Department of Informatics
School of Law and Economics
Göteborg University

Migration of an Administrative System to the Internet

- How to Improve the Current Situation.

Master of Science Thesis project
Spring 2001

Abstract

In this thesis paper we investigated a situation where two large corporations, Volvo Car Corporation and BOSCH, wished to renew the way they administer requests for changes in a software product. Our investigation covered what enhancements are needed to improve the administration, when migrating the system to the Internet. The purpose was to present a solution for a more ideal administration system. In the investigation we used interviews, observations and literature studies to get a clear view, more understanding of the situation and find solutions. We also used surveys and questionnaires to focus in on what an improvement could include. The results of the thesis were: 1. A model covering the activities in a future system, making the administration more effective, including an enhanced view of and access to information. 2. The important characteristics to concentrate on when implementing the system. These characteristics are functionality and maintainability.

Authors: Johan Andersson with Thomas Axelsson
Supervisor: Faramarz Agahi
Examiner: Kjell Engberg

Acknowledgments

We would like to thank Thanos Magoulas and Faramarz Agahi at the Department of Informatics at Göteborg University, for all their support and help throughout this thesis. We also would like to thank everyone at Volvo Car Corporation and BOSCH, especially our supervisors Andreas Nylander and Niklas Kvist.

Finally I, Thomas, would like to thank my girlfriend for being so wonderful!!

Johan Andersson & Thomas Axelsson

Table of Contents

1 Introduction

Along with the development of new technology innovations we change our way of doing things at the most fundamental levels. On an individual level, technological innovations have become a part of every aspect in our daily lives. Computers, networks, fax-machines, fibre-optics, automatic teller machines (ATM) has made us dependent on the access, services and flexibility that they provide us. They all play an important role in the way we work, have fun, do business and communicate.

From an organisational point of view both public and private institutions are experiencing an increase in the use of a range of different information technologies (ITs). It has become almost impossible for an organisation to function in a proper way without the use of several forms of ITs. People often consider IT as the medication for a variety of organisational problems like lack of performance or efficiency. Many large organizations today facilitate IT solutions such as electronic information sources like Lotus Notes, a database-driven groupware application, and Intranets (Welch, D.E. 1996). What is often forgotten is that new technologies could introduce their own problems and concerns into the organisational setting. Therefore it is important to perform a genuine investigation when introducing new ITs, or else there might be more troubles than enhancements.

In this thesis paper we are investigating such a situation where two large corporations, Volvo Car Corporation (VCC) and BOSCH, wishes to renew the interacting with each other regarding the administration of a software product used in different VCC engines. Our specific situation concerns the administration of requests for changes in the software, not the development of the software itself.

1.1 Background

Often when companies co-operate they are not physically close to each other, but in different cities, countries, or even on different continents. The distance makes it very important to have well functioning systems for the collaboration, or else friction, problems and misunderstandings might occur.

VCC in Sweden and BOSCH in Germany co-operate in many areas regarding different parts in many of VCC's car models. The VCC department "Implementation & System Verification" manages all software included in VCC's five and six cylinder engines. Software for the cars control unit is among the engine software that is produced by BOSCH. VCC orders, requests updates, tests and optimises the software.

It is the system for the administration of requests for changes of the control unit software, which is not functioning as well the involved parties would like it to do. The software coordinators working with this administration are of the opinion that something has to be done to improve the administration. There are no clear requirements from the software coordinators, or other people involved, as of what changes are needed. There is only a wish to improve administration system by replacing it with a web-based solution. This has brought up the need for this paper.

In this paper we will investigate what enhancements are needed to improve the administration, when migrating the system to the Internet.

1.2 Purpose and Problem Definition

The purpose of this master thesis is to present a solution for a more ideal administration system. The main question in the thesis is the following:

What is a more ideal web-based solution for replacing the current administrative system?

To be able to answer the main question, we have formulated two sub questions. The first sub question regards the issue of what possible conceptual changes there are to improve the current situation. The question we ask to answer this is:

What possible changes are there to improve the current situation?

By answering this question we will have an overview of what changes could be made. This gives us the information needed to make desirable conceptual improvements. To be able to find the improvements, we will later in our thesis follow a specific methodology suited for this kind of investigation.

The second sub question regards what important software quality characteristics the new system should include. The question we ask to answer this is:

What are the important software quality characteristics for the new administrative system?

We believe that most system software does not need all characteristics to fulfil all the quality characteristics. There are often irrelevant characteristics, which are unnecessary to consider. We will try to find the characteristics that are important to consider. When this is compiled, we will have an understanding of what characteristics a new system should have.

Together the two sub questions give a picture of the new system, both conceptually and characteristically. The conceptual part will show the changes of activities and the characteristics will show what qualities to concentrate on, when implementing a new web based system. The answers will give the information needed to answer the question regarding a more ideal system for substituting the old way of managing the situation.

1.3 Thesis Objectives

The main objective is to produce a kind of requirement specification that will meet both the scientific demands of a master thesis as well as generating a platform for VCC.

For students and other people with interest in the field of system development, this thesis will give an insight into a real world situation, and present a way of conducting the initial developing stages.

The solution we present to VCC will be a base to use if, or when, they decide to introduce a new administrative system.

We will ourselves gain knowledge within the thesis area, giving us valuable experience in the field of system development.

1.4 Delimitation

We have decided to focus our effort mainly to the VCC side of the system. There are several reasons for this. One reason is that the users on both sides, VCC and BOSCH, are using the system similarly, but there are users on the VCC side, which are primarily affected to possible changes. VCC is also the party wishing to investigate the situation. A final and important reason to the focus chosen is the time factor. There has been a limited amount of time at our disposal for this work.

In this paper we concentrate on the users needs. Traditionally there is an emphasis to meet all the technical requirements, leaving the users needs unanswered. This is something, which we would like to change.

Due to the narrow time span, the result of this thesis will not be a fully functional system, but merely an investigation regarding how to improve the current system. We will not take action; therefore any realisation of such a system will not be handled in this paper.

1.5 Disposition Outline

This thesis is divided into seven main sections. In the first section (**1 INTRODUCTION**), we give an introduction to the thesis. A background is given, the problem is stated and the purpose is presented. In the following method section (**2 METHOD)** there is a detailed description of the different procedures we use to collect data. Next section (**3 THEORETICAL FRAMEWORK**) covers the thesis' theoretical framework. Here we present frameworks for helping us find possible changes and characteristics. This will lead us to a description of the current system and explain how it works in section four (**4 THE CURRENT SITUATION**). Then our data results are compiled in the following section (**5 INTERVIEW AND DATA GATHERING**). We discuss the data results to give our own thoughts and answers to the given questions in section six (**6 A FUTURE SCENARIO: DATA ANALYSIS AND DISCUSSION**). In the final main section (**7 CONCLUSIONS**) we present our conclusions regarding what a more ideal web based solution for replacing the administrative system could be.

2 Method

According to Backman (1998), the purpose of the method section is to give a possibility for other people to replicate and evaluate our methods. It must be possible for a third part to replicate the different methods of procedure.

The method section can be divided into two parts, an investigation type how to collect data, and an investigation approach presenting the methods for the actual collection. We will give an overview over the investigation types available, then a presentation of the types we chose to use. In the next part of the chapter, we will give an overview over the different investigation approaches available, then a presentation of the approaches we chose to use. A deeper explanation then follows of how we used the methods as tools to collect data and information necessary to answer our set questions.

Finally we evaluate our methods and show the limitations that showed up during the application of the methods.

2.1 Type of Investigation

A premise for understanding what to study is to have a method for the investigation of what data there is to process. There are different types of investigations that decide how data is collected for processing.

Patel & Davidson (1994) mention three types of investigation commonly used. These can be classified based on how much is known about the problem before the investigation starts. The three main types of investigations they describe are:

- Explorative
- Descriptive
- Hypothesis Proving

2.1.1 Explorative Investigation

Explorative investigations are the type of investigation to use when there is an incomplete picture of which model to use and what characteristics and relations are important.

This type of investigation is characterised by its exploring nature. The investigations need to be elastic so that there is a possibility to adjust the work after the knowledge and results achieved. For this type of investigation different techniques to collect the information can be used. The objective of this type of investigation is to delimit a problem situation.

2.1.2 Descriptive Investigation

Descriptive investigations are conducted when the problem area includes a certain amount of knowledge that already might be a bit systemised. This kind of investigation is used to look into the relationships that already have taken place. The investigation is limited to only some of the aspects of the events. The goal is to get a thorough description of the events. For this type of investigation often interviews and surveys are being used and there is often a need for statistical methods. This type of investigation is the best when the problem it is used on is clearly defined.

2.1.3 Hypothesis Proving

This type of investigation is used where the amount of knowledge in the investigation area is extensive, and there are suitable theories developed. There is a need to find causes and effects behind the investigated.

Often it is solved with experiments and most often it starts from hypothesis about a connection between two or more variables. In this type of investigation there must be a clear structure on the problem and hypothesis and assumptions that one factor cause another.

2.1.4 Chosen Type of Investigation

In our investigation we mainly used one of the three types of investigation; the descriptive.

At the beginning of the investigation we used information that was collected from different sources to get a deeper understanding of the situation in order to get a perceptive on how to continue. The information was obtained through informal discussions and interviews, literature studies and observations.

After this part, an investigation was used with instruments such as questionnaires and surveys in order to find answers to some specific problems. For the surveys regarding software quality characteristics, statistical methods were used to get useful answers.

2.2 Investigation Approach

When you are about to conduct an investigation there are different methods to approach the investigation. What approach the investigation will use is depending on the investigators way to attack the problem and collect information. Backman (1998) and Patel & Davidson (1994) mention two different ways of approach the investigation: quantitatively and qualitatively.

2.2.1 Quantitative Methods

Quantitative methods are more structured than qualitative methods, and are used to systemise and explain situations from gathered information. Backman (1998) describes them as methods leading to numeric observations. Instruments commonly used for achieving this are written questionnaires, tests and surveys.

Quantitative methods are often accepted as methods giving reliable information, and are well suited for scientific research where data is finite and structured.

2.2.2 Qualitative Methods

Qualitative methods are informal and less structured than quantitative methods. They are more or less verbally based, and used to analyse situations to get an overview and understanding of the situation as a whole. Instruments for achieving this are informal interviews, discussions, observations and literature studies.

2.2.3 Chosen Investigation Approach

We used both qualitative and quantitative approaches in our investigation. In the explorative part we primarily used qualitative instruments, which is natural when a problem is hard to delimit and there is an incomplete picture of the situation. In the descriptive part we primarily used quantitative instruments, since the problem area included was already a bit systemised in the explorative part of the investigation.

2.2.4 Qualitative Instruments

The qualitative instruments we have used in the investigation are discussions, observations and literature studies. We have used this mix of method instruments to get a clear view and more understanding of the situation. By using more informal and unstructured ways to get information we also have more possibilities to discover not so obvious problems and information.

2.2.4.1 Discussions

We used unstructured discussions without predefined answers to let the person interviewed speak freely. This was made in the beginning of the investigation to get into what the system is about and to get a personal relationship with the persons interviewed. Trough these interviews we obtained useful information and insights not possible to achieve with more structured interviews. They were very helpful to understand the current situation, and the different activities occurring. We did a number of these unstructured discussions.

2.2.4.2 Observations

By using observations we got information about the system in under normal circumstances. Patel & Davidson (1994) describes the use of observations as very useful when there is a need to collect information from natural situations. T

The observations we did were to follow some of the users in their daily work, including their ordinary working tasks and attending at their various meetings at the department. We observed them when they performed their duties, without much interaction and discussion meanwhile. This was a helpful instrument to understand how the system flow works and how it is used in reality.

The main disadvantages with observations are that they often are costly and time consuming. In our investigation, where we knew little about the situation from the beginning, the observations had to be done to get enough information to understand the system, even if they took up much time.

2.2.4.3 Literature Studies

Through literature studies one find a background and an overview about what is known within a given problem area. The purpose with literature studies is usually to put together literature within a given area (Backman, 1998).

We have primarily used literature to get an introduction to, and understanding of, the theories and problem areas included in our investigation. Most of the literature has dealt with system methodology and the definition of software quality. Other literature concerns information regarding VCC, primarily company structure and figures.

2.2.5 Quantitative Instruments

The quantitative instruments we have used in our investigation are a survey and a questionnaire. It is a quantitative investigation approach, which uses structured instruments leading to fixed answers in the survey, and structured answers in the questionnaire.

2.2.5.1 Surveys

The surveys we used in the investigation were simple grading surveys. We wanted to find out which software quality characteristics were important and which were unimportant for this particular system and situation. We took inspiration how to perform the surveys and grading of characteristics, from Leung's article "Quality Metrics for Intranet Applications" (Leung, 2000).

The survey format was very suitable, because there is a finite amount of quality characteristics, and they are determined beforehand. One of the easiest and most used ways to carry out a survey is to use the "Likert scale" (Patel & Davidson, 1994). A Likert scale provides a way for respondents to quantify varying degrees of agreement or disagreement with a given statement. We used a variant of the Likert scale, which was graded from 1 to 5, where 1 is unimportant/not agree and 5 is very important/totally agree.

The surveys were handed out to all the different kinds of participants in the system. The software quality characteristics, which the survey covered, varied, depending on the type of knowledge the participant had. There were two types of participants, experts on software quality, and system users. The system users were given a survey covering user-accessible characteristics, and the experts were given a survey covering the other possible characteristics. This grouping was natural, due to the clear delimitation between the user-accessible and other more technical characteristics. Four system users and three experts did the surveys.

The results of the surveys gave us a picture of which characteristics are important, and which are unimportant.

Appendix B shows the survey forms used.

2.2.5.2 Questionnaires

Two types of questionnaires were used late in the investigation when we already had some information about the system, and thoughts about what an improvement could include.

One questionnaire was used to primarily help us to pinpoint and give more in-depth information about some of the most interesting areas. Such areas were, for example, people's personal wishes for changes and ideas regarding improvements. This questionnaire was given to the users at VCC and BOSCH. Complete user questions and answers can be found in Appendix A.

The other kind of questionnaire used, primarily helped us collect concrete information about how a future system could be built regarding software and hardware. We did not use the result vastly direct in our work, but it was very helpful to get a complete picture over all parts included.

This questionnaire was handed out to the experts. Complete expert questions and answers can be found in Appendix A.

The two questionnaires used were fairly structured and standardised with more or less the same questions to all people filling them out. The questions were openly formulated, giving us the questioned persons personal views. The system users' questionnaire was used on four different persons. The experts' questionnaire was used on three different persons.

2.3 Evaluation of Limitations

During the investigation we came across a number of different limitations in our work. These limitations restrict the investigation in various ways. Below follows a review of the restricting factors we found.

We have not been able to discuss improvements of the current system with the related senior supervisors. There has never been any possibility for such discussions, because of the low priority of this work on their schedule. There is a possibility that they have some important information regarding potential changes. In any event, after discussions with people working with the system, we have gained knowledge that the senior supervisors relation to the system is merely academic.

The observations of the users' daily work were done during a quite limited time period, making them less relevant. Even so, the time was enough to gather very useful intangible information in the beginning of our work.

