Titel /Title: Improvements in Organizational development

Författare /Author: Roger Aarenstrup

10 poäng (15 hp) /10 credits

Examensarbete/ Thesis

Magisterprogram med ämnesbredd
med inriktning mot elektroniska affärer

Study program in
Master of Business Administration in e-business
Abstract

Title: Improvements in organizational Development.
Level: Final assignment for Master Degree in Business Administration
Author: Roger Aarenstrup
Supervisor: Lars-Johan Åge
Date: 2009 June

Aim: The aim of this study was to investigate the relation between innovation and system complexity, and suggest improvements for an organization to handle innovation and complexity better.

Method: The analysis was based on theoretical models about organizational structure, development models and knowledge management. The models were selected to highlight theoretical extremes rather than practical usefulness to avoid practical obstacles in the theoretic evaluations. The work progressed as a strategy development flow based on a model including four phases; analysis, objectives and recommendations, options and Implementation.

Result & Conclusions: To significantly improve how complexity and innovation are managed it isn’t sufficient to focus on improvements in one part, such as processes. Organizational goals, external environment, organizational structure, development model, knowledge management and internal culture have to be considered and balanced to achieve significant improvements. For the organization studied it was clear that there was a difference in the official description of the organization and how it worked in practice.

Suggestions for future research: Metrics are important to measure value and improvement. Balanced metrics describing how well an organization is adapted to its goals and environment is an area for future work. The effect of Model-Based design on organizational structure is another interesting topic for further research.

Contribution of the thesis: The recommendations and objectives developed in this study can be used to improve an organization with respect to both internal and external environment.

Key words Model-Based Design, organizational development, development models, knowledge management
# Table of contents

1. Introduction
   1.1 Background ................................................................. 1
   1.2 Statement of problem ..................................................... 2
   1.3 Purpose ........................................................................... 2
   1.4 Disposition ..................................................................... 2

2. Method
   2.1 The organization ............................................................ 4
   2.2 Research approach ........................................................ 4
   2.3 Structure of the work ...................................................... 5
   2.4 Critical analysis of the Method ....................................... 6

3. Background to organizational development
   3.1 Organizational structure ............................................... 8
   3.2 Development models ...................................................... 12
   3.3 Knowledge management ............................................... 20
   3.4 Reflection .................................................................... 23
   3.5 Change Management .................................................... 25

4. The Organization
   4.1 Background ................................................................ 27
   4.2 People and what they do ............................................... 27
   4.3 The development process .............................................. 29
   4.4 Workflow ...................................................................... 30

5. Analysis, objectives and recommendations
   5.1 Analysis of the external environment ............................. 34
   5.2 Analysis of the internal environment ............................... 35
   5.3 Objectives .................................................................... 39

6. Option generation, evaluation and selection
   6.1 Options related Objective 1 - Task force layer .............. 45
   6.2 Options related Objective 2 – Business system layer ...... 46
   6.3 Options related Objective 3 – The knowledge layer ........ 47
   6.4 Options related Objective 4 – Standardize the simulation environment .... 48
   6.5 Options related Objective 5 – Process improvement .......... 49
   6.6 Options related Objective 6 – Knowledge creating enablers ...... 49
   6.7 Options related Objective 7 – Knowledge transfer ............ 50

7. Planning .......................................................................... 52
1 Introduction

Products like cars, trucks, mobile phones, airplanes and wind power plants all include many different technical solutions. The number of solutions in each product increase continuously driven by customer requirements, new laws and regulations and competition (Henkel, 2003) (Moore, 1975). The challenge of making all parts work together is becoming increasingly harder which can sometimes hinder implementation of new innovative features.

1.1 Background

The organization studied here has a long history of development and manufacturing of large machines. The past fifteen years they have used computer technology as an embedded part of their machinery for more efficient control and additional capabilities. The parts in their machinery are interrelated and needs to be designed to work as a coherent entity, a system. Their machinery, the systems, includes actuators, sensors, electronics, mechanical structures and software for control and supervision.

With new technology constantly available and with more capabilities implemented the number of parts such as sensors and actuators has increased significantly and as a result the control software has grown in size. This means that the complexity, the number of interrelated parts, of the system as well as the software has been growing continuously and is now on a level where it has become a problem.

