Relative Indicators
for
Success in Software Development

Jacob Bylund
Niklas Lundin
Pär Waldemarsson

School of Economics and Management
Lund University
Department of Business Administration
Box 7080
220 07 Lund

Lund Institute of Technology
Lund University
Department of Communication Systems
Box 118
221 00 Lund

ISSN 1404-1286

KFS i Lund AB
Lund 2001
Printed in Sweden

Abstract

Title: Relative Indicators for Success in Software Development

Authors: Jacob Bylund, Niklas Lundin and Pär Waldemarsson

Supervisors: Tore Eriksson, Department of Business Administration
Per Runeson, Department of Communication systems
Rikard Roth, Tribon Solutions AB

Problem: In the past, two different ways of valuation of the development process have been used, either an external or an internal valuation. The problem with both is that they do not take the other viewpoint into consideration. When valuating the internal activities the connection to the market demands, strategic issues and competitors are scarce. On the other hand, when valuating solely on external factors, the internal capabilities and the possibilities to fulfil new market demands are not considered.

Purpose: The purpose is to extract relative indicators for success in software development where strategic and operational issues are considered in one valuation. The purpose of this valuation is to compare different software developing companies with one other and to visualise the evolution of those indicators.

Method: An initial investigation of the development process at TS served as the starting point for the master thesis. The second step was the development of the indicators describing the development process. In the initial task a quantitative method was used and the indicators were then developed by the use of a qualitative method. Finally, a questionnaire was created to test and confirm our indicators.

Conclusions: The main conclusion of this master thesis is that the indicators, and the supporting reference models, do reflect internal and external factors and the interfaces between the market, the strategy and the development process. The final conclusion is that this master thesis highlights several research areas where much is still unknown to the software community and therefore these indicators have to be further refined and developed.

Keywords: Key success factors, Software development, Resource planning, Organisational structure, Information flow, Specification process.

Preface

This report is the result of a master thesis in the educational programme of Technology Management at Lund University. The report was written through the collaboration between the Institute of Business Administration and the Department of Communication Systems at Lund University.

We would like to use this opportunity to show our sincere gratitude towards those people who have made this master thesis possible. We would like to thank our supervisor at Tribon Solutions AB, Rikard Roth, for his commitment to the underlying activities. The assistance of Tribon Solutions AB is gratefully acknowledged as they contributed by providing company relevant information and the ability to give us the required amount of time for our surveys. Additionally, various discussions with Björn Regnell at Lund University and Peter Wittfjord at Tribon Solutions AB have contributed to the development of the reference models.

Finally, we would like to thank our supervisors at Lund University, Tore Eriksson and Per Runeson, who both have provided us with fast and useful feedback during the work with this master thesis.

Lund, August 2000

Jacob Bylund Niklas Lundin Pär Waldemarsson

Content

1 Introduction

> There are those who look at things the way they are, and ask why… I dream of things never were, and ask why not?
>
> - Robert F. Kennedy

This master thesis represents a fairly new emphasis in software engineering and business administration. It is based on the premise that software engineering has to consider strategic issues and concerns, as well as business administration has to consider operational aspects of software development. To give the readers a common base for understanding on these matters the background the problem and the purpose of this master thesis are presented in this chapter.

1.1 Background

The movement from the industrial era into the era of information technology has not gone unnoticed. Many companies have moved their focal points from selling physical products to selling knowledge and services. The company's value has therefore shifted from their tangible assets, including machines, buildings and stocks, towards their intangible assets, such as competence of their employees, processes and brand name.[1] Traditional methods have been developed to estimate a company's tangible assets, which hence can be accounted for in the balance sheet.[2]

Companies in the software industry now exist in an environment more turbulent than ever before. To be able to compete in these fast changing environments companies have to develop smart processes that are easily adapted to the changing environment[3]. These processes must have a built in ability to learn and to apply this new knowledge when structuring the processes. In other words: The ability of the company to implement new and better processes. There is no general way to evaluate the productivity and the quality of a certain process or how well performing a company is when it comes to recognise change and adapting to it.

> *"Corporate management accounting systems are inadequate for today's environment. In this time of rapid technological change, vigorous global and domestic competition, and enormously expanding information processing capabilities, management accounting systems are not providing useful, timely*

[1] Edvinsson 1997
[2] Ibid
[3] Business Process Reengineering

information for the process control, product costing, and performance evaluation activities of managers." [4]

Despite the challenges of increased productivity and quality, and the introduction of new software development processes, most software projects continue to run into schedule delays, cost overruns and quality problems. In an environment like this, methodologies, tools, management techniques and insights that lead to improved software productivity and quality are of fundamental interest.

The foundation of this master thesis is the scarce knowledge in the area between the business administration sector and the software engineering field. The curricula of business administration tend to evaluate the software developing companies on what is visible to the customers. Likewise, the software engineering field focuses on different methods and processes for the attainment of the goals. These two objectives are rarely united, even though they are actually two sides of the same coin.

Our education, Technology Management, is intended to bridge the gap between business administration and engineering practices. It is a joint venture between School of Economics and Management in Lund and Lund Institute of Technology. The idea of this master thesis stem from this joint venture and was initiated by one of our supervisors, Per Runeson, and the 6th AP-fond in Sweden. The master thesis was carried out at the software developing company Tribon Solutions AB (TS), which is owned by the 6th AP-fond.

1.2 Problem

New times demand new solutions and the analysts need new models to value software developing companies.[5] The fundamental question in the field of valuation of these companies is how the companies manage their development processes. In the past, two different ways of valuation of the development process have been used, either an external or an internal valuation. The problem with both the internal and external viewpoints is that they do not take the other viewpoint into consideration.

According to a recent study supported by the Royal Swedish Academy of Engineering Sciences (IVA) there are seven areas to examine in order to value fast growing companies. The seven areas are; level of consolidation, growth, barriers of entry, trademark, management, innovation and dependence on key-personnel[6]. This approach of performance valuation is much like the scoreboard in a football game. The scoreboard might tell whether a team is winning or losing a single match, but tells little of why the team is winning or losing. Furthermore, the scoreboard does not tell what the team is doing right or wrong in carrying out the tactics or the strategy of the entire season. If the software developing company only relies on the scoreboard, then the company will not be successful in the future market.

[4] Johnson 1987
[5] Lindvall 2000
[6] Frykman 2000

Different models have been developed to capture the internal key processes in organisations. They often concern the gathering of different metrics to improve their software engineering management practices[7]. The most widely known of these models is the Capability Maturity Model (CMM). This valuation approach is much like to valuate each football team without considering if they are on the right arena. If a team tries to play football by relying on the internal valuation it might be successful in different situations but it is not a prerequisite to win the game. CMM does not consider what to develop only how it is achieved and thus disregard the strategy of the company. Furthermore, might models like CMM be too bureaucratic to be effective for smaller companies.

1.3 Purpose

In the problem formulation two areas appear to be central to this master thesis. The first is how a company can manage the development process, and the second is how the strategic issues are connected with the development effort. By the identification of these two questions the purpose was set.

The purpose is to extract relative indicators for success in software development where strategic issues and operational issues are considered in one valuation model. The purpose of this valuation model is to benchmark different software developing companies with one another and to visualise the evolution of those factors.

1.4 Limitations

To avoid any misconceptions and to furthermore describe the research area, we would like to comment different areas not examined in this master thesis.

While this master thesis has its foundation in development processes in the software industry, we have not addressed the question on how the companies choose their development process. These issues are considered determined and are not questionable. We will not either address the motives to adopt different goals and systems for their evaluation. These are to the same extent not questionable.

As the approach is to focus on the supporting activities of the development process a discussion on the different steps in a generic software development process are not included in this master thesis. However, the connections between the supporting activities and a generic development process will be discussed where it is appropriate.

It is not our intention to give a fundamental theoretical study on process orientation or development of different evaluation systems. We will not analyse different reward-systems or how these reward-systems should be developed in a process oriented organisation.

[7] http://www.sei.cmu.edu/managing/managing.html 2000-04-10

We will, furthermore, specifically exclude the initial phases of software development including the development of basic technologies and market strategy. We assume that the company begins with a mission statement identifying a target market and the core benefit the product will provide to that market.

Additionally, we will not cover important aspects of software development, such as specific programming languages and supporting software to the development process. This is a consequence of the fact that specific implementing technologies and methods are replaced much more rapidly than the employees are.

1.5 Structure

This part is intended as an illustration of the structure of the report. In the next chapter a description of the method used during this master thesis is explained and ends the first step and the problem definition. The remaining steps made during the development of our master thesis are supported by the structure of the report. The left part of Figure 1-1 below describes each step and the supporting chapters are shown in the right part of the figure.

Problem: The basis of the master thesis is described and the underlying method is outlined.

Description of a real case: This step includes an investigation and an analysis of an existing development process.

Reference models: Our former knowledge together with the prior analysis transforms into a reference model.

Theoretical support: The reference model and its perspectives are supported by theories. Supporting indicators are developed.

Analysis: By using the indicators on an actual development process an analysis is accomplished.

Conclusions: A discussion on the findings of our research and future work and enhancements concludes this report.

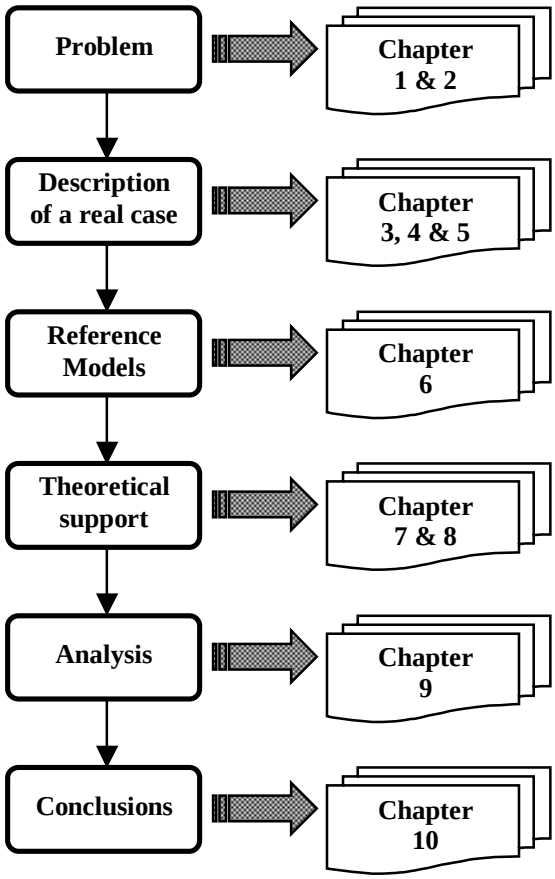


Figure 1-1

From our point of view, we would like to assure our readers that if you have read this far into the master thesis, the battle is half won. You are looking for answers to your questions. You have questions; the deficiency is in having none. Challenge yourself with challenging ideas.

2 Method

> There are three ways of doing things: The right way, the wrong way and "the army way".
>
> - U.S. Army[8]

We initially faced two different problems regarding the software development process. As the definition of our task evolved the two problems grew together to one task. We were to examine the internal software development process and thereby create a method for an external evaluation of this process. The aim of this chapter is to give the reader an understanding of the methods and problems we have faced during the development of this master thesis.

2.1 Initial Remarks

As we will discuss regarding the development processes in software development, the development of this master thesis can not be characterised by a particular process. The process described in this chapter is a combination of a predefined process and one that are revealed at the end of the project.

We start this chapter with a discussion on different views of research that were considered prior to the start up of the project. We then discuss the research approach that has been used in this master thesis. Finally, we discuss some criticism that can be applied to our research and the sources we have used.

Whether we tend to build hypotheses from experience or books, we all tend to have our own private assumptions on how things work and how to get things done. If the theories stem from our prior knowledge then, according to Whittington, it is what Argyris calls "theories of action"[9]. We will thus mention that the theories from our past studies are affecting the results in this master thesis. In chapter six we confront the aspects that we have found to be related "theories of action" in more detail.

2.2 Three ways of research

To create a solution to the task we faced, we first had to create a common vision or belief of how we would accomplish this. To discuss this further, we would like to point out three different scientific views of the world: positivism, hermeneutic, and

[8] According to the motion picture: The general's daughter
[9] Whittington 1993

the systemic view. Each of these views presents its own way for individuals and systems to manage the environment.

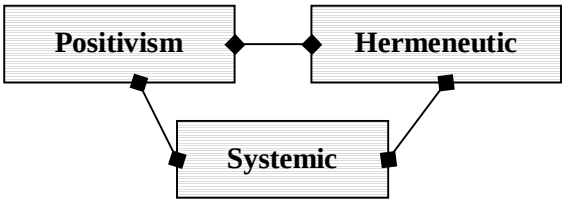


Figure 2-1: Three underlying views of the world.

As depicted by Figure 2-1, although we argue that it is a matter of different viewpoints, we indicate that there are historical and conceptual similarities between them.

2.2.1 Positivism

Comte is considered to be the father of positivism[10]. In the word positive he included such expressions as preciseness, security and reality. This was to him very different from the metaphysics' religious, idealistic and unfounded speculations. The social science should not speculate in matters that are not real or visual. Science has to be verified by empirical data.

The positivists say that cause and effect should be investigated in all science. Natural science as well as social science follows these laws[11]. The critics say that if this were true in social science, that you should base all your science on empirical data, all the results in research would be unavoidable and decreed by faith[12]. They say that there do not exist laws in social science as in natural science and that mankind can change what once has been established and thereby create their own future.

2.2.2 Hermeneutic

The main purpose of hermeneutic is to interpret and understand. The main difference from the positivistic way of thinking is that the spokesmen of hermeneutic mean that you have to make a difference between physical and social phenomena[13]. The hermeneutic way of thinking is also described as an historical viewpoint and is often connected to the qualitative theories[14].

[10] Lundahl 1999
[11] Ibid
[12] Halvorsen 1992
[13] Ibid
[14] Lundahl 1999

2.2.3 Systemic view

A third discipline of research, called the systematic view, has become more widespread, especially in recent decades. This theory of research accepts the positivists thinking that every cause has an effect. They think though that the effects of one's actions do not have to be visible in a near future and that these effects do not have to be a direct reaction of the cause. They say that you get different results depending on where and when you conduct your research, i.e. in which system you conduct your research.

2.2.4 Our opinion of these research theories

If you examine these different theories they are all quite extreme. There are, however, aspects that are acceptable in all of them, but it is the researchers own experiences of the world that determines how he or she views a certain problem.

We hope to work on the golden middle way of these theories, but perhaps closer to the hermeneutic and systematic ways of thinking. We do believe that there is a difference between social and natural science. People are constantly changing because they live in a constantly changing environment. For example, in business administration the theories have changed a lot the last hundred years from Taylor thoughts of efficiency in production to today's theories about efficiency in the market, i.e. focusing on customer demands. Taylor developed a theory that he thought was right in his society, in his period of time.

2.3 Our model of research

2.3.1 A qualitative/quantitative case study

To be able to create results that correspond with the purpose of our report we have chosen to do both a qualitative and a quantitative case study. The first task in our assignment is to investigate the software development process and we think that this will be done best with a qualitative method.

The target for this first task is to gain knowledge and understanding of how the development process is shaped at TS today. Eneroth, university lecturer at the School of Economics and Management in Lund, agrees to this when she argues that the target with a study is to gain knowledge and understanding of a real event, a qualitative method is then the best way to approach the problem[15].

It would have been preferred to use a qualitative case study when we where about to develop the indicators but time limits prevented us from doing so. The quantitative research method though is to prefer when it comes to statistical research of "non-living" instruments[16]. You can measure feelings, conceptions and values with a

[15] Eneroth 1993
[16] Merriam1994

quantitative method, but then you have to translate these factors into numbers and there is a risk that you loose the essential content of a feeling or a thought of the source.

2.3.2 Deductive vs. inductive approach to research

Deductive research means that the researcher from a given amount of sayings outlines new hypotheses.[17] As the hypotheses are created they are tested in the empirical context. A theory is never complete as long as the researcher can ask oneself the question ”why?”

Induction on the other hand means that the researcher draws general conclusions based on empirical facts.[18] There are much material that affect the empirical study and that is why the researcher never can be a hundred percent sure that his/her conclusions are correct.

In a researchers daily work the boundaries are not as distinct as stated above. During the work on this master thesis these issues have been recognised throughout the entire process from purpose statement to conclusion.

2.3.3 Course of action

We have in our research drawn conclusions from both empirical and theoretical studies and we therefore use both a deductive and an inductive approach. In this section our course of action is outlined and the transitions between these approaches are illustrated. This section also clarifies the importance of our theoretical and empirical studies.

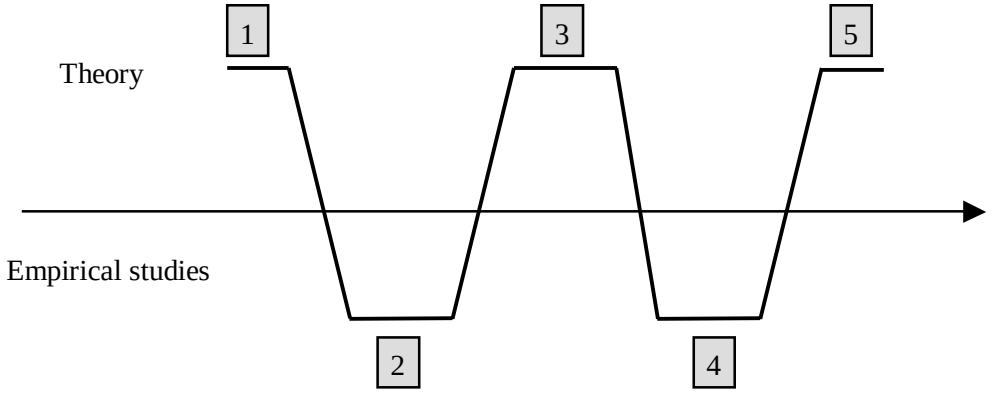


Figure 2-2: Course of action

As depicted in Figure 2-2, our course of action constitutes of five different steps and each step is now briefly described:

[17] Holme 1991
[18] Ibid

1. The first step in our course of action was to become more familiar with the software developing industry and the theories surrounding it, i.e. the methods that they use, the future of the markets etc.
2. The next step was to do an empirical study of our company, TS, and especially the software development process. To become more aware of the company's situation we conducted several qualitative interviews with key employees in the organisation. The interviewed employees are presented in the references and the initial open questions in appendix A.
3. By the use of available theories and our empirical studies three reference models were created in step 3. From the developed reference models applicable theories were chosen and investigated. These theories in combination with our reference models formed the basis for our indicators and their definitions.
4. In step 4 we made a survey in the form of a questionnaire to the personnel that where related to the software development process. The questionnaire was intended to confirm our indicators on an empirical level. The questionnaire and the results from the questionnaire are presented in appendix B.
5. Theoretical and empirical conclusions of our research.

2.3.4 Our secondary data gathering

Internet has been an important information source in this thesis. On the Internet we have searched for papers and working essays from universities and organisations around the world.

Books from courses in Technology Management have also played an important role in this thesis. It's quite natural that a thesis should be influenced by our past curriculum and we have found it very important that the foundations of our education should be reflected in our master thesis.

Further on we have had access to the library at the department of Telecommunication. From this library we have gained a great amount of literature in the field of software development.

We have also been given admission to TS internal library where we had access to books concerning the shipping yard industry.

2.3.5 Our primary data gathering

> *"The danger of too much flexibility are just as obvious as the truth in the argument that to give to everybody shoes of size 8 is to give the same thing to everybody, yet with different effect."*[19]

With this quote Galtung describes the biggest difference between a quantitative and a qualitative interview. The quantitative interview gives the same instructions to

[19] Galtung according to Holme 1991

everyone interviewed but the result among the persons interviewed differs. Of course the qualitative interviews have the problem that the interview can become too flexible. The advantage with this kind of interview is that it is more similar to a daily conversation, which makes the person more comfortable in the interviewing situation. The interviewer though, must have control over the situation and guide the "conversation" around his chosen themes[20].

We have conducted our interviews as qualitative interviews. We created a structure for the interviews[21] and from that structure we had a discussion with the person interviewed rather than a direct interview. With the assistance of our supervisor, Rickard Roth, we selected nine persons to interview from middle management and up. To get a better overall view of the software development process we chose people from different parts of the organisation.

The interviews were conducted by at least two of us, where one of us was the interviewer and the other was the primary interview secretary. Afterwards the secretary undertook the processing of the results from the interviews. The interviews were prepared by the sending of comprehensive questions to the interviewees. We sent these questions prior to the interviews, so that they knew the sorts of topics that the interview would cover. The questions were quite open and designed to let the interviewees start thinking of the different activities performed and which routines that were followed. The questions covered a number of topics including organisational policies, product and software development.

We have also conducted a quantitative survey among the software developers at Tribon Solutions AB's main office in Malmö, Sweden. The survey was conducted as a questionnaire that consisted of 59 questions that concerned the development process.[22] The questionnaire was handed out to 35 of the 40 software developers at the Malmö office. Due to vacation and other circumstances the remaining 5 employees were not available. Out of the 35 questionnaires handed out, 23 were returned and analysed.

2.3.6 From thorough to basic

It is hard to correctly define the development process in an organisation and people tend to have different opinions regarding what issues that are included. These different opinions introduced problems for us at an initial stage and made it hard to focus on the real problems in the development process. Additionally, we were influenced by the ambition to make a thorough investigation of the development process by investigating issues like lines of code, actual project time and customer satisfaction aspects.

As we continued to investigate different development models and development processes we realised that we needed to develop our own reference models. The

[20] Galtung according to Holme 1991
[21] Further information about the structure for the interviews are found in appendix A.
[22] The questionnaire and the result of this questionnaire are shown in appendixes B1 and B2.

problems of the former models are that they are either focused on process technology or management. Our belief is that they are actually the opposite sides of the same coin. While our developed models were connecting the strategic and operational levels of the organisation we recognised a need for a basic focus on the development process.

The basic approach and the creation of the reference models gave us invaluable insights into a software development process. It gave us the opportunity to reflect on the interconnections between strategic and operational levels including different activities in each level. Olsson supports this approach in his work on product development and argues that this approach is motivated in the definition and description of the activities in product development[23].

2.4 Criticism of the sources

As common with projects requiring the co-ordination between people, with different priorities and objectives apart from those of the project itself, the project stumbled at the interviewing stage. Our group had been keen to work to a tight schedule. It had been hoped to be able to go from the initial briefing, through to a completion of the interviews in a three week period: And completion of this first stage project within one month of that.

Unfortunately various people were unavailable at the pre-arranged interview times, and the period for this activity had to be extended. The Market and Sales Manager was the hardest to get hold of.

Some criticisms can also be made to the fact that we did not gain access to the company's Intranet until late in our research. An earlier access would have given us a more thorough view over the company and our study a better foundation to stand on in our initial analysis.

[23] Olsson 1976

.

3 Software for the shipbuilding industry

> The question is, then, do we try to make things easy on ourselves or do we try to make things easy on our customers, whoever they may be?
>
> - Erwin Frand

To have an initial understanding of the demands set on the development department at Tribon Solutions, TS, a brief description of the evolution of the industry, the company and the products is presented in this chapter. The chapter begins with discussing the market demands and continues with describing the background and development of TS. The final part of the chapter discusses TS' products, and different competitors that TS faces.

3.1 Market demands

The European ship builders have been facing declining order books and a fear of bankruptcy. Some shipyards are getting profit from their construction, but too often they are constructing the vessels at a loss.[24] In Western Europe, Germany is the only strong and profitable shipbuilding force remaining. However, the condition for survival has changed over the past years and causes many shipyards to suffer.

There is a growing trend that many ship owners prefer to build their ships in the Southeast-Asia. The Western European shipyards fear this competition and are thus demanding political and financial help to keep them alive.[25] They argue that the competition, from especially South Korea, is unfair and is built on government aid.

But even if South Korea would cut down governmental aid, the situation would not be that comfortable to the Western European shipyards. The shipyards in Poland and China are getting stronger and Japanese shipyards have a strong position on the market. Figure 3-1 illustrates this situation, where South Korea and Japan have an outstanding market share and where China is becoming a strong competitor.

[24] Trade Wind 2000
[25] Fairplay 2000

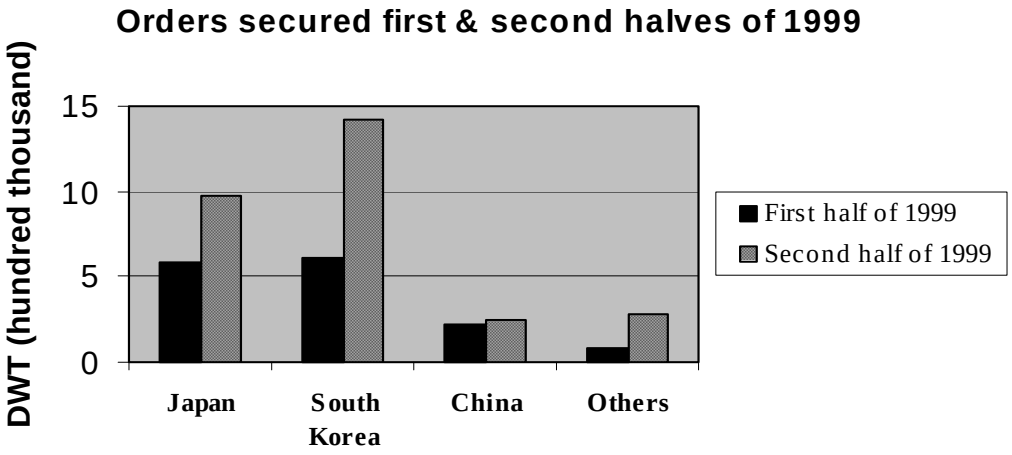


Figure 3-1: Values from Fairplay, February, 2000

Other indicators of the customer's market are the report "Newbuilding Requirement 2000-2015", by MSR-consult in Denmark[26]. This report states that the newbuilding requirements of commercial ships ought to continue on the present level until 2005. Beyond 2005 the market will increase by a factor of 25% by 2015.