A survey limitation is that in a system where there are a limited number of persons involved, like the system we are investigating, it might be difficult to point out the characteristics which are not relevant only from a low number of responses. This is because little statistical data often gives low accuracy. Another survey limitation is the subjectivity of each participant. There is always a possibility that the participants might interpret the questions differently.

When we conducted more formal interviews using a questionnaire, some of the participants were unable able to give answers to all questions due to lack of knowledge in the area.

It would also have been easier to get information about people thoughts regarding possible changes and wishes if more persons had answered the questionnaire. Once again, the amount of people available to us was the restricting factor.

3 Theoretical Framework

In this chapter we give a description of the theoretical framework we base our thesis upon. What we need is a framework, which allows us to get an understanding of the different elements involved in the thesis project.

We are interested in finding the answer to questions like, "is the way they work today really the best way of doing the job?", "should they do this kind of work at all?", "what are the needs in this situation?" and "what kind of system do they really want?". As we started to evaluate different system methodologies to use as a framework for achieving this, we came to the conclusion that a broader view of the situation would be to prefer.

In the first part of this chapter, there is a methodology section, which we use as framework to get the broad view we wish for when investigating what possible changes there are to improve. We need this to support and validate our course of action. By following a suitable methodology we will be able to include all necessary elements for shaping a picture of what can be do to change the current system to the better.

The second major part of this chapter deals with what potential frameworks there are to use when investigating what important quality characteristics there might be in the new administrative system. By finding the relevant quality characteristics for the software the system is composed of, we have a base upon which can formulate the requirements needed.

3.1 Introduction to Methodology

The following definition of methodology, by Avison & Fitzgerald, gives a good idea of what a methodology is.

"A methodology is a collection of procedures, techniques, tools and documentation aids.... But a methodology is more than merely a collection of these things. It is usually based on some philosophical view; otherwise it is merely a method. Like a recipe"

(Avison & Fitzgerald, 1995)

In short: Philosophy + Method = Methodology.

According the Checkland (1989), the traditional system methodologies are often described as using hard systems thinking. They start with the use of a carefully defined objective which is taken as a given. There is a clear problem stated, and you are using the methodology to find a solution to that specific problem. In 1981 Peter Checkland, Professor of Systems at the University of Lancaster, published "Soft Systems Methodology". This was the beginning of an alterative philosophy.

Where hard systems methodology typically addresses defined (quantified, quantifiable, measurable, engineered or otherwise specified) processes, soft systems methodology (SSM) almost always addresses people, subjective preferences, non-comparable data, and unknowable effects.

In this thesis, the “soft” parts of the system are very important to investigate, since we do not have clear picture of the system, or possible changes.

By using SSM as our methodology framework as a base for our analysis in the data analysis and discussion section (**6 A FUTURE SCENARIO: DATA ANALYSIS AND DISCUSSION**) of this paper, we will attempt to find what possible changes there are to improve the current situation.

3.2 Soft Systems Methodology (SSM)

In this section we will give a description of SSM, which will guide us through the work with this thesis. SSM is a problem solving philosophy. According to Peter Checkland (1989), the creator of the methodology, SSM is a learning system that “…articulates a process of inquiry, which leads to the action…”.

SSM gives an insight in both organisational learning and human activities in a system thinking way. Soft systems include human beings; hence the philosophy behind the methodology states that, in social systems, it is always contradictions in objectives. The learning is about non-structured complex human situations where the objectives are not clearly defined.

While in hard systems thinking where you start with problem solving, SSM helps you understand the existing problem in its context. Thereafter you move from one state that is experienced as problematic, to a state that is more satisfying than before. This is illustrated in figure 1 below. When the more satisfying state is reached, the cycle of learning and action is complete, and another cycle can begin in the challenge of a new and changed problem situation. This cycle continues until there is a new, problem free, solution.

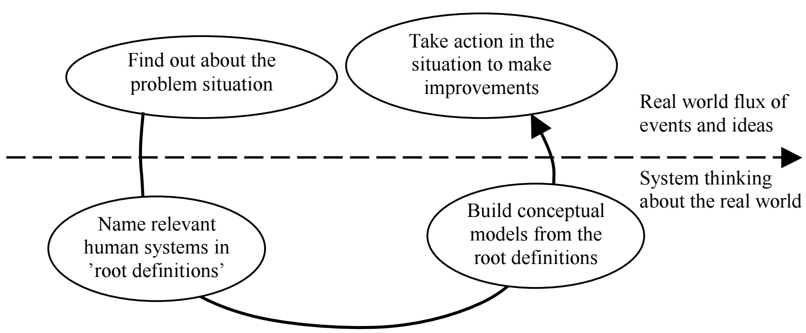


Figure 1 The basic structure of Soft Systems Methodology. (Checkland, 1989).

To perform this cycle, Checkland (1989) has divided it into seven smaller stages. He emphasises that it is important to understand that the linear progress from stage one to stage seven is not necessarily straight to users of SSM operating the learning life cycle. There can be flexibility among the different stages as long as the logical connection between them is kept in mind.

The seven stages of the learning life cycle will now be explained in detail.

3.2.1 The Learning Cycle

Figure 2 below shows all the seven stages in sequence. The different stages must not be followed mindlessly. The stages 1, 2 and 5-7 are real world activities and stages 3 and 4 are parts that handle the thinking about real world activities.

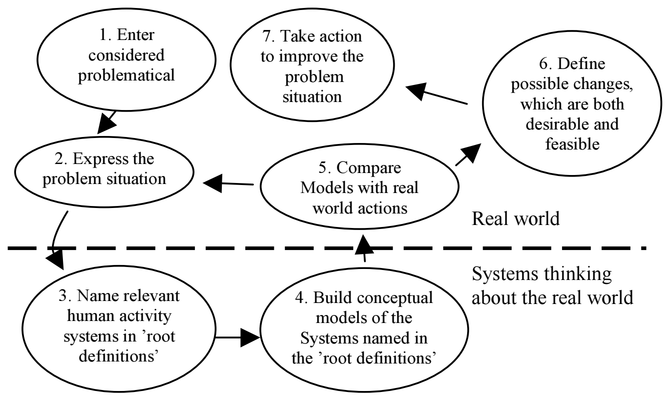


Figure 2 The advanced structure of Soft Systems Methodology. (Checkland, 1989).

3.2.2 The Seven Stages

The first two stages are concerned with finding out about the problem. One includes different views and looks at physical layout, reporting structures and formal and informal communication patterns. To express the problem situation, rich pictures can be used. They then work as an aid to discussion and communication around the problem situation.

3.2.2.1 Root Definitions

In stage three you formally start the system thinking by describing relevant human activity systems in the form of "Root Definitions". The systems described are those one thinks are relevant for a more extended investigation. A root definition is a description of what a system is and the purpose of it. The definition is meant to describe something relevant to the problem situation. It is not meant to be a model of the situation itself.

When making root definitions, it is important to take into account the "mnemonic" acronym CATWOE. The CATWOE assists when formulating root definitions, and lists the criteria that should be defined in the root definition. Below is an example explanation of CATWOE.

Customers (clients):	Includes everybody that in some way will be affected by the "purposeful activity", victims or beneficiaries.
Actors:	Everybody who does some form activity in the system.
Transformation process:	The purposeful activity articulated as Input ➔ T ➔ Output.
Weltanschauung:	Captures the different conceptions and views about the system and its purpose.
Owner:	Everyone who has the capability to put an end to or change the purposeful activity.
Environmental constraints:	Existing and planned limitations in the surrounding environment that affects the system.

Especially important is T, which defines the central transformation process, in which a defined input changes into a defined output. An example of a transformation process is a football match, where players (input) transforms to tired players (output) (Checkland, 1989).

The W in CATWOE stands for Weltanschauung, which is what makes the definition of the system meaningful. It is the interpretation of the purposeful activity given by the user. For example, a group fighting for a specific cause somewhere in the world might be seen, by some people, as freedom fighters and by others as terrorists.

It is important to take into account that there will be numerous possible ways to describe purposeful activities in the real world. The reason for using the German word Weltanschauung is, according to Checkland (1989), that it is the strongest word for describing the interpretation.

Below is a practical example of a root definition, built with CATWOE.

Situation: "An article and workflow system that enables us to handle the text electronically and manage the process. The text is marked going through the process. Management is the system owner as well as actors in the system along with coordinators, sales personnel, editors of text and clients."

C: Sales personnel, management, administrators, coordinators, editors and clients.
A: Sales personnel, administrators, coordinators, editors and clients.
T: Handle the text through the process, mark it, measure answer times and deliver the end product.
W: This is a "workflow system" for managing the process and electronical handling of the articles.
O: Management
E: Organisation - "That everybody does their job". IT-awareness internally. All clients don't have e-mail.

3.2.2.2 Building Conceptual Models

Conceptual models are used as a debating point to relate to real world activities. A conceptual model is drawn for each root definition. It is important that the model is a description of the root definition and not the real world activity. All verbs in the root definition are put together or structured following their "logical dependencies" (Checkland, 1989). One important aspect to have in mind is to use as few activities as possible in making the transformation process T work. Miller (referenced by Checkland, 1989) suggests that the human brain might only manage 7 ± 2 concepts at the same time.

Figure 3 beneath illustrates the common structure of a conceptual model.

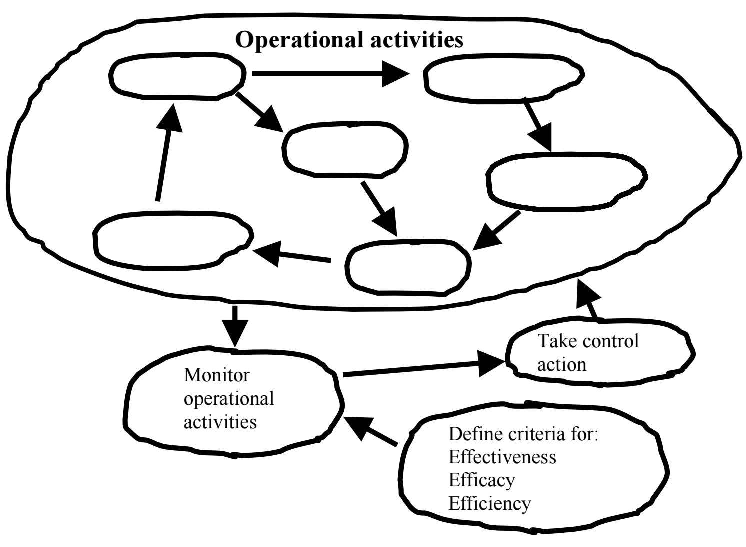


Figure 3 Common structure of a purposeful activity system. (Checkland, 1989).

To be able to monitor and control the final model of the system, some form of sub system needs to be defined. We need to ask how the system might fail. The answers can generally be categorised into three terms:

- *Effectiveness:* Are we doing the right thing?
- *Efficacy:* Do the selected means, expressed in the root definitions, really work?
- *Efficiency:* We might do the right thing and the system might work, but to what degree does the system use up resources. Is it economically defendable?

The reason for the monitoring sub system is to try to seek a "model, which is coherent and defensible rather than correct" (Checkland, 1989).

3.2.2.3 Comparing Conceptual Models With Reality

The conceptual models from the previous stage, shown in figure 3 above, are now taken into the real world again in stage five. The aim of the discussion is to compare the models with the real world settings and find out different changes that will improve the problem situation defined in stage two. If the root definitions and models that have emerged during step three and four do not fulfil their role as initiators of a problem solving discussion, there is a need to go back and work on step three and four again.

3.2.2.4 Assessing Feasible and Desirable Changes

The purpose of stage six is to draw up proposals, from the discussion in stage five, considering both desirable and feasible changes. They must be systematically desirable, for example in terms of effectiveness, appropriate resources or logical dependencies. Systematic and logical dependencies are not enough. It is also important that changes are culturally feasible, meaning keeping the human situation in mind. This is why the Weltanschauung is such an important ingredient in the root definitions in stage three. In this way, as Checkland puts it, "…cultural aspects cannot be completely ignored" (Checkland, 1989). The results of this discussion render a set of recommendations regarding change to help the problem situation.

3.2.2.5 Taking Action to Improve The Problem Situation

In stage seven it is time to take recommended action to help solving the problem situation. The cycle of SSM is concluded when the changes have been implemented. During the process of SSM, the initial problem situation has moved on and possibly been transformed into a problem situation more structured. SSM can again be used to address the new situation in another cycle of inquiry into the world.

3.3 The ISO Software Quality Model

It is becoming more and more important to measure the quality of a software product. To be able to improve any kind of software, it is uttermost important to know the required requirements. In this paper we concentrate on the users requirements needs. To help us find the quality characteristics needed, we will use a model, which defines software quality.

Traditionally there is an emphasis to meet all the technical requirements, leaving the users needs unanswered. This is something, which we would like to change. There has been several standards developed, that address users needs. One of the most accepted of these standards is the ISO 9126 international standard (ISO, 1991). This standard includes a list, and explanation, of software quality characteristics.

By using the ISO model as our quality characteristics framework in the data analysis and discussion section (**6 A FUTURE SCENARIO: DATA ANALYSIS AND DISCUSSION**) of this paper, we will have a base for finding the relevant characteristics for this kind of situation.

3.3.1 Description of the ISO 9126 Model

The ISO 9126 is the result of the joint committee of the International Organisation for Standardisation (ISO) and the International Electrotechnical Commission (IEC). It is published with the title "Information technology - Software product evaluation - Quality characteristics and guidelines for their use". According to ISO/IEC JTC1 (1997), the main content of this standard is the representation of quality of software as seen by software users.

The ISO 9126 standard covers the definition of software quality characteristics. The objective of this standard is only to provide a framework for the evaluation of software quality, not provide requirements for software. It defines a quality model, which can be applied to all software in all kinds of systems (ESSI-SCOPE, 1997).

This standard includes a model over quality characteristics needed to create a quality software product. By finding all the characteristics we have a base upon which we can formulate the requirements needed. The different characteristics of software quality are divided into six main characteristics. Below is figure 4 (ESSI-SCOPE, 1997) showing an overview of the ISO 9126 model.

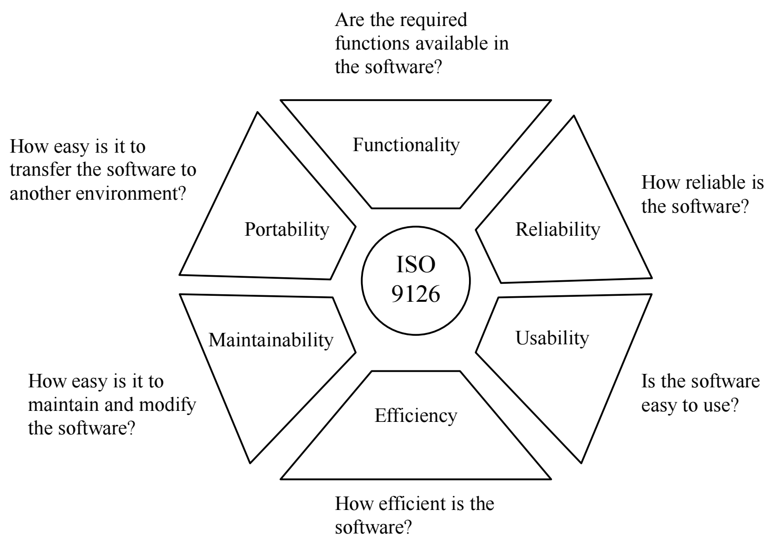


Figure 4 An overview of the ISO 9126 model. (ESSI-SCOPE, 1997).