Increasing the number of employees to handle the complexity of the system has only worked to some degree. A larger organization has added new challenges such as communication and knowledge sharing between departments and groups causing the innovation power to decrease. Most innovations that have been created despite of the internal problems are deferred or cancelled since there is no way to cope with the increased complexity of the system caused by it. Without innovation organizations will loose market shares to competitors (Blundell et al., 1999).

When complexity of the system isn’t managed properly there is a risk of delays and increased costs. This has been the case a number of times in the organization studied here and often with the result of removed features to decrease complexity of the system or just accept a lower performance of it.

Another effect of a more complex system and more people working with it is that change in technology and change caused by innovations have been harder since more people needs to be involved in decisions and it affects more parts. Although the ability to be able to change with new technology or with new innovations is important it has been identified in the organization to be less important than to be able handle complexity of the system. Large changes in technology are rare and when it happens there is usually a new project dealing with it rather than changing the existing system. As stated above new innovations rarely make it to the final product. This is likely why it is not experienced to be a problem.

The organizational structure, tools for development and development model used in the research and development department of the organization studied here have its’ roots in a time before computers were used. It was ok for some time but has now been identified to be an area of improvement. A development model can be described as a high level view of the steps that are taken during the development from requirements to the final product.
Some individuals in the organization have also identified model based design as an approach for improvement and brought the idea to the management. Model based design is a model centric approach for the development where computer models are created and used for simulation of the entire system, as a way to verify the complex system at an early stage. Models can also improve communication, as a common language, and as a result improve innovation power.

1.2 Statement of problem

The effort of understanding how the interrelated parts of the system work together, the system dynamics, and to test and verify the system as well as individual parts have grown to a point where it is unmanageable. The effects have been that in some projects parts of the system have been removed late in development, performance has been reduced and quality has been compromised. A secondary effect has been that new ideas have been discarded because the effort of implementing them is too high. This contradiction between innovation and complexity of the system is the concern addressed here. How can an organization be innovative, make use of the innovations and handle the complexity of the system?

1.3 Purpose

The purpose of this work is to propose a strategy for the improvement of the research and development department of the organization to handle complexity of the system and innovation better.

1.4 Disposition

The chapters are organized according to the method of strategic development described in section 2-3 and Figure 2-1. To support the analysis chapter 3 includes a theoretic background to organizational development. The background includes theory about organizational structure, development models and knowledge management. Chapter 3 also includes reflections on how the different parts are interrelated and how they affect each other. A model for evaluation of organizational structure and development models is set up to support the analysis in chapter 5. Last in chapter 3 is also a summary of how to manage a larger change project in an organization that is later used, in chapter 7, to help make a reasonable change plan for the improvements.

The results from the investigation of the organization are presented in chapter 4 as an organizational chart and process descriptions with reflections on how they communicate and handle knowledge.

In chapter 5 the results from the investigation of the organization in chapter 4 is analyzed with help of the theory in chapter 3. In the analysis a number of areas of improvement, according to the theory, are identified and a corresponding recommendation for improvement is developed. The recommendations are then gathered and formed to suitable objectives.
In chapter 6 different options for how to implement each objective is developed. The objectives are evaluated and the most suitable selected as input to the development of the implementation plan in chapter 7.

Chapter 8 concludes the development of the strategy for change and suggests some further work.

Figure 1-2 Disposition. To the right is the straight flow of chapters and to the left the theory of chapter two is shown in some detail. Theory about organizational structures, development models and model-based design is used to build a model for evaluation of an organization with respect ability to cope with change VS ability to cope with complexity. This model is mainly used in the analysis in chapter 5. The theory about change management is used in chapter 7 to support the planning of the implementation (Own construction).
2 Method & Scope

This chapter describes, motivates and critically analyses the research approach used in this study. A method of strategy development which gives a structure to how the work will progress is also included.

2.1 The organization

The organization investigated here is a merge between two different companies. Even if the companies work in completely different industries their challenges are similar. This work is limited to the software development parts of the research and development department.

The reason for the merge is that the companies are reluctant to make their strengths and weaknesses in development public. The goal of the merge is to make sure the two companies can’t be recognized but still the result of this project should be applicable to both. The resulting organization from the merge is called “the organization”.

2.2 Research approach

Initial discussions with the management of the organization showed that they didn’t have any metrics or quantitative data for how they operate. Since no proven ways to measure how well they cope with complexity or how innovative they are were found in the literature it was clear that a qualitative approach for this study was necessary.

