perspective on the analysis is the most appropriate. According to Saunders, Lewis and Thornhill (2003) there are many different inductive strategies when analyzing the qualitative data drawn from the empirical gathering. One of them is the template analysis which will be used in this thesis. The term template analysis is referring to a specific way of organizing the collected empirical material into categories or themes (King, 1998). The data can be any kind of text e.g. text from interviews conducted over e-mail, open-ended question responses on a written questionnaire (King, 2004). But the most common textual data used in this kind of analysis is interview transcript as we have done in the analysis of this thesis. A template is a number of categories that represents some themes revealed from the empirical data that have been collected. The template analysis combines a deductive and an inductive approach in the sense that the categories are predetermined and then modified or added as the empirical data is collected.

The template analysis was used to identify key issues in the empirical data derived from the interviews. This analytical strategy is a good fit with this thesis because of the fact that it gives the opportunity for adjustments in the template if the need arises during the analysis work of the study.

3.6   Quality assessment of the thesis

No matter what method you use for the collection of data it is always important to criticize the chosen method by giving attention to the scarcities of the thesis (Bell, 2006). There is a difference between the type of study which is intended to be conducted, in the matter of validity and reliability. In qualitative studies, reliability is not of the same importance as in a quantitative study whereas low validity is a lesser problem when conducting qualitative studies, mainly because the study is conducted on a closer level (Holme & Solvang, 1997). At the same time can this closeness to the subject be a problem according to Holme and Solvang (1997), if the respondent feels that there are certain prior expectations on possible answers to the questions. Despite these differences, concepts like reliability, validity and generalizability will be discussed and explained here and we have the intention to follow these recommendations in order to make this thesis valid, reliable and generalizable.

3.6.1   Reliability

There are many definitions of reliability but they all point in the same direction, the data should be collected through the selected collection technique absent of random errors under similar circumstances (Lundahl & Skärvad, 1999; Bell, 2006). Reliability is a necessary precondition for validity (3.6.2) and to make a study reliable, nothing must be left to chance (Lundahl & Skärvad, 1999). There are several ways to achieve this; one is to use an interview guide which is accurate and well designed. Another way is the compilation of the empirical findings, which should be done in as short time period as possible after the conducted interview. A fast compilation increases the chances of a correct interpretation of the empiric material but more importantly is to give the respondents an opportunity to review the compilation afterwards to correct any mistakes made in it (Holme & Solvang, 1997). In order to get more precise answers in the interviews, it is essential to send the questions in advance, giving the respondents time to prepare for the interview.

The interview guide of our study was based upon an initial discussion with the contact persons at Volvo IT, and it was sent to the respondents a couple a days ahead. The empirical

material was compiled shortly after the interviews was finished to increase the chances of a correct interpretation of the material.

3.6.2 Validity

The concept of validity is very complicated but in general it refers to if the stated interview questions succeed in describing what they are meant to do describe (Bell, 2006). This means that if a question have the intention to lead to a specific empirical finding, it is considered valid if that specific empirical finding came out of that question. To achieve that, the right kind of question must be asked to a fitting respondent. If a question is not reliable, it cannot be valid but if the reliability of a study is high it does not mean that the study automatically is valid (Bell, 2006). This is the connection between the two concepts which means that in order to achieve validity, a well designed data collection technique is vital. To obtain a high validity it is also important that the asked questions reflect the purpose and stated research questions of the study. Furthermore it is necessary to explain the question for the respondent thoroughly if needed, to avoid a misinterpretation of the asked question (Lundahl & Skärvad, 1999).

As mentioned before, the interview guide of this study was based upon a discussion with the contact persons at Volvo IT, to make sure that the right kind of questions were asked. The respondents were selected by the contact persons at Volvo IT to ensure that the respondents had the right experience and competence to be able to answer the questions in the best way possible. Due to the fact that Volvo IT is the main interest party of this thesis as mentioned in section 1.4, we are confident that our questions will be answered by fitting respondents.

3.6.3 Generalizability

The term generalizability is about how the conducted study in the thesis reflects the reality (Lundahl & Skärvad, 1999). To achieve generalizability, theories or conclusions must be generated which are generalizable to a whole population. In this thesis we will conduct a case study and as mentioned in chapter 3.5, there can be a problem to generalize a study which is based on a single event or a single object. In a larger research project there can be a number of case studies where the similarities of the different cases can increase the generalizability of the study. When using only one case study as in this thesis, it is argued that it is hard not to get results that are case specific and that it is hard to generalize the study. As Denscombe (1998) quotes; "how generalizable the results from a case study are, is dependent on the similarities of the different cases" (Bell, 2006).

George et. al. (2004) argues that the development process of different systems are more similar than they are unlike, and therefore we can claim that this study will achieve generalizable results applicable for other ERP system related development projects as well. The result of this thesis may not work as suggestions of improvements for other companies because of the fact that we do not exactly know their need of improvement. These suggestions will instead work as important issues to take into consideration for other companies' future ERP system development projects of their own which makes the results of this thesis generalizable.

4 Empirical Findings

In this chapter the empirical data drawn from the interviews conducted during the case study at Volvo IT will be presented.

In the beginning of the empirical work we discussed how the breakdown of the empirical material should be conducted. We decided to present the four conducted interviews, with five respondents which were a part of the case study on Volvo IT, as a summary. The reason for this is that the respondents requested to be anonymous as mentioned in section 3.4.2, and by presenting the empirical data in a summary valuable quotes could be used without exposing the identity of the specific respondent. All respondents will be referred to as Volvo IT and not as a single respondent in the summary. Although, if a respondent express a different opinion this will be made clear.

4.1 Empirical summary

The summary is based on a categorization of the four major parts drawn from the interview guideline; documentation characteristics, globalization, documentation routines, and key users. The complete interviews can be found in appendix 2.1-2.4.

4.1.1 Documentation characteristics

Volvo IT is of the opinion that documentation is important in pretty much all projects, not only in projects of an IT-nature. Some of the produced documents can be responsible for invoices worth more than 1 billion SEK, which makes it essential to document to make as well-based decisions as possible. Another fact that was mention was that globalization and outsourcing has become a well used phenomenon, with project groups consisting of people from all over the world. These people come from countries with different routines and cultures, for example consultants from low cost countries, and this puts even higher demands on the documentation than before. To document is not only important, it is a requirement when it comes to maintenance of the system. The documentation is needed in order to support the developed solution.

"It is important to do it properly, carefully and precisely"

According to Volvo IT, the most important part of any project is to deliver the solution to the customer. But, at the same time documentation meant for future users of the system must be presented in order for the future user to be able to; use the system in the first place, use it in the way it was intended which makes it an important complement to the solution itself.

"To document is not high on the priority list"

If the documentation is not adequate or sufficient negative effects will appear later in the system development process and it will be impossible to maintain the solution. It will be hard to upgrade and make an improvement to the system if you do not know why the specific solution was selected or if you are not sure what the customer's needs are. The documentation makes it possible to compare the final documentation with the original documents. When the documentation is lacking many different factors is underlying. For example is documentation not a prioritized task in the projects and can sometimes be scheduled to the end of the project because of the fact that people involved in the projects often believe to have more important task to complete.

If Volvo IT has to choose between delivering the solution in time, or writing about the solution, the choice is easy to make. It is impossible to finish the solution in time if all documentation needs to be completed first. The reason for that problem is that the time it takes to document is not part of the presented project budget, which may be a result of the difficulty to measure the needed time to document in the right way. Even if you could, it would be difficult to get the project budget approved if you cannot prove the positive or negative effects of documentation. It is according to respondents though easier to see the effects or inadequate user documentation because of the direct feedback from the users.

"Difficult to motivate documentation as a part of the budget"

Time or money is however, not the only reasons behind insufficient documentation. Motivation is playing an important role, or the lack of it. It is generally considered to be more fun to work with the creation of the solution that it is to actually write it down. One of the respondents mentioned its experience from other companies which had clear and stated rules around documentation.