The characteristics are all very broad and are therefore divided into more specific sub-characteristics. In the original ISO 9126 model, found in ISO (1991), there are twenty-one sub-characteristics.

3.3.2 The Extended ISO Model

A number of organisations extended the ISO 9126 characteristics further within the QUINT (Quality in Information Technology) project of the SERC institute (referenced by Hendricks, 2000). In the extended ISO 9126 model an additional eleven sub-characteristics has been added, making the model more user oriented.

We have two reasons for choosing using the extended ISO model as our reference. First of all, it is, according to E.P.W.M, Van Veenendaal (referenced by Leung, 2000), widely used and accepted by the software industry. The other reason is that is oriented towards the users quality needs. Below is figure 5 showing an illustration of the extended ISO 9126 model (Hendricks, 2000). Extended characteristics are showed in italics.

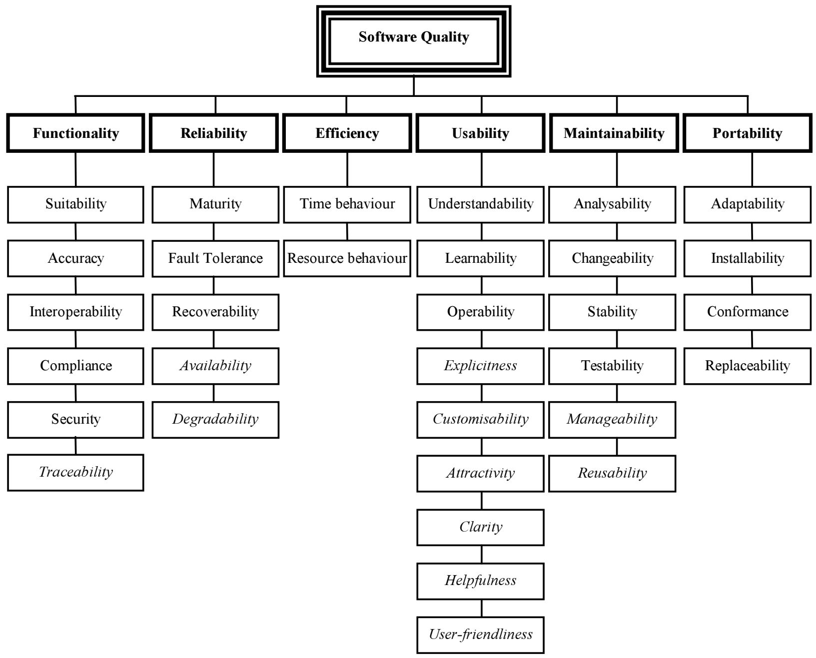


Figure 5 Illustration of the extended ISO 9126 model. (Hendricks, 2000).

3.3.3 Software Quality Characteristics

The following part is a breakdown of all the original characteristics and extended characteristics. Original characteristics and their sub-characteristics are according to ISO/IEC JTC1 (1997). The extended characteristics are according to Veldhuiozen (1999), and showed in italics.

3.3.3.1 Functionality

Functionality is the capability of the software to provide functions that meet stated and implied needs when the software is used under specified conditions. The sub-characteristics of functionality are:

- Suitability: The capability of the software to provide an appropriate set of functions for specified tasks and user objectives.
- Accuracy: The capability of the software to provide the right or agreed results or effects.
- Interoperability: The capability of the software to interact with one or more specified systems.
- Security: The capability of the software to protect information and data so that unauthorised persons or systems cannot read or modify them and authorised persons or systems are not denied access to them.

- Compliance: The capability of the software to adhere to application related standards, conventions or regulations in laws and similar prescriptions.
- *Traceability:* Attributes of software that bear on the effort needed to verify correctness of data processing on required points.

3.3.3.2 Reliability

Reliability is the capability of the software to maintain its level of performance when used under specified conditions. The sub-characteristics of reliability are:

- Maturity: The capability of the software to avoid failure as a result of faults in the software.
- Fault Tolerance: The capability of the software to maintain a specified level of performance in cases of software faults or of infringement of its specified interface.
- Recoverability: The capability of the software to re-establish its level of performance and recover the data directly affected in the case of a failure.
- *Availability:* The capability of the software to be in a state to perform a required function at a given point in time, under stated conditions of use.
- *Degradability:* The capability of the software to remain functional even when interacting with older versions of other software.

3.3.3.3 Usability

Usability is the capability of the software to be understood, learned, used and liked by the user, when used under specified conditions. The sub-characteristics of usability are:

- Understandability: The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.
- Learnability: The capability of the software product to enable the user to learn its application.
- Operability: The capability of the software product to enable the user to operate and control it.
- *Explicitness:* The capability of the software to state it's meaning to the user. Leaving no question as to meaning or intent.
- *Customisability:* The capability of the software to be adjusted to the demands of the user.
- *Attractivity:* The capability of the software product to be liked by the user.
- *Clarity:* It should be straightforward what the software is used for.
- *Helpfulness:* The capability of the software to support the user in the use of the software.
- *User-friendliness:* Attributes of software that bear on the users' satisfaction.

3.3.3.4 Efficiency

Efficiency is the capability of the software to provide the required performance, relative to the amount of resources used, under stated conditions. The sub-characteristics of efficiency are:

- Time behaviour: The capability of the software to provide appropriate response and processing times and throughput rates when performing its function, under stated conditions.
- Resource behaviour: The capability of the software to use appropriate resources in an appropriate time when the software performs its function under stated conditions.

3.3.3.5 Maintainability

Maintainability is the capability of the software to be modified. Modifications may include corrections, improvements or adaptation of the software to changes in environment, and in requirements and functional specifications. The sub-characteristics of maintainability are:

- Analysability: The capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified or to be identified. At the realisation phase adequate use of assertions, debug statements, logging and test methods is recommended.
- Changeability: The capability of the software product to enable a specified modification to be implemented. During the development phase a modular approach should be taken in order to minimise internal dependencies between object classes.
- Stability: The capability of the software to minimise unexpected effects from modifications of the software. See Changeability.
- Testability: The capability of the software product to enable modified software to be validated.
- *Manageability:* the structure of the software should be simple and straightforward in order to keep the software manageable.
- *Reusability:* Attributes of software that bear on its potential for complete or partial re-use in another software product.

3.3.3.6 Portability

Portability is the capability of software to be transferred from one environment to another. The sub-characteristics of portability are:

- Adaptability: The capability of the software to be modified for different specified environments without applying actions or means other than those provided for this purpose for the software considered.
- Installability: The capability of the software to be installed in a specified environment.
- Conformance: The ability of the software to act or behave in correspondence with current customs, rules, or styles.
- Replaceability: The capability of the software to be used in place of other specified software in the environment of that software.

4 The Current Situation

An important part of our work is to get an understanding of the current situation we are investigating. This is very helpful when working with SSM, to obtain a clear picture of the system, and understand what kind of situation we are investigating. We include different views and look at the physical layout, reporting structures and communication patterns. This is used as an aid to the discussion around the problem situation in the next chapter. This chapter can be viewed as stage one in the SSM learning life cycle mentioned in the previous chapter.

We begin by giving a general description over VCC, where we give an introduction to the company and show relevant division and department structure. After the general description we look into the current situation and explain how the system is working and who are involved. A brief description of BOSCH is as well included.

4.1 General Description of Volvo Car Corporation (VCC)

According to the Volvo Car Corporation Intranet web pages (VCCI, 2001), VCC develops, constructs, manufactures and markets cars. Since 1999, VCC is a part of the Ford Motor Company. Together with Aston Martin, Jaguar, Land Rover and Lincoln, VCC form the Premium Automotive Group, the most prestigious part of Ford.

Research and development is primarily carried out in Sweden. The major production units are situated in Born (the Netherlands), Gent (Belgium) and Torslanda (Sweden). The other existing production units are located in Sweden, Malaysia, the Philippines, South Africa and Thailand. VCC has approximately 27000 employees, whereof about 20000 in Sweden.

In 2000 VCC produced a total of 429600 cars, the most produced car model being the S/V40. The largest markets are Europe and North America.

4.2 Company Structure

VCC is divided into a number of different parts. These are:

- Human Resources
- Public Affairs
- Car Manufacturing
- Research, Development and Purchasing
- Project Management
- Marketing, Sales and Services
- Economy and Finance
- Quality

We are looking into the administration of the software for some of the car models control unit. This unit is a part of the engine.

The part, which is involved with all engine development, including the engine software, is named Research, Development and Purchasing. This development is carried out in the Chassi & Powertrain Engineering department, or more exactly in the Engine unit, which resides within Chassi & Powertrain Engineering.

The Engine Managements Systems (EMS) unit sorting under the Engine department is responsible for the management of the engine systems. The part of the EMS, which handles the engine management systems for the engines that use the control unit software in question, is EMS Europe (EMS-E).

4.2.1 EMS Europe Department Structure

EMS-E is responsible for the BOSCH engine management systems in VCCs five and six cylinder engines. In EMS-E we find several different departments. The majority of the relevant requests for changes in the software are produced here, but there is also change requests produced in other departments outside EMS-E. One of the departments of EMS-E is called Implementation & System Verification, and it is here where the actual administration of the control unit software is managed. Figure 6 below gives an overview over EMS-E.

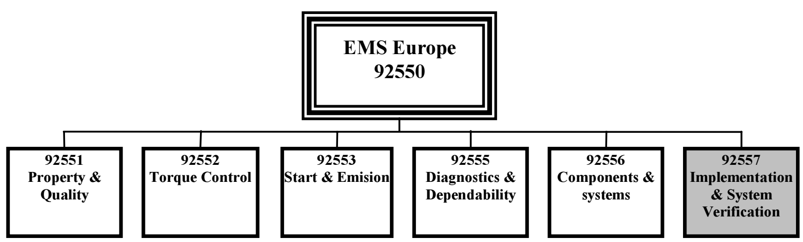


Figure 6 EMS Europe Department Structure. (VCCI, 2001).

4.2.2 An Overview of the relevant Company Structure

To give a final overview over relevant company structure we present figure 7 below showing the relation between the different parts. Only the parts related to the system are included to get an acceptable overview. The “VCC” part at the top represents the entire VCC, everything else is below. For each step down the structure, the part is more specialised.

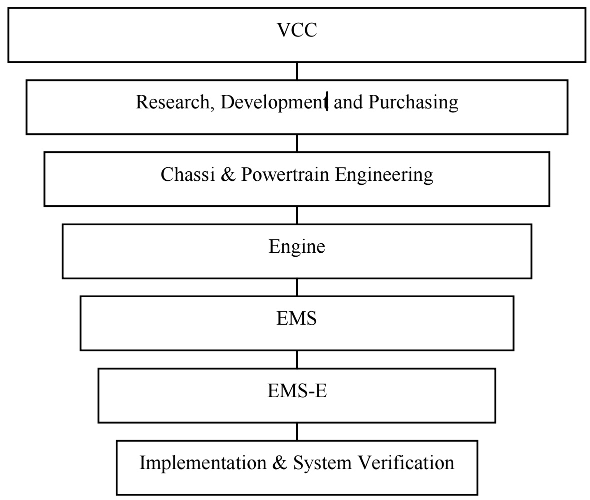


Figure 7 Relevant company structure of VCC.

4.3 General Description of BOSCH

Although we are concentrating our work on the VCC part of the current system, we will give a brief description of BOSCH as a company.

According to BOSCH (2001), Robert BOSCH AB in Sweden is a subsidiary of Robert BOSCH GmbH (BOSCH) in Stuttgart, Germany. BOSCH manufactures various technology related products from safety-critical systems for modern motor vehicles, through to domestic appliances and power tools, as well as manufacturing solutions for industry.

The main office is located in Stuttgart, Germany and the company employs approximately 195.000 people around the world.

4.4 The Current system

To get a picture of the current situation, we had number of informal discussions and observations with people in the system. People from both VCC and BOSCH use the system, and because of that we have had discussion with people from both sides. We had most discussions with the software coordinators, since they are the power users of the system, meaning they use it most extensively.

The reason for keeping the discussions informal was to let the users give their own picture of the activities, and not lead them with questions, thus giving a more relevant and broader picture of the activities and communication compared to formal interviews. We will look into the current situation, explain how the system is working and who are involved. This will lead us to a clear picture over the current situation.

First of all, a description over relevant terms follows to help the reader understanding the situation.

4.4.1 The Software Being Administered

The software involved in this system is software for the cars motor control unit. This unit controls most of the cars computer controlled functions and is updated frequently. The system we are looking into is working with the administration of requests for changes in the control unit software.

4.4.2 Change Request for Software (CRS)

A CRS is a Lotus Notes (LN) based form, which is used to request all kinds of changes in a software product, in this case the software for the cars control unit. The CRS is not the software itself, but only the form on which changes are requested. LN is a database-driven groupware application that manages information for many users on a network, to communicate, collect and share database documents, following a client/server model. A request is made when the software needs to be updated. Each time there is need for change in the software a CRS is made by people working with the optimisation of the control unit.

Appendix D contains an example of a CRS form.

4.4.3 Software Requirement List (SRL)

An SRL is a Microsoft Excel spreadsheet containing information regarding all CRSs for a certain program version. The SRL is the main information source for information regarding the CRSs. It contains information such as what kind of function a CRS concerns, planned release date, change description, person in charge for the change etc. It is extended each time a change request is made until the new version is released. Then it becomes an order list, containing all the included changes in the new version, and a new SRL is created.

Appendix E contains an example of a SRL form.

4.4.4 The People Involved

In the system there are three different kinds of people involved. They are Optimisers, Software Coordinators and Supervisors.

The different kinds of people are related to the system in various ways. Optimisers are primarily filling out CRS forms, Software Coordinators are “power” users which works with the system the most of their time. Supervisors have the final decision about what change should be implemented and are responsible for all the changes.

4.4.4.1 The Optimisers

The Optimisers work hands-on with testing the cars different functions. Standards, protocols and regulations have to be complied with, and functions need to be as well functioning as possible. There are a total of about 50 optimisers.

An example is the work with the different protocols that set the rules for the communication between the engine and different diagnostic tools. It is important to achieve good diagnostic results when developing engines. We are talking about protocols for the signals transferred between engine and different diagnostic devices, and that the protocols are correct in relation to the different services that the data will serve. There are standards set by e.g. the government, so that governmental organizations can use their test equipment on VCC's engines. Also the mechanics at the different garages and VCC retailers must be able to use their equipment to diagnose failures in the engines performance.

The Optimisers function in the system is to create and update CRS forms related to the control unit software produced by BOSCH. The CRS forms are used as a base for correcting, updating and extending the control unit software. BOSCH produces a new version of the software when enough CRS forms are examined and sent to BOSCH.

The Optimisers working with CRSs in this system are located both in the EMS-E department, and in other Engine departments.

4.4.4.2 The Software Coordinators

A considerable part of the Software Coordinator's work is to act as a contact person between VCC and BOSCH. There are two Software Coordinators in the system. The contacts mediated concern coordination of numerous different tasks related to BOSCH. Included in these tasks, is to attend and arrange numerous different meetings on various levels. The meetings primarily concern coordination of different kinds.