The purpose was to define a strategy for improvements by obtaining a big picture of how they operate and identify key areas of improvement. It is thus not known from the beginning what the investigation will lead to and in which direction it will go. This further supports the choice of a qualitative approach. Table 2-1 summarises the features of qualitative and quantitative approaches.

<table>
<thead>
<tr>
<th>Qualitative</th>
<th>Quantitative</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;All research ultimately has a qualitative grounding&quot; - Donald Campbell</td>
<td>&quot;There’s no such thing as qualitative data. Everything is either 1 or 0&quot; - Fred Kerlinger</td>
</tr>
<tr>
<td>The aim is a complete, detailed description.</td>
<td>The aim is to classify features, count them, and construct statistical models in an attempt to explain what is observed.</td>
</tr>
<tr>
<td>Researcher may only know roughly in advance what he/she is looking for.</td>
<td>Researcher knows clearly in advance what he/she is looking for.</td>
</tr>
<tr>
<td>Recommended during earlier phases of research projects.</td>
<td>Recommended during latter phases of research projects.</td>
</tr>
<tr>
<td>The design emerges as the study unfolds.</td>
<td>All aspects of the study are carefully designed before data is collected.</td>
</tr>
<tr>
<td>Researcher is the data gathering instrument.</td>
<td>Researcher uses tools, such as questionnaires or equipment to collect numerical data.</td>
</tr>
<tr>
<td>Data is in the form of words, pictures or objects.</td>
<td>Data is in the form of numbers and statistics.</td>
</tr>
<tr>
<td>Subjective - individuals’ interpretation of events is important, e.g., uses participant observation, in-depth interviews etc.</td>
<td>Objective – seeks precise measurement &amp; analysis of target concepts, e.g., uses surveys, questionnaires etc.</td>
</tr>
<tr>
<td>Qualitative data is more 'rich', time consuming, and less able to be generalized.</td>
<td>Quantitative data is more efficient, able to test hypotheses, but may miss contextual detail.</td>
</tr>
<tr>
<td>Researcher tends to become subjectively immersed in the subject matter.</td>
<td>Researcher tends to remain objectively separated from the subject matter.</td>
</tr>
</tbody>
</table>

Table 2-1, Features of qualitative and quantitative research approaches (Miles & Hauberman, 1994, p. 40) (table from http://wilderdom.com/research/QualitativeVersusQuantitativeResearch.html)

2.3 Structure of the work

To structure the work a method for strategy development (Grant, 2007) was followed throughout this project. The method includes four phases; analysis, objectives and recommendations, options and Implementation, see Figure 2-1.

In the analysis phase the focus was on the internal environment since much of the external analysis was given as an input. The internal environment or the current situation was investigated by studying documented development processes and models. In addition to this personal interviews were performed with people with different positions to capture aspects not included in the official processes. The interviews were informal and performed as an open discussion about how things work. In addition to this the author worked as a consultant for a few weeks within the organization to get a real insight in how things work and personal experience.

These three steps were chosen because the official documents would show the corporate initiative and then personal interviews will give an insight in how strictly the formal methods are followed. The interviews will show if the individuals have a common understanding and view of how things work or should work. Then by getting into the organization and do some personal work, the consultant can get a personal view, based on experience from other organizations, about how things work.

The current situation was then analyzed with theoretical models to capture areas of improvement supported by theory about knowledge creation, development models and organizational structure.

Based on the analysis a number of possible improvements, some with help of model based design, was generated and then selected in an order suitable to create a plan for improvements.
2.4 Critical analysis of the method

The qualitative approach was necessary because there was no empirical data available and obtaining such data would take a long time. The main problem is that conclusions are more open to debate compared to a quantitative approach with empirical findings. This can make the change effort harder. Empirical findings can work as change agents and are hard to dispute. For future continuous improvements of the processes it is important to
establish metrics to get necessary empirical data to supervise the operations and identify and prioritize areas of improvements.

A qualitative approach which partly involves previous experience of the researcher can also be criticized since the organizations culture and environment is not identical to previous situations when that experience was gained. A weakness in the analysis here is that the external environment was not included which makes it hard to re-use conclusions from other organizations and development models.