"There is not really a good excuse for not documenting properly"

Another factor is said to be the lack of availability, people who have important knowledge of the organization is often busy doing something else which makes it difficult to connect people with the right kind of knowledge to conduct the documentation. When there are people with critical knowledge missing from the project, it does not really matter how much available resources there is.

Many different opinions on how to individually improve the quality of the documentation was brought up. For example to change the jargon about the task itself or making a person who is responsible for specific part of a project also be responsible for documenting that part. Another idea was to making sure that it is always a sender and a receiver of the documents, who together should be responsible for the final version of that specific documentation. For them to be able to do that, the communication between the sender and the receiver must be improved. All in all, there must always be some kind of verification of the created documents from at least two people before a sign-off of the documents can be done.

4.1.2   Globalization

The formal language for global applications at Volvo IT is English where the project manager delivers an English version of the system. Sometimes there are certain needs for change of language but that change is up to the customer do perform.

"The project's deliverance is an English version of the system"

According to Volvo IT there is a certain risk involved in changing the language of the system. In projects which deals with many involved languages there is always a person dedicated to coordinate the translations. If there is a local translation of a system, that translation is based on a particular system version, version 1.0 for example. When the customer then wants to upgrade its system with its new system language, the system may already have gone through several updates since the translation which makes this local update very complicated. All and all, in order for the customer to actually run the system on another language than the original one, requires a very good motivation and most important, such translation is very expensive to conduct.

4.1.3   Documentation routines

"All development projects that are of some dignity use some kind of model or method"

The respondents all had the same opinion that if you work with projects of the size that Volvo IT does, a method and a model is vital. Volvo IT primarily makes use of two models, the IS-GDP (appendix 1.1) and the project control model or PCM (appendix 1.2). The IS-GDP is a "customer model" and is meant to be driven by the business processes and the business managers and not by the IT-solution. This model is also said to have a close commonality with the company's main process GDP. It should also provide guidance, support and some quality assurance in different information system projects by going through a series of gates. The PCM model is more thoroughly based on the fact that Volvo IT is the supplier for its different customers.

"Now we are trying to meet the customer methodology with ours"

The strategy Volvo IT has is to combine the customer model (IS-GDP) with the supplier model (PCM) which puts a demand on a higher level of involvement from the customer. The PCM model will be called the IS-GDP for IT, making the important connection between the models even more clear. In smaller projects Volvo IT uses the simple waterfall model, which lacks the iteration ability. But in all larger projects they use the OOSAD approach (2.1.1.2) with their IS-GDP model where iteration and integration of customers are two important attributes.

"This is something that is under development and they
want the involvement by the customer to increase even more."

In the PCM model there are documentation requirements between the phases. There is actually a built-in documentation routine with checklists (appendix 1.3, 1.4), and forms that have to be filled in and controlled by a delivery committee which controls that the required documents are in place and then lets the project move into the next phase. The respondents have a bit different opinions in the matter of actual control of the documentation. One says that the meetings lack of actual control while others say that the control is sufficient.

"The delivery meetings should be more of a control of the actual
quality of things rather than it is right now when they basically just
check the existence of all different documents."

"It is better to be involved too early, than too late."

The end user and maintenance team of the project are involved in an early stage of the project. This way the maintenance and the users can see what is happening and be well-grounded in the matter. This is not always the case though, in some projects neither the customer nor the maintenance has the chance to be involved in the project. It is in these projects that the documentation quality is even more important.

"The documentation method differs between the standardized
ERP-system and other non-standardized system."

When developing an SAP product, or SAP related system the models and methods for systems development does not always fit. To encounter this problem, Volvo IT has developed

a method called "Project Framework". By gathering best practices from different previous SAP projects they now have a working model which they use to develop SAP related systems with Accelerated SAP (ASAP). In the SAP related systems Volvo IT has developed a documentation method called End User Instructions (EUI). These are instructions that can be used for training, testing, and also day to day work. The EUI is a step-by-step walk-through of how to perform a so called transaction in SAP, together with a graphical description of what it should look like when you do it. This way Volvo IT has taken the SAP software testing to another level. Being able to have non-expert people testing the system just by following these EUI has improved the quality of both the documentation and the systems being developed.

"The problems in all project is time and priorities."

Something unanimous for all respondents is like we mentioned earlier, that they all think that documentation has a low priority. They all agree on that documentation is often something that is put together in the end of the project. This is why they want to improve the integration of customers in the projects, have more control of the documentation quality and also integrate the maintenance team earlier in the developing process.

"Documentation and EUI are of higher quality and easier to work
with if they have been worked by both developers and key users."

4.1.4 Key users

Volvo IT involves key users during the system development projects. The key users are sometimes people from the customer who learn the system during the development of the same. Most of the key users can be said to be positioned under a division called functional leads, but the key users are sometimes also project managers. Considering the user documentation, where the work instructions for different tasks in the system are written by the key users themselves which makes them very involved in the system development process.

5   Analysis

In this chapter the empirical findings derived from the qualitative interviews in the case study will be ana-
lyzed using the strategy template analysis.

The categories were decided upon through the template analysis (3.5) and were influenced by the theoretical framework (2.1 and 2.2). The predetermined categories were "documentation characteristics", "globalization", "documentation routines" and "key users". These categories are the main parts of the interview guideline, and were the result of a discussion with Volvo IT regarding appropriate focus areas of the empirical study. During the breakdown of the empirical material, these categories needed to be modified to "the perception of documentation" and "the integration of documentation routines" for making it fit better to the collected data derived from the respondents and to the content of the analysis. The reason for that is that the empirical data collected through the case study was not exactly what we expected. The answers we received on the "globalization" related questions, turned out to be more towards why documentation is important in ERP system development projects and the answers we received on the "key user" related questions, were pointing towards users' influence on the documentation routines in the development process.

The first category "the perception of documentation" refers to the respondents' attitude to and experience of documentation and is based upon the theoretical framework chapter documentation overview (2.2).

The second category "the integration of documentation routines" is about which models or frameworks Volvo IT uses today for managing their ERP system development projects and how documentation is integrated in these, and is based upon the theoretical framework chapter system development concepts (2.1) and documentation review and control (2.2.4).

5.1   The perception of documentation

According to Green (1996) there are several documents to be expected from a systems development project, such as quality assurance material, various specifications, source code, technical guidelines and user manuals etc. All these different kinds of documentation can be divided into two types; system documentation and user documentation (Dennis & Wixom, 2003). There are different factors that affect the documentation quality, one of them being the budget allocation. It is hard for a project manager to find a way to motivate why he needs a larger budget to cover the documentation costs. No budget for documenting often means that the documentation does not get that much of room in the project plan as it should. We have found numbers showing that up to 80 percent of the total cost of a system's development or implementation is connected to maintenance costs, which could be costs for having a person to find a solution on a specific problem which is hidden from the user of the system. This is a problem that could easily be solved with appropriate amount of high quality documentation at hand. Maintenance costs like these often occur after the project is finished and are registered as an IT cost in many organizations. We believe this to be a major problem for future improvements of a development projects. Due to that the negative effects of lacking quality of a project's documentation is never brought up to the project planners' attention, the documentation will not be recognized as an important part of a project's budget.

The problem is that it is hard for project managers to argue for increased funds for documentation expenses when there is no concrete evidence that documentation will reduce the future costs. Return on investment (ROI) of documentation would be of great interest for

project planners. We have not been able to find any material on this subject, and for a project manager to have the basic data to point out the need of documentation budget, to decrease the future maintainability costs is crucial.