More specifically related to the system we investigate, the Software Co-ordinator is the link between the Optimisers and BOSCH in the system. The Software Coordinator administers various things, including CRS handling, SRL handling, software delivery schedule and the ordering of software from BOSCH when needed. The Software Coordinators are located in the Implementation & System Verification department.

There is also a person at BOSCH in Sweden acting as a kind of Software Coordinator. He works as a link between VCC and BOSCH in the form of translator between cultures and languages and attends meetings at VCC concerning the change requests.

4.4.4.3 The Supervisors

The Supervisor related to this system is the "Construction Assignment" (KU) responsible person. The KU-responsible has the whole responsibility for a construction assignment, of which the software for the control unit is a part.

The KU-responsible has the final decision about what change should be implemented and are responsible for all the changes. When VCC orders a new software version from BOSCH, it is the KU-responsible who makes the decision.

4.4.5 Workflow Diagram over the Current System

Through interviews and discussions we have obtained picture over the current situation. Figure 8 illustrates the activities in the current situation. All events occurring in the current system are included.

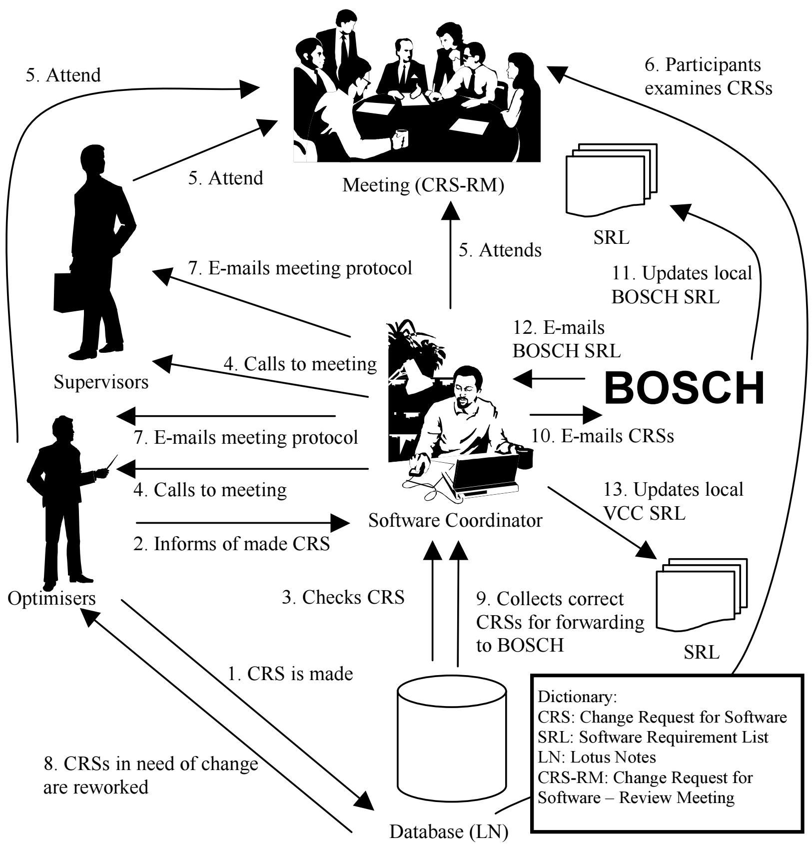


Figure 8 Workflow diagram over the current system.

4.4.6 The Order of Events

The order of events is the system is explained more detailed below. Each activity is explained, giving deeper information about the activities in the system.

1. *CRS is made by Optimisers*: When an Optimiser feels the need to make a CRS he fills in a CRS form and saves it to the LN database.

2. *Informs about made CRS*: After the Optimiser has done the CRS, he informs the Software Co-ordinator about the CRS. This can be made trough e-mail, phone or face-to-face communication.

3. *Software Coordinator checks CRS*: After receiving the e-mail from the Optimiser, the Software Coordinator opens the LN database and checks the CRS. If there are changes to be done, or if he has questions, he updates the CRS form.

4. *Software Coordinator calls to meeting*: Once a week the Software Coordinator calls to a Change Request for Software Review Meeting (CRS-RM). Usually it is made through e-mail, but also face-to-face. Participants are Software Coordinators, Optimisers who have CRSs being examined at the meeting, and related Supervisors.

5. *Software Coordinator, Optimisers and Supervisors attend meeting*

6. *Meeting participants examines CRSs*: On the meeting the CRSs in the LN database, which are up for examination, are reviewed and processed. A priority list is made, deciding which CRSs are to send to BOSCH after the meeting.

7. *E-mails meeting protocol*: After the meeting a copy of the meeting protocol is automatically sent to all meeting participants.

8. *CRSs in need of change are reworked*: If there are any irregularities, or need for other updates in the CRSs covered at the meeting, they need to be reworked. The Optimiser then corrects/completes the irregularities.

9. *Collects correct CRSs for forwarding to BOSCH*: The Software Coordinator processes the correct CRSs and before sending them to BOSCH. The processing consists of first converting the CRSs from LN format to Microsoft Word format. The CRS documents and eventual additional files such as pictures, or data documents, are then are zipped to a single file. The zipped file is finally encrypted by running the Pretty Good Privacy (PGP) program on the file.

PGP is a software product, which is used to protect the privacy of e-mail messages and files by encrypting them so that only the intended recipients can read them. There is also a possibility to digitally sign messages and files, which ensures their authenticity. A signed message verifies that the information within it has not been tampered with in any way.

10. *Software Coordinator mails the CRSs*: The encrypted zip file is sent to BOSCH as an attachment to an ordinary e-mail.

11. *Software Coordinator updates local BOSCH SRL*: BOSCH processes the CRSs sent by the Software Coordinator and updates the local BOSCH SRL with the new information regarding the CRSs sent.

12. *BOSCH e-mails BOSCH SRL to the Software Coordinator*: Whenever the Software Coordinator requests it, BOSCH sends an updated SRL to the Software Coordinator located at VCC. In this SRL all updates made at BOSCH are included.

13. *Software Co-ordinator updates local VCC SRL*: The software Co-ordinator imports the BOSCH SRL to the local VCC SRL and checks that they are corresponding. If there are any irregularities, then he marks them in the SRL to be checked again at the next SRL update.

5 Interviews and Data Gathering

In this chapter we will present a summary of the interviews made and present the data collected regarding the grading of the software characteristics. We will as well give a summing up of the comments made in connection with the grading. The results of the interviews and data will be used in our discussion and analysis chapter following.

In view of the fact that the different participants have knowledge in various areas, the questions we presented them varied. System users (Optimisers, Software Coordinators and the BOSCH representative) received different kind of questions compared to the questions given the Experts.

Out of security and consideration, we have decided to keep the participants figuring in the interviews and surveys anonymous. There is instead the following naming:

- Optimisers are named Opt1 and Opt2
- Software Coordinators are named SWC1 and SWC1
- The BOSCH representative is named BOSCHrep
- The Experts are named Expert1, Expert2 and Expert3

The interviews and characteristics comments were made in Swedish and translated into English.

5.1 Interviews

Below is a summary of the interviews we conducted. A complete list of questions and answers can be found in Appendix A.

5.1.1 VCC Interviews

The people we interviewed at VCC were all system users. Two of them were optimisers working in different departments. The other two persons interviewed were Software Coordinators, located at the same department. All the interviews were conducted at VCC in Gothenburg.

5.1.1.1 Questions Regarding the Participants

We asked the participants if they could give a description of their current assignments and activities. This was made to understand their role in the system. Opt1 is working with the protocols that set the rules for the communication between the engine and different diagnostic tools. It is important to achieve good diagnostic results when developing engines. There are standards set by e.g. the government, so that governmental organizations can use their test equipment on VCC's engines.

A big part of SWC1's work is to act as a contact person between VCC and BOSCH. He coordinates numerous different tasks. SWC1 also attends and arrange meetings on various levels. The meetings primarily concern coordination of different kinds. SWC2 work towards Siemens regarding the software for some of the car models engines. He also attends and arranges numerous various coordination meetings.

To get a picture of their role in the administrative system, we asked the participants what their roles were in the cooperation with BOSCH. Opt1 told us that, when he finds a problem with a protocol, he reports it to BOSCH in the form of a CRS. Opt2 also answered that he creates CRSs

SWC1 acts as a contact person between VCC and BOSCH, and also administers a variety of tasks, including CRS handling, SRL handling and software delivery schedules. SWC2 is not directly involved in the BOSCH related CRS handling, but works close to SCW1 with similar tasks as SWC1 towards Siemens.

5.1.1.2 Questions Regarding the Current System and Working Situation

With the next question we tried to get an idea about the problems with the current situation and the way they perform their work concerning the handling of CRSs and SRL. Opt1 does not get satisfying information about what has happened to the CRS after it is sent BOSCH. It is not possible to get a pleasant total view of the CRSs sent. Opt2 cannot easily obtain information about when his CRSs will be implemented in the code.

Both SWC1 and SWC2 think is that it is complicated to get and hand out information. People in various positions, both at BOSCH and VCC have questions regarding CRS. SWC1 say people often ask questions through e-mail, telephone or in person; then if he does not have the information they want, he has to contact someone else, e.g. people at BOSCH.

To probe the question of the availability of information we asked if they obtain enough and correct information for performing a satisfying result in the work they do. Both Opt1 and Opt2 referred to the question regarding problems with the current situation. SWC1 and SWC2 want the information flow from BOSCH more satisfactory.

Next question asked was if they sense there is the ability to influence the work they perform. Opt1 is of the opinion that he can influence, but experiences a form of inertia or slowness in the beginning of a transformation process

Both Software Coordinators feel it is possible to give suggestions, at least at VCC. It feels as if it a bit harder to get BOSCH interested in new ideas.

5.1.1.3 Questions Regarding a Future System and Working Situation

We asked if the participants had suggestions about what type of activities a future administrative system could have. Opt1 would like his own specific virtual place where he can have a better real time overview of the status of CRSs he has made, like his own CRS inbox. Opt2 wants to be able to obtain information about when a certain CRS will be implemented in the code.

SWC1 and SWC2 wish for better overview over the system. It should be easier to obtain information without having to contact various persons each time there is a question or need for information. The CRS handling needs to be arranged in a different way. More and better functions such as information regarding a CRS's status, estimated cost per CRS, estimated program version where a change will be included etc.

5.1.2 BOSCH Interview

The interview we did with BOSCHrep was conducted at the BOSCH Sweden office in Gothenburg.

We started with asking if BOSCHrep could give a description of his current assignments at BOSCH. BOSCHrep is a so-called Group Leader, with 3-4 people under him. His assignments include software developing, software modelling, and to act as activity coordinator. His role in the cooperation with VCC is to act as a link between VCC and BOSCH in the form of translator between cultures and languages. He also attends CRS related meetings at VCC.

With the next question we tried to obtain BOSCHrep's opinion about problems with the current situation. BOSCHrep thinks that CRSs sometimes tend to arrive late in the project just before a new release of software. There seem to be a problem with the communication of information. This lack of communication might lead to a crisis of confidence; quality problems and a "give and take" situation may arise.

Next question area was about BOSCHrep's thoughts about VCC's proposition to change the current way the two organizations administer the communication with each other. As a regional office placed in Gothenburg, BOSCH in Gothenburg is very interested in new ways of developing the communication between the participants in the projects. A future administrative system could have a joint database to increase the structure of multiparty projects like this. It would help BOSCH to structure the projects they are jointly working on.

5.1.3 Expert Interviews

We interviewed three experts on practical system structures to get a view of what a new system could look like. Expert1 works as Software Coordinator at VCC, but was interviewed in the role as expert because of his role in a web based change request system used by VCC towards Denso, another supplier of control unit software. He was involved with the development of the system.

Expert2 and Expert3 are working at Volvo IT, Gothenburg. Volvo IT is not related to VCC, but is an independent IT company, which previously belonged to Volvo.

5.1.3.1 Questions Regarding the Participants

First of all we asked if the participants could give a description about what their work consists of. Expert1 works with the supervision of the development of the software for the car engine's control unit, in the models having the Denso engine control unit. The working tasks include CRS handling, SRL handling, software delivery schedule, implementation planning and program testing co-ordination. Expert2 handles the administration of Oracle databases within a department at Volvo IT. They are running a database hotel for Volvo IT's customers. Expert3 works as a part of the local IT support at VCC. His working task is to support a part of VCC in their use of IT tools.

5.1.3.2 Questions Regarding the VCC/Denso System

To get an insight into how a similar system had been developed into a web based solution, we asked Expert1 to describe how system between VCC and Denso is structured. The system is built with the web and database server placed outside both Denso and VCC. There is an IP control access from VCC through terminals at VCC. The selected platform for the SRL handling between VCC and Denso consists of a PostgreSQL database manger and an Apache web server on a Linux operating system. The database is an ANSI SQL database. The browser is Netscape with a 128 bits RC4 encryption.

5.1.3.3 Questions Regarding a Future System

We wanted to get information regarding what the physical structure of a web based system between VCC and BOSCH could look like. Expert2 said there are different possibilities. According to him, Volvo IT has very good turnkey solutions for this kind of systems. Also Expert3 thought it would be a convenient solution to let Volvo IT handle the connection, if BOSCH agrees.

Another possible connection solution could be to build a completely new, separate system. This solution gives the possibility to freely put servers where you want. If VCC or BOSCH feels they have the need to store some data on their own side, on an own server, then this could be a way to do that. There are different ways to build this kind of system.

Following the question about physical structure, we asked what choice of software there is to consider. Expert1 mentioned one solution, similar to what is being used in the Denso system. It consists of a PostgreSQL database manger and an Apache web server on a Linux operating system. The database should be an ANSI SQL database, for compatibility and standard. The browser could be anyone, which has a good encryption support. Expert2's opinion was that if the choice were to use the services of Volvo IT, then Windows 2000 would be the operating system. Oracle would be the database software. In their web hotel they run Microsoft Internet Information Server as web server.

To match the questions about software and physical structure, we asked what possibilities there are when it comes to hardware. Expert1 mentioned the possibility to use a similar system as the one between VCC and Denso; which is running on an ordinary PC. This would probably suite the new system also.

Both Expert2 and Expert3 said that the hardware could be any PC with enough power and storage capabilities. It depends on how advanced the new system will be. If the system is hosted at Volvo IT, they have all necessary hardware.

The final question area we covered with the experts was the question about security. The view of Expert2 was that it is important to have very strict standards when it comes to security regarding trespassing into systems. Anyone who builds a system using the web should be very cautious. Nowadays there are good encryption standards, firewalls and proxy servers to choose from. Data security is another kind of security aspect. Data must be properly backed up maintained to secure data and system performance.

Expert3 mentioned that some important things to consider are user authorization and communication security. There has to be user account logins and some kind of encryption. A VPN (Virtual Private Network) between VCC and BOSCH sounds like good idea. A VPN connects Intranets together; allowing remote access to services as transparently and securely as the users locally connected to remote networks and hosts. It lets users access information on their corporate network (e.g. an Intranet) over a public network such as the Internet.