The four phase strategy development method includes the major steps in this work and provides a good and communicable structure. It is, however, important for the change work that individuals in the organization affected by the change take a more active role in the development of the objectives and the plan. The internal culture, financial situation and other changes should be taken into consideration before any major changes are implemented. Therefore the validity of this study is mainly to serve as a recommendation and guideline for the real change project.
3 Background to organizational development

Since every organization is unique with its internal capabilities and resources and its external environment, it isn’t possible to make one development model and organizational structure to fit all. The aim of this chapter is to provide some general background to organizational structure, development models, knowledge management and change management that can be used for analysis and establishing objectives for the strategy development.

The important question to be kept in mind is how to use the organization, the development model and knowledge to gain competitive advantage long term.

3.1 Organizational structure

To accomplish complex tasks humans need to cooperate. Cooperation can be split in two fundamental concerns; how to divide the work into smaller tasks and how to coordinate these tasks to accomplish the whole. In this part, two basic types of organizational structures are first described which are extreme cases of organic or mechanistic views (Dahlbom, 1993) (Robbins, 2007). After that the matrix organization and hypertext organizations are briefly presented as examples of ways to modify the extreme cases to fit different organizational needs. Further attempts (Robbins, 2007) to organizational structure are team structure, virtual organization (mainly outsourced) and boundary-less organization (Tichy, 2005), these are not discussed here.

3.1.1 The bureaucratic organization

The most well known way to divide and coordinate labor is the bureaucratic organization (Robbins, 2007). In a bureaucratic organization there are workers who do the work and a hierarchy of managers who make the decisions. Work is mainly coordinated by well specified and formalized tasks for the workers.

All organizations operate in an environment which is mainly not controllable from the organization. This includes competitors, suppliers, market, and political, technological and economic factors. The effectiveness of an organizational structure is strongly dependant on the environment of the organization. This means that it is not possible to determine one best way to do things.

The main disadvantage with bureaucratic organizations is its inability to cope with change in the environment. When things change, the formal procedures and formalized tasks are no longer useful which put more and more load on the management hierarchy. The bureaucratic organization requires a stable and predictable environment to be efficient. It is also generally recognized that bureaucratic organizations isn’t good for motivation and creativity.

3.1.2 The organic organization

When things are uncertain the organic organization (Robbins, 2007) steps in. In an organic organization the structure is more like a network or task force oriented than hierarchical. All members are supposed to participate in decision making which means
that everybody needs the tools to communicate and interact. Coordination in an organic organization can be viewed as a dynamic negotiation and creation of commitments. The role of the manager is to serve as intermediary between the planners and the workers. The main disadvantage with the organic organization is the problem that arises with coordination when the tasks grow in complexity.

3.1.3 **The matrix organization**

The matrix organization (Robbins, 2007) is an attempt to combine the functional bureaucratic organization with a cross-functional project oriented organization, see Figure 3-1. It is well suited for project oriented organizations where the project manager manages a cross functional team with specific time-bound projects. The functional manager has a long term responsibility to develop each functional group, their skills and knowledge transfer.

Advantages with the matrix organization are that specialists’ time can be used efficiently in different projects and it is flexible as individuals can easily be moved between different projects. The cross functional projects also improves communication and knowledge sharing.

The main disadvantage is ambiguity in command. Project members have two managers (at least) to report to, this can increase stress for the individual. There can also be conflicts between functional managers and projects managers in terms of how to use resources and priorities.

![Figure 3-1, the matrix organization (Own construction).](image-url)
3.1.4 **Hypertext organization**

The hypertext organization was presented by Nonaka and Takeuchi (Nonaka & Takeuchi, 1995) and is an attempt to take knowledge creation into consideration in the organizational structure. It is also an attempt to combine the matrix organization with bureaucratic organization to get the advantages of both and minimize the drawbacks.

The metaphor hypertext is used to emphasize that the organization is built up by different layers and the layers should be interpreted as different contexts that are available. The hypertext organization has a formal hierarchical structure, the business system layer, working in tandem with a nonhierarchical task force structure, the project team layer, and a knowledge base layer, see Figure 3-2.

![Project-system layer](image)

The knowledge base layer doesn’t exist as an actual entity but is captured in organizational vision, culture, people, databases and simulation models (simulation models added by the author to include model bases design).

Members of the project teams are taken out from the business system and only reports to project management, this is an important difference from the matrix organization. When projects are over the team members go back to the business system where they share knowledge gained in the project work. The fact that members only report to one layer at a time removes most weaknesses that the matrix organization has. Also by putting different members together temporarily the cycle of knowledge conversion and creation can flourish within the organization, see section 3.3.