George et. al. (2004) claim that all the resources and efforts put into documentation are a waste if it is not being used properly because of lacking quality. An increased globalization factor influencing many companies is another reason to why the quality of the documentation is so important. Volvo IT delivers their solutions to many other countries and customers. Although they deliver solely English versions of the system, the respondents told us that their internationalized business sets higher demands on the documentation quality.

Volvo IT has experienced weakly produced documentation more than once, and the reason for it is said to be the task itself which is not prioritized enough. There are many different factors contributing to that, according to the respondents. The motivation to perform the task is not simply there, the people who are part of the project believe that they have other important tasks to complete. This makes it difficult to connect the right kind of people to produce the documentation. We do not believe these reasons to be entirely true however. Some of the people may actually have other important tasks to complete, but the majority of those people are probably of the opinion that it is more exiting to work with the solution itself than it is to document on it and therefore seek an excuse not to perform that particular task. Of course there can be other reasons for people missing from the project, but if that is the case then the project is dependent on too few people with the right competence. The motivation problem can be related to the fact that many employees have the responsibility to complete a certain amount of tasks, but how and when they do it is up to them. This way of working makes it easier to overlook the documentation task. If the documentation is overlooked then the task is left to the very end of the project and according to the respondents it is a common occurrence. Sommerville (1992) and Eriksson (2004) both claim that documentation should follow the same development path as the system. This is seldom the case as mentioned before, when talking to the respondents we got the impression that they have very different experiences of how it actually works. Some said that the documentation is put together in the end of almost all projects while some said that the documentation is working very well and that the maintenance team is integrated in the process from the early stages of design.

To leave the documentation to the end is according to Green (1996) a serious mistake. He argues that in a project there are many risks involved, and by ignoring the documentation task the chances of a project failure increases or leads to a very expensive project. We think that this statement is a little bit exaggerated, the lacking quality of the produced documentation may increase the projects total cost, but it will not for certain lead to a project being considered a failure. But as Green (1996) argues, to avoid being in this kind of situation, documentation should be seen as an ongoing process throughout the entire project to avoid it being incomplete. As it is today, Volvo IT considers documentation to be this ongoing process, but it is not the reality in all of their ERP-system development projects.

If the documentation is left out to the very end of the project it is still possible to produce some kind of documentation, but not documentation of higher quality. If that is the case, the documentation does not fulfill one of the purposes of documentation, stated in the documentation standard ISO 12207 (2.2.3), which is to offer a historical view to enable traceability.

According to Volvo IT, a decision is enforced to either finish the solution, and delivering it in time, or taking the time to document on it. From an economic-point-of-view, without being aware of the consequences of lacking documentation quality, the choice is easy to

make. We cannot blame the company for overlooking the documentation task in a situation like this. The most important for Volvo IT is to make the customer satisfied, and by being late with the deliverance of the solution is not a good way to achieve that.

5.2   The integration of documentation routines

According to Avison & Fitzgerald (2003) the documentation work should be done parallel to the system development and with the same quality controls. The problem is that in reality the documentation is often slacking behind in the process. When the system finishes one phase, the documentation does not always follow. This makes it hard for the project managers to actually perform any control on the documents. One respondent told us about their delivery committee which controls the quality of the documents between the phases. But the respondent apprehends this as a mere control of what documents were made and which was not, rather than a quality control on the documents (see appendix 1.3-1.4 which are examples of checklists which are used between the different phases). Another respondent said that between the phases there is a "sign-off" between the person who has made the documentation and the person who is to use it. These are good tasks, but we believe that the control needs to be more of a milestone rather than something that is made if there is any time left. The control of the documentation has more importance in a project using the functional version of the waterfall method. Bennet et. al. (2002) compares the difficulty of returning to an earlier phase once it is complete, to the difficulty of swimming up a waterfall. Between the phases in Volvo IT´s IS-GDP model (appendix 1.1), which is Volvo IT´s version of a traditional life cycle model for IS development, there should be an interaction between the documentation creator, and the documentation receiver where they together complements, controls and views the documentation. Dividing the project into smaller parts will make it more manageable according to George et. al. (2004). Having the project divided into these smaller parts, will make it easier to control each activity within the project, including the documentation.

All of the respondents were of the opinion that it is fundamental to use some kind of model when dealing with the kind of complex projects that ERP system development projects are. George et. al. (2004) argues that it is important to use a model in disregard of the project size. The larger the project is the more risk is involved. Avison and Fitzgerald (2003) believe that in order to avoid these potential problems and weaknesses with the model, the ability to customize it is a necessity.

According to the respondents, Volvo IT has developed their own method for system development when dealing with the ERP system SAP and its related systems. The method is called Project framework and is a gathering of best practices from previous ERP system related projects. Volvo IT also uses a documentation method called End User Instructions (EUI) which is instructions for the end-user of the system. The main point with the EUI is the possibility for non-expert people to test the system, and have according to Volvo IT also improved the quality of the systems being developed as well as the documentation during the development. These methods can of course be seen as a step in the right direction. However, during the qualitative study on Volvo IT the respondents gave the impression that they did not have a complete method on how to produce high quality documentation. There were different opinions about to which extent the method was actually used during ERP system development projects. We believe that these obscurities around its utilization are an indication of either a badly-functioning method, or the lack of motivation to use it. The people involved in the ERP system development projects are probably aware of the existence of a preferred method of documentation. The problem is that they do not realize

its importance, when the lack of resources, time, and effort put into documentation, points in another direction.

When IT started to show its competitive advantage, the customers did not have that much knowledge of the possibilities within the area of IT. Today the customers have high expectations on the IT developers. ERP systems need to be more customizable, the integration of customers is a very important part of system development projects. Avison & Fitzgerald (2003) claim that involving the customer in the development process will decrease the risk of the project turning out to be something that the customer did not order. In the early era of IT this happened quite often due to misunderstandings between customers and developers. George et. al. (2004) argues that for example the lack of communication with the customer can be a costly experience. The communication with the customers is something that Volvo IT would like to improve. Having a representative from the customer will decrease the risk of missing the scope and increase the quality of the documentation. By involving the customer in the iterative process of developing a system, we believe that the possibilities of a successful ERP system development project will increase.

6 Conclusions

In the following chapter the conclusions derived from the analysis chapter are presented.

The purpose of this thesis is to examine the existing documentation routines at Volvo IT, in order to find suggestions of improvement for documentation routines in future system development projects at Volvo IT through a case study. For interested parties (1.4) the conclusions of this thesis will work as valuable issues to take into consideration for future ERP system development projects. The qualitative study, and the analysis work of the empirical findings, resulted in several ideas and suggestions which are outlined as follows:

- Although many people involved in the ERP system development projects are aware of the importance of documentation, it will not be an important part of the system development project budget until concrete evidence of its positive effect on maintenance cost is presented.

- In order to achieve high quality documentation, the attitude towards documenting as an individual work task needs to change and the priority of such a task must be higher in the system development projects.

- It is important that a control of the documentation quality is made between the different phases of the system development project, not only a control of its existence.

- The company must work with the supposition that documentation is strongly affecting the total cost of a system as a starting-point and from that create an appropriate method for documentation. The mentality should be *Why – How* (making clear why it is important before creating a method for documentation) instead of *How – Why* (creating a method for documentation before being completely aware of its impact on the project's cost).

- To improve the quality of both the final product and the documentation of any ERP system development project, it is important to involve the customer in the developing process. Customer involvement will help the people in the project to set the right level of abstraction on the documentation. This is a relatively new phenomenon in the IT business, but as systems change and software along with the documentation needs to be more custom made, this is a vital step.

7   Discussion

This chapter will summarize the thesis; discuss alternative approaches to the study, and present suggestions for future research.