5.2 Systemisation of Software Quality Data

We used surveys to find out which of the software quality characteristics found in the extended ISO model are important to consider when migrating the current system to a web based environment. The surveys used included explanations of the extended ISO model. The survey forms can be found in Appendix B.

5.2.1 Software Quality Characteristics Table

The grades given in the surveys were compiled into averages. The averages for the main characteristics were compiled by us; calculated from the averages for the sub characteristics. On the next page table 1 shows the grades and averages the participants gave when grading the software quality characteristics.

Extended ISO Model	**Experts**			**Average**
Quality Characteristic	**1**	**2**	**3**	
Functionality				3,86
Suitability	4	3	4	3,67
Accuracy	5	5	3	4,33
Interoperbility	3	2	4	3,00
Compliance	5	5	4	4,67
Security	4	4	4	4,00
Traceability	4	?	3	3,50
Reliability				3,00
Maturity	3	4	3	3,33
Fault Tolerance	4	4	3	3,67
Recoverability	4	2	3	3,00
Availability	4	2	2	2,67
Degradability	3	2	2	2,33
Maintainability				3,89
Analysability	4	4	5	4,33
Changeability	4	4	5	4,33
Stability	3	4	4	3,67
Testability	3	3	4	3,33
Manageability	5	4	4	4,33
Reusability	3	3	4	3,33
Portability				2,83
Adaptability	3	2	3	2,67
Installability	4	2	2	2,67
Conformance	5	3	3	3,67
Replaceability	3	2	2	2,33
Efficiency				3,33
Time Behaviour	4	4	4	4,00
Resource Behaviour	3	2	3	2,67

Extended ISO Model	**Users**				**Average**
Quality Characteristic	**1**	**2**	**3**	**4**	
Usability					3,11
Understandability	3	3	3	4	3,25
Learnability	4	4	2	3	3,25
Operability	3	4	4	5	4,00
Explicitness	3	4	5	5	4,25
Customisability	5	3	1	2	2,75
Attractivity	1	2	1	1	1,25
Clarity	4	3	3	2	3,00
Helpfulness	4	2	2	2	2,50
User friendliness	4	4	3	4	3,75

Table 1 The grades and averages of the software characteristics.

5.2.2 Comments Regarding Software Quality Characteristics

During the completion of the surveys, the participants could give their comments on the subject of each characteristic. Below is a summary of the comments the participants gave when grading the software quality characteristics. In this section we have included only the most interesting comments. A complete list of comments can be found in Appendix C.

5.2.2.1 Functionality Qualities Characteristics

Expert2 commented that in most solutions suitability can be adapted depending on needs. Accuracy was pointed out as "what the customers pay for". This needs to be at highest quality to gain the customer's confidence in the system. Expert 3 pointed out that security is a question of trust between solution provider and the customers. If there is no belief in the security of the system, it will probably not be used. Compliance was mentioned as a bit tricky to estimate by Expert3. The software has to comply to a certain extend, or else it will not be used at all.

5.2.2.2 Reliability Qualities Characteristics

Both Expert1 and Expert2 saw recoverability as a characteristic, which isn't directly related to the software itself. It is more a about background functions, related more to hardware and data security functions. Expert1 thinks that it is hard to get 100 percent availability, not least in a web-based system where the Internet connection itself might go down, completely unrelated to the software. It might also be a problem to find a long enough time slot when backing up data if the system is used around the clock.

5.2.2.3 Efficiency Qualities Characteristics

Consistent time behaviour is important according to Expert2. The users are expecting speed. This is a thing, which might become quite annoying if the user has to wait longer than he feels is acceptable. In a comment given by Expert3, he remarks that in this kind of systems software (e.g. browser) time behaviour often depends on external factors such as Internet bandwidth.

5.2.2.4 Usability Qualities Characteristics

SWC2 commented understandability as "it would be strange to work with a program if you do not understand its capabilities". Operability was very important to SWC1. He expressed that it's very important to be able to operate a program, or else it would be frustrating in the long run. Customisability was an important factor to Opt1. He finds it valuable to be able to customise the way his "work space" looks like. A comment regarding attractivity came from SWC2. The bells and whistles of the software is not important. It's much more important to have great functionality.

5.2.2.5 Maintainability Qualities Characteristics

Expert2 states analysability as important if there is a cause to believe that the software will be used for a longer time span. He applies this to all maintainability qualities. The software may need change in the future. It will be easier to maintain the system if there is good maintainability.

5.2.2.6 Portability Qualities Characteristics

Expert2 brought up conformance as he told that the software need to behave the way the user expects, or else unnecessary annoyances will occur.

6 A Future Scenario: Data Analysis and Discussion

In this chapter we will interpret the results and analyse the data, which have been collected through out our study. We will have a discussion around the two sub questions formulated, to help us find the answer to our main question.

In order to answer the first sub question, "What possible changes are there to improve the current situation?" we will follow the stages in Soft Systems Methodology (SSM).

The primary information sources we use to analyse the first sub question, are the interviews we made, both formal and informal, and discussions with the parties involved in the system. We will start by presenting a problem situation, which will be our base for the following SSM stages. The next part of the chapter will be the root definition, where the relevant human activities in the system are explained. After that conceptual models will shown, visualising what was expressed in the root definition. Following this, a comparison between the conceptual models and reality will be done. Finally there will be an assessing of possible changes. As we declared in chapter 1.4, we will not complete the full circle of stages in SSM. The seventh stage, which includes taking action to improve the problem situation, will not be brought up in this thesis. In this chapter we advance in a linear fashion, but as stated in chapter 3.2, the process to reach here has been iterative.

To help us discuss and analyse the second sub question regarding the important characteristics for a new administrative system, we use the surveys conducted, comments made and literature studies. We conduct a review of the characteristics grading and discuss the results. Then we will present a model where the importance of each characteristic is indicated and give a view over the characteristics, which should be considered.

The analysis and discussion in this chapter will lead us to the answers of the two sub questions. This will be used in the next chapter of conclusion and recommendations to help us finalise the thesis, thus give an answer to what a web-based solution replacing the current system could be.

6.1 Possible Changes to Improve the Current Situation

In this section we will, with the help of the learning cycle of SSM, define possible changes for improving the current situation in which VCC and BOSCH cooperate. We have already covered the first stage of SSM in chapter 4, where we entered considered problematical, illustrated in figure 2, to obtain a description of the current way that VCC and BOSCH administer CRSs and SRLs. During the interviews and discussions with both people from VCC and BOSCH, we were able to obtain a satisfactory description of the current system. We will start by describing the problem situation.

6.1.1 The Problem situation

In order to obtain a clear view of the problem situation we compiled a rich picture shown beneath as figure 9. It is based on the interviews with the optimisers and Software Coordinators at VCC and a group leader representing BOSCH. Full interview questions and answers can be found in Appendix A. The picture points to the problems that both people from VCC and BOSCH experience today.

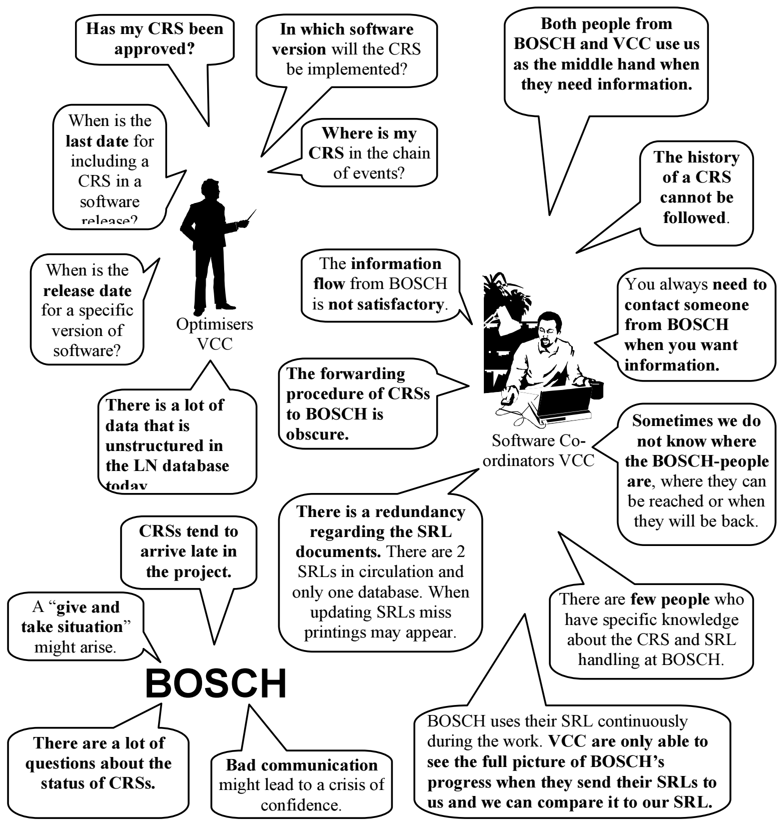


Figure 9 Rich picture of the current situation.

There are numerous problems with the current way of working. We sense the most imminent problem seems to be the ability to get access to the right information. The Optimiser who creates an initial CRS cannot obtain satisfying information about the status of it. Optimisers are as well not able to get a clear, structured overview of the CRSs that they have produced over time. When they have questions regarding release dates for a specific software releases, they have to contact someone who might know something. This person may be either from VCC, e.g. the Software Coordinators, or someone from BOSCH who may well not be reachable.

For the Software Coordinators the problem seems to be that they have become more of information coordinators than the former. Compared to the optimisers they have a better view of what is going on in the system, and acts as a link between the activities at VCC and BOSCH. Still, they sense the information flow from BOSCH is insufficient.

Another major problem concerns the SRLs that give VCC information about the progress of work at BOSCH. At present there are a two versions of the SRL in circulation, one at VCC and one at BOSCH. BOSCH continually uses them during their work with the software for the engines and emails them to VCC more ore less frequently. The time span between updates of the VCC SRL can sometimes reach several weeks.

The forwarding procedure of CRSs from VCC to BOSCH is another thing that could be done in a better way. This relates to the procedure of collecting CRSs from the database. They are first converted from Lotus Notes format to Microsoft Word, then compressed to ZIP format before encrypted with PGP and e-mailed to BOSCH.

The Software Coordinators, in the same way as the optimisers, do not perceive that they have a satisfactory view of the CRSs today. It is for example difficult to follow the history of a CRS.

The co-workers from BOSCH have similar problems as the people from VCC in regards to the flow of information. They too sense there are problems with the communication of information between the companies. CRSs from VCC tend to arrive late in the project, too close to the software release date. This problem correlates with the optimisers' problem of getting a clear picture over what dates are set as last dates for specific software releases. There are also a lot of questions from VCC that has to be answered, often over the phone, about the status of CRSs. This might turn into a "give and take" as mentioned in the BOSCH interview. This type of situation, with the lack of a good channel to communicate information, may in the long run lead to a crisis of confidence. During discussions we noticed that most of the participants felt the same way.

6.1.2 Root Definition

As stated before, VCC is interested in replacing the current system with a web-based solution. The creation of a root definition gives us a clear view what the system is and its purpose.

We used the criteria that the "mnemonic" CATWOE gives us when we constructed root definitions in cooperation with the optimisers, Software

Coordinators and the BOSCH representative. This was made with each person individually during the interviews and later summarised in the root definition found below. The root definition must define both the activities solving the problem situation, and at the same time will make the system function as supposed.

The transformation process (T) lines up what activities should be a part of a future system. There is also a view of the people involved, and their roles. They are represented in the form of customers, actors or owners. The Weltanschauung tells us why we need this type of system. The environmental constraints help us point out what is important to consider in getting the system to work properly.

CATWOE-analysis

A professionally manned system for the administration of CRSs and SRLs to and from BOSCH in the form of a web-based interface connected to a database. The system will give an up to date view of CRSs and SRLs for all people involved. The database houses all CRSs and SRLs. It will be searchable through an encrypted website. All users will be able to have their own customisable start page. VCC will be the owner of the system.

Customer: *Who would be the victims/beneficiaries of the purposeful system?*
Optimisers, Software Coordinators, project management (KU), everybody related to the project at BOSCH.

Actors: *Who would do the activities?*
Optimisers, Software Coordinators, project management (KU), everybody related to the project at BOSCH.

Transformation process: *What are the purposeful activities?*
Secure login with encryption to personal start page, search for information, create CRSs, edit CRSs, Approve CRSs, new CRS notification, create SRLs, update SRLs

Weltanschauung: *What view of the system makes this definition meaningful? What is the purpose of the system?*
The handling of SRLs is a quality requirement set by VCC. There is a need to adapt to the government laws and standards, and to other nodes in the company, Nodes that are concerned with other parts of the vehicles e.g. gearbox and chassis.

Someone is needed to make the ordering of new software, and administer CRSs. All administration of software related tasks go through the same channel. This is important for securing the quality of the products manufactured. There is a lot of different software for different engines developed, and someone has to control, coordinate and set the right priority for the development of this software.

The administration of CRSs and SRLs will be made more effective. The improved access to more structured information in a system like this will be of great help both to VCC and BOSCH.

Owner: *Who could stop this activity? Example: the Company?*
Management at VCC, Ford Motor Company, Volvo IT.

Environmental Constraints: *What constraints/problems in its environment does this system take as given?*
CRSs and SRLs must be correctly filled in.

There are few people who have specific knowledge about the CRS and SRL handling. People need to learn about the new system and also the whole cycle of events in the process.

Volvo IT who is responsible for the security must approve of a solution like this that is accessed from outside VCC.

6.1.3 Conceptual Models

Figure 10 below is a visualisation of the system previously defined in the Root definition. By using information from the expert interviews, see Appendix A for complete interview questions and answers, we were able to picture the web server and database.

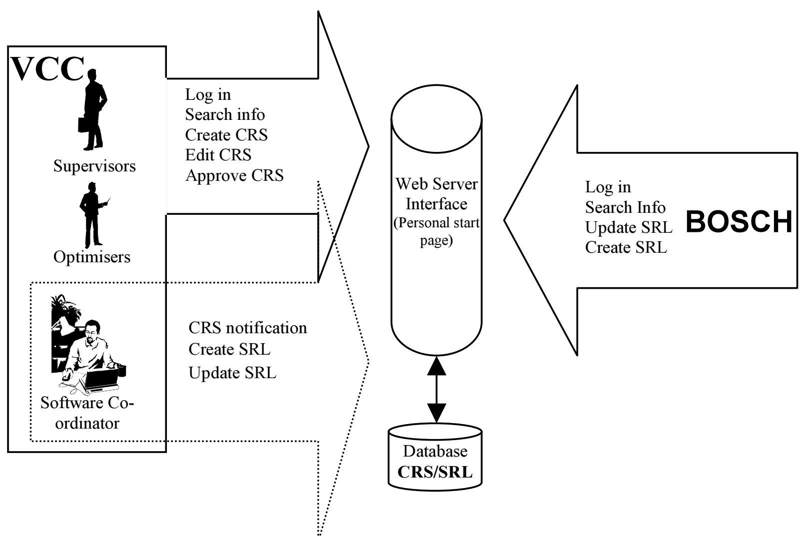


Figure 10 Visualisation of a future system.

We also present a model of the most important activities in figure 11 below, to show their logical dependencies.