Successes and failures in the project teams are evaluated re-categorized and re-contextualized and stored in the knowledge base, available for everybody.
3.1.5 Centralization vs. decentralization

This model summarizes the key points about how to choose the right level of centralization for an organization in a specific situation (Smedberg, 2001). When leaders are in control in a way that they have enough information about what is going on and have efficient enough communication, the organization can be more centralized. When situation awareness is low for leaders and communication is poor a more decentralized approach is more efficient since it allows people in the situation to react immediately. In the latter case it is very important that goals and strategies are clearly explained and understood by everybody; otherwise it is not possible to work for the same goal. Choosing the right level of centralization is a delicate matter. Too much centralization, even if communication and control is good, will decrease motivation.

![Centralization vs. Decentralization Diagram](Image)

3.1.6 Model for evaluation of organizational structure

To evaluate the organizational structure the model in Figure 3-4 has been created. It visualizes the organization in context of how it can handle complexity vs. change in the environment. As stated above, the bureaucratic organization is good for handling complexity but poor at handling change and vice versa for the organic organization. The matrix organization falls somewhere between the two extremes.

The project layer and the business system layer of the hypertext organization falls into different positions in the model since they are of different nature. This is one strength of the hypertext organization but the hypertext organization was in first hand created for knowledge creation and not to cope with complexity and change.
3.2 Development models

In addition to the organizational structure a development model is needed to define how to develop new products in an organization. The development model needs to fit to and be integrated in the organization to be efficient.

In this section the stereotypical development models construction and evolution are first presented then some more practical models are described. Last but not least model based design is presented and discusses in the context of development models.

3.2.1 Construction

Construction is the classical approach to problem solving (Dahlbom, 1993). Construction has been formalized as the waterfall model that fits well into the bureaucratic organization. It requires a well defined task and then the work starts with finding a strategy for finding a solution. This can also be seen as a top-down approach. System constructors develop systems in a stepwise fashion, decomposing and refining an abstract concept of the system as a whole into its concrete components, the final implementation.

The problem is divided into smaller tasks that are well defined and easier to cope with, this is the work division. Coordination is handled by well specified interfaces between different parts together with general company standards, rules and specifications. When the smaller parts are solved and put together the strategy to solve the problem is ready and the implementation can start. The implementation and the development of the strategy are two separate tasks.

Construction as a methodology has a disadvantage similar to the bureaucratic organization, its inability to adapt to change in specifications or environment and uncertainty. To be efficient it needs a well defined goal and specification that doesn’t change.
The main advantages are predictability and that construction can handle very complex tasks. Progress is made by rational thinking rather then empirical studies which means that people leading the projects needs to have enough experience to be able to move the project forward.

3.2.2 Evolution

The evolution approach recognizes uncertainties (Dahlbom, 1993). Developers identify and evaluate concrete solutions to partial problems, gradually approaching the system as a whole. Prototypes and experiments are heavily used and developers process information as they move along, as scientific investigators.

Evolution fits the organic organization and it is a bottom-up methodology. Because of the organic nature there are few permanent rules and structures prescribing or determining the behavior of the actors. To achieve coordination, each project has to establish and dynamically maintain explicit interaction between all parties.

Due to the nature of evolution it is hard to plan and design a development process in advance. The project structure reflects the learning process rather than the structure of the system. Less effort is spent on planning the process and designing the system and more effort is spent on fixing the process and testing and modifying the system.

One major disadvantage with evolution is that it is hard to coordinate complex projects. Progress is made by empirical studies rather then rational thinking which means that it isn’t necessary to have prior experience. The empirical studies include customers and users.

3.2.3 Development models in organizational context

As discussed above construction fits well with the bureaucratic organization and deals well with complexity. Evolution works well with the organic organization that handles change well but not complexity.

In Figure 3-5 the two stereotypical development models are inserted in the complexity vs. change model from section 3.1.6.
3.2.4 The waterfall model

The waterfall model is a sequential development model first suggested in 1970 by W.W. Royce (Royce, 1970). The model suggests that development should be sequential and the next step isn’t started until the previous is completed.

The steps in the model are requirements, design, implementation, verification and maintenance, see Figure 3-6.