The situation in China indicates another future demand. The ship owners are dissatisfied with the Chinese shipyards constantly delivering past due date. There is today a risk that some ship owners will give up their focus on cheaper prices in favour of guaranteed delivery dates. This accentuates the importance of efficient productions systems, instead of concentration on inexpensive workforce.

One indicator of the market development for computer support in the shipyard industry is the recently presented research programme "Shipbuilding and maritime technology for the 21st century". The main targets of the research programme are to reduce the design phases, standardise the construction parts and invent new assembly technology. The new programme includes the development of software-based tools for production. [27]

3.2 Company background

As a result of the crises in the Swedish shipyard industry during the 1970s only two parts of the shipyard Kockums in Malmö succeeded to survive; the production of vessels for defence purposes and a software company.

The software company referred to is Tribon Solutions, TS, and perhaps this is the company that will lead the Swedish traditions of the shipping industry into the future.

[26] Fairplay 2000
[27] Ibid

TS is the world leading developer of software in the area of construction systems for the shipbuilding industry. Systems developed by TS support the process from cost estimation and vessel design to logistics and assembly. TS possesses the thorough knowledge demanded by the shipping industry, including the requirements stated by the shipyards' customers, and the underlying reason is the company's historical roots inside the industry.

TS' head office is located in Malmö, Sweden, and the company is furthermore represented in the United Kingdom, Germany, China, Japan, Russia, Singapore, South Korea and the United States. The sole owner of the company is the 6th AP-fond, one of the seven Swedish National Pension Insurance Funds. The company has a total of 185 employees around the world. The number of employees in development and research departments is 65 and they are located in Sweden and in the United Kingdom.

3.3 Company development[28]

The idea to develop software for the shipyard industry was formed during the 1960's. The main purpose was to facilitate the entire shipbuilding process from conceptual design to manufacturing and assembly of different types of vessels. In the beginning this development was solely for internal use at Kockums Shipyard, but due to a growing interest from a number of shipyards in Europe at the end of the 1970's, the software was sold externally as well.

The first practical version of the software, STEERBEAR 1, was introduced in the middle of the 1960's and in the first half of the 1970's the second generation, STEERBEAR 2, was introduced. The second generation was in addition able to handle hull production activities, descriptions of piping systems and also information linked to a management information system.

The development of the shipyard system continued during the 1980's and through the acquisition of all rights to a complementary system, the AUTOKON system, in 1988 the system was furthermore enhanced. By acquiring the BMT ICONS and the rights to the marine design software systems of British Maritime Technology Ltd in 1994 more knowledge in defining hull forms and hydrostatic and hydrodynamic knowledge was gained.

As the development continued in the 1990's and the acquired units were merged into TS a new generation of the system was created. The present generation is called Tribon and is a combination of the old systems and new technology, including support for the Windows NT operating system.

Since the beginning of 1994, the number of customers using Tribon has increased from about 20 to the present number of 250. The systems developed by TS have

[28] Internal material describing Tribon

proven to be the most widely used system in the ship design and building industry. At the beginning of 2000 around 35% of all commercial ships built in the world were designed and produced by production systems using a TS system in one way or another.

3.4 The Tribon system

The Tribon system consists of an Application Programming Interface, Design applications, Design Management and Production Engineering as showed by Figure 3-2.

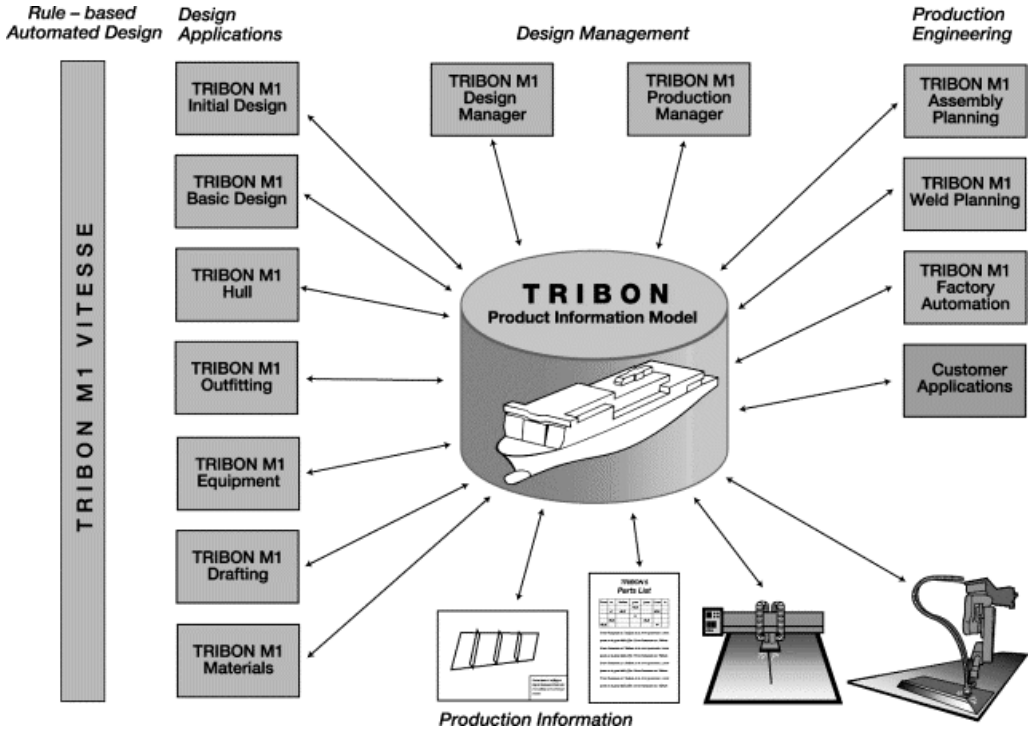


Figure 3-2: The TRIBON system

The Application Programming Interface, Vitesse, enables the users to write programs to control design development according to their own rules with direct access to the Tribon Product Information Model. The Product information Model database is the heart of the system.

The design applications are Initial design, Basic design, Hull, Outfitting, Equipment, Drafting and Material. The goals with these design applications are to, in shortest time, produce all the information needed for design and production and to store this information in the Tribon Product Information Model. TS describes the applications as:

- Initial Design — for initial hull geometry modelling and naval architecture calculations.
- Basic Design — for initial design of hull structure and major equipment items, which can be used to develop the building strategy.
- Hull — for design and parts manufacturing information for the main hull structure.
- Outfitting — for design and manufacturing information for piping, ventilation, electrical, and miscellaneous steel and equipment foundations in a vessel.
- Equipment — for defining all equipment items.
- Drafting — is a comprehensive 2D/3D drafting system for shipbuilding drawings based on the Tribon Product Information Model.
- Material — for all aspects of materials control from purchasing to invoice clearance

The Design Management aims to influence the efficiency of the design process for the customer and is applicable for Design Managers and Production Managers.

The Production engineering part of Tribon helps the shipyards to create an efficient production process. This is achieved through deriving extensive and accurate production information. The application areas are assembly planning, weld planning and factory automation.

Of all benefits achievable with Tribon those in the production area are the largest. These benefits are based on very extensive and accurate production information derived from the Tribon system. Applications in this area are assembly planning, weld planning and factory automation.

The Tribon system is ported to various hardware platforms such as Alpha AXP, SUN, HP-UX, TRU64 UNIX and Windows NT. Each platform can access to the same database server. In the future all Tribon systems will be developed for the Windows NT platform. During a transition period the customer can have a mixed configuration between Windows NT and any of the other platforms. To be able to transform the Tribon into only Windows NT technology, TS has been forced to rewrite parts of the product that depends on the technology used.

3.5 Tribon versus competitors

Available competitive systems to Tribon are characterised as advanced integrated CAD/CAM/CIM[29] systems. According to internal sources, the main competitors are the products: Foran, Catia, and Pro/Engineer. To make a comparison of these products, an independent description is necessary. This view of the different products is received from an external independent web site on the Internet.[30]

[29] CAD-Computer Aided Design, CAM-Computer Aided Manufacturing, CIM-Computer Interface Manufacturing

[30] http://sites.netscape.net/yachtdesigner/arsenal/highend.htm 2000-06-01

Tribon is characterised as an integrated design and information system created to fit the shipbuilding and offshore industries. Tribon contains all elements of design from initial design to production. Incorporated into the production aspects of the system are advanced features for the assembly phase and materials control. The core of the Tribon system is a database containing the product information model, which stores all information regarding a specific project.

Sener's Marine Division develops and maintains the Foran System. Foran is one of the businesses that Sener's Division manages and the other concerns the actual ship engineering processes. In the Foran system the different parts of the ship is divided into specific modules. These modules are characterised in a similar way as the elements in the Tribon system, from hull form generation to hydrostatic modules.

Another competitor is the Catia system. This system is like the Tribon system divided into different elements. The main element is the hull design and is intended to generate the hull structures for all kinds of vessels. A complementary product provided by the same company is the Catia Cadam, which provides tools for ship design and production. The system includes elements that create support to the assembly and detailed drawings.

The last competitor is basically a standard tool for mechanical design automation, Pro/Engineer. It is based on an architecture that is intended to deliver a complete suite of solutions for all areas of a development process, not only in the shipping industry. The range of the process supported stretches from the conceptual design and simulation of the product to manufacturing. TS does not see this product as a real competitor, at the moment. However, there is some resemblance between Tribon and Pro/Engineer and this could make it a future threat to the company.

4 Empirical study of TS

> Now I'm not looking for absolution. Forgiveness for the things I do. But before you come to any conclusions. Try walking in my shoes.
>
> - Martin L. Gore

The aim of this chapter is to describe the existing processes in the software development department at TS. Without an understanding and the ability to articulate the processes in use, it is not likely that they can be managed and improved. This chapter starts with describing organisational aspects at TS and ends with the concept development and software development activities.

4.1 Organisation

The management at TS is involved in deciding on the set of projects to be added to the "active list", how these projects are to be scoped and defined, and their objectives. They also decide when these projects are to be started and completed, what resources will be allocated in what time periods, and how they will accomplish, collectively, the firms' strategy. To be able to get an idea of how this is done a description of vital forums and procedures performed by the Product Team, Product Managers and General Management Team at TS are presented.

4.1.1 Product Team, PT

The PT consists of management from Research, Software Development and Sales/ Marketing. They act as a screen for different proposals that emerge from different parts of the organisation. They evaluate if the proposal is achievable at a reasonable cost and if the organisation can profit from it. They also consider if the proposal is aligned to the general development of the product. Depending on the magnitude and importance of the proposition, the procedure that follows differs. The first procedure handles proposals, which affect the current strategy of the company. If this is the case then it will be forwarded to the GMT group who makes a decision on the matter and then hands it back to the PT. If the decision by the GMT is to progress then the PT makes a further refinement of the proposition and transforms it into an internal work order[31]. The second procedure handles proposals, which stick to current strategy. If the PT gives a go then the proposition also turns into an internal work order.

[31] See internal work order under Departmental planning, section 4.3.2

4.1.2 Product Manager, PM

Product Managers are subordinated to the vice-president of the company, who has the overall responsibility for the software development. The PM's are accountable for different parts of the Tribon system. They are in charge of the planning of their resources and the overall development within their domain. Due to the relatively small size of the company, new concepts are often invented during informal meetings with the Product Team. Formal reviews are conducted 2-3 times per year to assess the total project plan.

4.1.3 General Management Team, GMT

GMT is the top counsel of the company. It handles strategic issues and is thus ultimately responsible for the internal and external development of the company. GMT consists of management of Sales & Customer Support, Finance, Software Development, Accounting & Administration, Research & Basic Software Development and the President of the company. They establish the desired future mix of projects. This entails balancing strategic choice against practical realities, in determining what percent of the critical resources should be committed to each project type.

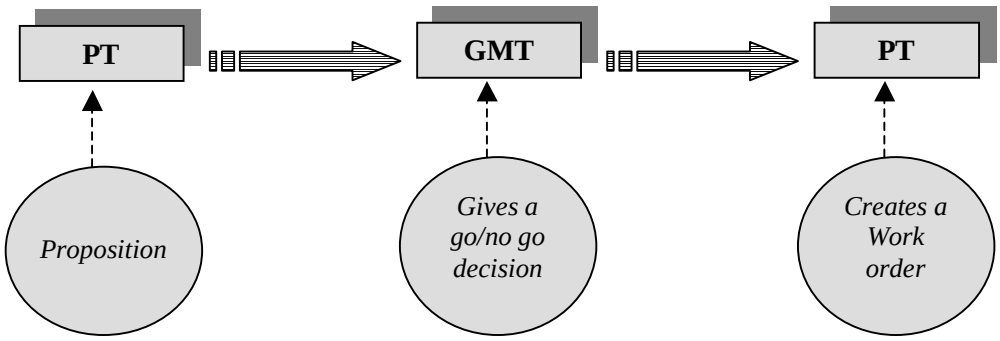


Figure 4-1: The project selection process at Tribon Solutions AB.

4.2 Research

One of the main assignments attached to the Research and Basic Software Development department is to generate new feature ideas and stimulate the evolution of software. In order to retain a creative environment too much control is not enforced on this department. The department gets their ideas from being well acquainted with the market and through socialising within the organisation.

One can divide the projects undertaken in the research department, described by Figure 4-2, into three different areas: minor improvements, functional specifications and major projects that stretch over several years.

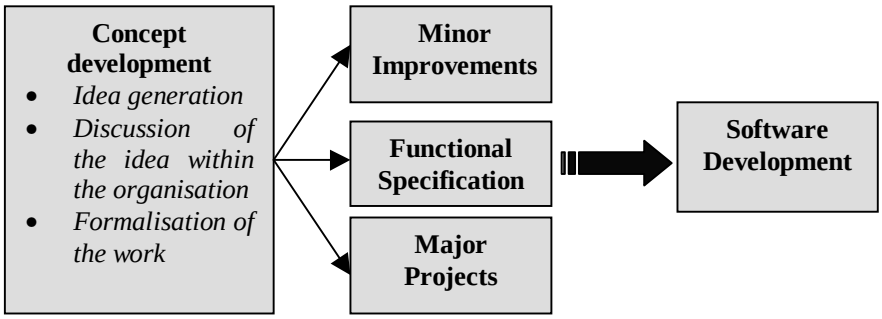


Figure 4-2: Different types of concepts in Tribon Solutions AB.

Minor Improvements: are often dealt with within the development group and there is no considerable need for an external contact with other departments. These improvements have a low impact on the surrounding system and are therefore easy to test. An example is a customer who would like to be able to choose the colour of a line instead of just having one alternative.

Functional Specifications: are developed to concretise ideas created within the research department. In this context prototyping is a common tool for specifying a product and to communicate the department's tacit knowledge to the development department. To ensure this, these kinds of projects have a transition period characterised by co-operation to confirm that the software development department is following the path laid out by the research department.

Major Projects: are ideas that have an overall impact on the product and must be, as earlier stated, ventured in the PT. Generally it takes 1-2 persons to develop a prototype for a major project.

4.3 Software development

The software development department is responsible for the main development of the Tribon system. The specification of the work is derived from the work of the research department and direct customer demands. This section starts from direct customer demands and continues via departmental planning to product releases and configuration management.

4.3.1 Direct customer demands

Direct customer demands placed on the product can be captured anywhere within the organisation. These demands will advance to the software development department through several different channels such as customer support reports, CSRs, informal meetings and the Product Team. When the demand has reached the software development department they will be classified as one out of three different project categories - A, B or C.

A- These kinds of project are developed within TS and there is no direct contact with a specific customer and therefore they are totally funded by TS.
B- These projects are developed in conjunction with a specific customer and the project will benefit both the customer and TS. The cost of the project is thus divided between TS and the customer.
C- These projects are carried out as a request from a specific customer and are only of interest to the specific customer. The cost is totally carried by the customer in question. The company tries to avoid these projects by turning them into B projects. The purpose for this behaviour is that it is more profitable to develop features that are needed by several customers.

Another category of projects is pirate development and these are carried out without the formal approval of the organisation. Pirate development is individuals' attempt to solve unattended problems and are stated as very important to the software development. The development cost is often spread over several projects or carried out on the individual's spare time.

4.3.2 Departmental planning

Product Managers are responsible for the planning of their department, this involves the planning of resources and the improvement of their employees' capabilities. Through internal work orders and CSRs product managers can get the overall picture of the resource utilisation and the individual workload. Furthermore work orders prevent misunderstandings between the sales and software development departments. There exist two different types of work orders, external and internal.

External Work Order: When an agreement has been reached with a customer, this is reported to the Contract Manager. The Contract Manager then creates an external work order and then distributes this to both the department responsible for carrying out the order and the financial department. The external work order contains information on customer contact, contract responsible, delivery date and the nature of the assignment.

The external work order also aims to keep track of the progress of the assignment. This procedure therefore works as an early warning system for time schedule deviation and as help for the Product Managers to update their total project plan.

Internal Work Order: When a product manager distributes an external work order to the developers, it transforms into one or several internal work orders. Through the internal work orders and the CSRs it is possible for the product managers to create an overall picture of the individual workload as well as the departmental workload.

To support the work orders, TS has developed an Intranet to give an overall picture of the company's activities and its structure. It contains information on routines, policies, bulletins and external work orders. In ambition to capture both the total workload in the company and the workload at the individual level there is an intention to include internal work orders into the Intranet as well.

4.3.3 Product releases

Tribon is continuously undergoing development and new releases are created at least once a year and delivered to all customers with a maintenance agreement. A new release could for example contain a program for handling continuous welding, where the product takes into account continuous varying bevel angles.

In addition to these releases, specific customers can get an upgrade of their product, if this is of major importance to them. There are several reasons for just upgrading the specific customer in question and not all customers. The first is that different clients use the product in different ways. This means that the problem encountered could be customer specific since no other clients have encountered the same problem. The second reason is that customers reluctantly want to upgrade their system during production. The cost of having a production stop is too high and customers are not willing to take the risk.

4.3.4 Configuration Management, CM

The goals of using CM are to ensure the integrity of a product and to make its evolution more manageable. Tribon is a very large and complex product. One cause of the complexity lies in the system is being written with an increasing number of programming languages and running on different operating systems. It has therefore become harder to control the development of the software. To manage the complexity the company use a software configuration tool called Clear Case. It is used to promote the use of version control, keep a history of changes in the data repository and to mark successful builds.

4.4 Test and support

The test and front line support department is responsible for the α- and β-testing and they are also accountable for the front line support to the customer. As a result they have a close relationship with the company's customers. Based on the information given by the customers they can produce the first evaluation of the software. When it comes to internal development front line support is occasionally used as a substitute for the customers. In addition, through front line support and CSRs, the company gets indirect information on which functions are being used and what new features the customer would like to use in the future.

α-Test, Integration test: α-test is an integration test foremost performed in order to verify that the system is executable and can be initialised and ended in a proper way. The tests performed as α-test are regularly not automatically derived tests. The result from the α-test is a certification of the product.

β-Test, Customer test: While the α-tests are made to confirm that vital functionality of the system are correct, β-tests make sure that the system works in a proper way. Due to the magnitude of different usage of the product TS cannot thoroughly test the

program themselves therefore customers perform the β-tests. In order to get detailed information on how the product is used by the customer TS tries to execute the β-tests on site in Malmö. This procedure is a lot more efficient since problems that occur can be dealt with much faster. The company has recently decided to start β-testing in Bulgaria. One of the reasons for choosing Bulgaria is that it is cheaper to hire people there than in Sweden.

Frontline Support: TS has divided the problems reported to the frontline support help line into four different classes, A, B, C and P. This classification should not be mixed up with the product classification referred to earlier. A stands for high priority, B stands for medium priority and C stands for low priority. The class P is a rather new class that is only used internally in the company. It classifies those problems that might cause production stop at the shipyard.

Local support: Since customers come from different cultures, TS has identified a need for locally situated support in a couple of countries. In USA, Japan, Russia, Germany, U.K, Korea and China the main support is most of the time handled by the local support. It is always the local support in these countries that have the direct contact with the customers. If the local support cannot solve the problems themselves then they are forwarded to TS in Malmö.

Customer Support Report, CSR: When a fault or an improvement request is reported to the test and front line support department an appropriate CSR co-ordinator is identified. The CSR co-ordinator is an experienced developer who is responsible for following up the reports. Based on the information from the CSR co-ordinator and the customer, a CSR is created and a priority is attached to it. A front line support's member or a CSR co-ordinator then forwards the CSR to an appropriate developer. If the project timetable slips, the priority escalates and the responsibility for getting it done moves up the organisational hierarchy.

5 Analysing the product development process

> Experience is not what happens to a man. It is what a man does with what happens to him.
>
> - Huxley

As a starting point to understand the complexity and identify the critical activities in software development the aim of this chapter is to present an initial analysis of the software development at TS. This analysis has emerged from the empirical study and our theoretical knowledge and no general conclusions are drawn at this stage.

5.1 First impression

In attempting to understand the development process of TS, we have found that the customers seem to be satisfied with the product performance in comparison with the competitors' products. This is due to the uniqueness of the product and the fact that TS has more than 30 years of experience within the development of software for shipbuilding. The first impression is then, that TS is a strong competitive company who knows how to deal with the shipyards. Additionally, they have the ability to develop products in accordance with the customer needs.

The software market for the shipbuilding industry is very unpredictable and the customers are continually shifting. There are a few countries and shipyards that seem stable and have a long history in shipbuilding. However, as governmental support for shipbuilding is diminishing in some parts of the world, it is increasing in other. This implicates that new shipyards emerge continuously and as long as the dead-weight tonnage for shipbuilding is constant, new customers will appear.

5.2 Organisation

We believe that the software development department at TS too fast acknowledge demands placed upon them from other parts of the company. The culture of the company is characterised by that the software development department rarely confronts the decisions brought on them. Software development managers accept the current demands placed upon the systems and the resources they control. In this way their decisions are based on a narrow functional perspective rather than on the welfare of the overall company. This is a situation that is amazingly stable, as there seems to be a resistance to changes within the organisation.

There are some routines established in the development process, but rather few of them are actually supported by the organisation. The problem is that there is a

moderate knowledge throughout TS of these routines and their application. Nevertheless, there is a thoughtful routine in the transition of the requirements from the research department to the software development department. This routine creates a thorough transition of tacit knowledge by the joint effort in development during the first stages in the software development.

The development approach at TS is very structured. The work is divided into tasks and then passed through a structured sequence of steps from concept specification through implementation to β-testing. As each step is completed, the project is passed on to the next step.

Despite the structure in the process no one is actually responsible for the overall performance such as specification fulfilment, schedule management, or financial performance. The developers are generally controlling their own part of the project and do not reflect the overall conformance to customer satisfaction. The work in other departments is not seen as affecting their own work. There are no clear priorities and responsibilities throughout the development process and there seem to be signs of sub-optimisation.

TS is very dependent on a few key employees. The dependence on key employees has, so far, not caused any problems. The main reason for this is that the workforce is not volatile and only a few employees have intentionally left TS. It is our opinion though that if these people were going to leave the company, TS would face a lot of problems. The importance of tacit knowledge is often underestimated and no information system could ever replace it.

5.3 Software development strategy

The development of Tribon has in later years been much controlled by the customer. To avoid becoming to market-oriented efforts concerning new technology opportunities and long-term strategic plans will be necessary. This is not the situation today since the company is working as an organisation that is reactive instead of proactive.

An example of this reactiveness is when the company decided to develop the Tribon system on a NT-platform. This decision was forced due to the apparent risk of diminishing market-shares. They were able to finish a primary version nine months ahead of time schedule. Unfortunately, the developed product could not fulfil all of its primary and initial specifications. This could perhaps have been avoided if the company's attitude would have been more proactive. There has to be a balance between reactiveness and long-term strategies. We think that the degree to which the software development is aligned to the customer needs in the market place is significant in order to create an overall success of TS.

There is a lack of shared perspectives within software development; this is especially true when discussing the role of testing and implementation. This lack of

understanding causes an insufficient cross-functional perception throughout the company.

The main problem in the formal communication is the shortcomings in the transfer of information between the market department and the software development and vice versa. Another difficulty in the communication is the lack of specification information on what to develop since this is made more or less informally. A problem area that could cause this insufficient understanding is the lack of formal ways of communication. The informal communication seems to be in complete contradiction as there is a lot of informal discussion throughout the TS.