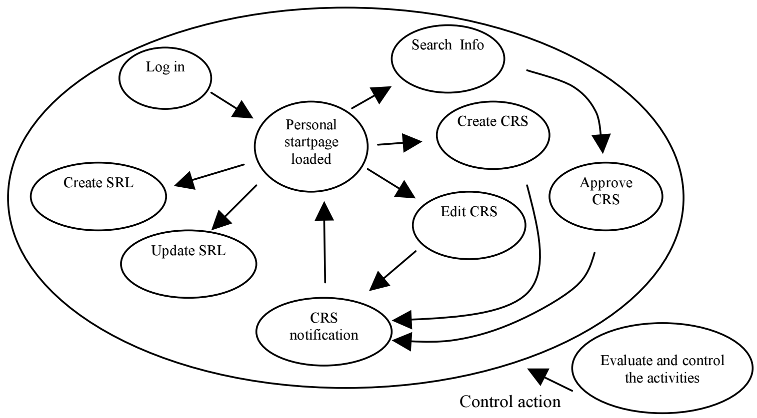


Figure 11 Activity model of a future system.

We used the conceptual model put together by Mathiassen and Nielsen (1989) as a starting point for building the activity model.

6.1.4 Comparing Conceptual Models with Reality

Until this part of the SSM process we have found out about the situation with the help of interviews, informal discussions and our own system thinking. Different users of the system have answered questions about their role and the problems they experience working in the current environment.

In order to compare the conceptual models with the current real world situation we organised another discussion with participants from both VCC and BOSCH. The aim of the discussion was to create a debate around possible changes that will improve the problem situation expressed previously. We concentrated the discussion around a future systems possibility to fail, by focusing in on the control action terms effectiveness, efficacy and efficiency brought up in chapter 3.2.2.2.

The outline for the discussion was to try to find answers to the following questions:

- Is a web-based solution the right way to go (Effectiveness)?
- Will it work, based on the activities in the root definition (Efficacy)?
- To what degree will the system use up resources (Efficiency)?

All parties agreed that a web-based solution would be the best way to go. Both VCC and BOSCH use similar solutions in other parts of their organisations. The participants in the discussion had difficulties seeing other kinds of solutions to the problems. One thing that stood clear was that everybody wanted to change the current system. They came to the conclusion that a web-based solution would be the natural way to develop and improve the co-operation between the companies.

The representatives from VCC also have the desire to move away from the use of Lotus Notes provided in the current setting. During the discussion different activities in the current and a possible new web-based system were examined. As help when discussing if a web-based solution would be sufficient in handling current and future activities a user/administrator manual for another similar web-based system where examined and discussed. This system is in use today by another group at the department who is co-operating with the Japanese company Denso.

Then the question might be raised, why VCC does not use the system already developed, in the case with BOSCH. The problem is that this system was developed by and is managed by Denso. The conclusion that all agreed of was that if the proposal for a new administrative system had the same functionality as the Denso system it would be good solution. The Denso system is described in the expert interviews found in chapter 5.1.3.

When it comes to the activities defined in the root definition (T) all agreed that they would make the new system perform the necessary tasks, also in a more efficient way, by simplifying the work.

Another thing that came out as a result of this discussion was that there might be a desire from other branches of the VCC organisation to incorporate activities in their organisations into a similar web-based database solution. This indicates that it would be important to follow VCC standards when developing a new system.

6.1.5 Assessing Feasible and Desirable Changes

In this section we will present the systematically desirable and culturally feasible changes that will improve the current way of working, as mentioned in chapter 3.2.2.4. This is the sixth stage in the learning cycle of SSM, and a continuation of the discussion in the previous stage. We will start by looking at the systematically desirable changes and move on to look at the changes from a cultural perspective.

6.1.5.1 Systematically desirable changes

The outline for the discussion starting in the preceding section was concentrated around three questions concerning effectiveness, efficacy and efficiency. These terms can generally be used when controlling the final model of the system, illustrated in figures 10 and 11. Together with the participants we came to the conclusion that the system shown in the conceptual models would be systematically desirable. Below is a description of the different activities shown in the conceptual models and their logical dependencies.

Log in to personal start page:
The secure login with encryption to and use, of a common interface and database for all actors in the system, will solve a lot of the problems relating to the flow of information and communication at present time. It will also meet the demands for security mentioned in the expert interviews, when talking about information being shared over a public network.

Search information:
All actors in the system, especially the optimisers, will be able to acquire information about ongoing activities and the statuses of CRSs. A personal start page may include an “inbox” with CRSs, as Opt1 expresses it in the VCC interviews, chapter 5.1.1.3. A lot of the uncertainties that the optimisers are expressing in the rich picture, figure 9, will be resolved in this way. The “give and take situation”, that BOSCH rises concerns about, will most likely not appear, and the time they put on answering questions from VCC will probably, to a great extent, be reduced.

Creation of CRS, edit CRS, new CRS notification and approval of CRS:
The creation and editing of CRSs is an activity that is necessary and will not change much from before. This will be done through the web interface and saved on the jointly shared database. The Software Coordinator will automatically get a notification that a new CRS has been created, e.g. by e-mail. The CRS will, at this point in time, not be visible to BOSCH. In order to be visible the CRS has to be approved of in the CRS review meeting. During or after the meeting a specific CRS can be approved of instantly and made available to BOSCH just by changing its status in the database. This will render a CRS notification to the co-workers at BOSCH. The Optimisers at VCC will at the same time be able to see the status of their CRSs change.

The forwarding procedure of CRSs by e-mail from VCC to BOSCH will in this way be unnecessary. Software like Microsoft Word, PGP and WinZip do not need to be used anymore and can be taken out of the system.

Update SRL and create SRL:
BOSCH does not have to e-mail VCC their SRL anymore. They will work against the commonly shared SRL, which will give VCC instant access to information about the activities at BOSCH.

6.1.5.2 Culturally feasible changes

Checkland (1989) also stresses the importance that the changes are culturally feasible. In this case you move from a quite unstructured way of working with a lot of tools and channels used in performing the activities, to a system that incorporates most of the activities. For this to work, there is a need to emphasise the purpose of the system, and why it is meaningful. The Weltanschauung (W) declared in the root definition play an important role in this. W Captures the different conceptions and views about the system and its purpose to include the cultural aspects of the changes. By defining W, described with the help of two questions below, in co-operation with the people involved in this system, we also ensure that changes are culturally feasible.

What view of the system makes this definition meaningful?
There is a need to administer CRSs and SRLs more effectively. The improved access to more structured information in a system like this will be of great help both to VCC and BOSCH. Everybody that in some way will be affected by the new system will benefit from it. All administration of software related tasks go through the same channel. This is important for securing the quality of the products manufactured. There is a lot of different software for different engines developed, and someone has to control, coordinate and set the right priority for the development of this software.

What is the purpose of the system?
The purpose of this system is to reach set quality requirements. The handling of SRLs is a quality requirement set by VCC. There is need for adapting to government laws and standards, and to other nodes in the company that are concerned with other parts of the vehicles, e.g. gearbox and chassis.

6.1.6 Summary of Possible Changes

We have now identified possible changes to improve the current situation in a both systematically desirable and culturally feasible way.

The conceptual models presented in 6.1.3 give an overview of the activities being present in the future system. By implementing such a system, the administration will be improved by:

- Easier and more effective way of managing the activities related to the CRSs and SRLs.
- Enhanced view and access of information.

The people involved in the system will be affected in various ways:

- The optimisers will create and edit CRSs much in the same way as they do today but using another interface. The largest benefits for them will be the enhanced access to and better view of information.
- The Software Coordinators at VCC will be the ones who benefit the most from the new situation. They will have an improved working routine and will not have to mediate information to the extent of today.
- The Supervisors will benefit from the improved information availability and also be affected, as responsible, in their work through the improvements available to Optimisers and Software Coordinators.

- The BOSCH side of the system will gain similar benefits as stated above.

6.2 Important Characteristics for an Administrative System

In this section we will review the grading of the characteristics and discuss the results. Then we will present a model where the importance of each characteristic is indicated. Finally, we will give a view of the characteristics, which should be considered important. This will give us an understanding of what characteristics a new system should have, and give us the answer to the question regarding what the important characteristics could be.

A complete overview of the grades given can be found in table 1, chapter 5.2.1.

6.2.1 Interpretation of Software Quality Data Results

The characteristics have been marked from the viewpoint of a new system that consists of a document database with a web-based interface. Users will be able to access the documents for creating, viewing and updating. Search functions will also be available. The database will be updated in real-time when a new document is created or updated.

The grades were divided into the following five categories:

5 = Uttermost important. This has to be of highest possible quality
4 = Important. Need to be of high quality
3 = Should be considered, but isn't a priority
2 = Not important. May, or may not be considered.
1 = No importance at all. No need to consider.

There was also a possibility to grade with a "?", if the participant did not understand the meaning of a characteristic. One participant used this grade on one characteristic.

Grades were only given to all the sub characteristics, and we compiled an average of the six main characteristics (Functionality, Reliability, Maintainability, Portability, Efficiency and Usability) to find out if there are general conclusions to be drawn from each main characteristic.

By looking at the grading result, found in table 1 in 5.2.1, we can draw some direct conclusions. Five out of the six main characteristics had averages indicating that they at least should be considered. The only main characteristic not regarded important to consider, was the portability characteristic. We consider the low importance of portability as fairly expected; a future system will only be used at VCC and it's suppliers, making the need of moving or implement the system in other environment minor. There was only a single portability comment made by the experts. Expert2 commented the conformance sub characteristic, when he stated that the software need to behave the way the user expects, or else unnecessary annoyances will occur.

The two main characteristics with the highest average are functionality and maintainability. All of their sub characteristics were graded with an average of at least 3,00.

This shows that the participants regard maintainability characteristics such as being able to maintain and modify the software as important. Expert1 felt it important to keep good manageability, since a simple and straightforward structure is very helpful for all kinds of maintainability needs. Expert2 regarded all maintainability characteristics as important if there is a cause to believe that the software will be used for a longer time span. It will be easier to maintain the system if there is good maintainability.

The high functionality average points to the importance of having all required functions available in the software. We also consider this characteristic very important. If there is a lack of functionality, it will be tremendously frustrating for the users to use the software. Expert2 pointed out the accuracy sub characteristic, as "this is what the customers pay for. This needs to be at highest quality to gain the customer's confidence in the system.". Compliance was also mentioned as very important to consider. If the software does not follow at least minimum rules and standards, it will never be possible to use the software. This is an important remark when investigating a new system at VCC. There are numerous standards and guidelines to follow to be able to bring any new system in use. Security was as well highly graded. This is a question of trust between solution provider and the customers. If there is no belief in the security of the system, it will probably not be used at all.

The efficiency characteristic only has two sub characteristics, time behaviour and resource behaviour. Time behaviour were graded as important with a high need of consideration by all the participants. A consistent time behaviour is important according to Expert2, because the users are expecting speed and do not wish to wait unnecessarily. Expert3 also comments this characteristic. He remarks that in this kind of systems software (e.g. browser) time behaviour often depends on external factors such as Internet bandwidth.

The most important usability characteristics to the participants are operability and explicitness, two closely related characteristics. We think this shows that it is important to consider the characteristics that enable the user to operate, control and easily understand the intentions of the software's behaviour. The characteristics, not essential to consider, were attractivity, helpfulness and customisability. We view this as an indication that the users see a new administrative system's software as a functional tool, and they do not care too much about how the software appears. As the SWC2 put it: "I do not care too much about the bells and whistles of the software. It's much more important to have great functionality".

A look at the grades for the reliability sub characteristics shows they are all close to the average area, where they should be considered but are not a priority. This is reflected in reliability average of 3,0. An interesting detail here is that both Expert1 and Expert2 saw recoverability as a background function related more to hardware and data security, than a software characteristic. Another comment related to availability. Expert1 presented the reflection that it is always hard to get 100 percent availability in a web-based system where the Internet connection itself might go down, completely unrelated to the software. This shows how a web-based system is depending on other systems around it, which is significant to understand when trying to build such a system.

6.2.2 The Revised Extended ISO Model

After the review of the characteristics, we have a representation of how important the different sub characteristics are. We have divided the sub characteristics into three groups. The first group contains the characteristics, which have an average above 4, indicating they are important to consider and ought to be of high quality. The middle group includes those, which have an average between 3 and 4, showing they should be considered, but are not a priority. The last group contains the characteristics with an average below 3, illustrating they are not important to consider. To reflect the grouping, we have compiled table 2 below.

Most important to consider		**Lesser important to consider**		**Not important to consider**	
Quality Characteristic	**Average**	**Quality Characteristic**	**Average**	**Quality Characteristic**	**Average**
Compliance	4,67	User friendliness	3,75	Customisability	2,75
Accuracy	4,33	Conformance	3,67	Adaptability	2,67
Analysability	4,33	Fault Tolerance	3,67	Availability	2,67
Changeability	4,33	Stability	3,67	Installability	2,67
Manageability	4,33	Suitability	3,67	Resource Behaviour	2,67
Explicitness	4,25	Traceability	3,50	Helpfullness	2,50
Operability	4,00	Maturity	3,33	Degradability	2,33
Security	4,00	Reusability	3,33	Replaceability	2,33
Time Behaviour	4,00	Testability	3,33	Attractivity	1,25
		Learnability	3,25		
		Understandability	3,25		
		Clarity	3,00		
		Interoperbility	3,00		
		Recoverability	3,00		

Table 2 The quality characteristics by consideration importance.

6.2.3 The Essential Characteristics

During the analysis of the characteristics' grades and the comments given related to them, we have discovered several important characteristics to consider when introducing a new web based administrative system.

Out of the more technical characteristics, graded by the experts, there were two main characteristics standing out, functionality and maintainability. Time behaviour was as well given high grades. By assessing these factors, we have come to the assumption that it is important in a future system's software to have all required functions available in the software, or else it will be tremendously frustrating for the users to use the software. There has to be a capability of the software to provide functions, which meet, both stated and implied requirements when the software is used. The system software should also have ability to be altered quickly and cost effectively in the case of corrections, enhancements or adaptation of the software. All these characteristics have to be working in a swift manor.

The users side of the characteristics concerned the usability of the system. Their grading indicates there are two very important usability sub characteristics, operability and explicitness. This shows the importance of having a system, which is easy to control and operate. It in addition to that, there is a need to effortlessly understand the intentions of the software's behaviour. These aspects are indications that the users see a new administrative system's software as a practical tool, where they can understand and control the functions of the system software.

After studying the grading result and comments we have come to the understanding that the most important main characteristics of a new system ought to be:

- Functionality
- Maintainability

Within these two, all but three important sub characteristic is represented, but it is a concentrated view of the characteristics needs. Below is a more specific view over the important characteristics to consider, including time behaviour, operability and explicitness missing from functionality and maintainability.

- Compliance
- Accuracy
- Analysability
- Changeability
- Manageability
- Explicitness
- Operability
- Security

- Time Behaviour

The characteristics mentioned above are those, which are most important to bear in mind when replacing the current system with a web-based solution.

We see the reason for the concentration on functionality and maintainability as an expression of a professional point of view from both users and experts. They are concentrated to get a functional system; therefore they find characteristics as functionality and maintainability important. They do not have a focus on more aesthetic characteristics such as attractivity or clarity. The system kind, a web based administrative system, might be the reason for them to regard reliability less important. This is a non-critical system, where reliability does not have to be uttermost important. This is also a system, which probably only will be used at VCC and it's suppliers, which have a standard environments, making the need of moving or implement the system in other environments minor.