7.1   Reflections

The empirical material of this thesis is drawn from a case study carried out on a company called Volvo IT. It started with a meeting with two people from Volvo IT who would be our contact persons throughout the writing of this thesis. Volvo IT and our contact persons had identified a problem which concerned the documentation during systems development. At first we wanted to do the study with SAP R/3 as the main system of investigating. When this did not work out as planned, we started to focus on ERP systems as a general subject. We had hopes of being able to see actual work done and to observe the different steps in an ERP systems development cycle take place. This did not happen due to that the consultants most often did not work in office, or the work was confidential.

When we realized that we were not going to see any actual work we had to change the scope and focus of the study and we realized that it is way too complicated to bring out any kind of "best practice" methods for systems development projects. Instead we decided to name it an investigation of what might be the cause of the problems, possible solutions for the problems, but even more we wanted to focus on the general beliefs and attitudes toward documentation. We think this worked very well and we conducted interviews with five employees at Volvo IT with various positions, job descriptions and areas of responsibilities. We believe that the change of scope makes the result of the study generalizable.

We are a bit disappointed with the result of the conducted interviews. We e-mailed the interview guideline a couple of days in advance to the respondents and we asked them to look through the questions and prepare for the interview. We came to the interviews with the belief that they would be prepared and with a professional approach, but the reality was that we spent as much time looking for emails and power-point slides as actual time asking and answering questions. Valuable time for effective interviewing was lost since the interviews were on a time constraint. The respondents had answers to many of the questions but they also shared information which was redundant to our scope. We ignored some of the information during the interviews, and tried to steer the respondent in the right direction.

During one of the interviews we had some electronic issue with our sound recorder. We learned that it is very important to control the gear previous to each interview. Instead we had to write down as much as we could possible remember from that interview. This of course can be seen as a lack of validity but if we were the least bit of insecure of the actual information we did not use it.

Concluding the empirical findings in a shorter summary was not an easy task. It took a very long time and we struggled hard to get the "red thread" through the summary, at the same time as we wanted to point out some very important quotes and information found in the material. In the end we are satisfied with the result of the empirical summary and after our change in scope and focus; the material is much more comprehensible.

Due to the "narrow" area of research in ERP systems development documentation routines, we found it very hard to conduct a comprehensive literature study. We have therefore based most of our conclusions and results on the empirical findings along with knowledge gathered throughout the writing of this report. We are pleased with the outcome of

our thesis and believe that with the help of theories, the empirical findings and some of our own thoughts have managed to present a number of important suggestions of issues that are important when working with documentation in ERP system development projects.

7.2   Future research

During the thesis work a number of suggestions for further research have derived. We have thought of suggestions that we believe would be a good complement to the conclusion of this thesis. First, we believe that in order for the companies to actually realize our suggested changes to the documentation routines depends on the ability to show the return on investment (ROI) on high quality documentation, which we believe is an interesting approach for future research in the area. It could be difficult to do so, but an alternative way could be to conduct a study from a project perspective where you compare two or more projects of the same size and which covers the same amount of estimated work time. Then you could investigate how the documentation is handled in these different projects, how much time and resources are put into it, and calculate the cost for the total effort. Thereafter you would compare this cost with the maintenance cost of the specific project to see the return of investment on documentation.

Second, a study could be conducted with the aim to investigate the attitude several companies have towards documentation as an important work task. An interesting approach to that kind of study could be to see what kind of reasons lies behind a particular attitude.

References

Avison, D. & Fitzgerald, G. (2003). *Information systems development - Methodologies Techniques and Tools* (3rd Edition). London: McGraw-Hill.

Avraham, S. (1999). *Enterprise Resource Planning (ERP): The Dynamics of Operations Management.* USA: Kluwer Academic Publishers.

Bell, J. (2006). *Introduktion till forskningsmetodik* (4th edition). Danmark: Narayana Press.

Bennet, S., McRobb, S., & Farmer, R. (2002). *Object-Oriented Systems Analysis and Design – Using UML* (Second edition), London: McGraw-Hill.

Berg, G. (1987). *Dokumentationens juridiska betydelse.* Stockholm: Institutet för rättsinformatik.

Buck-Emden, R. (2000). *The SAP R/3 system: an introduction to ERP and business software technology.* London: Pearson Education Limited.

Dennis, A. & Wixom, B. (2003). *Systems analysis and design* (2nd edition). New York: Wiley.

Eriksson, U. (2004). *Test och kvalitetssäkring av IT-system.* Lund: Studentlitteratur.

Eriksson, U. (2007). *Kravhantering för IT-system.* Lund: Studentlitteratur.

Fuller, M., Valacich, J.S. & George, .J.F. (2008). *Information Systems Project Management - A Process and Team Approach.* Prentice Hall: New Jersey.

George, J.F., Batra, D., Valacich, J.S. and Hoffer, J.A. (2004). *Object-oriented systems analysis and design.* New Jersey: Upper Sadle River.

Green, S. (1996). *Information systems design.* London: Thomson.

Holme, I.M., & Solvang, B.K. (1997). *Forskningsmetodik.* Lund: Studentlitteratur.

IRM - Information resources management. (2003, January). *The Department of Justice Systems Development Life Cycle Guidance Document.* Retrieved 2007-09-20, from http://www.usdoj.gov/jmd/irm/lifecycle/ch1.htm

Kaplan, S. (2002, March 15). *Now is the time to pull the plug on your legacy apps.* CIO Magazine. Retrieved 2007-09-20, from:
http://www.cio.com/article/30930/Why_Now_s_the_Time_to_Tear_Down_Old_Legacy_Apps_and_Rebuild

King, N. (1998). Template Analysis in: *Qualitative Methods and Analysis in Organizational Research*, London: Sage, p.118-134.

King, N (2004) *Template analysis - What is Template Analysis?* Retrieved 2008-01-03, from http://www.hud.ac.uk/hhs/research/template_analysis/whatis.htm

References

Lundahl, U., & Skärvad, P-H. (1999). *Utredningsmetodik för samhällsvetare och ekonomer* (3rd edition). Lund: Studentlitteratur.

Mathiassen, L., Munk-Madsen, A., Nielsen, P. A., & Stage, J. (1998). *Objektorienterad analys och design.* Lund: Studentlitteratur.

Merriam, S.B. (1994). *Fallstudien som forskningsmetod.* Lund: Studentlitteratur.

Nah, F., Lee-Shang L. J., & Kuang, J. (2001). Critical factors for successful implementation of enterprise systems, *Business Process Management Journal, 7* (3). p. 285-96.

SAP. (2007). *About SAP.* Retrieved 2007-09-20, from http://www.sap.com/about/index.epx

Saunders, M., Lewis, P. & Thornhill, A. (2003). *Research methods for business students* (3th edition). Harlow: Financial Times/Prentice Hall

Seigerroth, U. (2007). *Teaching materials for the course Knowledge development within the social sciences.* Retrieved 2007-10-01, from http://jibsnet.hj.se/documents/files/download/548121620/9127545582834985137/Semserie%20VT06%20anv%E4nda%20bilder%20sem%201.pdf

Sommerville, I. (1992). *Software Engineering* (4th edition). Harlow: Addison-Wesley.

Strand, L. (2003). *UML & RUP: att lyckas med 00-projekt.* Sundbyberg: Docendo.

Trochim, W. (2001). *The Research Methods Knowledge Base.* Cornell University: Atomic Dog Publishing.

Yin, R.K., (2003, Vol. 34). *Applications of case study research* (2nd edition). USA: Library of Congress Cataloging-in-Publication Data.