Another factor that could explain this lack of understanding is that software managers often are not involved in the creation of the corporate strategy and policies until the decisions almost are made. This creates a situation where these managers have little to contribute to strategy alternatives and, as a consequence always appear to be complaining. The complaints often regard unrealistic demands placed on them regarding specifications and timetables.

5.4 Research

In general, at TS, software development is seen as a highly creative work. This is especially true when it comes to the research department's work. Creativeness is concentrated to this department and therefore a relatively loose control is preferable - this is also the case. This department seems to be good at capturing different ideas, both internally and externally, and to realise them. It is thus also very good at transferring ideas to other departments through the joint development in the beginning. However, this transition period could also be taken as a sign of that the specifications are not clear enough to be efficiently carried out.

An impression that concerns the entire software development department is that there seems to be no loss of prestige to unravel both personal and departmental weaknesses. If problems arise, there is thus an open attitude to discuss probable causes and solutions. A clear sign of this is that TS is an active member of a group called SPIN-SYD. This group constitutes of several large software companies that have a close co-operation with Lund Institute of Technology. TS has good experience from these kinds of joint ventures and it shows that they as a company are not afraid of showing of their weaknesses in order to improve their development process.

5.5 Software development

5.5.1 Departmental planning

At the department level, managers have the walk around approach towards management. That is, they rely on informal communication with their employees. This approach work fine when the number of staff members is sufficiently low and the complexity of the tasks is manageable.

To support the decision process, the information system contains the following data: projects currently undertaken, their estimated time duration and how much time that has been spent on them so far. What is not found in the system is the estimated time left before project completion. This might not be a problem as long as the informal approach towards management works and problems are found early. However, as it is today, time schedules seem to be slipping and since the time reporting system is used ad hoc and not representing the actual time spent on different projects, this system doesn't support the management very well. This could perhaps be connected to that TS is a very action oriented company where no excessive documentation can be found. This can however be good since the amount of administrative work therefore gets minimised. But it has its shortcomings in the ability to evaluate past decisions.

Both projects and requirements lack priorities with exception of the CSRs. Besides those, priorities are left for the individual developers to decide since they know their areas best and are therefore better suited to spot requirement dependencies. However, a prerequisite for setting requirement priorities is that their interdependencies are mapped. Setting priorities can thus help the developers to make an early detection of interdependence and thus avoid sub-optimisation.

Along with the informal approach towards management, decisions and evaluations are often based on past experience and gut feelings. Due to the fact that there is a low turnover in staff and that most people have been working within the company for an extensive period of time there ought to be an extensive experience within the company.

5.5.2 Cost

The primary essence in the cost structure of the company is the time it takes to develop the individual projects. There is no division of overhead corresponding to financial departments and other supporting departments. This is, however, requested from various departments. Furthermore, the market department would like to see a direct cost attached to each requirement/project and not just the time. They believe that a direct cost would increase the cost awareness within the company and the knowledge of how much different requirement cost to develop.

When viewing the process from specification, implementation, alpha testing, work added due to that tasks have not been approved by the β-test, and CSR, the company

knows their overall development cost, and the costs related to CSR. CSR costs are allocated at department level and they seek no further division.

Equally important as keeping track of costs is to know where the revenues are created. As with costs, the revenues are not divided into different departments, programs, or functions. TS has no thorough understanding of which of the functions that are used, and are thus not able to make this division.

5.5.3 Pirate development

Pirate development is development that has not been formally approved. This kind of activity is a hot issue within the software development department and is considered to be sacred by some developers. Due to its nature it is hard to assess how much time that is spent on this activity. The danger of having this sort of activities, outside the research department, is that resources could be spent on tasks that are not aligned to the overall corporate strategy and that these steal resources from those that are aligned. However, in contrast it is argued that too much control limits the creativeness of the developers' work and several of these outputs have turned into major success.

The performance of TS is low, this with respect to that target and set schedules are repeatedly not fulfilled. This is remarkably as the projects are continuously probed in order to make sure that the projects are keeping the schedules. There seems to be a problem in estimating how much time that is actually needed to finish a project. Another problem is the interconnectedness between projects, which is not managed in a proper way.

5.6 Test and support

The employees in the test and support group are overworked. There are probably several reasons why this situation has emerged. The primary reason might be that the company is focusing more on the quality dimension now than before and that this has lead to more work. Another reason might be that the responsibility boundaries between the software developing departments and this group have been unclear. In order to solve this situation a new β-testing department is under establishment in Bulgaria. One of the obvious benefits is the lower cost of salary. But compared to bringing customers to TS in Malmö this solution might not be as efficient or effective as the direct customer contact.

6 Reference models

> We cannot solve problems by using the same kind of thinking we used when we created them.
> - Albert Einstein

In this chapter we present the key concepts and models that are necessary to understand and to analyse in the software development process. The purpose is to outline some of the basic characteristics of the structure and dynamics of software development and to create a support for the reality these models represent. The aim is furthermore to set the scene for the theoretical part of the master thesis and introduce ideas that will be further studied in succeeding chapters.

6.1 Perspectives of strategic level

The reality of software development is too complex to characterise in exact terms. In order to manage the complexity of the development process certain aspects are considered more essential than others. We argue that there exists a complex set of aspects, or perspectives, that have to be encountered and managed to control the product performance. In this section these findings are divided into four different perspectives: resource planning, organisational structure, specification process and information flow.

The four perspectives should not be seen as independent activities to manage. Instead an understanding of the interactions between them must be recognised. This is the complex reality for the managers in a software developing company and forms the basis of the product development process. In Figure 6-1 the different interactions that have to be considered in order to create the desired product performance are presented.

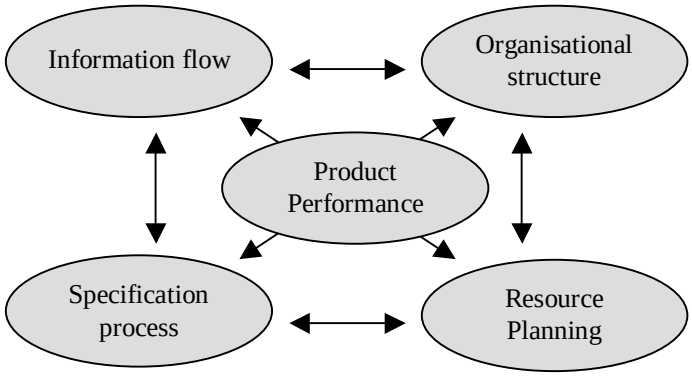


Figure 6-1: Potential interactions between process driven factors.

Controlling these aspects at strategic level is, as stated before, not the only prerequisite for achievement in a software developing company. However, as the tools and platforms are continuously changing in the software industries this implicates that the implementation techniques play a less significant role to the overall picture.

6.1.1 Resource planning

The resources within a company have a strategic impact on the company and therefore they need to be matched to the opportunities that arise on the market. As these resources often are shared between different functions and projects in an organisation, the management of these resources is essential to the company. According to Grant, the greater the rate of change in the company's external environment the more likely the internal resources are to provide a secure foundation for long-term strategy[32].

6.1.2 Organisational structure

No matter what business a company is involved in, there is always a need to innovate and learn in order to maintain competitiveness. One might argue that an organisation, where the roles are specialised and thus has a clearly defined division of tasks, is to prefer. The consequences of specialisation are rather that the company gets bureaucratic and stiffening than that the company is nurturing the flexibility of the organisation.[33]

This aspect has to be considered to be able to fit the software to customer demands. Most organisations now recognise that various functions of the company are not independent. Instead they are closely interconnected. This interconnectedness implicates that employees from various functions must work towards a common goal to achieve a prosperous company.[34]

[32] Grant 1998
[33] IVA 1991
[34] Cooper 1995

6.1.3 Information flow

To be able to compete on the market it will not suffice that only the market department knows what the customer demands are. There has to be an overall understanding throughout the company of both customer demands and the resource situation. How management provides information on the actual standings on the market and the future direction of the company reflects the efficiency of the information flow within the company.

To create and implement a business strategy is not an issue for the management alone. The realisation of the strategy has to be supported by a certain individual and departmental educational level inside the company by those who have to execute it. Otherwise, the strategy implementation is futile and thus are visions and strategy statements not translated into actions at the operational level. A broad communication from top to bottom and vice verse is essential to share the strategy and critical objectives with all employees if the strategy of the company is to succeed.

6.1.4 Specification process

In the work of creating a strategy Kaplan and Norton argues "Lofty vision and strategy statements do not translate easily into action at the local level".[35] This is basically the same when arguing on how to write requirements on what to develop and how to achieve these requirements. These two perspectives of requirements are combined into what is called the specification process. The specification process also handles the issue of controlling that the requirements are fulfilled and thus includes the areas of verification and validation.

The customers continuously demand new and extended requirements on the product. It is therefore crucial to the specification process that the requirements held by the software developing company itself, are primarily the same as those held by the customers. The management of the company is significant to the specification process while these set the direction and articulate the goals of the product.

6.2 Perspectives of operational level

An awareness of corporate goals at management level is not enough to manage the employees' behaviour. Somehow, the company's overall strategic objectives and measures must be translated into objectives and measures for operating units and individuals.

The theoretical and empirical studies have revealed that the TS' development process is not describable by a generic development process. The development processes differ between companies. The reason for this is that there are no development process or standards of behaviour that can be reasonably applied at all times and in all places. Instead, the question of whether or not a development process is adequate

[35] Kaplan 1996

must be answered relative to the market structure and organisational size in which it takes place.

The approach at this level will thus focus on the supporting activities of the software development process. By supporting activities we classify those activities that are vital to manage in order to create a software product in accordance to actual customer demands. The supporting activities are resource planning, organisational structure, specification process and information flow. These supporting activities assist the whole software development process from concept development to integration- and β-testing. The ideas are presented in Figure 6-2 as our proposed value-chain in a software developing company[36].

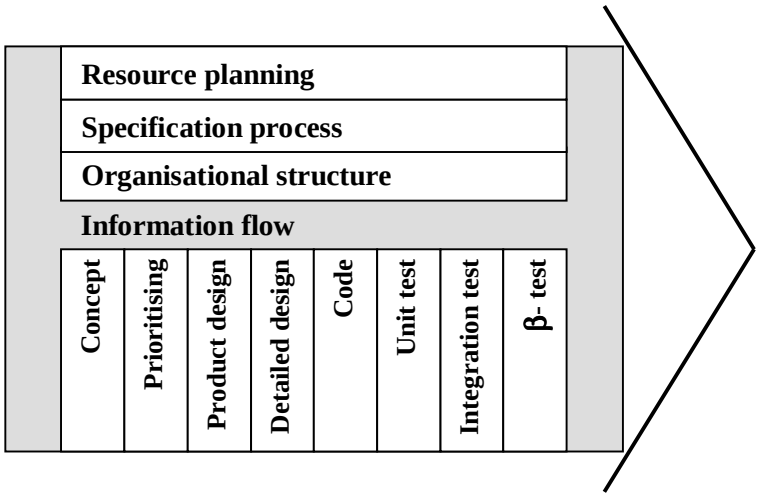


Figure 6-2: A proposed software development value-chain

An important remark to the Figure 6-2 is that the steps from concept development to β-test describe a generic development process, which is not time dependent. Instead, the relationships between the steps are merely linear dependent.

6.2.1 Resource planning

Resource planning at this level is the stage where the strategy and policies are translated into activities to be undertaken over a predefined amount of time. The projects that have been derived from the strategic level can vary in size from small projects as building special editions of the software product to new product development.

Making optimal use of available resources, by planning and tracking, is vital, as software development is an expensive task. To control the resources a couple of obvious obstacles must be handled. First, specialised skills are often required to complete a project and the aim of the company is to make sure that essential skills are

[36] Influenced by the value-chain of Porter 1998

available. Additionally, one or more key team members may have other unavoidable responsibilities. It is thus important to carefully consider how and when these resources have to be available. Another obstacle to consider is that project work tasks are not constant over time. This is especially important to have in mind if several projects are to be concluded on a common release date and if they use shared resources.

6.2.2 Organisational structure

The organisational structure includes a company's formal reporting structure, the controlling and co-ordinating systems, as well as informal relations among projects within the company and between the projects and its customers. The organisational structure and the management style that operates within the company are one of the most obvious ways of discovering the company culture. The culture is additionally dependent on a number of factors including the size, history and tradition, technology, environment, leadership and management style.

The culture in a software development company distinguishes the effort of the employees to use developed standards and methods. This is, however, not specific to a software development company and one conclusion is that it is more or less a matter of attitudes or a question of mental resistance. This implicates that spending time and effort on human resource management and employee motivation amplifies the productivity of the company and its projects.

6.2.3 Information flow

Once strategic goals have been set and communicated, managers at the project level are responsible for producing plans to achieve them. In addition, the process by which the projects are conducted should then be communicated throughout the project. Communication of plans and processes are critical to the effective implementation of projects. An environment of open communication and sharing of information is seen as essential to achieve project goals. This requires high levels of trust, respect and confidence.

Successful innovation and product development have been linked repeatedly to a creative corporate climate. In order to form this creative climate, communication between different functions throughout the entire company and informal and lateral interactions between them are essential. By division of the company functions into different areas of the company, building the communication is less effective and will not support informal communication. However, by gathering members in the project team from different functional areas this can be avoided.

6.2.4 Specification process

A critical step in the software development occurs when the strategic goals specified in the requirements are transformed into how the requirements are to be achieved. Not until this has occurred can the feasibility of the software product be analysed and the

requirements be verified. The requirements are then the initial statements of all feature deliverables.

In current practices the verification and validation of the developed software products is an inconsistent activity, generally not supported by refined software tools.[37] At the best a set of test cases are defined at the beginning of the project, before the initial design has taken place. Typically the test cases are developed past implementation. This is a crucial area to manage and an early definition of test cases limits the question on what to develop and the effort is thus on implementation.

To document the requirements and tracking the changes due to conditions during the project are important issues if the software product is to be used for further development and enhancements.[38] By documenting these aspects the efforts in succeeding development projects are optimised and the management of the software product is simplified.

6.3 Interconnecting the strategic and operational levels

The development processes of today generally do not provide adequate information to support organisational learning and improvement.[39] In a turbulent competitive environment the managers and operators need to have timely and accurate information in order to make the performance of the processes more efficient and customer focused. The question of whether the performance of the software development department is determined by the operational level or the strategic level might be debatable.

A twofold link between the strategic level and the operational level is presented and the approach is to describe the interconnections between them. As the four perspectives are discussed individually in the previous parts, in this part we discuss the perspectives in an overall context and it is mainly influenced by the systemic view. This implicates that the visions and strategies of the top management together with the executioners are important to the success of the organisation[40].

One important issue for the software development department to handle is how to link current and future projects in the product portfolio. As previously described, the strategic and operational levels are interconnected and no cause and effect can be distinguished. Additionally, there is a risk of sub-optimisation in the software development if the overall picture of the company can not be viewed. In Figure 6-3 the links between current and future projects and strategic and operational levels are described. The model shows that there is no unambiguous matter of cause and effect. Instead both strategic and operational levels must be managed in order to create the required software. That is, for the possibility to value the software development

[37] Brathall 2000
[38] Regnell 1998
[39] Rus 2000
[40] Whittington 1993

process, it is important to investigate how current projects affect the possibility to decide on what tasks to undertake in the future.

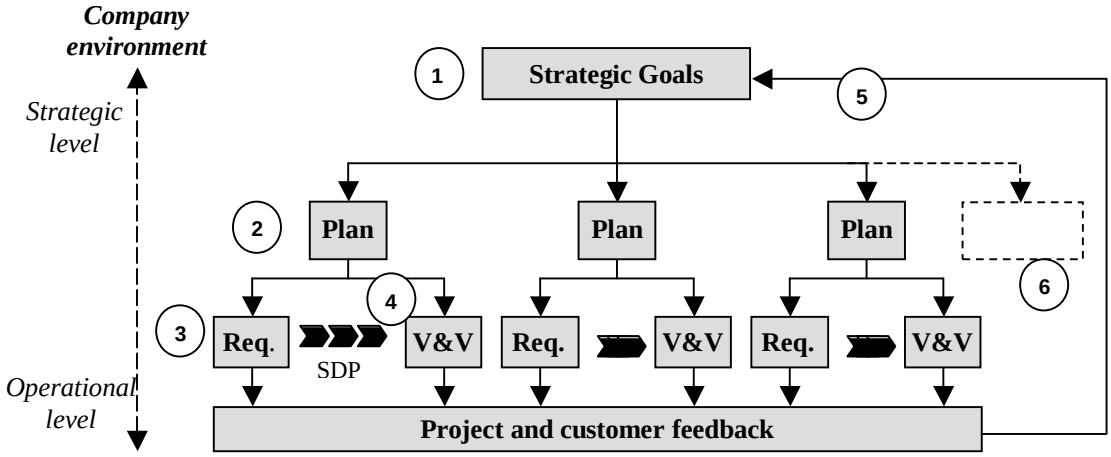


Figure 6-3: Our proposed feedback-loop of a software developing company.

The feedback loop starts with the specification of the strategic goals of the company and an overall plan on which projects to undertake, the project portfolio plan (1). The second step is to plan the individual projects according to the resources available and the objectives from the top management (2). The project is then divided into different phases, which are individual to each software development company. However, both requirements elicitation and verification and validation are vital to the development process and are represented by step 3 and 4 in the model above. As the testing and initial customer feedback is gathered, the overall project portfolio plan is affected in step 5. If the steps from project level to verification and validation are not handled correctly a sixth step occurs, which is projects concerning correction of past failures at strategic and operational levels (6).

7 Theories supporting the reference models

> When you steal from one author, it's plagiarism; if you steal from many, it's research.
>
> - Wilson Mizner

The objective of this chapter is to present theories that support the reference models and clarifies the introduced perspectives in the previous chapter. The aim is furthermore to create the theoretical background to the indicators developed in the succeeding chapter. The theories presented in this chapter reflect all levels of software development from the viewpoints of managers and middle management down to the individual level. It is the combination of the different levels that is of importance and not the specific theories in isolation.

7.1 Theoretical structure

This chapter is organised around three levels for managing the software development process. The three levels are outlined from the reference model presented in the previous chapter. Due to the magnitude of this chapter, the different parts, and supporting sections, of this chapter are described in Figure 7-1.

Background: A discussion on process diversity and development processes.

Strategic level: The strategic level including the four perspectives.

Operational level: The operational level including the four perspectives.

Feedback between levels: Interconnecting the strategic and operational levels.

Measurement: Discussion on different measurement techniques.

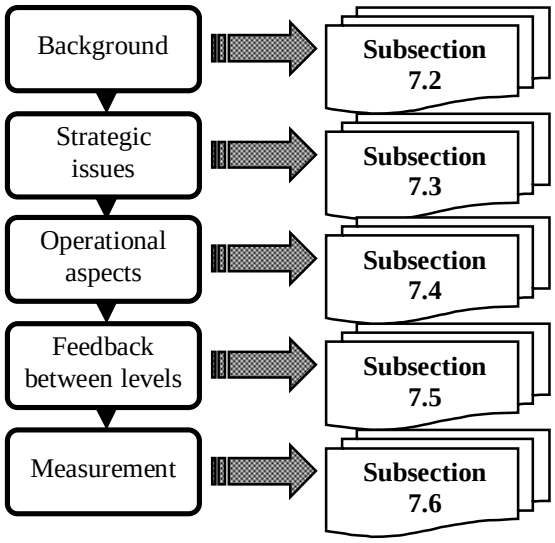


Figure 7-1

7.2 Process diversity

According to Ulrich and Eppinger a process is a sequence of steps that transforms a set of inputs into a set of outputs. They state that many of these steps and activities are intellectual and organisational rather than physical.[41] This is particularly true when discussing the development process in a software development company. The used development process affects the research and development of new products and reflects the undertakings of an organisation.

The development processes used in each software development project can be characterised by a particular process. Some of the processes are predefined, but some might not be revealed until the end of the project and others will not ever be revealed. Furthermore, the process used for each of several different types of development projects in a company might differ and could be changed between similar projects. Brooks discusses these issues of process development and diversity and describes the situation in 1987 as follows:

> *"But as we look to the horizon of a decade hence, we see no silver bullet. There is no single development, either in technology or management technique, which by itself promises even one order of magnitude improvement in productivity, in reliability, in simplicity. Not only are there no silver bullets in view, the very nature of software makes it unlikely there will be any."* [42]

In the 1995 edition of the book *The Mythical Man-Month: Essays on Software Engineering*, Brook concludes that no silver bullet has been found during the twenty-five years his book has been published[43].

According to recent research in software development the reasons for the process diversity is manifold. Firstly, the process varies in accordance with the project's goals and available resources. Secondly, time-to-market, cost restrains and quality are all factors that affect the actual work procedure. Lindvall and Rus argue that some of the most significant factors in the decisions of the process are the company's size, knowledge and experience of the workforce. The application domain and the corresponding software and system requirements together with other constraints are other reasons.[44]

Because all the process components, such as different activities, products and tools, and all interactions between them can vary, processes will differ. The complexity of the process is evident and one general process is therefore not apparent in the software development industry.

[41] Ulrich 1995
[42] Brooks 1987
[43] Brooks 1995
[44] Rus 2000

7.3 Perspectives of strategic level

7.3.1 Resource planning

Out of a processual approach towards strategy, company strategy should be more incremental in its nature[45]. Instead of focusing externally, the company should concentrate internally and cultivate their core competencies. The processual perspective, which is more pragmatic, is more aligned to the resource-based strategy, described by Grant. Grant argues that when the external environment is in a state of flux, the firm itself, in terms of its bundle of resources and capabilities, may be a much more stable basis on which to define its identity than a market oriented strategy[46].

The aggregated project plan initially concerns the definition of the types of development projects the project portfolio should contain. Furthermore, it focuses on continuous identification of the existing available resources, the capacity utilisation and the future resource demands needed to complete active projects. This identification clarifies the ability to start new projects and to complete existing projects and thereby estimating the future mix of product and project types. [47]

The degree of fit between the created aggregated project plan and the current resources decides how well the company is able to carry out the project portfolio. This since each project requires certain resources to be effectively carried out. The amount of resources and the scarcity of these set a limit for the content of the aggregated project plan during a specific time period.

Time-to-market and software productivity have become driving forces in product development process reengineering in both manufacturing and software development. In many instances, the pressure for shorter and shorter schedules pushed quality issues to the background and instead has the focus shifted towards timely release intervals.[48] The research performed by Brown and Eisenhard points out to the significance of releases in timely intervals and successful product portfolios.[49]

In general, each product release constitutes of several different simultaneous projects. Good project management will support the handling of one specific project but to effectively support the company strategy, the overall project portfolio must be well aligned to that strategy[50]. Hence, the company's active and future projects have to support the strategy chosen. In virtually all companies ideas for new development projects, far exceed the capacity of the available resources within the company. The aggregated project plan aims to assure that the development resources within the

[45] Whittington 1993
[46] Grant 1998
[47] Ibid
[48] Hantos 2000
[49] Brown 1997
[50] Wheelwright 1992

organisation are applied to the appropriate types and mix of projects in order to carry out the strategy of the company.

7.3.2 Organisational structure

The knowledge-oriented company must develop an organisation that can cope with an ever-changing environment. This creates difficulties both for leaders and employees in the organisation. One of the most vital processes in the organisation is the decision making process. Of course, the leadership plays an important part in this process and it is vital that all aspects of the organisation are considered when making decisions.

7.3.2.1 Decision making process

Making the right decisions on strategic and operational level in the company will always be important and is vital for the company to stay alive on a competitive market.[51] How these decisions are taken is necessary to understand and there are many reasons why it is so important to consider. First, there can be severe consequences for the company if you do not consider a decision in a reasonable way. Second, many decisions are made almost automatically. If you are aware that there is a process of decision-making then groups and individuals have a greater possibility to affect the decisions made[52].

Authority decisions are made by a controlling part of a group or the leader. The group follows the leader's decision and keeps quite about what they really think. Supporting decisions occurs when one person makes a suggestion and one or two other persons support this decision but not the whole group. Majority decisions are very common in today's companies. But majority decisions have a difficulty in the sense that the minority in the group often has very hard to accept the decision of the majority.

There are also different kinds of decisions where a group is united. Either you discuss a suggestion until all members of the group are united or you could change the decision so that it satisfies all the members. The last one we would like to address is when decisions are made only when we have total unity. We do not change the suggestion and one member's disagreement is enough for not making a decision. One organisation that has this decision process is NATO.

7.3.2.2 Organisational culture

The organisational culture is normally divided into four parts as illustrated by Figure 7-3.[53] The first part is the dominant ideas and values and this is the company's apprehension of things that are desired or not desired and what the company should aim at and to avoid. These ideas and values affect the working procedures for the people in the organisation. The second part is the significant actors and role models and these are the people, within or outside the organisation, that have the power to affect ideas and values in the company.