7 Conclusions

The purpose of this master thesis was to present a solution for a more ideal administration system, being web-based. We chose to use two sub questions to achieve this. In this final chapter we will present our solution by taking the answers to our two sub questions and compile them into an answer for a more ideal administrative system.

7.1 A Solution for a More Ideal Administrative System

We have identified possible changes to improve the current situation, by using soft systems methodology. The conceptual models presented in 6.1.3 give an overview of the activities being present in a future system. By implementing such a solution, the administration will be improved by attaining an easier and more effective way of managing the activities related to the CRSs and SRLs. There will also be an enhanced view and access of information through the proposed changes.

By using the extended ISO model, showing software quality characteristics, we investigated what important software quality characteristics a new administrative system ought to have. The end result from the investigation concerning the important software quality characteristics shows that there are two main characteristics, which should be especially considered when introducing a future system. These characteristics are functionality and maintainability A categorisation into sub characteristics reveals nine important characteristics to consider. Six of these are covered by the two main characteristics above. These characteristics are compliance, accuracy, analysability, changeability, manageability and security. The others are time behaviour, operability and explicitness.

We have now showed the changes of activities needed and the qualities characteristics to concentrate on, when implementing a new web based system; Therefore we now deem the purpose of this thesis is reached.

7.2 Personal Reflections

We would like to conclude this paper by giving some reflections we have made throughout the thesis work.

When we started to use SSM as our methodology a feeling of uncertainty overcame us. It was quite difficult to completely comprehend how to work with it. After a while we gained more and more understanding, making it easier to use, although we are still novice users of this methodology.

We are both entering the real world of working after finishing this thesis. It felt important to tune in on the users and in some way put them in the centre of the investigation by the use of SSM. We think it will help us not to forget them later on in life, when working with other projects.

We discovered some difficulties finding certain kind of references when needed. Therefore we had to use some indirect sources. This is unfortunate, but we consider this solution the best available.

One thing which was especially uplifting, was to get the feel of how planning for a new system could be carried out. When doing interviews and having discussions it was as if we were working staff conducting a study. To be able to do the thesis with a real company as one part gave us this experience.

Hopefully this thesis can be inspiring to others and it would be great if VCC chooses to continue on and implement a web-based system based on our conclusions.

As final words we would like to express that this thesis work has given us experience and knowledge in some areas we were inexperienced in previously. During the voyage we have had some difficulties, but we manage to ride out the storms and anchor in the land of enlightenment.

References

Books

Avison, D. E. & Fitzgerald G. (1995). *Information Systems Development: Methodologies, Techniques and Tool*s, 2nd Edition, Blackwell Scientific Publications, Oxford.

Backman, J. (1998). *Rapporter och uppsatser.* Lund: Studentlitteratur.

Checkland, P. (1989). Soft Systems Methodology. In J Rosenhead (ed.), *Rational Analysis for a Problematic World*. Chichester, UK: Wiley.

Hendriks, P.R.H. (2000). Quality of Software Development - On the Road to Total Quality of Information Technology. In J. van Bon, Hagen & Stam Publishers (Eds.), *World Class IT Service Management Guide 2000* (pp. 263-272), Den Haag, The Netherlands.

ISO - International Organisation for Standardisation. (1991). *ISO/IEC 9126:1991 - Information technology -- Software product evaluation -- Quality characteristics and guidelines for their use*. International Organisation for Standardisation, Geneva, Switzerland.

ISO/IEC JTC1 - Joint Technical Committee. (1997). Part 1 - Quality Characteristics and sub-characteristics. In International Organisation for Standardisation and International Electrotechnical Commission (Eds.), *CD 9126-1 - Information technology - Software quality characteristics and metrics*. Quebec, Canada.

Patel, R., Davidson, B. (1994). *Forskningsmetodikens grunder*. Lund: Studentlitteratur.

Articles

Leung, Hareton K.N. (2000). Quality Metrics for Intranet Applications. *Information & Management*, 38 (3), 137-152.

Mathiassen, L., & Nielsen, P. A. (1989). Soft Systems and Hard Contradictions – Approaching the Reality of Information Systems in Organizations. *Journal of Applied Systems Analysis*, 16.

Internet

BOSCH - Robert BOSCH AB. (2001). *http://www.BOSCH.se/* [WWW Document]. URL http://www.BOSCH.se/

ESSI-SCOPE. (1997). *ISO 9126: The Standard of Reference* [WWW Document]. URL http://www.cse.dcu.ie/essiscope/sm2/9126ref.html

Welch, D.E. (1996). *Intranets the New Engine of Corporate Productivity* [WWW Document]. URL http://www.welch.com/wpapers/intranet.html

Veldhuiozen, J, et al. (1999). *D11 - Applications scenarios and user requirement* [WWW Document]. URL http://www.txt.it/progetti/w3i3/d11.pdf

Volvo Car Corporation Intranet (2001). Multiple intranet pages.

Appendix A – Interview Questions and Answers

Out of security and consideration, we have decided to keep the participants figuring in the interviews and surveys anonymous. There is instead the following naming:

- Optimisers are named Opt1 and Opt2
- Software Coordinators are named SWC1 and SWC1
- The BOSCH representative is named BOSCHrep
- The Experts are named Expert1, Expert2 and Expert3

The interviews were made in Swedish and translated by us into English.

VCC Interviews

Can you give a description of your current assignments/activities at the department?
Opt1: I'm working with the protocols that set the rules for the communication between the engine and different diagnostic tools. It is important that we can achieve good diagnostic results when developing engines. We are talking about protocols for the signals transferred between engine and different diagnostic devices, and that the protocols are correct in relation to the different services that the data will serve. There are standards set by e.g. the government, so that governmental organizations can use their test equipment on VCC's engines. Also the mechanics at the different garages and VCC retailers must be able to use their equipment to diagnose failures in the engines performance.

Opt2: We did not ask this question.

SWC1: A big part of my work is that I act as a contact person between VCC and BOSCH. The contacts I mediate concern coordination of numerous different tasks. I also attend and arrange numerous different meetings on various levels. The meetings primarily concern coordination of different kinds.

SWC2: One part of my work is that I work towards Siemens regarding software for some of the car models engines. I also attend and arrange numerous different coordination meetings on various levels.

What is your role in the cooperation with BOSCH?
Opt1: When I find a problem with a protocol, I will report this to BOSCH in the form of a CRS.

Opt2: I create CRS

SWC1: I act as a contact person between VCC and BOSCH. The contacts I mediate concern coordination of numerous different tasks related to BOSCH. I also administer various things, including CRS handling, SRL handling and software delivery schedule. I also order software from BOSCH when needed.

SWC2: I am not directly involved in the BOSCH CRS handling, but working with similar tasks as SWC1 towards Siemens. I also follow SWC1 closely in his work, which gives me an insight into his work regarding the CRS.

Do you see any problems with the current situation and the way you perform your work, concerning the handling of CRS and SRL?
Opt1: At the moment I can't get any satisfying information about what has happened to the CRS after sending it to BOSCH. It is not possible to get a pleasant total view of the CRSs that I have sent. You have to look through the whole database where all CRSs from all departments are lined up after each other.

Opt2: I can't easily obtain information about when my CRS will be implemented in the code.

SWC1 and *SWC2:* A major problem is that it is complicated to get and hand out information. People in various positions, both at BOSCH and VCC have questions regarding CRS. Often they ask us, through e-mail, telephone or in person; then if we don't have the information they want, we have to contact someone, e.g. people at BOSCH. Sometimes they might be away, even for several days; the person asking for information then starts to wonder what happened with his/hers question.

Do you sense that you obtain enough and correct information, for performing a satisfying result in the work that you do?
Opt1: No. See the above question.

Opt2: No. See the above question.

SWC1 and *SWC2:* No. The information flow from BOSCH does not work satisfactory. Sometimes there is no information at all, and sometimes there is information overload, e.g. when a new program version is about to be released.

Do you sense that you have the ability to influence the work you perform in the department?
Opt1: Yes I do, but you might experience a form of inertia or slowness in the beginning of a transformation process, but this is to my mind a good quality.2

Opt1: We did not ask this question.

*SWC1*and *SWC2:* It is possible to give suggestions, at least at VCC. Your thesis work is a great example of an idea we had which now has been heard. It feels as if it a bit harder to get BOSCH interested in new ideas.

What type of activities would you want a future administrative system for CSR's to have?
Opt1: I would like my own specific place where I can have a better real time overview of the status of CRSs that only belongs to me. It could be information like; where is my CRS in the chain of activities; has the CRS been approved? in which software version is my CRS going to be included? I would see it as my own CRS inbox.

Opt2: I want to be able to obtain information about when my CRS will be implemented in the code.

SWC1 and *SWC2:* A great general improvement would to have better overview over the system. It should be easier to obtain information, without having to contact various persons each time there is a question or need for information.

The CRS handling needs to be arranged in a different way. More and better functions such as information regarding a CRS' status, estimated cost per CRS, estimated program version where a change will be included. Some kind of history log where there is possible to follow a CRS through all stages- what has happened with it from start until it is included in the program?

It would be great if the new system could include a kind of discussion area. In the discussion area it would be possible to have discussions and ask questions and receive answers.

BOSCH interview

Can you give a description of your current assignments/activities at BOSCH?
BOSCHrep: I am a so-called Group Leader. I have 3-4 people under me. My activities include software developing, software modelling, and to act as Activity Coordinator.

What is your role in the cooperation with VCC?
BOSCHrep: I act as a link between VCC and BOSCH in the form of translator between cultures and languages. Attend KU-meetings at VCC.

Do you see any problems with the current situation and the way you perform your work, concerning the communication with VCC and the handling of CRS, SRL etc?
BOSCHrep: CRSs sometimes tend to arrive late in the project just before a new release of software. The release of a new software version is set to a specific

date. Therefore BOSCH has a date set for when the last CRS must arrive if it will be a part of that release. There seem to be a problem with the communication of information. This lack of communication might lead to a crisis of confidence; quality problems and a “give and take” situation may arise.

We are also getting a lot of questions about the status of CRSs from personnel at VCC. It can be a little bit too much support over the telephone occasionally.

What are your thoughts about and interest in, VCCs proposition to change the current way the two organizations administer the communication with each other?
BOSCHrep: As a regional office placed here in Gothenburg, we are very interested in new ways of developing the communication between the participants in the projects.

What type of characteristics would you want a future administrative system for CSR’s to have?
BOSCHrep: A joint database would increase the structure of multiparty projects like this. It would help us structure the projects we are jointly working with. The coordination would increase with a better view of updated information e.g. deadlines, answers to CRSs, test results etc.

Expert Interviews

Can you give a description of your current assignments/activities at the department?
Expert1: My work is to supervise the development of the software for the car engine’s control unit, on the models having the Denso engine control unit. The working tasks include CRS handling, SRL handling, software delivery schedule, implementation planning and program testing co-ordination

Expert2: I am working with the administration of Oracle databases within 8295. We are running a database hotel for Volvo IT’s customers, who choose not to administer their databases themselves.

Expert3: I work as a part of LITS (Local IT Support). I am employed by Volvo IT, but work as local IT support at VCC. My work is to support a part of VCC in their use of IT-tools. Support means to offer competence such as; Local IT-support, installation, field support, print support, hardware service and planning of purchase and installation.

Question only to Expert1: Can you give a description of how system between VCC and Denso is structured (server placements, type of servers, firewalls etc)?
Expert1: The system is built with the web and database server placed outside both Denso and VCC. There is an IP control access from VCC through terminals at VCC.

Question only to Expert1: Can you give a description of the software used in the VCC/Denso system (browser, type of database etc.)?
Expert1: The selected platform for the SRL handling between VCC and Denso consists of a PostgreSQL database manger and an Apache web server on a Linux operating system. The database is an ANSI SQL database. The browser is Netscape with a 128 bits RC4 encryption.

Can you give information regarding how the physical connection/structure VCC – BOSCH could look like?
Expert1: Did not have enough information to answer this question.

Expert2: There are different possibilities when it comes to the connection between BOSCH and VCC. Volvo IT has very good turnkey solutions for this kind of systems. We have an Extranet, which is a complete set of security arrangements that makes it possible to in a secure way, connect external users to selected resources on the VCC intranet. The basic configuration is:

One part of the Extranet, which could be used, is the WSS (Web Security Service). This is an infrastructure component that enables access of web-based applications in a secure way from Internet to VCC's Intranet.

The main purpose of the WSS is to access web based applications or a simple web server located inside the VCC Intranet from the Internet. This is achieved in a more or less restricted way depending on the security level of the web application.

WSS serve web applications with the function web proxy, encryption, authentication and authorisation and enables users using a web browser outside VCC to use web applications inside VCN (Volvo Corporate Network). Depending on the security classification of the data, different authentication methods will be used.

Another possible connection solution could be to build a completely new separate system. This solution gives the possibility to freely put servers where you want. If VCC or BOSCH feels they have the need to store some data on their own side, on an own server, then this could be a way to do that. There are different ways to build this kind of system.

Expert3: If BOSCH agrees, then it would be a convenient solution to let Volvo IT handle the connection. It's possible to rent space on our machines and use our connections.

Choice of Software - What possibilities are there for a new system?
Expert1: One solution could be similar to what we are using in the Denso system. There is a PostgreSQL database manger and an Apache web server on a. Linux operating system. The database should be an ANSI SQL database, for compatibility and standard. The browser could be anyone, which has a good encryption support.

Expert2: If you choose to use the services of Volvo IT, then Windows 2000 will be the operating system. Oracle will be the database software. In our web hotel we run Microsoft IIS (Internet Information Server) as web server in the web runtime environment.

Expert3: Oracle would be my choice for database. Volvo IT has complete solutions for this kind of system. Another functional and cheap solution could be to use a Linux OS running some kind of SQL server and Apache web server.

Choice of Hardware - What possibilities are there for a new system?
Expert1: The quite similar system between VCC and Denso is running on an ordinary PC. This would probably suite the new system also.

Expert2: The hardware could be any PC with enough power and storage capabilities. It depends on how advanced the new system will be. If the system is hosted at Volvo IT, we have almost unlimited capabilities.

Expert3: This answer is like last the one. If Volvo IT is to provide with the hardware, then we have all necessary hardware available. Generally speaking, an ordinary PC should be enough to run this kind of system.

What is important to think about regarding the security for a news system?
Expert1: Did not have enough information to answer this question.

Expert2: We have very strict standard when it comes to security regarding trespassing into systems. Anyone who builds a system using the web should be very cautious. Nowadays there are good encryption standards, firewalls and proxy servers to choose from. Data security is another kind of security aspect. Data must be properly backed up maintained to secure data and system performance.

Expert3: Some important things to consider are user authorization and communication security. There has to be user account logins and some kind of encryption. A VPN (Virtual Private Network) between VCC and BOSCH sounds like good idea. A VPN connects Intranets together; allowing remote access to services as transparently and securely as the users locally connected to remote networks and hosts. It lets users access information on their corporate network (e.g. an Intranet) over a public network such as the Internet.