Appendix 1.1 Volvo IT - IS-GDP
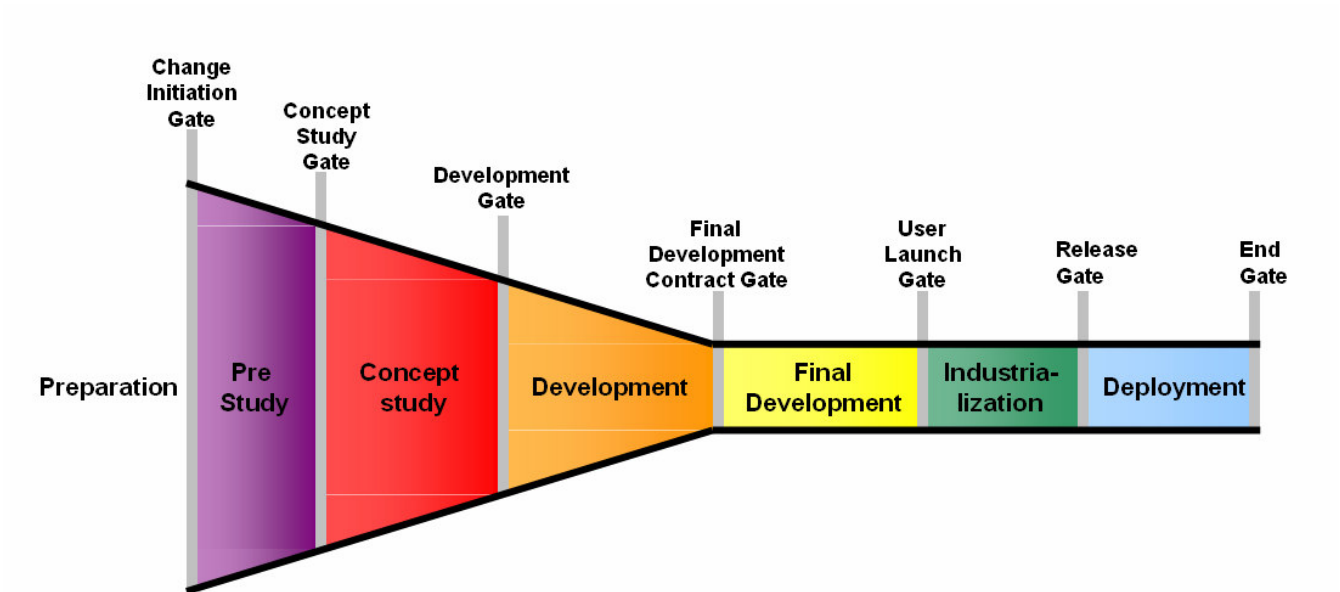
Demand side control model focused on "business matters".

Appendix 1.2 Volvo IT – PCM
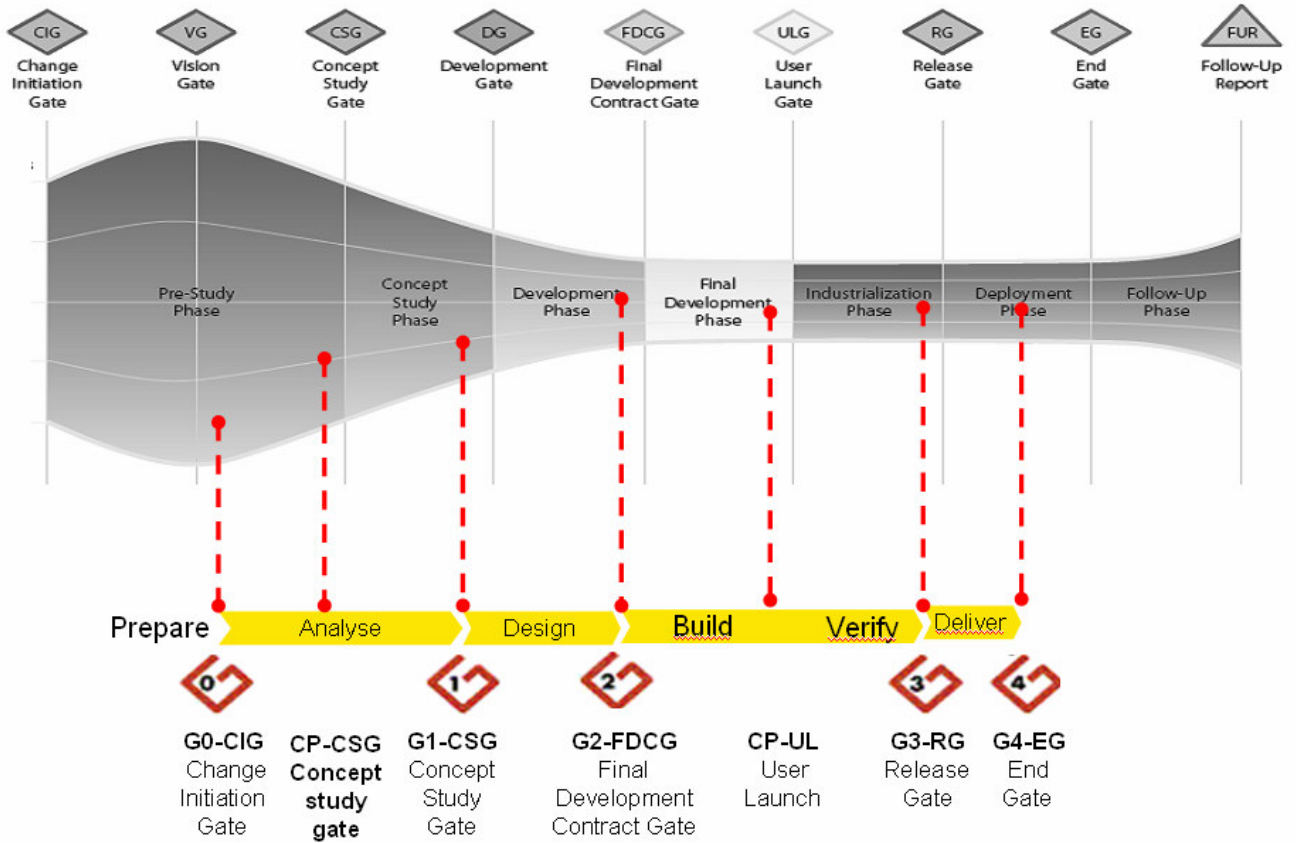
Supply side control model for "internal IT matters" in
relation to the IS-GDP model.

Appendix 1.3 Volvo IT – G3 Release gate checklist

	A	B	C	D	E	F	G	H
1	G3 - Release Gate Preparation Checklist Items	Project Sponsor	IT SC Chairman	IT Project Manager	Application Software Architect	Technical Architect	Maintenance Owner	Delivery Coordinator
2	I am committed to the contents of the assignment agreement.				N/A	N/A	N/A	
3	I am committed to the contents of the IT steering committee plan.				N/A	N/A	N/A	N/A
4	I am committed to the contents of the project charter.				N/A	N/A	N/A	N/A
5	I am committed to the risk management.	N/A					N/A	
6	I am committed to the communication management.	N/A			N/A	N/A	N/A	
7	Project key resources and resources for the next roadmap (G3 to G4) have been secured.	N/A		N/A	N/A	N/A	N/A	N/A
8	HR Agreements have been signed with project key resources and resources for the next roadmap (G3 to G4).	N/A	N/A		N/A	N/A	N/A	N/A
9	I agree with the calculated cost for G3 to G4 and the estimated cost for the entire project.				N/A	N/A	N/A	
10	I agree with the estimated cost for the operation, support and maintenance of the final result.				N/A			
11	The MG0 meeting has been held, and a maintenance and support organization has been established.	N/A			N/A			
12	The user documentation and the documentation for operations, maintenance & support have been completed.	N/A			N/A			
13	The end users have been trained according to the training plan.	N/A			N/A	N/A	N/A	N/A
14	I am committed to the roll-out (deployment) plan.	N/A						
15	The audit and review plan for the project has been updated, as needed.				N/A	N/A	N/A	N/A
16	The construction has been reviewed and any issues from the review have been addressed.	N/A						
17	The management of the project has been reviewed and any issues from the review have been addressed.	N/A			N/A	N/A	N/A	N/A
18	The Project Catalogue-Dashboard has been updated.	N/A			N/A	N/A	N/A	N/A
19	The produced and tested results address the customer requirements agreed upon.						N/A	N/A
20	The customer has approved the produced result for deployment.				N/A	N/A	N/A	N/A
21	The ISGDP Deliverables for the Release Gate have been produced.	N/A			N/A	N/A	N/A	