[51] Drucker 1996
[52] Socialhögskolan, 1985
[53] Bruzelius 1995

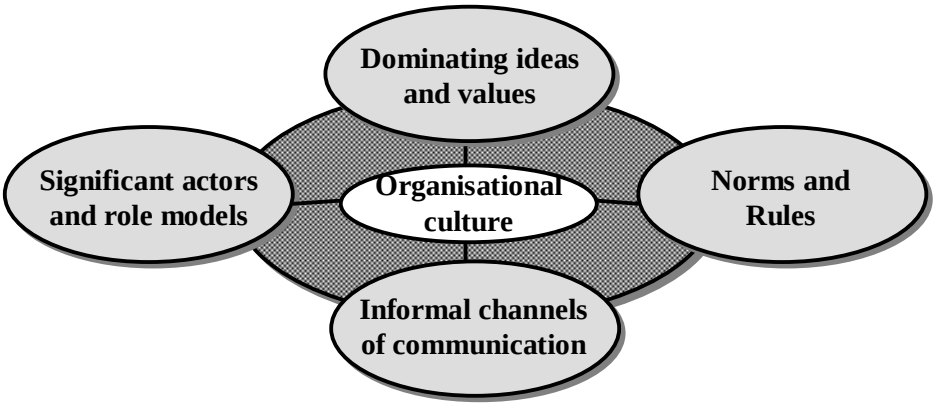


Figure 7-3: The organisational culture.

Norms and rules is the third part and this is the etiquette at the company to how you perform an assignment. Bruzelius and Skärvad state that a formal rule or norm is only efficient with organisational control. This means, that there are built in incentives in the system, for example rewards and punishments of some kind. The authors also state that there is a form of social control within the organisation. Social groups make up informal norms and rules that the group members must follow or they will be punished. The fourth part of the company culture is the informal ways of communication. According to Deal and Kennedy 90 percent of the activities in the organisation are spread by the informal network[54]. These four parts creates the culture in the company.

From Deal and Kennedy's viewpoint there are different types of cultures in different areas of business, Figure 7-4. The starting-point for the four different types is the way the organisation handles risk and how fast the organisation gets feedback on its actions.

The "macho cultures" characterised by large risk exposure and fast feedback. This kind of culture suits environments that think and plan on a short time basis. On the other hand "Bet your company cultures" also is characterised by large risk exposure but then the feedback is slow. The employees identify themselves more strongly with the company and the rank in company is often based on experience. The "Process culture" is based on low risk and slow feedback and the business is constantly developing in a controlled environment.. These companies are leading when it comes to documentation and technical perfection. The last culture is the "Work hard/play hard culture". As seen in Figure 7-4, this is where the computer-oriented companies are placed. The culture is considered having low risk but fast feedback. The organisations are market oriented and the customers' needs are in centre.

[54] Deal 1985

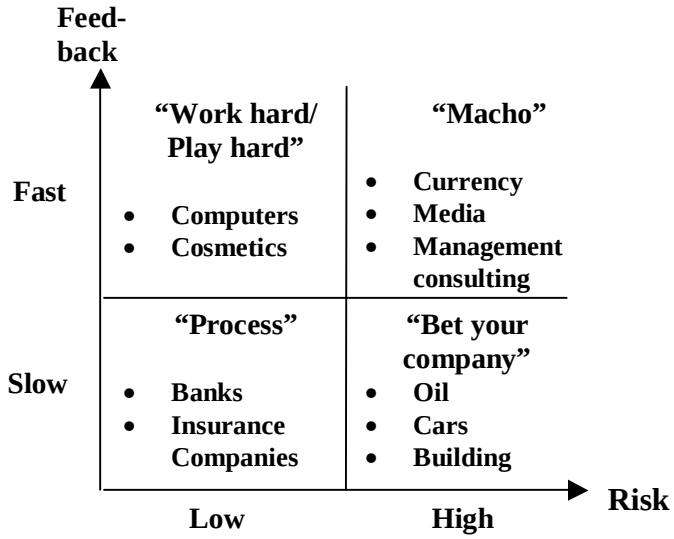


Figure 7-4 Four cultures according to Deal and Kennedy.

Building on Peters and Watermans observations that ”firms with sustained superior financial performance typically are characterised by a strong set of core managerial values that defines the ways they conduct business”, Barney identifies organisational cultures as a firm resource that is potentially very valuable and of great strategic importance.[55]

7.3.3 Information flow

Previously we have discussed the importance of organisational structure and in particular organisational decisions and culture. Equally important is it to have a good information flow within the company.

”Communication is the fundament for accomplishing co-ordinated activities”[56]

The information flow in companies today is mostly characterised by modern information technology. The information technology gives the opportunity to implement an effective co-ordination of the organisational activities. The new technology also gives the opportunity to spread important strategic information to more people and thus get a better awareness of strategy in the organisation[57]. This new swiftness in how the company spreads information is an important factor in a competitive environment.

[55] Grant 1998
[56] Bruzelius 1995
[57] Zuboff 1988

The information flow is necessary to identify and anticipate external changes. This is dependent on a firm's environmental scanning capability. As the pace of change has accelerated, environmental scanning activities have changed with them: firms are less dependent on conventional analysis or economic and market research data and more dependent on "early warning systems" through direct relationships with customers, suppliers, and competitors.

For companies in the software industry today, the employees' knowledge is the most vital resource. This makes exchange of knowledge between employees especially important. If the company cannot handle the technology that controls this it will be difficult to manage the knowledge exchange.

7.3.4 Specification process

The ability to ensure that the developed system satisfies the need of the customers is the principal challenge in any industry and is a broad research area in software engineering. According to recent research, incomplete specifications are still a considerable source of misunderstandings between the customer and the individual developer.[58] To solve this IEEE tried to define completeness of the specification process in 1984 in the *Guide to Software Requirements Specifications* as:

> *"A specification is complete when all the requirements relative to functionality, performance, constraints on system structure, attributes and external interfaces are written and if all the terms used in these requirements are defined."*[59]

This definition is in itself a statement of the problems in the specification process. It is too abstract and there are too many possible interpretations of the definition.

In the customers perspective it is the product performance that matters in the end. At the time of product-release it is often too late and too expensive to make required changes in the product. It is therefore not surprising that the focus on current research in software engineering is on the initial stages and the specification process and requirements elicitation. In addition, Cugola and Ghezzi state that these issues cannot be delayed to the end of the development.[60]

A major obstacle that must be managed is the fact that in many cases the customer does not know exactly what they want the software to accomplish. The customer has a perception of the problem to be solved, but is unable to translate this perception into precise and verifiable requirements.[61] The initial effort of the specification process at strategic level is thus to identify a set of representative users, who will provide basic software requirements proposals.[62] The finished requirements and specifications are

[58] Álvarez 1996
[59] IEEE 1984
[60] Cugola 1998
[61] Ibid
[62] Boehm 2000

then described by a complete and detailed specification of each function that the software must perform and each criterion that it must meet.[63]

An open book as in the Figure 7-5 below is one possible explanation of the differences between customer and developer perceptions of the specification. The different perceptions are characterised by the two different pages. On the left side are the customer perceptions and on the other are the perceptions of the software developing company. These two pictures are not only unequal; they are in some aspect contradictory.[64]

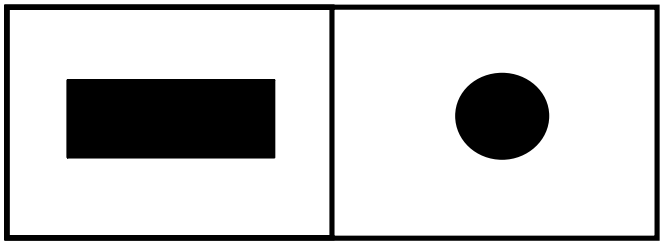


Figure 7-5: Two perceptions

These views seem impossible to join and have very few similarities of each other. However, what happens if the left page is placed orthogonal to the right, as in Figure 7-6? Then the two former two-dimensional pictures describe projections of a three-dimensional object, the desired software product. The contradictory pictures then create an entity. This is fundamental to requirement elicitation and is initially the most difficult tasks to manage.

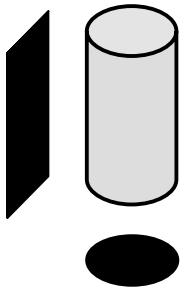


Figure 7-6: A Unified Perception

The requirement elicitation description above leads directly to the discussion on which of the requirements are essential to implement and which can be left out due to possible time-constraints. One golden rule is presented in a NASA-report and is to

[63] Landis 1992
[64] This description is influenced by a model created by Frankl 1995

implement only what is required. They have found a tendency of software developers and analysts to add features and make changes that would make the system better in their point of view. According to that research are these features and changes a cause for schedule delays and might cause requirement-deviations.[65]

In addition to the discussion on the importance of adhering to the decided requirements, the need to gain a detailed understanding of the current and future market is essential. Without this understanding, the company is not able to win orders in the market place. The strategic task is therefore to provide a better product, relating to certain requirements, than those of the competitors.[66]

As new requirements emerge during the development process and other requirements become obsolete there is a need for prioritisation among the different requirements. If the company uses prioritisation, Brown and Eisenhardt mean that it provides autonomy and accountability for significant aspects of the task. These aspects create intrinsically motivating jobs and high company performance.[67]

It is essential to assess the change's impact on the specification prior to change approval. According to a research study funded by NASA it is stated that prior to specification change it is important to estimate the cost and schedule impact of each change to requirements and specifications even if the project can absorb it. Small changes have an ability to add up over time. In cases where changes or corrections are proposed during the development process, documentation of the proposed changes is utterly important.[68]

7.4 Perspectives of operational level

7.4.1 Resource planning

The project portfolio is established at a strategic level in the company to assure that it is aligned with the company strategy. To be able to carry out the content of the aggregated project plan efficiently, good project management is needed. At the operational level projects are rarely developed on its own. Co-ordination and effective project management between different projects are critical to the result of individual project teams. In the construction industry there is a great awareness of these matters, but in software development projects this is not the case.[69] This section is directed towards project management and particularly on cost and time management.

[65] Landis 1992
[66] Hill 1995
[67] Brown 1997
[68] Landis 1992
[69] Hantos 2000

7.4.1.1 Project management

Project Management is an ever-expanding field and can be summarised as the application of knowledge, skills, tools, and techniques to control project activities in order to meet or exceed stakeholder needs and expectations on a project[70]. Traditionally, project management focused on the handling of the three different dimensions time, cost and requirements as illustrated in Figure 7-7[71].

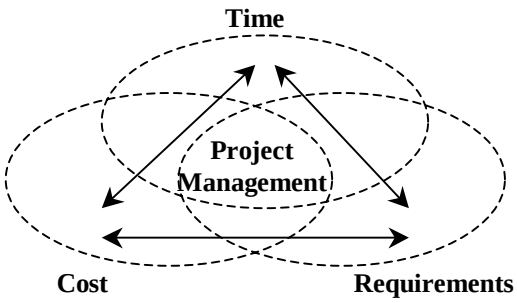


Figure 7-7: The project triangle.

The triangle consists of two resources, time and cost. The third factor, requirements, represents what the project is supposed to deliver. The interactions between time, cost and requirements are important because they are the linkage between the internal and the external factors. The triangle implies that there are trade-offs between the different dimensions. The centre of the gravity of any project will always be somewhere within the triangle.

7.4.1.2 Project time management

Project time management includes the process required to ensure timely completion of a project. Project managers have to define and estimate the time resources needed for the different activities to produce the project deliverables. Through acknowledging their interdependencies an activity sequence can be established. With the help of this work brake down structure and expert opinions or analogies managers can achieve better time estimates[72]. When the activity sequence and its time duration is established, project managers have to assure that the right resources are available at the right time. The outcome of this process is a time schedule with appropriate milestones attached to it.

Time has become an important competitive factor. Especially as product life cycles have shrunk and missing the window of opportunity could be devastating for a project and the company as a whole. If deadlines are not kept, “bad will” can be incurred externally and due to project interdependencies the aggregated project plan could fall

[70] PMI 1999
[71] Briner 1997
[72] Kerzner 1979

apart. To reach the window of opportunity, the development time can be decreased by, for example, parallel development, prototyping or reuse of already existing software.

7.4.1.3 Project cost management

Project cost management aims to assure that the project is completed within the approved budget. If the budget is not kept this can undermine the existence of the project and decrease the company's overall profitability. Project management is primarily concerned with the cost of the resources necessary to complete the different activities within the project schedule. However, managers should also consider effects of project decisions on the cost of using the product. This broader view is commonly known as "life-cycle costing".

With the resources needed for carrying out the project activities identified and the time estimates for the different activities, managers can estimate the project cost. This estimation will serve as a foundation for the project budget, which allocates the overall cost estimates to individual work items. The budget serves as a baseline for measuring project performance and if any changes are made, either positive or negative, the reasons why they have occurred must be identified in order to assure organisational learning.

Decreasing the costs of software development is primarily realised by decreasing the software development labour, and therefore cost reduction will mostly aim at executing the software development process more efficiently. A first step to this is identifying current effort and expenditure. Examples from practice in which cost are expressed are costs per source of line, cost per function point, cost per life-cycle phase, cost per product or cost per subsystem.

Cost estimates in the software industry tend to be less accurate than for those in manufactured goods. One reason is that projects are less standardised, and cost information that has been accumulated for similar work is therefore not a valid basis for comparison[73].

7.4.2 Organisational structure

7.4.2.1 Decision process

Many of the decisions in an organisation are made at the operational level. Regardless of the dimension of the decisions, they all play an important part in the development of the organisational structure. Some authors state that one of the most important prerequisites for an organisation survival is the ability to make rational decisions. The rational decision process is characterised by the following steps[74].

- Define the problem

[73] Anthony 1998
[74] Drucker 1996

- Establish the goals that needs to be obtained
- Seek different alternatives that can help you to achieve the goals
- Value the alternatives by analysing the consequences if they are executed
- Chose the alternative that best fulfils the goals
- Follow up and control the actual result of the decision

7.4.2.2 Organisational culture

The culture at the operational level is often not so clear. Though there is often a certain culture in groups that have been together for a long time. In the culture, conscious and unconscious, assumptions and attitudes are embedded and this is a part of the every day life.

In this quite psychological level the literature defines two different sets of cultures: the rule and the mind cultures. The culture of rules is characterised by the fact that everyone minds their own business. The action to set individual goals seems pointless and too complicated for the group to practice. In the group the work is organised by rules, order and tradition. This individual feeling of powerlessness often creates a culture of resistance. The group members react instead of acts. They also tend to think more about how they do things instead of thinking on what they are doing.

The mind culture is an organisational culture that has its basis in the teambuilding organisation. The group is in centre when it comes to decisions and problem solving. It is also goal oriented with a high degree of responsibility. The management of the group is concentrated on the content and process of the work[75]. Keywords are co-operation, feedback and creativity. In this kind of culture development of the organisation is common and therefore creates a natural connection between project goals and project process.

Out of a processual perspective it is not surprising that organisational politics and personal objectives influence projects[76], these phenomena have always been present in companies but are now being acknowledged. The same acknowledgement applies to the commercial pressure on projects as the world competition has intensified. But even though these factors are not revolutionising and have always existed to some degree, it is of great importance to acknowledge them, as the project approach has spread across the whole spectrum of business and organisational activities.

As the development usually takes place over a number of years, the project team is in most case volatile. The workforce may change completely over the lifetime of the project. Therefore, particularly skilled or talented individuals do not usually have a dominant effect over the lifetime of the project.[77] To manage this volatile workforce a number of supporting tools must be in place in both project and product development.

[75] Svedberg 1997
[76] Whittington 1993
[77] Sommerville 1995

7.4.3 Information flow

The word communication comes from the Latin word “communicare” and basically means making something mutual. Sommerville argues that in large project integration, project management and communications are the major difficulties. When teams are large, there is usually a difference of abilities and experience between team members. [78] By having a good dialog and efficient information sharing tools the employees and management creates a mutual understanding on what the project’s goals and the project’s status are.

Good informal communication is one of the mechanisms most useful in breaking down individual and organisational barriers to cross-functional co-operation. The most usual measure used to enhance the informal communication at a work place is to bring together the core members of a project or workgroup into the same workplace. Open office space makes the communication easier. Electronic mail and voicemail are also important factors but leave out the very important personal contact in the communication. This kind of correspondence is most effective among people that are already well acquainted with one another.

The most important formal communication mechanism probably is meetings. The frequency of the meeting is important. They have to be frequent enough to satisfy the group members’ need of information. Of course time spent exchanging information in meetings is time not spent completing other project tasks. Different techniques are used to keep the meetings as effective as possible. Some companies have morning meetings without chairs. The meetings are more effective since it is harder to stand than sit for a longer time. It is also important that the meetings are well prepared, i.e. they have a written agenda, and an appointed chairman.

Another form of formal communication is different type of information systems. With information systems we do not only mean a computerised program that control and communicate the actions of the team but also schedules that can co-ordinate the information between the members of a group. If the schedule of a project is co-ordinated and updated with the rest of the organisation it becomes an important source of information about the different activities in a project. A weekly status memo of the project could also be an important way of communicating in the organisation.

7.4.4 Specification process

The specification process at the operational level is primarily concerned with defining and controlling which requirements that are included in each individual project. One of the problems of the traditional approach to software development has been the lack of clarifying requirement goals in the development process. Proper scope definition is thus critical to project success. Davies and Layzell argues that the underlying problem

[78] Sommerville 1995

in scope definition is that specifications and designs are represented in natural language, which leads to imprecision, ambiguity and difficulty in understanding.[79]

As the human thinking is the background to how requirements are elicited and decided upon, the outcome of the specification process must be clear without misconceptions. The human perception of the world is, according to Larsen, divided into two parts. One part regards the assimilation between new requirements and former knowledge. The other regards the revision of former knowledge as new experiences develop.[80]

Figure 7-8: Human Perception

Figure 7-8 is an example on revision of former knowledge as new experiences develop. If the picture is interpreted as a big-nosed witch or a young woman is based on our former knowledge and experiences. However, if we get the information concerning the two ways of observing the picture a revision of our former knowledge occurs. This has to be considered in the specification process as well. The revision of former knowledge as new experiences develop, is mostly affecting the specification process in the software development.

System building is the process of combining the efforts of the individual project teams into a program that executes on a particular target configuration. According to Callahan and Easterbrook, it is in the responsibility of the test group to work with the development organisation and to build revisions to the release.[81]

As the building is accomplished and the result is an executable product, it is in the responsibility of the company to test if the program adheres to the requirements. Whittaker argues that if a product has high testability, it is easy to test and,

[79] Davies 1993
[80] Schultz-Larsen 1994
[81] Callahan 1997

consequently, easier to find bugs in. In addition, low testability would require more tests to gain the same knowledge of the product.[82] However, it is important to have in mind that tests can only reveal the existence of errors, not the absence.

7.5 Interconnecting the strategic and operational levels

Interconnecting the strategic and operational levels concerns learning from past experiences. The feedback process within a company creates this interconnection and is like probing the future. Reel describes the common-sense area of feedback and continuing learning in software development as:

> *"If you don't take time to figure out what happened during a project, both the good and the bad, you're doomed to repeat it."* [83]

The Experience Factory defined by Basili aims to control the feedback through evaluation of the software development process and product.[84] Unfortunately, few companies institutionalise a process for learning from their mistakes. The reasons appear to be made up of deeply rooted cultural and institutional factors that obstruct organisational change. At the same time, there is the need to continually update software competencies. The required software skills change much faster than software process improvements and reengineering techniques. This continual updating is illustrated by the specification loop, which Juell-Skielse and Askerfelt outline as Figure 7-9. The figure describes the learning process of an organisation in three steps.

The first step concerns the learning in combination with the four perspectives, the quality demands and needs of the customers. In the next step the concepts and knowledge, which has passed the first filter, forms new products and services. The filter concerns several factors such as insufficient information flow or the lack of organisational structure to handle the knowledge. The third step concerns the releases of products and services to customers. The filter between step two and three obfuscates the specifications and complicates the transfer of the products to the customers. When the software product is delivered, the last filter is introduced in the specification loop. This filter concerns the distribution of knowledge on which functions, programs and services that are actually used by the customers.[85]

[82] Whittaker 2000
[83] Reel 1999
[84] Basili 1994
[85] Juell-Skielse 1997

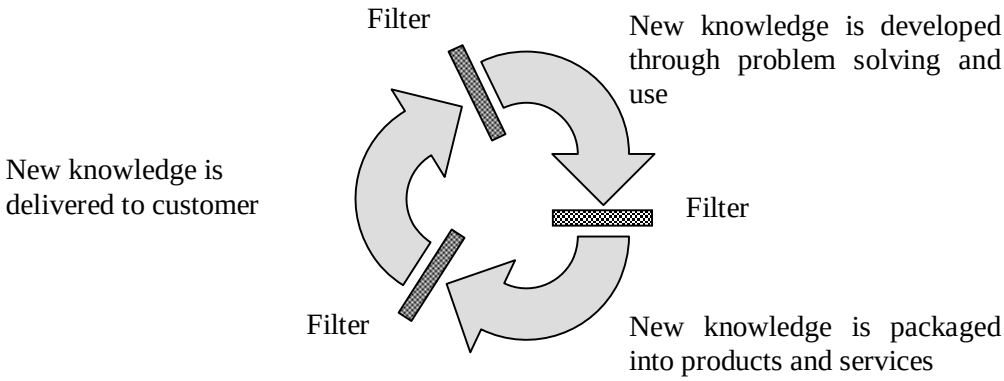


Figure 7-9: The specification process cycle

As the time-to-market has become an important issue in software development, there is a need to continually make post-mortem analysis of the projects and releases. An implication of this is that the company is able to learn why the schedule estimates could not be achieved. Compensating for those factors in the next project will dramatically improve the estimating techniques. A post-mortem analysis will also help in developing a profile for how teams and companies develop software systems. Knowing the areas of improvement allows the company to circumvent or at least manage these problems in adjacent projects.[86]

A constant problem in the software industry is the volatile workforce. It can be disastrous for software project, because replacing employees must quickly get up to speed on software that is not complete, not tested, and probably not well-documented. The lag time between when an employee quits and when a replacement is hired can wreak havoc with even the most pessimistic schedules.[87]

As specifications are tightly concerned with the customers' expectations of the quality of the software, this implicates that a discussion on quality in software development is appropriate. Unfortunately, perceptions of quality in software development differ in what is included in the quality concept. Giertz pictures the different views on quality concepts as more or less overlapping circles, perceptions, as in Figure 7-10.[88]

The quality concept is often just a question of how much is included, as in perceptions 1 and 2. At other times the disagreement goes deeper and concerns relative value of different elements. In some cases the most important feature to someone is totally irrelevant or even objectionable to someone else, so that some quality concepts include aspects that not accepted in others, like perceptions 2 and 3.

[86] Reel 1999
[87] Ibid
[88] Giertz 2000

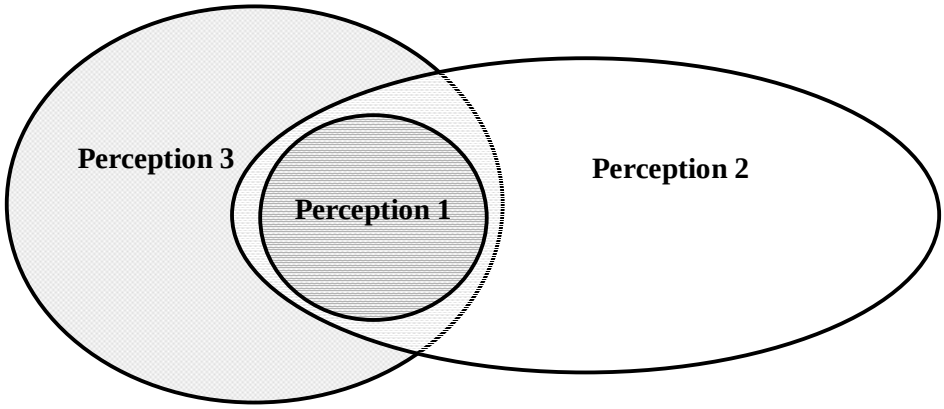


Figure 7-11: Different quality perceptions

Differences in opinion as to what aspects that should be included in the quality concept in software development are of importance for the possibility to agree on a common definition. One common definition of quality is "the degree in which the whole of the properties of the product, process or services meets the requirements, which result from its practical purpose."[89] This is, however, not generally accepted. From a measurement point of view, Fenton and Pfleeger argue that quality is defined in the terms of "specific software attributes of interest to the user".[90] As the definitions often are not the same, the least common denominator of conceptions in software development quality is instead pictured as perception 1.[91]

7.6 Performance measurement methods

The competitive environment requires that managers have timely and accurate information to help them make processes more efficient and more customer-focused.[92] According to Sage and Jensen this information can only be obtained through systematic measurement and the development of models that support the processing of the information.[93] Currently many software development practitioners and researchers are involved in Software Process Improvement. Several improvement models, methods and techniques are available, and they are divided into two major approaches, either top-down or bottom-up.[94]

The top-down approach is mainly based on assessments and benchmarking and CMM, SPICE and BOOTSTRAP are all examples of this approach. In the other case, the bottom-up approach, measurement is mainly applied as the basic guide for

[89] van Weele 1998
[90] Fenton 1997
[91] Giertz 2000
[92] Kaplan 1996
[93] Sage 1999
[94] van Solingen 1999

improvement. The Goal-Question-Metric (GQM), the Quality Improvement Paradigm (QIP) and the Balanced Scorecard (BSC) are all representing this approach.