Appendix B – Survey Forms

User Survey Form: Ranking of Software Quality Characteristics - Usability

Name:
Position (Optimiser, Software Coordinator etc.):

Please rank the software quality characteristics below from your point of view-in your work related to the system. Use the scale 1-5, where a characteristic with the grade of 5 is uttermost important to consider according to you, when choosing software for a new system, and where a characteristic with the grade of 1 is not important at all. If there is characteristics which you do not understand, then please mark the grade "?".

Below is an explanation of the grade system:
5 = Uttermost important. This has to be of highest possible quality
4 = Important. Need to be of high quality
3 = Should be considered, but isn't a priority
2 = Not important. May, or may not be considered.
1 = No importance at all. No need to consider.

? = I do no understand the meaning of this characteristic

If you have any comments regarding a characteristic, please write it in the "Comments" field.

The situation: A web based system for administration of CRS is to be developed. The system will consist of three parts:
1. A database where all the documents are stored.
2. A web based interface to the database
3. Other web based functions, e.g. a discussion area where it would be possible to have discussions, ask questions, and receive answers.

The system will be used by BOSCH, both in Sweden and in Germany, and by VCC in Gothenburg. The database will be updated several times each day.

Usability Qualities

- **Understandability**: The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.

Grade:
Comments:

- **Learnability**: The capability of the software product to enable the user to learn its application. How easy it is to learn to use the software.

Grade:
Comments:

- **Operability**: The capability of the software product to enable the user to operate and control it. Examples of operability are consistency in user interface and self-explanatory messages.

Grade:
Comments:

- **Explicitness**: The capability of the software to state it's meaning to the user. Leaving no question as to meaning or intent. How straightforward it is to understand what the software is used for.

Grade:
Comments:

- **Customisability**: The capability of the software to be adjusted to the demands of the user.

Grade:
Comments:

- **Attractivity**: The capability of the software product to be liked by the user. Example of attractivity is the capability to support the satisfaction of latent user desires and preferences, through services, behaviour and presentation beyond actual demand.

Grade:
Comments:

- **Clarity**: The capability of the software state it's intended use. The clarity of making the user aware of the functions it can perform.

Grade:
Comments:

- **Helpfulness**: The capability of the software to support the user in the use of the software. Examples helpfulness are effective user documentation and help functions.

Grade:
Comments:

- **User-friendliness**: Attributes of software that bear on the users' satisfaction. How friendly the software is for the user to use. Includes factors as screen composition, vocabulary, application of colour and sound.

Grade:
Comments:

Expert Survey Form - Ranking of Software Quality Characteristics

Name:
Position:

Please rank the software quality sub-characteristics below from your point of view- in your work related to the system. Use the scale 1-5, where a characteristic with the grade of 5 is uttermost important to consider according to you, when choosing software for a new system, and where a characteristic with the grade of 1 is not important at all. If there is characteristics which you do not understand, then please mark the grade "?".

Below is an explanation of the grade system:
5 = Uttermost important. This has to be of highest possible quality
4 = Important. Need to be of high quality
3 = Should be considered, but isn't a priority
2 = Not important. May, or may not be considered.
1 = No importance at all. No need to consider.

? = I do no understand the meaning of this characteristic

If you have any comments regarding a characteristic, please write it in the "Comments" field.

The situation: A web based system for administration of CRS is to be developed. The system will consist of three parts:

4. A database where all the documents are stored.
5. A web based interface to the database
6. Other web based functions, e.g. a discussion area where it would be possible to have discussions, ask questions, and receive answers.

The system will be used by BOSCH, both in Sweden and in Germany, and by VCC in Gothenburg. The database will be updated several times each day.

Functionality Qualities

Functionality is the capability of the software to provide functions which meet stated and implied needs when the software is used. This characteristic is concerned with what the software does to fulfil needs, whereas other characteristics are mainly concerned with when and how it does fulfil the needs.

The sub-characteristics of functionality are:

- **Suitability**: The capability of the software to provide an appropriate set of functions for specified tasks and user objectives. Example of suitability is task-oriented compositions of functions from constituent sub-functions, capabilities of tables.

Grade:
Comments:

- **Accuracy**: The capability of the software to provide the right or agreed results or effects. This includes the expected data within the needed degree of precision of calculated values.

Grade:
Comments:

- **Interoperability**: The capability of the software to interact with one or more specified systems.

Grade:
Comments:

- **Security**: The capability of the software to protect information and data so that unauthorised persons or systems cannot read or modify them and authorised persons or systems are not denied access to them.

Grade:
Comments:

- **Compliance**: The capability of the software to adhere to (follow) application related standards, conventions or regulations in laws and similar prescriptions.

Grade:
Comments:

- **Traceability**: Attributes of software that bear on the effort needed to verify correctness of data processing on required points. The degree to which each element in a software development product establishes its reason for existing, for example, the degree to which the requirement and design of a given software component match.

Grade:
Comments:

Reliability Qualities

Reliability is the capability of the software to maintain its level of performance. The ability of the software to keep operating over time.

The sub-characteristics of reliability are:

- **Maturity**: The capability of the software to avoid failure as a result of faults in the software. How well the software can handle software faults without failing.

Grade:
Comments:

- **Fault Tolerance**: The capability of the software to maintain a level of performance in cases of software faults .

Grade:
Comments:

- **Recoverability**: The capability of the software to re-establish its level of performance and recover the data directly affected in the case of a failure.

Grade:

Comments:

- **Availability**: The capability of the software to be in a state to perform a required function at a given point in time. Availability can be assessed by the proportion of total time during which the software product is in an up state.

Grade:
Comments:

- **Degradability**: The capability of the software to remain functional even when interacting with older versions of other software. Example of this is a web page which remain functional even when viewed by older browsers or browsers not preferred.

Grade:
Comments:

Efficiency Qualities

Efficiency is the capability of the software to provide the required performance, relative to the amount of resources used. The ability of the software to respond with appropriate speed to a user's requests.

The sub-characteristics of efficiency are:

- **Time Behaviour**: The capability of the software to provide appropriate response and processing times and throughput rates when performing its function.

Grade:
Comments:

- **Resource Behaviour**: The capability of the software to use appropriate resources in an appropriate time when the software performs its functions.

Grade:
Comments:

Maintainability Qualities

Maintainability is the capability of the software to be modified. Modifications may include corrections, improvements or adaptation of the software to changes in environment, and in requirements and functional specifications. The ability to make changes quickly and cost effectively in the software.

The sub-characteristics of maintainability are:

- **Analysability**: The capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified or to be identified.

Grade:
Comments:

- **Changeability**: The capability of the software product to enable a specified modification to be implemented.

Grade:
Comments:

- **Stability**: The capability of the software to minimise unexpected effects from modifications of the software. See Changeability.

Grade:
Comments:

- **Testability**: The capability of the software product to enable modified software to be validated.

Grade:
Comments:

- **Manageability**: the structure of the software should be simple and straightforward in order to keep the software manageable.

Grade:
Comments:

- **Reusability**: Attributes of software that bear on its potential for complete or partial re-use in another software product.

Grade:
Comments:

Portability Qualities

Portability is the capability of software to be transferred from one environment to another. The environment may include organisational, hardware or software environment.

The sub-characteristics of portability are:

- **Adaptability**: The capability of the software to be modified for different specified environments without applying actions or means other than those provided for this purpose for the software considered. Adaptability includes the scalability of internal capacity (e.g. screen fields, tables, transaction volumes, report formats, etc.)

Grade:
Comments:

- **Installability**: The capability of the software to be installed in a specified environment.

Grade:
Comments:

- **Conformance**: The ability of the software to act or behave in correspondence with current customs, rules, or styles.

Grade:
Comments:

- **Replaceability**: The capability of the software to be used in place of other specified software in the environment of that software.

Grade:
Comments:

Appendix C – Complete Characteristics Comments

Out of security and consideration, we have decided to keep the participants figuring in the interviews and surveys anonymous. There is instead the following naming:

- Optimisers are named Opt1 and Opt2
- Software Coordinators are named SWC1 and SWC1
- The BOSCH representative is named BOSCHrep
- The Experts are named Expert1, Expert2 and Expert3

The characteristics comments were made in Swedish and translated by us into English text.

Functionality Qualities Characteristics

- Suitability

Expert2: In most solutions this can be adapted depending on needs.

- Accuracy

Expert2: This is what the customers pay for. This needs to be at highest quality to gain the customer's confidence in the system.

- Interoperability

Expert2: There are standard from Volvo IT, which works. If a certain standard is followed, this should work quite flawlessly.

Expert3: Would be good to consider well if there is a plan to extend the system and work with other systems, and software.

- Security

Expert2: This is question of trust between solution provider and the customers. If there is no belief in the security of the system, it will probably not be used.

- Compliance

Expert2: This is a bit tricky to estimate. The software has to comply to a certain extent, or else it will be used at all! Volvo IT has a certain routine to make sure a software product complies with all relevant standards. There is first a design review before coding, and then there is an operational review after implementation, but before the software is used.

Expert3: One assumes that the software follows standards and other regulations. It will not be possible to use the software if it does not follow at least minimum rules and standards.

- Traceability

The participants gave no comments.

Reliability Qualities Characteristics

- Maturity

The participants gave no comments.

- Fault Tolerance

The participants gave no comments.

- Recoverability

Expert1: Looking to the database I see this capability more as a hardware question, then a software characteristic. Data should be backed up often enough to be able to recover the lost data.

Expert2: This looks like a quality, which isn't directly related to the software itself. It is more a about background functions, related more to hardware and data security functions.

- Availability

Expert1: It's hard to get 100 percent up time, not least in a web based system where the internet connection itself might go down, completely unrelated to the software. It might be a problem to find a long enough time slot when backing up data if the system is used 24/7.

Expert2: This also looks like a quality, which isn't directly related to the software itself. This is more a question about its possibility to work with other parts of the system.

- Degradability

The participants gave no comments.

Efficiency Qualities Characteristics

- Time Behaviour

Expert1: A simple thing as a form loaded from the server might cause bad time behaviour if it is too big. One idea could be to use client based forms.

Expert2: Consistent behaviour is important. The users are expecting speed! This is a thing, which might become quite annoying if the user has to wait longer than he feels is acceptable.

Expert3: In this kind of systems software (e.g. browser) time behaviour often depends on external factors such as Internet bandwidth.

- Resource Behaviour

Expert1: This is a question about smart code vs. computing power. If the code is good, there will be a lower use of resources. If there is enough resources, the software can be less efficient without causing trouble.

Usability Qualities Characteristics

- Understandability

SWC2: It would be strange to work with a program if you do not understand its capabilities.

- Learnability

The participants gave no comments.

- Operability

SWC1: To be able operate a program correctly is very important, or else it would be frustrating in the long run.

Opt2: Of course you expect to have control over what you do!

- Explicitness

SWC1: This feels quite close to operability. You need to feel that you have control over what can be done with the software.

- Customisability

Opt1: This is an important factor to me! I would like to customise the way my "work space" looks like.

- Attractivity

SWC2: This is not important to me. I don not care too much about the bells and whistles of the software. It's much more important to have great functionality.

- Clarity

The participants gave no comments.

- Helpfulness

The participants gave no comments.

- User friendliness

The participants gave no comments.

Maintainability Qualities Characteristics

- Analysability

Expert2: This is important if there is a cause to believe that the software will be used for a longer time span. The software may need change in the future. Sometimes licenses are expired and support is both expensive and hard to get-then if there is a good analysability, it will be easier to maintain the system. This applies to maintainability qualities.

- Changeability

Expert2: Look Expert2's "Analysability Comments".

Expert3: This is an important quality to concentrate on if you wish to have a future proof product.

- Stability

Expert2: Look Expert2's "Analysability Comments".

- Testability

Expert1: There is software, which could be used to manage and thoroughly test software changes. An example can be found at http://www.mccabe.com.

Expert2: Look Expert2's "Analysability Comments".

- Manageability

Expert1: A simple and straightforward structure is very helpful for all kinds of maintainability needs.

Expert2: Look Expert2's "Analysability Comments".

- Reusability

Expert2: Look Expert2's "Analysability Comments".

Portability Qualities Characteristics

- Adaptability

The participants gave no comments.

- Installability

The participants gave no comments.

- Conformance

Expert2: The software need to behave the way the user expects, or else unnecessary annoyances will occur.

- Replaceability

The participants gave no comments.

Appendix D – CRS Form

VOLVO
Volvo Car Corporation

Change Request

Registration no: XXX **Issue:**X **CR status:** Ended

Subject: CARB on K, Header byte
Project: 12345 **Release week:** P26 PTO 9950
Issuer: 98765 First Surname 1234567 **Creation date:** 99-04-23
Reason for CR: Error description **Reason, SW/HW/Spec:** For SW

DESCRIPTION:

ECU: XXX VB **SW issue:** C2.2.3.1 **HW issue:**

Change Request/Fault description: Fault Description goes here

.

ANALYSIS:

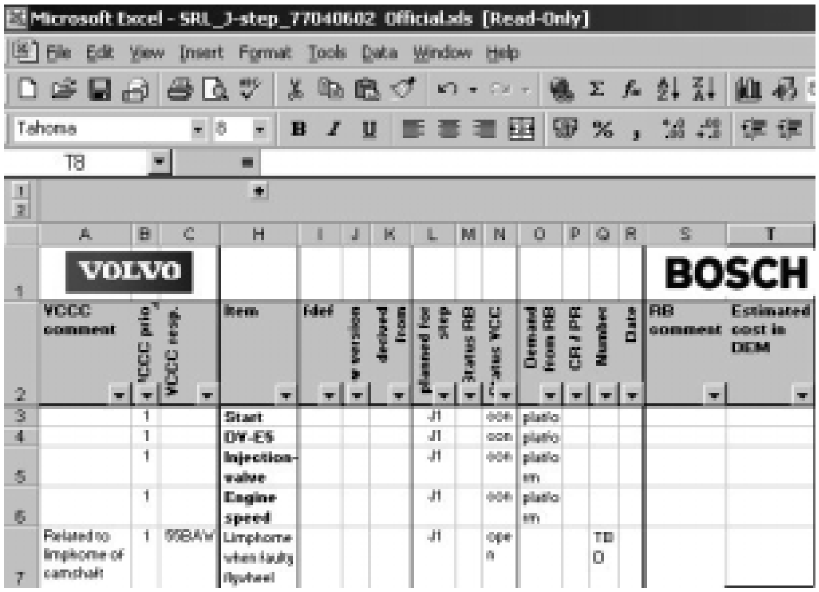
Measure proposed:

Comments:

KU approved:

Impl. in issue: F3 **Estimated effort:**

Appendix E – SRL Form



	A	B	C	H	I	J	K	L	M	N	O	P	Q	R	S	T
2	VCCC comment	VCCC prio	VCCC resp.	Item	Fdef	w version	devlored from	planned for step	Status RB	Status VCC	Demand from RB	CR / PR	Number	Date	RB comment	Estimated cost in DEM
3		1		Start				J1		oon	platfo					
4		1		DY-ES				J1		oon	platfo					
5		1		Injection-valve				J1		oon	platfo m					
6		1		Engine speed				J1		oon	platfo m					
7	Related to limphome of camshaft	1	55BAW	Limphome when faulty flywheel				J1		ope n			TB D			