Appendix 1.4 Volvo IT – G4 End gate checklist

	A	B	C	D	E	F	G	H
	G4 - End Gate Preparation Checklist Items	Project Sponsor	IT SC Chairman	IT Project Manager	Application Software Architect	Technical Architect	Maintenance Owner	Delivery Coordinator
2	The customer has approved the delivery of the final result.				N/A	N/A	N/A	N/A
3	The MG1 meeting has been held and the final result has been handed over to the maintenance organization.	N/A			N/A	N/A		
4	All remaining issues have been taken care of in agreement with the maintenance organization.	N/A			N/A	N/A		
5	The user documentation and the documentation for operations, maintenance & support have been handed over to the maintenance organization.	N/A			N/A	N/A		
6	I am committed to the contents of the final report for the project.	N/A			N/A	N/A	N/A	
7	I agree with the summary of the actual cost for the project.				N/A	N/A	N/A	N/A
8	I agree with the calculated cost for the operation, maintenance and support of the final result handed over by the project.				N/A	N/A		
9	The hand-over has been reviewed and any issues from the review have been addressed.	N/A						
10	The management of the project has been reviewed and any issues from the review have been addressed.	N/A			N/A	N/A	N/A	N/A
11	An evaluation of customer satisfaction has been done and any needed corrective actions have been agreed.				N/A	N/A	N/A	N/A
12	The Project Catalogue-Dashboard has been updated.	N/A			N/A	N/A	N/A	N/A
13	The ISGDP Deliverables for the End Gate have been produced.	N/A			N/A	N/A	N/A	

Appendix 2 Interview guideline

Introduction

- Can you give a short description of your company?
- Which are your main responsibilities/tasks in the company?
- For how long have you worked for the company?
- During this time, did you work with the same tasks all along?

Documentation characteristics

- How important do you believe documentation to be in a system development project and why?
  - What are the consequences of inadequate documentation?
  - Which factors contributes to insufficient documentation?
- How would you define documentation of high quality?
- How important do you believe the quality of the documentation to be?
- How important is the language respectively the content in the documentation?
- What can be done on an individual level to improve the documentation?
  - How do you motivate the individual to improve the documentation?

Documentation routines

- Are you familiar with the term System Development Life Cycle (SDLC)?
  - Do you make use of SDCL in your system development project work?
  - Which phases of SDCL can you identify within your system development project?
  - Are you familiar with the with the object oriented/functional oriented approach?
    - Which one do you use in the organization?
  - Are you familiar with the Work Breakdown Structure (WBS)?
    - Is it applied in your project work?
    - If yes, in which phase is it created?
    - How is it followed up?
    - Do you have distinctive areas of responsibilities?
- Which documentation needs can you identify in regard to the phases in the SDLC?
- When and how is user documentation and system documentation differentiated in the SDLC?
- Is there a deliverance of documentation between the different phases of SDLC?
  - When and how is the deliverance taking place?
  - Which problems do you believe can derive from the deliverance of the documentation?
  - What is important to take into consideration concerning the deliverance?
- What do you believe is not working so well in your documentation routines?
  - Why do you think that is?
- In what way is the documentation saved?
  - Which media is used to store the documentation?
  - Who has access to the stored documentation?
- In what degree is a review or control of the documentation conducted?

- - Who perform this review or control?
  - If a review or control is not conducted, what will the consequences be?
- If changes are made in the system e.g. changes in the functionality or an addition to the units, how do you manage these changes in the documentation?
  - Who have the responsibility for managing the changes?
- Regarding documentation, does SAP differ from other ERP systems according to your own experience?

Key users

- Do you make use of key users in your system development projects?
  - Which advantages and disadvantages can you identify when using key users?
  - How are the key users involved in the system development process?
  - Which phases are owned by the key users?

Globalization

- Do you make use of outsourcing of your system development project in some way?
  - How would you describe this outsourcing?
- How is the documentation affected by the possible cultural and linguistic differences?
  - How do you solve these issues?

Appendix 2.1 Interview 1

According to the respondent, documentation is indeed very important. Especially when creating the requirement specification and in the design phase of the system development project. If it is not done properly, there will be negative effects drawn from that fact later in the system development process. When the customer is not satisfied with a solution, documentation of the solution is reviewed to see if the solution was made as agreed upon. If a solution was created as planned and the customer still is not satisfied, the requirements probably have been too wide and not precise enough.

The respondent also believes documentation being very important. In system development projects, the most important part of the project is to deliver the solution to the customer. But, in parallel with that, user material must also be presented in order to make sure that the future user will be able to use it in the right way, as intended, and is therefore an important complement to the solution itself. The most important thing when writing user documentation is to making clear what, when and who is going to do a specific task. This is not always the case according to the respondent. What Volvo IT has done is to connect the written instructions for specific activities to the processes in the customer's system, which has been a rather successful way of working.

The respondent believes that one factor that can explain insufficient documentation is availability. People who have important knowledge of the organization are often busy doing something else, so it is therefore difficult to get resources with the right kind of knowledge. The respondent claims that documentation has a lower priority, and that people involved in a project often think that they have other, more important tasks to complete. The respondent is sure that there is not a difference in importance between different tasks before a project starts, but when choosing between finishing the solution, and writing about the solution, the documentation is always the secondary option.

When confronting the respondent about the fact that most of the total cost of a system development project is related to activities taking place after the solution is delivered, the respondent is not surprised. The respondent explains that in reality, there is a deadline to consider, and to be able to deliver the solution in time, the documentation becomes lower prioritized. The respondent although says that it is a battle; you want to finish the documentation as well, but sometimes it is just impossible. The respondent does not think that time is the only factor behind this, but that there are many other factors as well. One of them is motivation, meaning that it is considered to be more fun to work with the creation of the solution than it is to write it down. Another aspect is according to the respondent the lack of resources. But the respondent does not agree that you could use the amount of money which is spent after the solution is delivered, on getting more resources. The respondent claims that it is not all about the money. Sometimes it is just difficult to replace one resource that has critical knowledge with another, no matter how much money you can spend.

The respondent argues that is a question of ownership. If someone in the project is responsible for a specific part of the project, this person should also be responsible for the related documentation. By having someone work solemnly with documentation, the person in question may not take that responsibility seriously, because the person knows that someone else is working on it.

Another factor is said to be that documentation is not people's main task, which is always prioritized first.

Regarding the actual quality of the documentation and how it should look like, the respondent says that it is always a sender and a receiver of the documentation. They should together in one way or another come up with the requirements for the documentation. The receiver will inform the sender what is needed, and will then also give his approval to the presented documentation. To keep track of this process Volvo IT has an index of all the documents which includes information about who has written it, who the receiver was, the status of the document etc. The respondent makes it clear that the there are different receiver depending on if its system or user documentation. A customer for example, cannot be expected to understand all of the system documentation.

A way of improving the documentation on a more individual basis could be to improve the communication between the sender and the receiver of the documentation according to the respondent. The respondent is more into that a verification of the material must be done. The material must be correspondent with the related process or system. By doing this a document sign-off can be done. Because of the delivery of a massive amount of documentation, there is a need for this kind of correspondence. This is not always done and can and should be improved according to the respondent. The reason for that is again a time issue.

The system development life cycle (SDLC) is something the respondent is familiar with, but refers SDLC to a method for dividing a project into different phases. This method is based upon the customers from a requirements point-of-view, and what the people at Volvo IT uses, the project control model (PCM). This model contains of a number of phases with different gates. The strategy Volvo IT has according to the respondent is to combine the customers' "model" with the PCM which puts a demand on a higher level of involvement from the customer. The respondent claims that this is something that is under development and that they want the involvement by the customer to increase even more.