Although both approaches complement each other to a great extent, they are rarely applied together. It is recommended to use a top-down approach to create a first overview on the status of a software organisation. Based on the result of this assessment, improvement goals should be identified that suit the organisation in the best possible way. Guidance of the improvement activities towards these goals should be created by applying some form of activity measurement. Measurement provides an overview and the opportunity to evaluate whether implemented improvement activities toward clearly defined goals, a so-called "goal-oriented" measurement method should be selected to implement measurement.

According to the corresponding literature, the CMM helps organisations to improve the maturity of their software processes through an evolutionary path from ad hoc and chaotic to mature and disciplined. A low level of maturity incorporates a high level of risk in applying a process. As organisations become more capable, risks decrease and productivity and quality are expected to increase.[95]

Each maturity level is intended to add further enhancement that software organisations typically master as they improve. The expected usage of CMM is to determine the most important areas for immediate improvement. For this purpose the CMM literature argues that it provides an assessment method to objectively and consistently assess the capability of software organisations and place them on one of CMM's five maturity levels. After executing these actions organisations can then return to re-assessing the established process, as process improvement is a continuous cycle.
Many measurement programs fail due to that they measure what is convenient to measure without assuring that the data extracted is useful to the developers of the software. To avoid this GQM is designed with the goals of the project in mind. The GQM approach provides a framework involving three steps[96]:

1. List the major goals of the development or maintenance project.
2. Derive from each goal the questions that must be answered to determine if the goals are being met.
3. Decide what must be measured in order to be able to answer the questions adequately.

[95] Paulk 1993
[96] Fenton 1997

8 Development of indicators

> Our life is frittered away with detail … Simplify, simplify
> - Henry David Thoreau

The science of measurements is called metrology and the aim of this section is to define the metrology of the software development process. The basis for this definition lies in the sections describing the reference models and the supporting theories. The aim of this chapter is furthermore to present the extracted indictors.

8.1 Measurement definition

An entity is an object or an event in the real world and there is often a need to describe the entity by identifying characteristics that are important to us in distinguishing one entity from another. The area of measurements is intended to capture information about the attributes of these entities. An attribute is a feature or property of an entity and typical attributes include cost, time or quality. One possible measurement definition is:

> *"The process by which numbers or symbols are assigned to attributes of entities in the real world in such way as to describe them according to clearly defined rules."* [97]

A model is an abstraction of reality, allowing us to strip away detail and view an entity or concept from a particular perspective. Models come in many different forms and show how the component parts are related to one another, and thus make it possible to examine and understand these relationships and make judgements about them.

8.2 Measurement implications

Measurement is needed for assessing the status of the company's projects, products, processes, and resources. Companies need to document trends, the magnitude of corrective action, and the resulting changes. In other words, they must control their projects, not just run them. Without measurement, methods such as CMM and GQM would loose their meaning. Tom DeMarco, a strong supporter of the need for measurement in software development, asserts that:

> *"You cannot control what you cannot measure"* [98]

[97] Fenton 1997

Furthermore, every measurement action must be motivated by a particular goal or need that is clearly defined and easily understandable. That is, it is not enough to assert the need to measure to gain control. The measurement objectives must be specific and tied to the needed knowledge of managers, developers and users. Thus, these objectives may differ according to the kind of employees involved and at which level of the organisation they are generated. But it is the goals that tell us how the measurement information will be used once it has been collected.

As already mentioned, there are many different arguments why an organisation should make the effort of measurement and the attached investments in capital and resources. One implication is that measurement can aim to establish the company position and its development. Without this knowledge, strategic and operational decisions get insecure and hard to implement in the organisation. Equally important it is to internally communicate the company's position and development in order to create a shared motivation to achieve goals and improve the product, the process and the company culture.

8.3 Possibility to measure

With history in mind, it has always been of human interest to measure different aspects of an event. In accordance with this Galilei once stated that:

> "*What is not measurable make measurable*" [99]

Time, cost and quality are the most common aspects that are measured in an organisation and these aspects are also reflected in the project triangle described in the theoretical part of this report.

The reason why time is a common attribute is because it can be used in order to assess internal efficiency as well as external efficiency. The implication of measuring time is the possibility to shortening cycle-times and to make cost reductions and thus increase customer satisfaction. This is the major theme behind Just In Time (JIT) theories and practices.

Productivity of a process can be assessed by the relationship between its input and its output. Typical outputs are physical volume, accrued profit. Inputs are for example, investments, human resources or the process total cost. Since cost can be hard to trace to specific processes or activities methods such as Activity Based Costing can be of great help.

The quality is an attribute of a process's external effectiveness. There are several dimensions of what characterises quality. They can only be assessed indirectly and they all ultimately aim to capture customer satisfaction. Some common metrics are;

[98] DeMarco 1982
[99] Galilei 1564 – 1642

the number of customer complaints, time to solve a customer problem, or mean time between failures.

8.4 Extraction of indicators

According to Mossberg there are some fundamental characteristics that indicators should fulfil.[100] First, a number should be the expression of the indicator. Secondly, they should comprise information and third, what is measured should be of high value for the total picture. The fundamental characteristics of indicators is described by Mossberg as:

> *"If the user of the indicators had an unlimited ability to absorb and use information, indicators would not be needed"* [101]

Behind every indicator there ought to be a well-formulated hypothesis. This hypothesis should contain information on what the user expects the indicator to show. It can be compared to a model for how to perceive the indicator. The hypothesis should be continuously evaluated and possibly changed in line with the experience of it. An indicator should in the beginning be dealt with in isolation, and only in later stages can several indicators be merged together with one another. Naturally, the more indicators that aim to highlight a specific entity the better since this will increase the richness of the measurement.

To increase the usefulness of the indicator the indicators should live up to the following criteria[102]:

- They should be significant compared to what they want to establish. That is, there should be a clear connection between the indicator and the entity.
- They ought to be quantifiable. However, these criteria can be modified to a graded scale, relative or ordinal. Even subjective evaluations can be quantified by, for example, points or grading system.
- The indicators should be independent from one other and if there is a strong dependence between different indicators these should be presented together and their dependence clearly described.
- The indicators should be useful over a longer time period. This means that they ought to be available and constant in their definition over time.
- Indicators should be "objective" in the sense that everybody involved should agree on the meaning of the indicator even though the opinions of the result may differ.
- The indicators should be easily understood and coupled to the original data even though this aspect is relative due to the competence of the individual.

[100] Mossberg 1977
[101] Fenton 1997
[102] www.balancedscorecard.org 2000-08-01

8.5 Extracted supporting indicators

The four perspectives; resource planning, organisational structure, information flow and specification process are the areas where the organisation has to direct its attention. These perspectives are, as described in the reference models, divided into three levels; strategic level, operational level and interconnecting the strategic and operational levels.

The indicators are different measures of certain aspects of the perspectives and several indicators are interconnected, which is indicated in the definitions. The developed thirty-nine indicators are presented by name, perspective and level in sections 8.5.1 through 8.5.3 and the indicators are then defined in section 8.6.

To evaluate the development process according to the definitions of the indicators stated in section 8.6 the questionnaire in appendix B was created. The questionnaire reflects different aspects of the development process and each indicator analysed is supported by one or several questions in the questionnaire. When the questionnaire are used in other organisations the questions have to be changed in order to align to organisational specific vocabulary and business environment. The basis of the questions, however, remains the same.

8.5.1 Strategic level

Resource planning	Organisational structure	Information flow	Specification process
Aggregated project status	*Company reactiveness*	*Strategic information*	*Development goal*
Customer response	*Department representation*	*Priority of information*	*Priority of requirement*
Reactive or proactive development	*Project priority*	*Formal meetings regarding company strategy*	*Requirement change window*
Resource estimation	*Customer specific dependant*	*Departments alignment to overall strategy*	*Requirement cost*
			Documentation standard

8.5.2 Operational level

Resource planning	Organisational structure	Information flow	Specification process
Release date accuracy	*Support of decisions*	*Project co-ordination*	*Requirements motivation*
Release content		*Individual dependence*	*Requirements testability*
Release workload		*Supportive tools alignment*	*Building software*
Individual burden			*Degree of automated test*
Clarity of project interdependence			

8.5.3 Interconnecting the strategic and operational levels

Resource planning	Organisational structure	Information flow	Specification process
Resource evaluation	*Documentation alignment to actual work procedure*	*Organisational learning*	*Feedback on quality at department level*
Time report system			*Feedback on quality at individual level*
			Overall system knowledge

8.6 Definition of strategic indicators

In this section the indicators at the strategic level are defined. The definitions originate from the extracted theories and the developed reference models. These indicators can be used when analysing different development processes and are analysed in the following chapter. For further information see section 6.1 and 7.4.

8.6.1 Resource planning

Aggregated project status indicator
Questions of relevance: 6, 7 and 20

The indicator captures how the software department handles its current aggregated project plan, not what types of projects it should contain. Therefore it will be of interest to map what kind of information that this plan has attached to it and how often this information is formally updated. Information that could be of interest is: the projects' budgeted time and cost, what human resources and during what time period they are assigned to a specific project, who is the project manager, what project dependencies are there and what is the current status of each project. The status of a project includes how much time that has been consumed by the project so far and thus the amount of sunken costs and finally how much that is estimated to be left of the project.

The relevance of this indicator increases with the size of the development department. The more people that have to be updated on the progress of the projects the more formal the process of controlling the aggregated project plan must be. In smaller departments it might be enough with an informal approach towards management since the complexity there is more manageable than in larger departments.

Adherent indicators are project priority indicator, project interdependence indicator and the estimation of resources indicator. The indicator can be of interest if one wants to get a quick insight in how the development process is structured and controlled and what information that is easy available to the employees.

Customer response indicator
Questions of relevance: none

This indicator will try to capture how fast the company responds to problems occurring at the customers. This response time is important in order to make customers feel that the company cares about them and takes their relationship seriously. We believe that if a company perceive itself well treated it will be more inclined to make further business with the company and recommend it to other potential customers. In fact we believe that customers who have felt well treated when they have reported a problem is more likely to make re-buys and recommendations than customers who have never reported a problem. As long as the customers bother to report problems, the company is given a chance to a better

understanding of customer needs. This new insight can help the company to make better decisions in the future. Finally, in order to have a well functioning customer support system the company needs a very good system for selecting the most important reports to handle.

The relevance of this indicator will always be high as long as the company does not have an error free and perfect product.

If the company spends a lot of it resources on customers reports than this indicator must be very good. Furthermore, the more reactive a company seems to be, the more important this indicator is. Naturally all companies have to handle their customers well, but the importance will increase whit the amount of customer reports.

Reactive or proactive development indicator
Questions of relevance: 17, 18 and 26

The indicator captures if the software development department is reactive or proactive. It focuses on the amount of resources that the company spends on customer support reports, development related to that the product has been β-tested and development occurring before β-testing. The underlying idea is that one can get an approximate evaluation of these three major activities and how reactive/proactive the development process is. By taking the relationship between software development and the two remaining areas one can get a first impression on how well this development works. Similarly, one could get a first impression on how effective the β-testing is if one looked at the amount of resources spent on customer support reports. It is important to note that the primary goal is to remove the need to spend resources on customer support reports since errors found at this stadium will affect the customers directly and will cost more to correct.

The relevance of this indicator gets higher the longer a company's software product has been around. This since we believe that a product's complexity will increase with its age and that this indicator might show how well the company has dealt with this complexity.

Resource estimation indicator
Questions of relevance: 10, 21-24

This indicator looks at the most important factor for decisions, time. Time, in the form of salaries to employees, is by far the largest cost for software development. Computers, office material etc. just cost a small sum compared to salaries to employees. Resource estimation is an important part in the decision process and managers have to have great knowledge and experience to make a correct estimate of time.

The ability to estimate resources will always be of relevance for resource planning and the strategic decision process in a company. Any business decision weighs costs against revenues, in the long run or the short run. The resource planning decisions has

to well substantiated so they support the overall strategy plan. The balance is to be found when the actual time spent coheres with estimated time.

If this indicator shows a negative trend then it might be of interest to take a look at the individual deadline indicator, the individual burden indicator, the time report system indicator and the project dependence indicator in order to figure out if there are any correlation between them. For example, if the individual burden indicator is too high then the time estimates will suffer as a consequence.

8.6.2 Organisational structure

Company reactiveness indicator
Questions of relevance: 13, 17, 18 and 58

How reactive the company is, is also a measurement of how well their decision process functions. A small strategic group should be able to take fast and solid reactive and proactive decision in the constantly changing environment called the competition. The balance between the two is very important. If you give in too much to the customer there is a big risk that you will spend to much resources on solving the problems on the market today when you can be proactive and solve the problems of tomorrow.

Department representation indicator
Questions of relevance: 11

This indicator is one of the most important for the decision process. The indicator shows whether the employees support the decisions or not. If they are dissatisfied with the decisions then they do not think that they are well represented either. As we said earlier in the information flow chapter, the employees will support a decision if it is well founded, i.e. can be motivated.

It is also a question of making the right decision. If everybody in the company feels well represented they can at an early stage in the decision process signal threats and possibilities that can effect the decision. In other words, the employees act as watch dogs that support the decision process. On the other hand if they do not feel represented, they do not feel any involvement and keep the information to themselves.

Project priority indicator
Questions of relevance: 12, 42, 43 and 44

This indicator indicates the existence of a prioritising system of projects and requirements. The existent of such a system is the first step towards new ways of thinking. If the company is going to start prioritising they have to motivate why one project or requirement is more important then another. To do this you need better decision material. We believe that by starting to prioritise the company builds up a better decision process.

Customer specific dependant indicator
Questions of relevance: 54 - 58

This indicator captures the spectrum of customers' needs that the company satisfy. In other words, how much of the company's resources are dedicated to satisfy it's total customer portfolio compared to how much resources they spend on their main customers' needs.

The relevance of this indicator will always be of interest since it gives an idea of what the actual company strategy is. The reason behind this is that this indicator reveals how the company's resources are truly used and can be compared with the estimated values from overall project planning.

When analysing this indicator it is of interest to take into account the departments alignment to overall strategy indicator in order to get a better picture of the company strategy.

8.6.3 Information flow

Strategic information indicator
Questions of relevance: 4, 5, 14, 15 and 16

By this indicator it is possible to detect the ability in the organisation of sharing information concerning the overall strategy. It is important that information of the strategy is spreading throughout the organisation and that the employees have possibility to reflect and give feedback on the strategy. There has to be a good balance between the employees' interest in strategy and the willingness of the management to give out information about the strategy.

This indicator is, as indicated by the name, of most concern at the strategic level of the company, but it also important at the operational level. Without the possibility of the management to share the strategy and visions throughout the company, the strategic decisions are futile. The management must create an overall interest in strategy so that all members of the company at the operational level can have a clear view of the business. By introducing the employees in the strategy development they become more confident in their daily activities. The members of the organisation have to know their role in the strategy development.

This indicator is focusing on the fundamental reasons why the organisation exists. The indicator is therefore affecting the development goal indicator and the reactive or proactive development indicator. As this indicator improves the result of the development process is more aligned to the actual business.

Priority of information indicator
Questions of relevance: none

This indicator captures whether the company prioritises information or not. Prioritising information within the company is very important. This due to the simplicity by which information gets spread at present, through for example e-mail. The use of e-mail and other technological advances has imposed a huge amount of information on employees that must be processed and thus there is a great need for prioritising this information.

The relevance of this indicator is at present of high interest due to the technological advances that aims to facilitate the communication between individuals. However, people will learn how to use the simplicity by which information gets spread the moment when new technologies get developed to aid the selection of information.

Formal meetings regarding company strategy indicator
Questions of relevance: 15

This indicator captures the number of formal meetings within the company regarding the strategy of the company per a quarter of a year. These formal meetings assure that the employees receive strategic information and are given an opportunity to actively participate in the creation of the company strategy. At formal meetings all the different departments within the organisation should be present in order to make sure that all aspects regarding company strategy are reflected. The number of attendants should also be as high as possible.

The relevance of this indicator increases with the size of the company. This since in smaller firms informal discussions regarding company strategy might be adequate to confirm that all employees participates and thus all aspects are covered. In larger organisations formal meetings are necessary to achieve the strategy alignment.

When analysing this indicator it is of interest to investigate how well the employees' personal views of the company strategy is aligned to the decisions made at the strategic level. The more actively involved employees are in the creation of the company strategy the easier it is to get them to strive towards a common strategic goal and thus implement the strategy of the company.

Departments' alignment to overall strategy indicator
Questions of relevance: 11, 15, 16, 52, 57 and 58

This indicator describes more thoroughly the interaction between the overall strategy in the company and different departments/individuals activities. It is important to know how well the organisation is following the strategy that the management has outlined. With other words, the indicator examines how good the company is at implementing the decided strategy of the company.

8.6.4 Specification process

Development goal indicator
Questions of relevance: 27 and 30 - 32

This indicator focuses on the ability to forecast the outcome of the development process. If this indicator is satisfactory then the developers are able to better fulfil the commitments of the company. Another aspects is the possibility to affect the content of the requirements and thus produce a clear goal of the development effort which both the organisation and the individual developers accept.

As the turbulence in the software industry is ever increasing and new competitors emerge constantly, the importance of developing the demanded products is increasing. It is therefore of strategic concern to the company that there are clear goals of what to be developed in the development process. In order to manage this it is crucial that the entire organisation is able to comment and influence the requirements and thus consider all aspects of the development.

The indicator is closely connected to resource planning and prioritisation and thus the outcome of the ability to estimate resource requirements indicator, project priority indicator and requirement motivation indicator affects this indicator. As this indicator improves, the development result will be better aligned with the demands of the customers.

Priority of requirement indicator
Questions of relevance: 42 - 44

This indicator points to the amount of prioritisation between the requirements undertaken by the development department. Without clear prioritisation between the tasks at hand, relevant functionality might not be implemented within the time for the next release. When the prioritisation is well defined, the developers are able to distinguish between requirements that have to / ought to and might be implemented.

It is utterly important at the strategic level that the aspect of prioritisation is well established in the organisation. This could sometimes be achieved through informal communication in small companies, but as the company is growing, prioritisation has to be made formally in the requirement definition. The insufficiency in the information flow from the customers to the development department is one vital obstacle for dividing the requirements into different priority levels.

The prioritisation is important to the overall performance of the company. The most affected indicators are the ability to estimate resource requirements indicator, project priority indicator and department alignment to overall strategy indicator. The requirement motivation indicator is on the other hand affecting the result of the indicator.

Requirement change window indicator
Questions of relevance: 13, 20 - 24

It is important that no new requirements are introduced close to or after the initial deadline of the projects in order to have a secure release management. The indicator is a reflection of this matter and if it receives a bad result there is an indication of problems in the specification process. How the requirements change in the development process is also a reflection of how reactive the company is.

Several different causes affect the outcome of this indicator. Most of them is concerning indicators at the strategic level of the company. As stated before, if the company is too reactive this could be problematic to both the specification and the development processes. Time estimations and management of release dates are other aspects that are of concern when investigating this indicator. If the result of this indicator is unsatisfactory the long-term strategy might get damaged.

This indicator and the long-term strategy is of concern to many different indicators and is mostly affecting other strategic indicators such as the reactive and proactive development indicator and customer response indicator. The release content indicator and the project co-ordination indicator are other influenced indicators.

Requirement cost indicator
Questions of relevance: 13 and 48

The performance of the product is connected to the total cost of developing it. This indicator reveals the understanding that the employees have on this matter in the development department. Without prioritisation ability in the development process then this indicator has to be adequate. If not, there is a risk that new requirements are implemented without consideration on the costs of including the requirement into the project or release.

This indicator is important at the strategic level of the organisation as it is of concern to the projects to be implemented and to the content of requirements included into them. The awareness in the development department as well as in the marketing department of the costs to introduce new requirements is therefore essential. If the costs to implement the requirements are revealed then it becomes more understandable to implement according to priority levels.

The cost awareness and particularly the indicators at strategic level such as ability to estimate resource requirements indicator, the departments' alignment to overall strategy indicator and the project priority indicator are influencing and are influenced by this level of the development process.

Documentation standard indicator
Questions of relevance: 50 and 51

This indicator focuses on the ability to document the activities in the development process. The documentation is of concern in both the specification process and the overall information flow in the organisation. Without a proper documentation in the specification process, the organisation is not able to decide when the requirements are fulfilled. However, if too much structure is used, there is a risk in smaller organisations to loose the vitality and creativity.

It is of concern to the strategic level of the company that this indicator is satisfactory. If the company is not able to document the decisions made and the product requirements in an orderly manner, the specification process might get volatile. The procedures at operational level are thus also affected by the results on this indicator.

A number of different indicators are connected to this indicator, such as the strategic information indicator, the release content indicator and the development goal indicator.

8.7 Definition of operational indicators

In this section the indicators at the operational level are defined. The definitions originate from the extracted theories and the developed reference models. These indicators can be used when analysing different development processes and are analysed in the following chapter. For further information see section 6.2 and 7.5.

8.7.1 Resource planning

Release date accuracy indicator
Questions of relevance: none

The purpose of this indicator is to give an idea of how good the company is at keeping release dates. How good the company is at this will affect the company's ability to gain new customers and to keep old ones.

The relevance of this indicator depends on how the market is structured, how and when the customers can update their software. In other words what window of opportunity that is important not to miss.

There are naturally other indicators that will affect the outcome of this indicator. For example, the aggregated project status indicator and the ability to estimate resource requirement indicator. As with project there might be a trade off between the time factor and the content of the product, the release in this case. Therefore it might be of interest to take a closer look at the content indicator.

Release content indicator
Questions of relevance: none

This indicator captures the extent of a predefined release content that actually gets delivered to the customers. To be able to extract information supporting this indicator it is easier if the company analysed has periodic release handling with a frozen requirement content attached to each release.

This indicator is also of interest at the strategic level within the company. The reason behind this is that the indicator captures the realisation of the aggregated project plan and thus how well the company strategy is fulfilled.

When analysing this indicator it might be of interest to consider the outcome of the release date accuracy indicator. The share of the company resources that are committed to correction of past failures is also of interest.

Release workload indicator
Questions of relevance: none

This indicator captures how the upcoming of a release date affects the organisations overall work load. By examine the time report system one can map accumulated overtime against time and thus illustrate how the total overtime is allocated over time. If the amount of accumulated overtime is too large in conjunction with a release this can be a sign of that the company's resource planning process is not well functioning. If the resource planning process was well functioning then there would be no need for making the employees work overtime. Furthermore, if the accumulated overtime per employee is too large this can affect the time that the employees stay within the company negatively. Which means that the company is at risk of loosing valuable tacit knowledge.

A number of indicators are of interest when analysing this indicator. For example, the resource estimation indicator, the release date accuracy indicator and the release content indicator in order to create a better understanding of the outcomes of these indicators.

Individual burden indicator
Questions of relevance: 19

The underlying assumption, which this indicator relies on, is that the work efficiency per employee increases in the range from zero to three projects carried out in parallel. This because, if one project gets stuck another one can be continued with while the solution to the former project takes shape. However, if the number of project rises

above 3-4 projects then the efficiency is believed to drop. The reason is that the switching costs[103] gets too high.

If one is to believe the different studies carried out in this field then this indicator will always be considered relevant. However, some people might handle more than just three to four projects concurrently but these people should consider at what price they are stretching themselves, both personal and business wise.

As stated in the ability to estimate resource requirements indicator this indicator should be of special interest to take a look at in order to see if there is a correlation between constant project overruns and this indicator.

Clarity of project interdependence indicator
Questions of relevance: 29, 40 and 45

This indicator is assembled through the information on the project interdependencies. The indicator points out the relevance to ensure the scope of the project in the specifications. Without clear information on this matter, the developers might negatively influence the product, as different functionality is demanding change in the same code. A specification on the modules and files that are affected is therefore important.

This indicator is positioned at the operational level in the development process. Although it is at the strategic level that the projects are decided upon, the interdependence is normally managed at the operational level. This interdependence affects several aspects of the accomplishment of the projects as some projects might be sequentially implemented and others might be implemented in parallel.

This indicator is dependent on the strategic decisions and thus is the ability to estimate resource requirements indicator connected to this indicator. Other indicators influenced are building software indicator, individual deadline indicator and release content indicator at the operational level.

8.7.2 Organisational structure

Support of decisions indicator
Questions of relevance: 15, 16

This indicator captures how well the operational decisions within the software department are supported by the employees. A prerequisite to get the support of the employees is to back up the decisions with well-founded arguments and to have a clear purpose. If this indicator turns out well then the employees` motivation to carry out the assignments is believed to be high. Well-motivated employees striving towards a common goal increases the productivity. On the other hand if the

[103] Switching costs: the time it takes to close down one project in order to reopen another project.

employees feel insecure in the development of the working environment they also fell insecure in the leadership[104].

The four parts of the organisational culture also play an important part in the support of a decision. It harder to get a hundred percent support from the organisation if the members are used to a total democracy when making decision about the company's future.

This indicator is ultimately connected to the Department representation indicator and the Formal meetings regarding company strategy indicator since the operational decisions originate from the company strategy.

8.7.3 Information flow

Project co-ordination indicator
Questions of relevance: 6, 7 and 20

This indicator emphasises the importance of co-ordination between different activities and projects. The indicator is thus closely linked to project time management and project cost management. Specifically, the indicator captures the number of project co-ordination meetings per quarter of a year, since this gives a good estimate of how seriously the software department takes this co-ordination of different activities and projects. Middle management and top management should be able to co-ordinate different projects and activities with the help of a functioning report system.

Individual dependence indicator
Questions of relevance: 53

Experts are often overworked and as a result they have little time to share their knowledge with other employees. If the company does not take this problem seriously, vital organisational learning will not take place and the company will face serious problems if these experts decide to leave. The positive effect with having a few known experts is that the other employees will know where to turn to when facing specific obstacles. The organisations goal should be to transfer the experts knowledge to a few others and thus decrease the dependence on an individual.

Supportive tools alignment indicator
Questions of relevance: 19, 28, 33, 50 and 51

The implication for this indicator is that the specification process needs to have supportive tools in the development process, which reflect the actual work performed. If the indicator is satisfactory then the supportive tools are able to handle division of requirements, aspects of different versions and the total requirement documentation effort.

[104] Lennerlöf 1991

The requirements have to be supported during the initial development as well as in the documentation and feedback phase of the specification process. The outcome of this indicator concerns the operational level of the company. The indicator is based on the project repetitiveness and the ability to learn in the organisation. The ability to manage different versions of a product is important if different customers have considerable different versions of the product.

As the indicator is based on the ability to learn, it involves the feedback process in the development department. The indicators in the specification process and in the other perspectives are therefore influencing the result. Due to the learning ability, an indicator that affects the result is the aggregated project status indicator. The project co-ordination indicator also affects the outcome and this is based on how to co-ordinate the different projects into different versions.

8.7.4 Specification process

Requirements motivation indicator
Questions of relevance: 29, 46 and 58

This indicator is intended to show if the company knows the origin of the requirements. Different customers are demanding different aspects of the implemented requirements. The outcome of this indicator shows how well understood the demands of the requirement are.

Requirements motivation indicator is important at operational level as this gives an indication of the importance to implement the requirement. This indicator is especially important if the company is mostly relying on externally funded projects. The indication is also important if the company is mainly reactive. If the requirements are of interest of several customers, then the customer-segments should instead be indicated. The customer motivation gives the developer an idea of how the requirements should be used in a broader context.

The indicator depends on several different indicators, such as the customer specific indicator, customer reactive indicator, requirement prioritisation indicator and project co-ordination indicator. The dependency concerns aspects such as co-ordination and prioritisation of demands from different customers and different market-segments. The indicator shows how broadly the customer segments are defined in the requirements and is usable in order to develop from a reactive to a proactive organisation.

Requirements testability indicator
Questions of relevance: 30, 31 and 32

This indicator describes the ability to test the developed requirements for completion. It is intended to show the ability to forecast the actual result of the development. If the implemented requirements are tested and confirmed to a great extent, then the product performance is better aligned with the initial requirements. The maintainability of the

product is another aspect of this indicator. If this indicator is satisfactory then the maintenance and regression testing is improved.

This indicator is of greatest concern at the operational level. The ability to test the requirements is also crucial to the strategic level, as it is at the operational level that strategic visions are implemented. This indicator is intended to verify if the company is able to test and confirm the requirements. It is also intended to verify how well the company is using the possibility to test the requirements at present and in future regression tests.

This indicator is focusing on the last phase of requirement handling, when the requirement is actually implemented. The indicator is therefore dependent on the development goal indicator. Other aspects are the resource planning and thus the status of the testing in the company. As this indicator improves, the requirements definitions are improved and then the result of the implementation.

Software building indicator
Questions of relevance: 39
This indicator reflects the work performed at the preceding steps within the development process. If this indicator gives a positive result then the process developing the product is more likely to be thoroughly completed than otherwise. It is a receipt on how the requirements are handled among interconnected projects and how different interfaces are specified. There is a great risk to the development process if this indicator is negative.

The operational level is most affected by this indicator. The indicator is a reflection of how the operational level has accomplished its undertakings in creating a product from the initial requirements. Furthermore, this indicator is important at the strategic level while it is of concern when estimating the release date of the product and many different projects are relying on this result.

As stated before, the indicator is important both at the operational and strategic level. The different perspectives that are interconnected to this indicator are the estimate resource requirements indicator, project co-ordination indicator and organisational learning indicator.

Degree of automated test indicator
Questions of relevance: 27, 30, 31 and 41

In order to ensure the performance of the product it is important that the testing of the product is made thoroughly and in a controlled manner. This indicator reveals the amount of testing that has been automated and if the tests are easily completed. A satisfactory result would be a testing of the requirements in more than fifty percent of the total amount of requirements that can be tested.

The specification process is considered in this indicator and the result is relevant to the whole operational level in present and in future releases of the product. It is

relevant to the work performed in all of the different perspectives at the operational level as well as at the strategic level. As the product matures, more and more of the tests executed to ensure the performance of the product is more or less automated.

Several indicators influence this one and the most obvious of them are of concern to the specification process, such as the requirements testability indicator and feedback on quality at department level indicator. Other indicators are the release content indicator and aggregated project status indicator.

8.8 Definition of interconnecting indicators

In this section the indicators interconnecting the strategic and operational levels are defined. The definitions originate from the extracted theories and the developed reference models. These indicators can be used when analysing different development processes and are analysed in the following chapter. For further information see section 6.3 and 7.6.

8.8.1 Resource planning

Resource evaluation indicator
Questions of relevance: 52

This indicator captures how much time that the employees within the software development department spend on project reflection and feedback. The time spent on this activity should not be too small in comparison to the total project time in order to facilitate organisational learning and thus improve the overall development process.

As stated above project reflection and feedback is important in order to promote organisational learning and thus this indicator will always be of significance.

When analysing this indicator it is of relevance to regard the outcome of the feedback on quality at individual level indicator and the total amount of time that each employee spend in training per year since those also concern organisational learning.

Time report system indicator
Questions of relevance: 25

The indicator tries to capture how accurate the time report system is and thus how useful this system is as basis for making new decisions and evaluating past decisions/projects. If the information contained in this system is of high accuracy then managers can use this information in combination with their own and their employees accumulated experience in order to make better estimates in the future.

The relevance of this indicator is very high in larger organisations since it gives a hint of how much control the company has of its development process. To be more precise, what knowledge the company possesses on how the developers spend their time and thus the costs to carry out different tasks.

The indicator can be linked to the status report indicator in order to tell how valuable the time information given at these occasions might be.

8.8.2 Organisational structure

Documentation alignment to actual work procedure indicator
Questions of relevance: 2, 25

To be able to introduce new employees into the organisation and to align the individual work procedure to the common standard in the organisation, the use of organisational control can be interpreted in different ways. One way is to have thorough documentation of the different work procedures gathered by a standardisation department. Another way is to leave it up to the individuals to align their common work procedure to each other in an organic manner.

This indicator is intended to show how the usage of the formal reporting system in the company is constructed and how the resources spent coheres with what is actually reported. As the company grows there is a need for the organisational control mechanism used within the company to be more standardised and specified. In a smaller company the specified work procedures will be more or less minor guidelines.

This indicator affects several indicators as it resides within the feedback loop. The work procedures will be enhanced in each iteration of the development process from strategic decisions to the release of the product. The most immediate indicators affected are the feedback on quality at both individual and departmental levels. Another indicator is the individual dependence indicator.

8.8.3 Information flow

Organisational learning indicator
Questions of relevance: none
The organisation has to mobilise tacit knowledge and experiences created and accumulated at the individual level and make them explicit. This indicator reflects the ability of the organisation to create organisational knowledge and improvements by transformation of tacit knowledge into explicit knowledge. Organisational knowledge creation is a process started at the individual level and expanded through the organisation by interactions that crosses departmental, divisional, and organisational boundaries.

The products created by the organisational knowledge process shall be reviewed for its coherence with both operational and strategic goals. Even if the newly created product has superior quality, it may conflict with the strategic or operational goals that are stated at the different levels of the organisation. The indicator is thus relevant to the feedback of the development process.

This indicator is closely linked to a variety of indicators as it concerns how the organisation learns and creates knowledge. Indicators that are affected are the

development goal indicator, the ability to estimate resource requirements indicator, and department alignment to overall strategy indicator. Furthermore, the reactive or proactive development indicator is affecting this indicator.

8.8.4 Specification process

Feedback on quality at department level indicator
Questions of relevance: 7, 27, 34, 36 and 37

To improve the performance of the product and align it to the expectations of the customers, there is a need for feedback on the perceived quality at departmental level. The product performance is connected to the fulfilment of requirements and it is the feedback on the requirements that is the important aspect of this indicator.

The indicator concerns the feedback into the development process and is thus relevant to the feedback perspective of the development process. This is the link between the operational and strategic level concerning the requirements. If the feedback at departmental level is inadequate the strategic feedback might be neglected and constant improvements of the development process are restrained.

As the indicator connects the strategic and operational levels, several other indicators influence this indicator. The influencing indicators reside in all perspectives, such as aggregated project plan indicator, departments' alignment to strategy indicator and project priority indicator.

Feedback on quality at individual level indicator
Questions of relevance: 8, 9, 35, 36, 37 and 38

This indicator reflects the ability to receive feedback on individual performance in the development process. Without feedback at the individual level there is no possibility to evaluate the performance and make corrections and adaptations. Another aspect is the search for responsibility in the organisation, if the individuals are striving to gain knowledge and learn from the process.

The indicator is mostly concerned with the connection between past, present and future performance at the individual level. It is thus the link between the operational and strategic levels in the organisation. There must be a support at strategic level for progress by the individuals and hence the performance at operational level.

As previously mentioned this indicator is the link between strategic and operational levels and therefore are several indicators affected and influenced by this indicator. The most immediate indicator is the feedback on quality at departmental level indicator. Other indicators are individual deadline indicator and individual dependence indicator.

Overall system knowledge indicator
Questions of relevance: 11, 40 and 49

This indicator evaluates if the employees know how development of new functionality affects other parts of the software. With a thorough knowledge of the overall system, it is more probable that implementation of new requirements are made without sub-optimisation of the total system. An overall system knowledge indicates that the employees know how their own effort affects the entire system.

The feedback system within the software development department affects this indicator. Through organisational learning, the individuals' knowledge of the overall system will increase. There must be a support at the strategic level for individual progress in this area hence an improvement at both strategic and operational level.

A few indicators influence this indicator and the most obvious concerns the specification process, such as the requirements motivation indicator and development goal indicator. Other indicators are the individual dependence indicator, strategic information indicator and the resource evaluation indicator.

9 Analysis

> I have a dream that my four little children will one day live in a nation where they will not be judged by the color of their skin, but by the content of their character.
>
> - Martin Luther King

The purpose of developing indicators is to actually use them. This section is a presentation of an attempt to use the indicators, as intended, as an evaluation tool of the development process in a software developing company. The aim of this section is therefore to make a first evaluation of a company by the indicators presented in section 8.

The structure of this section is straightforward and follows the structure of preceding sections. The analysis is divided into four parts; strategic level, operational level, an interconnection between strategic and operational levels and concluding remarks. Each level is then divided into the different perspectives presented in the reference model.

The indicators are in most cases analysed through several questions stated in the questionnaire, found in appendix B, and in some cases the empirical study is also included in the analysis. An inclusion of the results from the questionnaire at TS is as well presented in the appendix B.

9.1 Perspectives of strategic level

9.1.1 Resource planning

Aggregated project status indicator
Questions of relevance: 6, 7 and 20

The questionnaire indicates that employees report their individual work status once a week to their department manager. It also shows that the aggregated project plan at the department level gets updated once a month. Furthermore there seems to be no formal forum where the individual project statuses, in the concept of how much work that is estimated to be left before project completion, can be found. For example, in the time report system, where all the projects are included, one can find information on project budgets, in hours, and amount of time spent on it but not the estimated time left. Another thing that has been noticed in our empirical studies at the company is

that there is no cost attached to each project more than indirectly through the budgeted time.

According to our judgement the frequency of which the individuals report their work status is satisfactory. However, we think that the aggregated project plan should be updated every time a new project has been added and that the estimates of the amount of work left before project completion should be formally updated every week. This shall take place in order to create a better insight in the development process and to detect problem areas at an earlier stage. We also think that there ought to be a cost attached to each project in order to create a better cost-awareness within the company. In other words we believe that there has to be a more structured approach towards management of the aggregated project plan.

One reason why there is no formal update on each project might be that the development department is relatively small and that it therefore can succeed relatively well with a more informal approach towards management. However if the number of employees in the software development department grows, then a more formal structure might definitely be preferable in order to control the process.

Customer response indicator
Questions of relevance: none

In our empirical studies we have observed that once the company has decided to help a specific customer to solve a certain problem they are very fast to do so. The handling of customer support reports is considered to be high priority work within the software development department.

Based on the above information our opinion is that the company has a well functioning process from that a customer support report has been decided to be dealt with until the problem has been solved. This might bring with it that even though the customer has run into a problem with the product, the total impression will still be good and the customer will continue to do business and even recommend the product to other customers. However, the way that the system is constructed will also disturb the developers' normal work since they probably have to close down another project in order to deal with the customer report in question. The result might be that the individual burden indicator will rise above what can be considered to be efficient.

The good result of this indicator is not very surprising since our impression of the company is that it is mainly reactive in its approach towards customers and quality. The result of being reactive has meant that it has been extremely important to them to optimise this procedure. The reason why the individual burden indicator might be affected negatively depends on that these kinds of reports are very hard to take into account when making plans, due to the swift response needed.

Reactive or proactive development indicator
Questions of relevance: 17, 18 and 26

In the questionnaire there are two questions that are directly linked to this indicator, number 17 and number 18. The outcome of these questions indicates that the company is relatively reactive and that the outcome would have been the same one year ago. Another way to indirectly assess the reactiveness of the company would be to look at how the companies' resources are believed to be spent on different activities. Question number 26 can give some guidance in this case. The result from this question indicates that about 20 % are spent on fixing customer support reports, 16 % of the resources are spent on software development related to that the product has gone through β-testing. That is that amount of resources are spent on fixing errors detected through β-testing. The remaining 62 % of the development department's resources are spent on "pure" software development.

According to our judgement, the company is relatively reactive in its approach towards software development and the quality of their product. Naturally it would be better if the company was found to be more proactive than reactive since this would imply that they were very good at seeking out both stated and latent customer needs. However, this can be a very hard thing to achieve. To be able to have a proactive development there has to be optimised information flow through out the entire company and its environment. Furthermore the company seems to have a reactive approach towards quality since the amount of resources spent on fixing problems occurring either at β-testing or at the customers. If this amount increases the company will spend less time on new development.

There might be several reasons why the company is mostly reactive in it approach towards software development. The product Tribon is a very complex product and it can therefore be hard for the individual developers to judge how much and where their own development of software will affect the total product. The result will be that many problems will not arise until the product is β-tested or before the customers are using it. In addition, the fact that the customers can use the product in so many different ways that are hard to predict during the development of a feature does not exactly help this situation.

Resource estimation indicator
Questions of relevance: 10, 21-24

74 % of the employees think that the time estimation in the projects is dissatisfactory and in approximately 60 % of the projects deadlines are expanded with an average of 30 %. These are not acceptable levels for a high performing software developer. 13 % of the projects are finished earlier then estimated.

9.1.2 Organisational structure

Company reactive indicator
Questions of relevance: 13, 17, 18 and 58

In question 13 more than 60 % say “often” or “always” when we ask them how often new requirements appear after initial deadline. This is a very high percentage and indicates that TS is a very reactive customer oriented company. 78 % also believe that TS is more reactive than proactive. For TS it can be both good and bad to have a reactive organisation. The market is changing though and we believe that the company that is most proactive will win in the long run and draw the longest straw.

Our empirical studies have indicated that TS has the ambition to cut down on the amount of projects that are financed by a specific customer. This way they cut down the reactiveness on specific customers and the needs of all customers can be satisfied. This could also mean that they want to have a more proactive attitude towards the customer.

Our empirical studies have also shown that the potential of being proactive in the decision process. Developers in the company have many times opened the eyes of the customers to new solutions to the problems. If these activities are organised, TS could be much more proactive then they are today.

Department representation indicator
Questions of relevance: 11

47 % of the people that answered the questionnaire do not think that they are well represented when decisions are made. As stated in the Departments alignment to overall strategy indicator there are many reasons to why they do not feel represented. When it comes to making decisions we believe that middle management does not understand the importance of involving the employees. A discussion on a morning meeting or during the coffee break is perhaps all it takes to get the employees views on the matter and then they also can support the decision better.

Project priority indicator
Questions of relevance: 12, 42, 43 and 44

In question 12 we simply ask if the company prioritises their project. 70 % say yes. Notable but not statistically provable is to see that half of the remaining 30 % are people that has a management position. Developers clearly believe that there is an outspoken prioritisation between projects.

A majority of 53 % though believes that there are no different levels of prioritisation when it comes to requirements. Our belief is, as we mentioned in the previous chapter that prioritisation build up a better decision process. If TS are going to start prioritise they will have a good ground to stand on.

Customer specific dependant indicator
Questions of relevance: 54 - 58

9.1.3 Information flow

Strategic information indicator
Questions of relevance: 4, 5, 14, 15 and 16

We have found that the employees are reviewing strategic pages on the Intranet more than once a week. Additional information regarding the strategy is sent out monthly by the management of the company. The view of the strategy by corporate management and the views of the individuals seem to be aligned according to the questionnaire. However, in the question concerning the implementation of the strategy about one-third of the employees are dissatisfied.

The result of this indicator is satisfying and this is mostly an effect of the employees' eagerness to gain knowledge of the strategy development. The alignment of corporate strategy and the individuals' view of the strategy are comforting. There are, however, discrepancies on how the strategy should be implemented and how it is actually made. The employees seem to have a clear view of the current business of the company.

An explanation to the misfit in the implementation could be the frequency of information from the management. This could be explained by the repeated change in the strategy and that the management does not find it extensively important to inform the minor changes.

Priority of information indicator
Questions of relevance: none

Formal meetings regarding company strategy indicator
Questions of relevance: 15

Departments alignment to overall strategy indicator
Questions of relevance: 11, 15, 16, 52, 57, 58

Approximately 50% believe that the different departments are well represented when it comes to taking strategic decision. 50% though believe that they are less well or badly represented when it comes to taking strategic decisions. For the management the numbers are 75 / 25 %. Why almost 50 % think they are less well represented can have many reasons. One reason could be that top management does not prioritise the software development departments' needs or requirements as high as others. Another reason could be that middle management has not found ways to express the departments' concerns or they have difficulties to inform developers about the development. If the decisions are supported by good arguments then developers will find that they are represented when it comes to strategic decisions.

When it comes to question 52 we believe that more work effort should be put into documentation, reflection and feedback. These are important factors to the company's learning process, both towards the customer and internally. The documentation also prevents new employees to make the same mistakes again. A goal with clearer and more effective documentation system should be put into the overall strategy.

9.1.4 Specification process

Development goal indicator
Questions of relevance: 27 and 30 - 32

Through the results from the survey we have found that the amount of requirements that are formally defined is more than 60 %. Another finding is that the amount of requirements that the developers are able to influence is approximately 50 % in the case of internal requirements and approximately 30 % in the external case. This is then connected with the actual test effort on the amount of requirements that are tested and confirmed prior to release, which resulted in that only 20 % is tested.

Our findings indicate that the actual formalism in the requirement definition is satisfactory, as the research department is included in this survey. The formalism is, however, too static and the amount of requirements that the developers are able to influence is not satisfactory and might cause divergence between goals. Additionally, the confirmation of goal fulfilment is not adequate in the development process. The trend is however positive and has evolved from fifteen to twenty percent.

The complexity of the product developed and the multitude of possibilities to achieve the same result affect the result of this indicator. The change of operating system at a rapid pace has influenced this indicator as well. Another obstacle is the importance of time-to-market, which has decreased the development time and thus the effort of testing.

Priority of requirement indicator
Questions of relevance: 42 - 44

Without reflecting on the actual result of the questions referring to this indicator in the survey, most managers are stating the absence of prioritisation of requirements in the development department. Those of the employees and managers stating the existence of prioritisation are mostly arguing that four different levels are available. Unfortunately, the total sum of answers on the number of levels is a bit below statistical certainty.

From these results we draw the conclusion that the prioritisation is unsatisfactory and there seems to be disparate answers between both managers and employees. In those cases where there has been a positive answer on prioritisation most of them are determined that the requirements are implemented according to these levels.

However, the equal number of positive and negative answers on prioritisation indicates a defective priority management.

In the supporting questions to this indicator we are not able to completely ignore that the questions might be interpreted in more than one way. Additionally, if the management of the company is solely assigning tasks that the software department is actually going to complete, prioritisation is not a matter for this department.

Requirement change window indicator
Questions of relevance: 13, 20 - 24

More than 50 % of the employees state that new requirements often come forth after the initial deadline. In accordance with this more than 50 % of the projects conducted by the company are finished after deadline. Additionally, around 10 % of the projects are finished early or on time. In both cases the difference between estimated and actual time is about 25 %.

According to the results we find that the company is more reactive in the specification process than stated in strategy discussions. This affects the estimation of time and other resources and causes the company to continually race against time. If this indicator remains at current level, the company will have difficulties in reaching the release dates in future products as it is today. To compete on the market new requirements occasionally have to be introduced late in the development process. However, the changes after deadline are at present more a rule than an exception.

Some difficulties have developed in conjunction with the change of operating system for the Tribon system and are thus explaining the low result. Other obstacles to this indicator lie in the introduction of new tools and this might cause a non-stable development process.

Requirement cost indicator
Questions of relevance: 13 and 48

According to the survey, approximately 50 % of the employees state that requirements are often introduced past the initially set development deadline. On the question of the cost awareness of the employees about 50 % of them declare that they have an adequate level of understanding on the costs to implement the requirements.

The survey gives a satisfactory result on the question of awareness of the costs in the development department. Some problems in the cost awareness are articulated, as new requirements are introduced more than occasionally past initial deadline. Without any proper prioritisation in the organisation this situation could be hazardous to the return on investment in the development department.

Documentation standard indicator
Questions of relevance: 50 and 51

In approximately 20 % of the projects are documentation standards used to define events and decisions made in the development process. Most of the employees, more than 50 %, find these documentation standards unsatisfactory.

We argue that the results on the amount of used documentation standards are satisfactory according to the size of the company. There is, unfortunately, an inadequate result on the alignment of these standards to the actual work performed in the development department.

There might be several reasons for the misalignment of the documentation standards and one possible reason is the informal structure in the company.

9.2 Perspectives of operational level

9.2.1 Resource planning

Release date accuracy
Questions of relevance: 13

This indicator can only be analysed indirectly through other indicators and the outcome of question 13 which deals with how often new requirements are accepted after the original freezing date. Our observations indicate that the company's ability to estimate resource requirement is not very good and new requirements often are added to a release after the original freezing date.

We think that there are reasons to believe that the company has some difficulties with keeping major release dates. How severe these delays are and what consequences they have on their relationships towards their customer are hard for us to judge and therefore there is no further analysis of this indicator. However, we do believe that incurring new demand after the freezing date is of strategic importance and therefore decisions to do so must be taken by the appropriate board and not taken too lightly.

Individual burden indicator
Questions of relevance: 19

The answers to question 19 relating to this indicator are ranged widely and they are ranged from working on one to twenty different projects per week. Due to this wide range one might wonder if the question has been specific enough to be able to draw any conclusions from the answers given. However, due to our empirical research at the company we find that an average of 4.5 projects might be a reasonable outcome of the question.

If stated average is representative, then our judgement is that it is a little bit too high. At least if they are carried out in parallel, which the questionnaire unfortunately does not answer. But if this is the case then the result might indicate why the project estimates are a little bit of the mark. Projects are probably budgeted as they were isolated events, which they apparently are not.

One reason why the average is slightly above what is considered to be optimal could be that customer support reports are hard to predict and thus to plan. Since these reports must be dealt with swiftly they have a tendency to interrupt ongoing work. This means that it will be hard to estimate the amount of project that an individual will be forced to carry out in any given time period and thus the switching costs between different projects.

Clarity of project interdependence indicator
Questions of relevance: 29, 40 and 45

The survey points out that the developers often have the information on which parts of the product that are affected by the projects. According to the results this information is gained through individual knowledge of the product and not in the specification of the project. The employees have knowledge, gained through development or testing, of approximately 50 % of the total product.

The company seems to have a satisfactory knowledge of the overall product and new employees are introduced to the basis of the product at the beginning. The specification of the project scope is thus not that important as otherwise. Without an extensive new-recruitment the scope could be managed in the same way as at present, but in the long-range a specification is preferred.