When talking about work breakdown structure (WBS) the respondent recognize this as something Volvo IT calls MOM, which is dealing with invoices to the customers (the time spent by the employees at Volvo IT). Sometimes there is a separate WBS for every single activity, but to make it work practically, the respondent mention that every one reports to the same WBS during a specific period of time to avoid unnecessary administration.

The specific needs of documentation related to the different phases in the PCM, is something the respondent claims is linked to the checklists which must be completed at every gate. There is a need for the system documentation to be ready at a specific gate, but it is not a requirement because of the fact that the project stops at every gate anyway. The goal is for the system documentation to be completed in the end of the project but if it is not, the maintenance team still can support the system. But, the project cannot be "abandoned" before all of the system documentation is completed. The respondent claims that this goes for the user documentation as well. The user documentation must be completed before the so called release gate so the users can start working with the system. But, the respondent claims this to be important mainly because there will be less resources dealing with support for the users which would be the case with inadequate documentation.

In projects that are running out of time or are having other kinds of problems, user instructions/documentation can be delivered before the system is ready, or before the system even left the design phase. The reason for this is according to the respondent that it is a necessity in order to being able to finish with the user documentation in time. It is impos-

sible finish in time with the documentation if it is being written after the system is completed. To work so far ahead with the documentation is not according to the respondent an effective working method. It takes a lot of time to figure out in advance how the system is going to look like and function when ready. It also requires a lot of resources to document all the changes which was made to the system along the way. The best way of working according to the respondent would be to have a close collaboration with the system development team, in order to truly understand what is going on and to avoid being involved to late into the system development process. It is better to be involved too early, than too late.

The project manager cannot leave the project until the system documentation is complete according to the respondent. Neither could (should) a project Go Live with the system until the user documentation is complete, still it happens. There are a lot of criteria's which need to be fulfilled at a certain gate or checkpoint. If not all of them is complete at that certain time, a decision needs to be taken whether to continue with the project as planned or delay it with a couple of weeks with an increased total cost of the project with several millions of SKR. According to the respondent, that decision can be difficult to make and is depended on the persons affected by it. The respondent cannot leave the project until someone else in the customer's organization continues the respondent's work. The most important part of the completion of the system documentation is that the maintenance team then has the knowledge and competence enough to continue to maintain the system without the project manager and the other project members.

When receiving a document, a control of it takes place to see that the information is correct. That control also involves a technical control and a pedagogical control although the respondent admits that the pedagogical part of the control is not that developed yet. The user who in reality is going to make use of the documentation has not been involved in this control process, so the estimation of the final user's ability to understand the documentation is made by the actual writer. The consequences of documentation which is not controlled can be a risk that there will be an uncertainty of how things work etc. The respondent also mentions that technical aspects can be harder to solve in a good way.

The key users test the instructions written by them, and therefore minimize the chances of existing documentation with errors. If something is updated in the system, the documentation must follow or else the instruction will not work. This process is and must be an ongoing procedure according to the respondent.

If the project has been handed over to the maintenance team it is them who have to update the system documentation if there has been a change in the system. Regarding the user documentation, if it is not entirely complete, the respondent must stay for example.

It is up to each and everyone to provide them with updated documentation through an online source where the documentation is collected and updated. But the documentation is not regularly published on the online source but is updated in packs. It is practically impossible to sustain the online source with updated documentation all the time, but have to be collected into a single new release or version of the documentation.

The problems in all project is time and priorities. If a manager in the system development team was forced to work with the documentation part, it would have been prioritized in another way and would have let to a different quality of documentation according to the respondent. The documentation would have been up-to-date and more detailed. To work in a uniform manner is to prefer. When using InfoPak (which is a standard tool own by SAP) one can record different steps in the application, which is a way of writing documen-

tation. The problem is though that this way of working may not be what one want. This documentation is not conducted in a uniform manner really, with so many different people writing the same kind of documents. In the end, the documents do not resemble each other. But, the great amount of documentation which there is today may be because of the easy and fast documentation method this recorder tool provides. The respondent also claims that if there were a number of writers who manually did the documentation, the problem with consistency still exists and there is no guarantee that the documentation quality would be better that way either.

Another issue brought up by the respondent was the fact that just because all of the documentation is in place, does not mean that everyone actually make use of it. The respondent gave an example of a seminar where a new specification or a change in the system was presented. The problem with this kind of seminary was to actually get everyone to go there and listen. The proposal was then to document the entire presentation, with a voice as a complement to the presentation material itself. This proposal was not approved so it is now an idea of the past. According to the respondent, it would have been very useful for both the end users who practically then could use it as an online course, and the key users which could have used it to get valuable work instructions.

The documentation is stored centrally on a network unit, but in order to change in an InfoPak document InfoPak needs to be installed with special settings on the computer. It was however unclear which people who actually have access to the documentation according to the respondent. Some of the documentation is also stored and accessible for all project members on the so called team place. The respondent is sure that the documentation method differs between the standardized SAP and other non-standardized system.

Even if the system is standardized such as SAP is, the system must be adapted to the specific organization which makes it a unique application. There is not a lot of previous documentation for unique applications which puts a totally different demand on the documentation method. The question if SAP is like other system regarding the documentation method is if an auto recorder tool such as InfoPak is used. InfoPak is adapted to and own by SAP but works with other systems as well as far as the respondent knows. Key users are involved during the system development project, although one project can differ from another. Most of the key users are positioned under the division functional leads but some of the key users can be project managers. Most of the work instructions are written by the key users themselves which makes them more than involved in the process.

The language used at Volvo IT for global applications is English. The project manager delivers an English version of the system and after that it is up to the customer to change it to preferred language. There is a risk that the system is wrongly translated according to the respondent. In a project which deals with many different languages there is always someone who is responsible for coordinating the translations. If there is a local translation of a system, that translation is based on a particular version of the system. When the customer with the local translation of the system later wants to upgrade the system to the present version, the system may already have gone through several updates which can make the local update very difficult.

The respondent believes that a very good motivation is needed to run the system on another language that English, and most important of all, it is very expensive to do so.

Appendix 2.2 Interview 2

The interview started with a more broad discussion about documentation and the respondent said that documentation is very important in all projects, not only in IT projects. The world becoming more globalized and outsourcing is a well used phenomenon. Considering this the documentation is even more important when someone across the world is supposed to understand what you just made, or how it works. Today companies have a high focus on internationalization, and development of the company. This means that the companies have projects groups consisting of people from all over the world and this means different language different routines and cultures. Many large companies make use of consultants from low cost countries such as India or china, this, according to the respondent is the most important reason to why documentation is so important. We asked the respondent to give us an example of why and when documentation is not sufficient. The respondent admitted that documentation is not always as high on the priority list as it should be, and also that when planning a project it is way too often that the planning is scheduled to the end of the project when the is some time left. These factors, together with that Volvo IT does not have a clear method of how documentation should be done in the specific case is all contributing to that the documentation sometimes is not sufficient in the finished project.

The respondent have experiences from working for other companies where documentation is much more prioritized and where they have clear and stated rules for how documentation should be done and when. The respondent whishes to get this kind of mentality into his present company and so that the documentation becomes something that has a higher priority and is more accepted as a work task. To improve documentation quality, the respondent thinks that the jargon about documentation needs to change. One alternative would be to make the documentation required by the project members to be able to go to the next step.