9.2.2 Organisational structure

Support of decisions indicator
Questions of relevance: 15, 16

9.2.3 Information flow

Project co-ordination indicator
Questions of relevance: 6, 7 and 20

The questionnaire has shown an average of 2,8 co-ordination meetings/per quarter of a year. Is this enough though? When it comes to larger projects this is probably enough but there should be a possibility to arrange more frequent meetings for those projects that are not covered by these meetings. If the projects are not co-ordinated the risk of sub-optimisation is apparent.

Status reports are given on an average once a week. We believe that this is proper for a formal report system. We have also found that within TS there is an extensive informal network of communication that supplies any other important information.

Of course meetings are not the only way to co-ordinate project. In question 20 we ask how often the projects at the department are updated? The average here is once a month. This indicates in our meaning that there is a lack of ways to co-ordinate the projects and activities in the company. A more functioning report system would raise this indicator positively.

Individual dependence indicator
Questions of relevance: 53

Only nine persons have answered the question and thus we are not able to draw any considerable conclusions regarding this indicator. Why they have not answered is hard to say but one reason could be that they have misunderstood the question. Another could be that they are unsure about their answer and instead of answering it randomly they do not answer at all.

A positive effect of getting the same answer is that the company knows were to find the expertise. It also says something about the informal communication network, i.e. that it is working very well.

If these key persons were to leave the company today we believe that TS would get a lot of problems in the initial independence. Actions to prevent this from happening is to involve more people into the critical phases of the process or/and create a formal documentation system that can absorb the tacit knowledge in a better way. These actions we find absolutely necessary if the company has the ambition to grow in number of employees.

Supportive tools alignment indicator
Questions of relevance: 19, 28, 33, 50 and 51

The number of projects where management tools are used is fairly about 50 %. In the aspect of how different versions are handled in the company, it seems that most of the employees are very satisfied with the configuration management tool Clear Case. The documentation aspect of this indicator is vaguely supported and there is an indication that the documentation is non-satisfactory.

According to our judgement the alignment of the configuration management tool, Clear Case, to the development process is vital to the company. This aspect of the development process is well managed. Although the use of project management tools is only fifty percent, this is also satisfactory. Many projects are only of minor size and are thus not handled by a defined project management technique. The use of documentation standards in the development process is, however, unsatisfactory and might be fatal to the management of requirements.

One explanation of the lack of documentation standards is the pace at which the company is competing. There is not enough time to create a common standard of documentation. The overall finding of this indicator is satisfactory to some extent. Nevertheless the lack of documentation standards is influencing our view negatively.

9.2.4 Specification process

Requirements motivation indicator
Questions of relevance: 29, 46 and 58

We have found that the motivation of which customers that will use the implemented requirements are less understood than expected from our initial survey. Our findings indicate that the company is relying mainly on externally and joint funded projects. The insufficient specification of which customers are demanding the requirements should be analysed according to the dependence on externally funded projects. Another aspect in this matter is that the ability to influence the requirements is scarce.

Due to our judgement this indicator gives an inadequate result. According to the survey, the company is mainly reactive and the projects are funded externally to a great extent. It is therefore not satisfactory if the origins of the requirements are not defined. There is a risk in this situation that requirements are not fulfilled in accordance to the intention of the customer.

One possible explanation to this result lies in the individual understanding in the development department of different aspects that are of concern to a broad range of customers. In other words the customers do not differ to a great extent. Another explanation is that the requirements emanate from only the main customers and their interests and aspects are thoroughly understood.

Requirements testability indicator
Questions of relevance: 30, 31 and 32

In the survey we have found that the amount of cases where it is possible to test the requirements are over 80 %. The survey indicates that 16 % of the past functionality has formal regression tests. In the current development process this figure has increased to over 20 %, according to the survey.

The trend of this indicator seems to be positive. However, past performances in the development of formal tests still affect this indicator. This is due to the possibility to do formal regression tests of the product are way below 20 % of the functionality. If the company were able to automate the requirement testing to half of the requirements that are testable this would be satisfactory.

As this indicator is closely connected with the development goal indicator the same obstacles affect this indicator. One possible explanation for the inadequate result of this indicator might then be the multitude of operations that causes the same result.

Software building indicator
Questions of relevance: 39

We have examined the level of linkage problems in the compilation of the program. In this we have found that there are seldom problems in the linkage due to a single module. If problems do occur then the problems are generally taking approximately seventy hours to correct. This indicator could have been supported by similar questions in the survey. This is, however, one of the most essential.

According to our judgement the values presented to us on the linkage accomplishment are very satisfactory. Some problems do occur from time to time in the compilation and linkage of the product, but they are rare. This indicates that the current process has benefits to the outcome of the development process. In those cases where problems appear these are handled within ten man-days. There is a range in the time to correct the errors, from one day to hundred man-days, which is a rather disturbing difference.

This indicator could have been more thoroughly supported, as previously stated. We have although gained enough knowledge of the difficulties that emerges when problems are revealed. Ten days might not seem dreadful when referring to the time it takes to accomplish a project, but the better the specification process is the better this indicator will become.

Degree of automated test indicator
Questions of relevance: 27, 30, 31 and 41

When examining the results of the survey we found that in 7 % of the α-testing were made by automated methods. In 15 % of the available functions, formal tests are available to control the functionality. The total amount of testable functions in the current development process is more than two-third. The same amount is indicated in the question on the amount of the functions that have formal goals to fulfil.

While the company has developed the product during an extensive time period, it is surprisingly that they get low values on the amount of automatic tests performed at α-test level. However, the amount of automatically tests performed at α-level is about half of the available tests developed. This could have been a satisfactory figure if not the total amount of developed formal tests had differed from the total possible tests of functionality to current extent.

One contribution to the problem in this indicator might be the change of operational system made during 1999. The management forced this change and several reductions of the product were inevitable to launch the product in the limited window of opportunity. Other obstacles might be the complexity of the product and the multitude of possible approaches to gain the same end-result.

9.3 Interconnecting the strategic and operational levels

9.3.1 Resource planning

Resource evaluation indicator
Questions of relevance: 52

Time report system indicator
Questions of relevance: 25

The questionnaire indicates that the time report system reflects the actual work within the software development department well.

According to our opinion the result implies that the managers and developers within the software development department have a good foundation to learn from and evaluate past decisions. However, this system alone can only be used reactively since managers only can tell when a project has overdrawn its time budget, not before due to the absence of future project estimates.

The reason why this system only can be used reactively is because most time report systems are designed to be merely administrative and not a part of the active management system.

9.3.2 Organisational structure

Documentation alignment to actual work procedure indicator
Questions of relevance: 2 and 25

The usage of the present system differs to a great extent in our questionnaire. The majority uses the time report system either once a month or more then five times a month. Empirical studies have shown that the eager to fill out the time report could be an age issue. Younger personnel have a tendency to just fill it out once, i.e. when it is time to get paid. Perhaps they feel like it is unnecessary to enter the system and instead writes it down on a piece of paper.

70 % of the employees believe that the actual time spent coheres with the reported time. If this is the case then the time report system could, besides being useful when the salary is being calculated, provide an important status check of spent resources in a project. This gives the middle management more information on the amount of resources that the department has available.

9.3.3 Information flow

Organisational learning indicator
Questions of relevance: none

9.3.4 Specification process

Feedback on quality at department level indicator
Questions of relevance: 7, 27, 34, 36 and 37

According to our survey we have found that the amount of developed functionality that the department gets feedback on is approximately two-third. Close to the same value is collected when we investigate the amount of functionality where formal goals are defined. The department gets reports of the status on the projects almost once every week. If the feedback on projects is considered to reflect the work in the department, then the support of feedback at departmental level is about 50 %.

Due to our findings we conclude that the feedback process of the company is satisfactory. If we compare the amount of formal defined requirements and the amount of feedback to the department this indicate a high level of feedback to the department. However, there are some differences between the amount of feedback to the department and the individuals. By improving this aspect then this indicator would gain an even higher level.

The reason why the departments are not getting feedback on all of the work might lie in the information flow from the marketing department. Another reason is the complexity of the product and that the possibility to detect all improvements and changes are not easily done. Additional is the rate of formal goals to the requirements affecting the value of this indicator.

Feedback on quality at individual level indicator
Questions of relevance: 8, 9, 35, 36, 37 and 38

There is willingness within the development process to be evaluated at an individual level. Most of the employees would like to be evaluated at least at the end of every project. The value of the amount of work that the individuals get feedback on is less than 50 %. Accordingly, most of the projects are not gaining feedback from the individuals on the performance of the work in the project.

We think the amount of feedback to the individuals is below what is appropriate. This is indicated by the fact that the individuals are seldom getting feedback on how the projects were concluded. Additionally, more than half of the work at individual level is evaluated and there is demand for more feedback during and after completion of each project. To improve the quality at individual level this indicator has to be enhanced.

One possible reason for the result of this indicator is the amount of minor projects in the company, which are not easily evaluated. Another reason is the informal attitude in the organisation and that the feedback is gained in an informal and non-structured way. The result on the feedback at the departmental level is also affecting the outcome of this indicator.

Overall system knowledge indicator
Questions of relevance: 11, 40 and 49

The employees have experience from approximately 50 % of the total system due to either implementation or testing. As new projects are initiated knowledge on what other projects that are dependent on the new project is often identified. In addition the representation of the different departments when decisions are made is good according to the questionnaire.

According to the results we find that the employees have a sufficient knowledge on the different parts of the total system. Although there are more than 50 % stating that they have knowledge on what other projects depending on their own projects, 25 % state that they never have this knowledge at project start. This is connected to that the departmental representation when decisions are made is slightly negative.

One possible reason for the result of this indicator is the size of the total system, which emphasises the difficulty to posse's knowledge on the overall system to a greater extent. Another reason for the result of this indicator is the amount of minor projects within the company, which might not be known by the rest of the organisation.

9.4 Aggregated matrix of indicators

9.4.1 Strategic level

Resource planning	Organisational structure	Information flow	Specification process
Aggregated project status	*Company reactiveness*	*Strategic information*	*Development goal*
Customer response	*Department representation*	*Priority of information*	*Priority of requirement*
Reactive or proactive development	*Project priority*	*Formal meetings regarding company strategy*	*Requirement change window*
Resource estimation	*Customer specific dependant*	*Departments alignment to overall strategy*	*Requirement cost*
			Documentation standard

9.4.2 Operational level

Resource planning	Organisational structure	Information flow	Specification process
Release date accuracy	*Support of decisions*	*Project co-ordination*	*Requirements motivation*
Release content		*Individual dependence*	*Requirements testability*
Release workload		*Supportive tools alignment*	*Building software*
Individual burden			*Degree of automated test*
Clarity of project interdependence			

9.4.3 Interconnecting the strategic and operational levels

Resource planning	Organisational structure	Information flow	Specification process
Resource evaluation	*Documentation alignment to actual work procedure*	*Organisational learning*	*Feedback on quality at department level*
Time report system			*Feedback on quality at individual level*
			Overall system knowledge

9.5 Aggregated results according to the analysis

Very good	★★★★
Good	☺☺☺
Less well	😐😐
Bad	☹
No result	?

9.5.1 Strategic level

Resource planning	Organisational structure	Information flow	Specification process
😐😐	☺☺☺	☺☺☺	😐😐
☺☺☺	😐😐	?	😐😐
😐😐	😐😐	?	☹
?	?	😐😐	☺☺☺
			😐😐

Operational level

Resource planning	Organisational structure	Information flow	Specification process
☹	☺☺☺	☹	😐😐
?		☺☺☺	😐😐
?		☺☺☺	☺☺☺
😐😐			😐😐
☹			

Interconnecting the strategic and operational levels

Resource planning	Organisational structure	Information flow	Specification process
?	☺☺☺	?	☺☺☺
😐😐			😐😐
			☺☺☺

9.6 Concluding remarks on the analysis

Before we conclude this analysis we, the authors want to say that we believe that TS has all potential to change its performance in the software development. We are not supposed to give a grade or a judgement. With this analysis we just want to give directions to TS or an outside investor where the company is heading. The result of the indicators reflects the temperature of the company.

At the strategic level two similar indicators are shown the reactive or proactive development indicator and the company reactive indicator. These two express the same aspects about strategy but on different levels in the organisation. We believe that it is better to be reactive as a company than being fully reactive at the development level. At a company level you have to consider other departments reactiveness towards the customer. But when it comes to the software development department they have to be proactive in order to predict the future needs of the customer. We believe though that TS on an overall level could become more proactive. Perhaps this would decrease the numbers of CSR's.

Aggregated project status indicator is one of the most important indicators on the strategic level. This indicator is not satisfactory within TS. Three indicators are clearly connected to this indicator, i.e. Resource estimation indicator, Department alignment to overall strategy indicator and Requirement change window indicator. The aggregated project plan is a large part of the resource planning strategy. Since time is one of the most important resources in TS it has to be correctly estimated in order to create a resource strategy. This strategy has to be ventilated with the employees so that they support and work according to the aggregated project plan. Too many late requirements create an imbalance in the resource strategy and thereby the aggregated project plan. We believe that TS lacks this kind of project plan and will suffer in the long run if they do not consider these aspects.

Since resource planning is one of the main aspects that we consider in this thesis, we believe that Project co-ordination indicator is the most important indicator at the operational level. TS's project co-ordination indicator is not satisfactory. The indicator is closely connected to the resource planning indicators and the Clarity of project interdependence indicator. The resource planning indicators are all measurements of how well the resource planning function at the operational level. We believe that this has something to do with the information flow and how you co-ordinate the projects. The knowledge about connections between different projects is there though and this gives us good hope that the problem will be overcome if TS looks closer into resource planning.

An extensive use of supportive tools at the operational level would perhaps also be a factor that could help them document and structure the project. Today TS uses such tools but there could be many more purposes of these tools.

Feedback is one of these areas where the tools could be useful. The most important section at the Feedback level is the specification process. The indicators show that the departments get the satisfactory feedback on their work but not the employees. There is a willingness of the employees to improve their work at an individual level. Direct concrete feedback is a way to improve individual work. Why TS has not responded to this need from the employees is not clear. Perhaps they are satisfied with the work done? Individuals want to improve and develop new skills. They can not become better if they do not know if they are doing the right thing.

At an overall level we believe that the indicators describe the software development process at TS well. Some of the indicators have been misunderstood and not answered the way they should have been, but when we summarised them all we are satisfied. TS has difficulties in their efforts to estimate the resources the development process takes. This is a common but very serious problem of software companies today. The industry has much to learn by looking at other domains, as we have done.

We do sense a presence of a strong information flow in the company but the efforts to formalise and gather this information are few. From our results we draw the conclusion that a formalisation is needed to support the resource planning process. This is of course without any intrusions on the creative parts of the development process.

In order to be able to compete on a larger international market TS has an intention to grow in numbers of employees. We believe it is important though that the culture of the company is kept intact to preserve the positive things about the organisational structure and the information flow. Much can be lost and gained with this strategy.

When it comes to an overall view we think that TS is heading in the right direction when it comes to improving the specification process. We believe though that they still have much to learn when it comes to fixating the requirement specification at an early stage in the development process.

The matrix of indicators could be further developed, but this is an issue for further reference. We believe that the indicators that we have stated here are adequate enough to make the analysis we hereby presented.

10 Conclusions

> This is not the end. It is not even the beginning of the end. But it is, perhaps, the end of the beginning
> - Winston Churchill

In the beginning of this report a purpose was set and the efforts were then directed towards this goal. In this final chapter of the report the aim is to discuss the findings made during the research and the theoretical contribution we claim to present. The focus is on a discussion if the indicators are able to reflect the actual capabilities of the development process. In other words, the company's ability to develop products that outperform the competitors.

As we draw conclusions from our observations of the analysis of TS by the developed indicators, we evaluate the probabilities and estimate whether they are sufficient to establish proofs. Unfortunately, we cannot claim that the developed indicators are sufficiently tested to statistically define them as real indicators of the capabilities of the development process. What are then the conclusions and theoretical achievements if not the indicators are statistically confirmed?

There is a need for our indicators: By developing and using the indicators supporting the reference models that are suggested in this master thesis, the company is forced to learn new things about product development and its environment. At first glance, the reference models and the corresponding indicators could be yet another capability measurement technique. However, we state it is actually a new form of evaluation of the management system that links strategic and operational levels of the organisation to the needs of the customers.

Development processes differ: As supported research indicates that no development process is applicable to every situation, then aspects of resource planning, organisational structure, information flow and specification process play a vital role in the product development. To manage the differences, the indicators are based on basic questions and assumptions and are thus an applicable tool throughout the organisation. It requires both the strategic and operational levels of the organisation to reflect its position from the four different perspectives and to answer basic questions regarding the development process.

The analysis of the indicators is manageable: Having understood the basic assumptions behind the four perspectives, the company is able to create a method for collecting the required data. In the analysis of TS an elementary questionnaire was developed and by slight adaptations it could be used in other organisations. If an

appropriate data requirement method is at hand, the management is able to set targets on the four perspectives and then measure the achievements and improvements. These targets are then communicated to management and staff within the company in order to create an understanding throughout the organisation how individual efforts contribute to company development and vice verse.

Theoretical achievements: In a survey on the matter of evaluating the software development process several research areas are approached. We have found it unreasonable to confront each and every one of them thoroughly. However, we state that the four perspectives are sufficiently described to support the current definitions of the indicators. By the development of the reference models we argue that new applications of available theories are concluded. It is foremost in the bond between strategic issues in the software development department and operational aspects of the software development process that the main contributions are laid. We argue that there is generally no direct connection between cause and effect in this system.

Unbalanced theoretical support: In the development of the theoretical background, we have found ourselves questioning whether different theoretical aspects shall be contained in one perspective or in another. From frequent questioning we have drawn the conclusion that the four perspectives are closely interconnected. As a consequence one can, in some cases, argue that too much attention is laid in one area while other areas seem to be incomplete.

Preferred refinement of the indicators: The current status of the indicators is that they need refinement and feedback by application on a wider population. We do not regard the indicators as static. Instead we claim by using them as tools for both benchmarking and internal evaluation they will appear to be slightly dynamic. Although, the definitions might change to some extent and the substance might, in some cases get questionable, we state that the specific indicators are meaningful to the overall understanding.

Remaining questions: In future work there is a need to refine the indicators through a systematic research in a broad range of development processes in different software industries. By this there will be an ability to refine the scope of the indicators, so that they can be expressed in a more complete and systematic manner. In the end, there will be a possibility to propose evaluation methods or techniques that are more general. A method of evaluation would include appropriate indicators, forms of data, forms of analysis and techniques of interpretation to produce the valuation of the development process.

Finally: To conclude the last part of the report we would like to make some final remarks. Do not use these indicators too simply in the development process. Each software development company is unique and each has different obligations towards its customers and employees. The developed indicators are solely intended as relative indicators, and not as absolute measures. Correctly used they can provide a guideline to both the management of the development process and its employees.

References

Bibliography

Álvarez	J. Álvarez, N. Castell and O. Slavkova, *Combining Knowledge and Metrics to Control Software Quality Factors*, Achieving Quality in Software, Chapman & Hall, 1996
Anthony	R. Anthony, V. Govindarajan, *Management control systems: ninth edition*, McGraw-Hill, 1998
Basili	V. R. Basili, G. Caldiera and H.D. Rombach, *The Experience Factory*, Encylopedia of Software Engineering, vol. 1, John Wiley & Son Ltd, 1994
Ben-Ari	M. Ben-Ari, *Principles of Concurrent and Distributed Programming*, Prentice Hall Europe, 1990
Bergman	B. Bergman and B. Klefsjö, *Quality – from Customer Needs to Customer Satisfaction*, Studentlitteratur, Lund, Sweden, 1994
Boehm	B. Boehm, *The Art of Expectations Management*, IEEE Computer, January 2000, IEEE Computer Society, 2000
Bratthall	L. Bratthall, P. Runeson, K. Adelswärd and W. Eriksson, *A Survey of Lead-Time Challenges in the Development and Evolution of Distributed Real-Time Systems*, To appear in Journal of Information and Software Technologies, 2000.
Briner	W. Briner, C. Hastings and Michael Geddes, *Project Leadership*, Ashridge Managagement College, Gower, 1997
Brooks	F. Brooks, *No Silver Bullet – Essence and Accidents of Software Engineering*, IEEE Computer, April 1987, IEEE Computer Society, 1987
Brooks	F. Brooks, *The Mythical Man-Month: Essays on Software Engineering*, Addison-Wesley, Reading, Massachusetts, 1995
Brown	S. Brown and K. Eisenhardt, *The Art of Continuous Change: Linking Complexity Theory and Time-paced Evolution in Relentlessly Shifting Organizations*, Administrative Science Quarterly, March 1997, Cornell University, 1997
Bruzelius	L. Bruzelius and P-H Skärvad, *Integrerad organisationslära*, Studentlitteratur, Lund, 1995 (In Swedish)
Callahan	J. Callahan and S. Easterbrook, *Verification and Validation in a Rapid Software Development Process*, NASA Goddard Software Engineering Workshop, December 3-4, NASA, 1997
Cooper	R. Cooper and M. Press, *The Design Agenda: a guide to successful design management*, John Wiley & Sons Ltd, 1995
Cugola	G. Cugola and C. Ghezzi, *Software Processes: a Retrospective and a Path to the Future*, Software Process Improvement and Practice, 4, John Wiley & Sons Ltd, 1998

Davies	C. Davies and P. J. Layzell, *The Jackson Approach to System Development*, Chartwell-Bratt, 1993
Deal	T. Deal and A. Kennedy, *Corporate cultures,* översättning Företagskulturer av Christina Hornberger, Stockholm Sv.Dagbl., 1985. (In Swedish)
DeMarco	T. DeMarco, *Controlling software projects : management, measurement & estimation*, Yourdon Press, New York, 1982
Drucker	P. Drucker, *The effective executive*, HarperBusiness, 1996
Edvinsson	L. Edvinsson and M. Malone, *Intellectual capital : the proven way to establish your company's real value by measuring its hidden brainpower*, Piatkus, 1997
Eneroth	K.Eneroth, *The dynamic side of destinctive competence – a relational resourcebased perspective,* Working paper – Lunds University, 1993.
Fenton	N. Fenton and S. L. Pfleeger, *Software Metrics – A Rigorous and Practical Approach*, International Thomson Computer Press, 1997
Fairplay	Fairplay, *Will the fortress fall?*, Fairplay, February, 2000
Ford	D. Ford, L-E Gadde, H. Håkansson, A. Lundgren, I. Snehota, P. Turnbull and D. Wilson, *Managing Business Relationships*, John Wiley & Sons Ltd, 1998
Frankl	V. Frankl, *Viljan till mening,* Natur och Kultur, Stockholm, 1995 (In Swedish)
Frykman	D. Frykman and J. Trolleryd, *Värdering av tillväxtföretag – en handbok med modeller och metoder anpassade för tillväxtföretag*, Ingenjörsvetenskapsakademin (IVA), 2000 (In Swedish)
Giertz	B. Giertz, *Uppfattningar av kvalitet - En genomgång av litteratur om kvalitet och kvalitetsarbete*, Rapportserie från Enheten för utveckling och utvärdering, Uppsala universitet, 2000 (In Swedish)
Grant	R. Grant, *Contemporary strategy analysis : concepts, techniques, applications*, Blackwell, 1998
Green	D. Green and others, *Cognitive Science – an introduction,* Blackwell Publishers, 1998 (In Swedish)
Halvorsen	K. Halvorsen, *Samhällsvetenskaplig metod*, Studentlitteratur, Lund, 1992 (In Swedish)
Hamlyn	D. Hamlyn, *Filosofins historia*, Månpocket, 1998, (In Swedish)
Hammer	M. Hammer, *Beyond reengineering – how the process-centered organization is changing our work and our lives*, Harper Business, New York, 1996
Hantos	P. Hantos and M. Gisbert, *Identifying Software Productivity Improvement Approaches and Risks: Construction Industry Case Study*, IEEE Software, January/February 2000, IEEE Computer Society, 2000
Hill	T. Hill, *Manufacturing strategy: text and cases,* Macmillan: Basingstoke, 1995

Holme	I. M. Holme and B. K. Solvang, *Forskningsmetodik – Om kvalitativa och kvantitativa metoder*, Studentlitteratur, 1991, (In Swedish)
Holsapple	C. W. Holsapple and A. B. Whinston, *Recent Developments in Decision Support Systems*, Springer-Verlag, 1993
Humphrey	W. S. Humphrey, *Managing the Software Process*, Addison-Wesley, 1989
IEEE	IEEE Std 830, *IEEE Guide to Software Requirements Specifications*, Institute of Electrical and Electronics Engineers, New York, 1984
IVA	IVA, *"Bäst i världen": vad kan vi lära av världens ledande företag?*, Ingenjörsvetenskapsakademin (IVA) : Solna, 1991 (In Swedish)
Johnson	T. Johnson and R. Kaplan, *Relevance Lost: The Rise and Fall of Management Accounting*, Harvard Business School, 1987
Juell-Skielse	G. Juell-Skielse and K. Askerfelt, *Organisationens väg till en kompetensekonomi*, Liber Ekonomi, Malmö, Sweden, 1997 (In Swedish)
Kaplan	R. Kaplan and D. Norton, *Using the Balanced Scorecard as a Strategic Management System*, Harvard Business Review, January-February, 1996
Kerzner	H. Kerzner, *Project Management: A systems approach to planning, scheduling and controlling*, Van Nordstand Reinhold Company, 1979
Landis	L. Landis, S. Waligora and F. McGarry, Recommended Approach to Software Development (Revision 3), NASA SEL, 1992
Lennerlöf	L. Lennerlöf, *Människan i arbetslivet*, Beteendevetenskaplig arbetsmiljöforskning, Allmäna förlaget, Stockholm, Sweden, 1991 (In Swedish)
Lignell	M. Lignell, H. Andersson, M. Arblom, B. Backström, S. Molin, G. Nilsson and Ö. Sundblad, *Att värdera operativ försvarsförmåga – definitioner och förslag*, Division of Defence Analysis, Defence Research Establishment (FOA), 1998 (In Swedish)
Lindvall	M. Lindvall, *CMM för dem som konkurrerar med processer*, Computer Sweden, 20, 2000
Lundahl	U. Lundahl and P-H Skärvad, *Utredningsmetodik för samhällsvetare och ekonomer*, Studentlitteratur, Lund, 1999 (In Swedish)
Merriam	Merriam, Sharan, *Case study researc in education*, Swedish title: Fallstudien som forskningmetod, Studentlitteratur, 1994.
Mossberg	T. Mossberg, *Utveckling av nyckeltal*, Ekonomiska forskningsinstitutionen vid Handelshögskolan, Stockholm, 1977 (In Swedish)
Olsson	F. Olsson, *Systematisk konstruktion: en studie med syfte att systematisera innehåll och metoder i samband med produktkonstruktion*, Department of Design Science, Lund Institute of Technology, 1976 (In Swedish)

Olsson	F. Olsson, *Produktförnyelse: Förnyelseplanläggning Integrerad produktutveckling*, Department of Design Science, Lund Institute of Technology, 1985 (In Swedish)
Olsson	J. Olsson and P-H Skärvad, *Företagsekonomi 99*, Liber ekonomi, 1997 (In Swedish)
Paulk	M. Paulk, B. Curtis, M. Chrissis, and C. Weber, *Capability Maturity Model, Version 1.1*, IEEE Software, July 1993, IEEE Computer Society, 1993
PMI	PMI, *A Guide to the Project management Body of Knowledge*, Project Management institute, 1999
Porter	M. Porter, *Competitive advantage: creating and sustaining superior performance*, Free Press, 1998
Prigogine	I. Prigogine and I. Stengers, *Order out of chaos: man's new dialogue with nature*, Bantam Books: New York, 1984
Reel	J. Reel, *Critical Success Factors in Software Development*, IEEE Software, May/June 1999, IEEE Computer Society, 1999
Regnell	B. Regnell, P. Beremark and O. Eklundh, *A Market-Driven Requirements Engineering Process Results from an Industrial Process Improvement Programme*, Journal of Requirements Engineering, 3:121-129, Springer-Verlag, 1998
Rus	I. Rus and M. Lindvall, *Process diversity in software development*, IEEE Software, July-August, 2000
Sage	A. Sage and A. Jensen, *Systematic Measurement*, Handbook of Systems Engineering and Management, John Wiley & Sons, 1999
Sawyer	P. Sawyer, I. Sommerville and G. Kotonya, *Improving Market-Driven RE Processes*, International Conference on Product Focused Software Process Improvement, Department of Information Processing Science, University of Oulu, Finland, 1999
Schultz-Larsen	O. Schultz Larsen, *Utvecklingspsykologi*, Studentlitteratur, Lund, Sweden, 1994 (In Swedish)
Senge	P. Senge, *The fifth discipline: the art and practice of the learning organization*, Century Business, 1993
Socialhögskolan	Department of Social science at Lund University, *Desicions and decisions processes*, Handout from Departmet of Social science at Lund University, 1985 (In Swedish)
Sommerville	I. Sommerville, *Software Engineering*, Addison-Wesley, 1995
Starkey	K. Starkey, *How organizations learn*, Routledge, 1996
Svedberg	L. Svedberg, *Gruppsykologi – Om grupper, organisationer och ledarskap*, Studentlitteratur, Lund, 1997 (In Swedish)
Sveiby	K-E Sveiby, *Kunskapsflödet – Organisationens immateriella tillgångar*, Svenska Dagbladet Förlags AB, 1995 (In Swedish)
Trade Wind	Trade Wind, 18, February, 2000
Ulrich	K. Ulrich and S. Eppinger, *Product Design and Development*, McGraw-Hill, 1995

van Solingen R. van Solingen and E. Berghout, *The Goal/Question/Metric Method - A Practical Guide for Quality Improvement of Software Development*, McGraw-Hill, 1999

van Weele A. van Weele, *Purchasing Management*, International Thomson Business Press, 1998

Weick K. Weick, *The Social Psychology of Organizing*, McGraw-Hill, 1979

Wheelwright S. Wheelwright and K. Clark, *Revolutionizing Product Development Quantum leaps in Speed, Efficiency and Quality*, Free Press, 1992

Whittaker J. Whittaker, *Process diversity in software development*, IEEE Software, January-February, 2000

Whittington R. Whittington, *What is strategy - and does it matter?*, International Thomson Business Press, 1993

Zuboff S. Zuboff, *In the age of the smart machine – the future of work and power*, Oxford Heinemann, 1998

Årzén K-E Årzén, *Real-Time Control Systems*, Department of Automatic Control, Lund Institute of Technology, 1997

Electronic sources

http://www.balancedscorecard.org 2000-08-01

http://www.holidays.net/mlk/ 2000-08-31

http://www.sei.cmu.edu/managing/managing.html 2000-04-10

http://www.sqi.gu.edu.au/SPICE/ 2000-04-10

http://www.eb.com 2000-02-25

http://sites.netscape.net/yachtdesigner/arsenal/highend.htm 2000-06-01

http://www.famous-quotations.com/ 2000-08-04

Interviews

Dahlqvist, Jan **Date:** 000525
Department/Title: 6th AP-fond, Investment Manager.

Edahl, Patrik **Date:** 000324
Department/Title: Accounting and administration

Eriksson, Ulf **Date:** 000327
Department/Title: Software development, product manager Hull.

Hamitogullari, Semra	**Date:** 000328 **Department/Title:** Support and Configuration Management
Lindahl, Ingemar	**Date:** 000327 **Department/Title:** Contract Management.
Lundh, Holger	**Date:** 000525 **Department/Title:** Sales and marketing
Nilsson, Per-Olof	**Date:** 000322 **Department/Title:** Research, vice President.
Ridderman, Mats	**Date:** 000404 **Department/Title:** Configuration Management
Roth, Rikard	**Date:** 000404 **Department/Title:** Software Development, vice President.
Westenius, Mats	**Date:** 000322 **Department/Title:** Research

A. Initial interviews

Introduction
Personal background

What are your tasks?

Company specific
How do you relate TS to…?
Strategy
Market
Customers
Competitors

Development process
Describe the development process at TS?

Draw a flowchart on the development process at TS?

To what content does TS investigate the development process of other companies?

How would you define product development?

In what way does the company take care of risks in product development?

Does financial requirements affect the product development?

Which departments are collaborating in and affecting the product development?

Product
Is there a development plan to each part of the product?

To what content does the company locate resources on maintenance?

Management
How are the employees evaluated?

Company culture
What values are affecting the employees at TS?

B. Indicators

In the following section is the questionnaire used for evaluation of the development process at TS presented. The results are then presented in the second section of this appendix.

B.1 Questionnaire

Questionnaire submitted in a final thesis at KCS

This questionnaire is a part of a final thesis initiated by the management at KCS and the 6th AP-fund. The authors are three students studying Technology Management in Lund.
The purpose of the final thesis is to illustrate the development process within a software company.
Our goal is to develop business ratios which captures this process and can be used when evaluating a company in both economic and in comparison with other companies in the same line of business.
The purpose of the questionnaire is to support our extracted business ratios

Important to note! The questionnaire is anonymous and will under no circumstances be traced to individuals. ***Important to note!***

Some questions inquire you to state Minimum/Average/Maximum the underlying reason is that we want you to estimate the worst case, in the ordinary cases and in the best case. If there is only one alternative then estimate the ordinary case i.e in average cases.

1 Which position is most aligned to your work within the company?

Manager ☐ Developer ☐ Other position ☐

2 How often do you use the company report systems(for example the time report system, the status report system ect)?

Once a month ☐
2-3 times per month ☐
4-5 times per month ☐
More than 5 times per month ☐

3 Does the company report system lack something that would support your work(For example, i the time report system, status report system ect)?

Yes ☐ No ☐

If yes, what do you lack, in priority order?

1 ☐
2 ☐
3 ☐
4 ☐
5 ☐

4 How often do you visit sites on the intranet that you consider are of strategical importance?

State minimum/average/maximum amount of times per month!

Minimum ☐
Average ☐
Maximum ☐

5 How often does the managerial body send out strategic information to the staff?

State quarterly Minimum/Average/Maximum amount of times!

Minimum ☐
Average ☐
Maximum ☐

6 At department level, how many project co-ordinating meetings do you have quarterly? på er avdelning?

State Minimum/Average/Maximum per quarter of a year!

Minimum ☐
Average ☐
Maximum ☐

7 How often do you report the status of your work to your nearest superior?

State Minimum/Average/Maximum amount of times per month!

Minimum ☐
Average ☐
Maximum ☐

8 Would you like to be evaluated by your project co-workers?

Yes ☐ No ☐

9 If yes on the prior question! How often and on what grounds (for example work result, ability to co-operate ect) would you like to be evaluated?

monthly ☐
quarterly ☐
yearly ☐

On what grounds!

1 ☐
2 ☐
3 ☐
4 ☐
5 ☐

10 How good are the project time estimates, budgeted compared with final outcome?

Very good ☐
Good ☐
Less well ☐
Bad ☐

11 Do you think that the product departments are well represented when decisions concerning different projects within the company are made?

Very well represented ☐
They are well represented ☐
Less well represented ☐
They are badly represented ☐

12 Is there any priorities set between different projects within the company(Customer Support Reports excluded)?

Yes ☐ No ☐

If yes, how well does this prioritization work?

Very good ☐
Good ☐
Less well ☐
Bad ☐

If no, do think that there ought to be one?

Yes ☐ No ☐

13 How often are new functionality requirements added after the original freezing date?

Never ☐
Somtimes ☐
Often ☐
Always ☐

14 How often do you recieve information from your nearest superior concerning the company's development taken at a strategic level?

State quarterly the Minimum/Average/Maximum amount of times!

Minimum ☐
Average ☐
Maximum ☐

15 How well do you think that the decisions taken at the strategic level within the organisation reflects your view of the company strategy?

Very good ☐
Good ☐
Less well ☐
Bad ☐

16 How well do you think that the ongoing projects within the organisation are aligned with the the decisions made on the comapany's development?

Very good ☐
Good ☐
Less well ☐
Bad ☐

17 Do you think that the company's decisions are proactive or reactive, i.e are the decisions a a product of having investigated customer needs or a reaction to urgent customer demands?

Mainly proactive ☐
Mainly reactive ☐

18 Has the share of reactive projects increased or is it indifferent comparing to last year?

Increased ☐
Indifferent ☐
Decreased ☐

19 How many ongoing projects do you have per week?

State Minimum/Average/Maximum number of projects per week!

Minimum ☐
Average ☐
Maximum ☐

20 How often updates, formally, what projects that the department has undertaken to carrie out?

Every week ☐
Every month ☐
Every year ☐

21 How often are projects finished earlier then estimated? (Number of project finished earlier then estimated/ total amount of projects)

State the average percentage of the projects ☐

22 How much in per cent is usually a project earlier then estimated?

(Estimated time-actual time)/Estimated time Average ☐

23 How often do projects take longer to complete then estimated? (Number of projects that take longer to complete/total amount of projects)

Ange procentuellt medeltal av projekten ☐

24 How much longer in percent does a project usually take to complete?

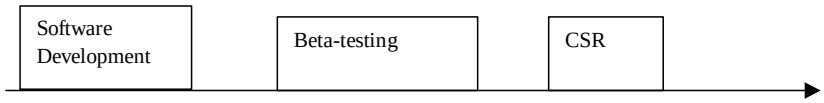
(Actual time-estimated time)/estimated time Average ☐

25 How well, at the project level, does the time reports reflect how the time is actually used?

Very good ☐
Good ☐
Less well ☐
Bad ☐

26 How big share of the resources do you consider that the software department spends on the three different areas development including alfa-testing, development ***as a result of*** that the product has been beta-tested and finaly customer related CSR development?

Software Development
Beta-testing
CSR



Software Development: includes all the work that is performed before the product is sent to beta-testing. That is project planning, actual coding(impementation), alfa testing ect.
Beta-testing: in order to estimate how much of the resources that are linked to beta-testing we will for the sake of simplicity seemingly freeze a project's cost when it has been forwarded beta-testing. The remaining development work until that the product has been delivered to the customer will be allocated to beta-testing.
CSR: how much of the software department's resources are devoted to correcting errors originates from customer related CSRs.

27 To what extent of the new functionality are there formal goals stating what functionality is to be developed?

Ange i procent/projekt

Minimum
Average
Maximum

28 To what extent of the requirements being implemented are there written/formal project management tools stated to be used?

State in per cent/project!

Minimum
Average
Maximum

29 To what extent of the requirements do you have a possibility to affect the composition of a requirement?

State in per cent!

Customer specific
Internal

30 To what extent of the existing functionality are there formal test developed in order to control that functionality goals are met?*(that the functionality is fulfilled*)

State the average per cent

Average

31 To what extent of the new functions being implemented is it possible to test whether the functionality is fulfilled?

Ange medel i procent

Average

32 To what extent of the new functionality are test cases developed in order to guarantee that the demands placed upon the new functionality are fulfiled?

State the average per cent

Average

33 How well is the version handling tool, Clear Case, adapted to the company's working practice?

Very good ☐
Good ☐
Less well ☐
Bad ☐

34 To what extent of the developed functions does the department recieve feedback on?

State the average per cent Average ☐

35 How often would you like to recieve formal feedback on your work?

After every project closure ☐
During and after project closure ☐
Once a month ☐
Once every half a year ☐
Once a year ☐

36 At what level within the company would you like to se that a verified failure report was investigate/traced?

Department ☐
Project ☐
Individual ☐
Not at all ☐

37 To what extent of your work do you recieve feedback?

State the average per cent Average ☐

38 To what extent do you give feedback on project performance after project closure?

Never ☐
Somtimes ☐
Often ☐
Always ☐

39 How often is a linking disrupted due to that a none buildable module has been checked in on the branch?

Never ☐
Somtimes ☐
Often ☐
Always ☐

How long time does it take to fix this error?

Minimum ☐
Average ☐
Maximum ☐

40 At project start, how often do you possess knowledge of what other projects that depend on your own project?

Never ☐
Somtimes ☐
Often ☐
Always ☐

41 To what extent are alfa-tests performed with the help of automated methods?

State per centually Minimum/Average/Average

Minimum ☐
Average ☐
Maximum ☐

42 Are requirements subdivided into different priority classes?

Yes ☐ No ☐

43 If yes, how many different priority classes are there?

Number ☐

44 If yes on question number 42: Are the requirements implemented in this priority order?

Yes ☐ No ☐

45 Is it specified to what extent programs will be affected by each requirement?

Yes ☐ No ☐

46 Is it specified how many customers that are affected by each requirement?

Yes ☐ No ☐

47 How much of a project has been completed when the last requirements are being added?

State the average per cent! Average ☐

48 To what extent do you have knowledge of how much it costs to implement the incomming requirements?

Very good knowledge ☐
Good knowledge ☐
Less well knowledge ☐
Bad knowledge ☐

49 In how big share of the total system(Tribon) have you either tested or developed within?

Share ☐

50 To what extent are project documentation standards used in order to document events and decisions?

State the average per cent Average ☐

51 How well do you think that these documentation standards supports your work?

Very good	
Good	
Less well	
Bad	

52 Allocate 100 points on the five areas requirement specification, implementation, testing, documentation and reflection and feedback to the degree you feel that you devote your time in a project

	Internal	Customer specific
Requirement specification		
Implementation		
Testning		
Documentation		
Reflection and feedback		

53 Who do you think are the key personell within the company?

1	
2	
3	
4	
5	

54 State the five most important projects within the company today!

1	
2	
3	
4	
5	

55 State the five most important customers to your work!

1	
2	
3	
4	
5	

56 State the five customers that you consider influence TRIBON the most!

1	
2	
3	
4	
5	

57 Estimate the time share that you as an individual and the company spends on the most influential customers(stated in question 56) requirements!

The average per cent as an individual! Average ☐

The company´s average per cent! Average ☐

58 How is the origin of your work tasks spread?

Financed by customer ☐
Financed by customer and KCS ☐
Finance by KCS ☐

59 Allocate 100 points on resource planning, requirement process, decision process and information flow to the extent that you think that the company should stake in order to improve! (where are the flaws?)

Resource planning ☐
Requirement process ☐
Decision process ☐
Information flow ☐

Resource planning: How the resources are spread on different projects and departments.
Requirement process: How the requirements are created, developed and tested.
Decision process: How decisions get made and how they are executed
Information flow: How the information is spread between management and the organisation and between individuals and departments.

B.2 Results

The results presented in this section are from the completed questionnaires by the employees at TS during the summer of 2000. The questionnaire was handed out to 35 employees in the software development department and 23 were received more or less completed. The questions referring to each answer is not presented in this section while they have been presented in section B.1.

In the result of each question is the amount of answers presented beneath the column containing the abbreviation Ans. In the case of requesting a numerical value by the respondents the total amount of answers is presented in this column and in the other cases the amount of answers to each choice is indicated.

To the questions requesting numerical answers both the mean value and the 95 percent confidence interval from the amount of relevant answers are calculated. The confidence interval is presented as the three values: minimum, maximum and the breadth of the confidence interval.

2.	Ans.	Percent
Once a month	7	30%
2-3 times per month	3	13%
4-5 times per month	5	22%
More than 5 times per month	8	35%

3.	Ans.	Percent
Yes	4	17%
No	19	83%

4.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	15	1,7	0,3	3,2	1,4
Aver.	19	6,7	3,4	9,9	3,3
Maximum	14	12,7	4,7	20,7	8,0

5.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	14	1,4	0,5	2,4	0,9
Aver.	17	3,5	1,9	5,0	1,6
Maximum	13	4,8	2,7	7,0	2,1

6.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	12	1,2	0,2	2,2	1,0
Aver.	15	2,8	1,2	4,4	1,6
Maximum	12	3,2	1,0	5,3	2,2

7.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	13	1,8	0,8	2,7	0,9
Aver.	17	4,4	2,0	6,8	2,4
Maximum	13	7,3	3,1	11,5	4,2

8.	Ans.	Percent
Yes	15	65%
No	8	35%

9.	Ans.	Percent
Monthly	7	47%
Quarterly	5	33%
Yearly	3	20%

10.	Ans.	Percent
Very good	0	0%
Good	6	26%
Less well	16	70%
Bad	1	4%

11.	Ans.	Percent
Very good	0	0%
Good	10	53%
Less well	8	42%
Bad	1	5%

12.	Ans.	Percent
Yes	16	70%
No	7	30%

12.	Ans.	Percent
Very good	0	0%
Good	8	50%
Less well	7	44%
Bad	1	6%

12.	Ans.	Percent
Yes	7	100%
No	0	0%

13.	Ans.	Percent
Never	0	0%
Sometimes	8	36%
Often	10	45%
Always	4	18%

14.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	13	1,8	0,7	3,0	1,1
Aver.	17	3,9	2,4	5,3	1,4
Maximum	13	4,9	3,1	6,7	1,8

15.	Ans.	Percent
Very good	1	4%
Good	20	87%
Less well	2	9%
Bad	0	0%

16.	Ans.	Percent
Very good	0	0%
Good	15	65%
Less well	8	35%
Bad	0	0%

17.	Ans.	Percent
Proactive	5	22%
Reactive	18	78%

18.	Ans.	Percent
Increased	1	5%
No Change	19	86%
Decreased	2	9%

19.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	14	2,6	1,2	4,0	1,4
Aver.	16	4,5	2,7	6,3	1,8
Maximum	13	7,5	4,2	10,8	3,3

20.	Ans.	Percent
Weekly	5	26%
Monthly	9	47%
Yearly	5	26%

21.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	21	12,8	8,5	17,1	4,3

22.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	15	24,1	9,9	38,4	14,3

23.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	22	60,8	49,8	71,9	11,1

24.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	15	27,8	22,5	33,1	5,3

25.		Ans.	Percent
	Very good	1	4%
	Good	16	70%
	Less well	3	13%
	Bad	3	13%

26.	Software Development	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	17	62,4	54,4	70,4	8,0
26.	β-testing	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	17	15,6	10,0	21,2	5,6
26.	CSR	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	17	22,0	15,9	28,1	6,1

27.		Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Minimum	10	23,2	2,6	43,8	20,6
	Aver.	18	67,0	53,7	80,3	13,3
	Maximum	10	71,5	53,1	89,9	18,4

28.		Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Minimum	8	16,0	-4,6	36,6	20,6
	Aver.	13	46,1	26,6	65,5	19,5
	Maximum	7	61,4	25,6	97,3	35,8

29.	Customer specific	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	21	32,2	19,8	44,6	12,4
29.	Internal	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	21	56,8	40,7	73,0	16,2

30.		Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	19	16,1	8,4	23,8	7,7

31.		Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	20	81,2	70,0	92,4	11,2

32.		Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	18	21,3	8,7	33,9	12,6

33.		Ans.	Percent
	Very good	5	23%
	Good	16	73%
	Less well	1	5%
	Bad	0	0%

34.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	22	65,5	53,4	77,6	12,1

35.	Ans.	Percent
After every project closure	5	22%
During and after project closure	11	48%
Once a month	3	13%
Once every half a year	1	4%
Once a year	3	13%

36.	Ans.	Percent
Department	3	14%
Project	8	36%
Individual	9	41%
Not at all	2	9%

37.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	23	45,7	33,7	57,6	11,9

38.	Ans.	Percent
Never	4	18%
Sometimes	15	68%
Often	3	14%
Always	0	0%

39.	Ans.	Percent
Never	4	19%
Sometimes	16	76%
Often	1	5%
Always	0	0%

39.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	11	8,3	3,5	13,0	4,8
Aver.	14	52,1	34,0	70,3	18,1
Maximum	11	823,6	403,5	1243,8	420,2

40.	Ans.	Percent
Never	5	23%
Sometimes	5	23%
Often	12	55%
Always	0	0%

41.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Minimum	9	3,3	-3,2	9,9	6,5
Aver.	15	6,9	2,0	11,8	4,9
Maximum	10	14,0	3,0	25,0	11,0

42.	Ans.	Percent
Yes	9	47%
No	10	53%

43.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	6	3,5	2,8	4,2	0,7

44.	Ans.	Percent
Yes	8	89%
No	1	11%

45.	Ans.	Percent
Yes	6	29%
No	15	71%

46.	Ans.	Percent
Yes	3	14%
No	18	86%

47.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	13	78,4	71,8	85,0	6,6

48.	Ans.	Percent
Very good	0	0%
Good	11	50%
Less well	7	32%
Bad	4	18%

49.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Aver.	23	48,8	36,4	61,2	12,4

51.	Ans.	Percent
Very good	0	0%
Good	5	38%
Less well	5	38%
Bad	3	23%

52.	Specification	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Int	Aver.	20	14,3	7,6	21,0	6,7

52.	Implementation	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Int	Aver.	20	58,1	50,6	65,5	7,4

52.	Testing	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Int	Aver.	20	15,1	11,5	18,7	3,6

52.	Documentation	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Int	Aver.	20	7,7	5,9	9,4	1,7

52.	Reflection and Feedback.	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Int	Aver.	20	4,9	2,4	7,4	2,5

52.	Specification	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Ext	Aver.	21	16,3	8,0	24,6	8,3

52.	Implementation	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Ext	Aver.	21	53,8	45,7	61,9	8,1

52.	Testing	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Ext	Aver.	21	17,2	13,4	21,1	3,8

52.	Documentation	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Ext	Aver.	21	7,8	6,0	9,6	1,8

52.	Reflection and Feedback	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
Ext	Aver.	21	5,4	2,9	7,8	2,5

57.	Individual	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	16	42,8	28,9	56,7	13,9

57.	Company	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	14	38,8	26,1	51,4	12,7

58.	Customer	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	22	27,7	13,5	42,0	14,3

58.	Company and Customer	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	22	40,7	23,8	57,5	16,8

58.	Company	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	22	31,6	17,5	45,7	14,1

59.	Resource	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	21	28,9	23,2	34,5	5,7
59.	Specification	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	21	27,1	20,2	34,0	6,9
59.	Decision	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	21	20,5	14,2	26,9	6,3
59.	Information flow	Ans.	Aver.	Conf, min	Conf, max	Conf. 95%
	Aver.	21	24,0	18,6	29,4	5,4