We asked the respondent if Volvo IT use a development model when starting new development projects. The respondent answered that all development projects that are of some dignity uses some kind of model or method. Volvo IT has their own Project framework, they use ASAP (accelerated SAP) and a model called waterfall. The waterfall model is a step-by-step model where you finish one step and then moves on to the next step. This model integrates the customer in the first phases but not so much in the late phases where the actual developing takes place. The opposite of this model is the Scrum model. The scrum model is a more iterative model where there is interaction with the customer throughout the whole project. Volvo IT has some unique customizations to ASAP to make it fit their organization. According to the respondent, using the waterfall model will generate more documentation than it will in the Scrum model. Since the customer is part of the scrum model the customer make much of their own observations and documentations while in the waterfall when the customer get a finished product and the required documentation after it is done. The next thing that we wanted to discuss with the respondent was whether or not they used some kind of WBS (Work Breakdown Structure). In Volvo IT they use a WBS regardless what project method or model they use. There is always a WBS with clearly stated tasks for each phase or step in the model. Between the phases Volvo IT has something they call: "gate meetings", between every phase there are gates with some milestones that need to be done before proceeding to the next phase, and those tasks and milestones are controlled at these meetings. The members of this meeting is the senior projects manager and his managers for each part of the project along with a delivery commit-

tee who's task is to control the project team and what they have done. The respondent points to a flaw in these meetings, according to the respondent these meetings should be more of a control of the actual quality of things rather than it is right now when they basically just check the existence of all different documents and items.

During the projects on going the project team has their own team space where they keep all the documents. It is not until the project is finished that they move these documents to a document portal where the customer of the project has access to the documentation and a control is made of the documents. But if the documents is not good enough when reaching this state of the project it might be too late to make them good since most of the project people has moved on to their next project.

The respondent once again emphasizes that it is the planning which is important to get high quality documentation and also that more resources would help. But since it is very hard to make a ROI (Return On Investment) on more money to a project to be spent on documentation which is not a real tangible item of the project it is hard to get more money to a project for documentation purposes.

Volvo IT makes good use of key-users; these key users are people from the customer of the project who learns the system during the development of the same.

Appendix 2.3 Interview 3

Volvo IT has one system which they use to overview all projects and their people, cut-load, budgets, time reporting and much more. The respondent is using this system to stay in control of all information he gets from all the projects that he is responsible for.

The respondent tells us that the project the respondent is program manager for is a customer within the organization and that the respondent's team have to decide what hardware, software and staff to buy and hire. His project sends invoices worth approximately 1 billion SEK per year. He tells us this, to point out that the documentation is very important and is the underlying documents for these kinds of decisions.

When the respondent started working for Volvo IT he was involved in a project which purpose was to find a way to improve the documentation routines. They collected all existing documentations and looked over the routines. What they realized quite instantly was that it was hard to get an overview of all the existing documentation within a project. So what they did was to make a new model for how documentation should be saved and they also created a new documentation overview. So that a consultant could come to a project and relatively fast get a grip of what was going on just by reading one document.

When we started to talk about changing, updating and storing the documentation the respondent explained that it is only two groups of people that can and are allowed to change in the documentation; the maintenance group, and the project group.

The documentation can be changed when one of these four criteria's is fulfilled; either it is a change request, meaning that something is wrong and needs to be fixed. VINST-case, someone found a way where the system can be more effective and needs to be changed. The last two changing criteria is if e.g. SAP sends out an upgrade pack which needs to be installed to maintain the systems functionality, or the testing phase when someone finds something that might not be sufficient in the documentation or the system.

Volvo IT has one business unit called IT Governance; this unit decides what software to use when integrating different projects. It also takes the decisions about which rules to apply when testing the software and the documentation. The IT Governance unit is owned by both the customer and by Volvo IT. The respondent states the obvious when he says that the documentation would be better in most of their projects if they would just get more funds to work within the project budget. Though the respondent can see areas where the documentation could be better he is still very happy with what they achieve in terms of documentation. Following the IS-GDP which is Volvo's project model for projects in the business view and then applying the IS-GDP for IT model in the projects development point of view. The respondent is confident that the documentation is "ok" and that it isn't any crisis to improve it. When we asked the respondent whether or not they are using UML notification for documentation in the planning and analysis phase the respondent answered that they do not. But that is something that the respondent wants to change, developing a standard for the planning documentation using UML notification such as use cases, data flow diagrams, activity diagrams would be a good improvement.

Volvo IT has developed their own type of User documentation called End User Instructions (EUI). The EUI is used in different ways, booth Testing, and Training. The EUI is user documentation where you can first read what the specific function or transaction is suppose to do, a step by step instruction of how you do it and a picture of what it looks

like when you do it. This way anyone can help testing the system just by using these EUI. Before only a person who already knew the system could test it which wasn't a very good thing according to the respondent.

The EUI is written by key-users and systems developers together in the process of finishing a part of the system. The respondent says that there are still occasions when the developer does the whole documentation but they want to involve the key users more and more. It has also shown that documentation and EUI is of higher quality and easier to work with if they have been worked by booth developers and key users.

When a phase is finished the documentation and the system is tested in the acceptance phase, here the system is tested, the documentation is tested and reviewed, and the program manager, the system owner, and the key users sign off on that the deliverable is good.

There is a separate model for the handover work, which is done between the project team and the maintenance team. There are different requirements depending on in which phase the project is. There are different documentation templates and handover templates for this know what is important in which phase.

This work starts very early in the cycle of developing a new system, already in the starting phase of developing there are worked out plans of how the system is going to be handed over to maintenance. According to the respondent this is a step in the right direction and a good improvement of how the documentation is handled. Earlier the documentation was a low priority and put together in the end of the project and handed over to the maintenance without any checklist of what was present in the documentation and what was missing.

Now the project team is using a documentation handover checklist where they can see what is done what is not before they even set up a meeting with the receiver.

The respondent says that during the last two years of project work, there has not been one single push-back of a release date. This is, according to the respondent, thanks to their new way of working with the handover checklist, the way they integrate handover team, key users and the development of EUI. After the acceptance phase and if the documentations is sufficient it is important that the project team hands over all the documentation instantly. By doing this the project team can end their project and start on a new one. All documentation is sent over in one "package" with the specified characteristics which were decided early in the project cycle, in the meetings between the project team and the maintenance team. The system or project can "go-live" first when it has all of the documentation tested and approved. The respondent says that he is satisfied with what the team delivers to its customer. But of course there can always be improvements.

Appendix 2.4 Interview 4

The respondent believes that it is important with documentation in projects. It is important for being able to take care of the solution from a maintenance perspective. When dealing with a productive solution the main reason to document is to create the ability to support the solution.

The respondent thinks that lacking quality of the documentation makes it more difficult to support the solution and make changes. There is no possibility to go back to see the reason behind a specific choice of solution and you do not which original demands the customer had. According to the respondent it is important to have some kind of documentation so you can compare the solution with what the customer from the beginning ordered to see that everything matches.

The reason behind lacking documentation in the ERP-system development projects is according to respondent El because the time needed for documenting never is part of the project budget. There is a focus on functions, and the main goal is to deliver the solution in time. Another aspect is that the project budget is small so documentation is not prioritized enough to fit in. According to the respondent it is perhaps hard to estimate the time needed to document in the right way. It can also be difficult to motivate such a task in the project budget when you cannot prove the effects of the documentation and the business advantages are unknown.

It is a little bit easier according to the respondent to see the effects of bad user documentation compared to system documentation. Regarding the user documentation, feedback is received from the users when they are having problems using the system in the right way. Another problem is the absence of a documentation standard, a model which gives direction on how to perform this task.

According to the respondent, it differs between what kinds of system being developed. For non-standardized system which is developed from scratch by Volvo IT, RUP is used. But for one of our ERP-systems SAP, there is not a single standard within the organization at the moment. It depends on the project which standard is being used. According to the respondent, a lot of time has been spent on to decide what type of documentation that should be included and what kind of template to be used.

The standard closest to being a common standard, is not complete at the moment according to the respondent and does not cover all the phases of the system development projects.