Some advantages suggested (Wikipedia - Waterfall model, 2008) are that the waterfall model is a strict approach to development and easy to manage and understand. It is also document based which means that if a person leaves the project the information isn’t lost (explicit information).

The simplicity of the model is probably the reason why it’s been so widely used. It is interesting because Royce suggested the model as a flawed model that he later in his
article transformed to a more iterative approach (Royce, 1970). He stated that it was “risky and invites failure”. Despite of this it became widely used, perhaps because of its simplicity or that necessary tools for more iterative approaches were not available.

Some of the major problems with the model are its inability to cope with change in requirements and that for many software problems it isn’t possible to know all details upfront in the design. For the process to work well; the requirements need to be fixed and very skilled and specialized people need to work in each phase. But that basically requires that people in the organization has solved very similar problems before.

The main advantage with the model is that it handles complexity well, as long as things remain constant.

### 3.2.5 The spiral model

The spiral model, Figure 3-7, is a development model that combines both bottom-up and top-down approaches (Wikipedia - Spiral Model, 2008). It was initially intended for large and expensive projects but with help of development tools it can also be used for moderate sized projects. The model was defined by Barry Boehm in 1988 (Boehm, 1988).

In the spiral model the development progress in a similar way as in the waterfall model; requirements, design, implementation and test. They are, however, done more quickly and only with some of the features of the whole system. Then the project is evaluated, preferably together with the customer, in terms of risks and changes in requirements, and a decision is made to continue or not. If the project continues another iteration is made. For each iteration the risk is reduced and more features of the final system are implemented.

![Figure 3-7, the spiral model (Wikipedia - Spiral Model, 2008).](image)

The spiral model was primarily designed to handle risk in projects because of its iterative nature it also handles change well. The drawback is its inability to cope with complexity. Larger projects may need high level system design where the parts depend on each other and can thus not be implemented separately by iterations.
3.2.6 The V-model

The V-model, Figure 3-8, is very common in the automotive industry. It can be seen as an extension to the waterfall model and is bent upwards after the coding phase to get the V-shape (IABG in Ottobrunn, 1993) (Wikipedia - V-Model, 2008). Each level in the model shows the relationship between each step of the development and its corresponding test phase.

In the first step requirements are collected by analysis and discussed with customers. The concern is to establish the ideal performance of the system without any concern about how it will be achieved. The customers’ acceptance test is also designed in this phase. The main output from this phase is a requirements specification.

In the system design phase, the requirements specification is used to set up a system design. Different possibilities are considered and evaluated. A software specification document is created with the general system organization and other things like interface descriptions are included. Documents for system tests are also created.

In the architecture design phase the high-level software design is made which splits up the software in modules and describes their interfaces and interdependence. The output is a high-level design document with lists of modules, brief description of their functionality and an interface design.

In the module design, which can be done in parallel, details about each module is specified. Modules can also be divided in sub-modules with their own specification. The output for each module is a module design description. Individual module test descriptions are also created at this stage.

In the coding phase each module is implemented as the module design specification describes.

After coding the test and system integration phase starts. Each module is first tested separately in module tests, then the modules are integrated and the integration test is performed. The integration test is usually a “black-box” test with dummy inputs and so on just to verify interfaces in the integration.

In system tests the entire design is tested with help of the system test description to verify functionality and performance of the system. In automotive applications this can be to test an engine control system against a real engine for example.

In the acceptance test real users or customers are testing the system against the requirements of the user or customer.

It is not obvious where to put the V-model in terms of how it handles change and complexity, it will depend on the particular implementation. It is chosen here to put it somewhere between evolution and construction. It doesn’t have any real iterative parts that handle change well but the continuous tests make it possible to capture some changes. It is also possible to implement the V-model to be able to iterate between two steps. Developing the entire system design in the beginning makes it hard working with complex systems where some things might not be well known up front.

There are variants to the V-model that improves parts, such as the dual-Vee model, but that is not considered here.
3.2.7 Agile development models

Agile development models such as scrum, extreme programming (XP) and test driven development are gaining popularity (Wikipedia - Agile software development, 2009). Agile methods have some interesting concepts that might develop into more capable development models in the future, but are not fully explored according to the author’s opinion. Agile methods are not considered further here.

3.2.8 Model based design

Model based design (MBD) is a model centric approach to the development of embedded systems by automating error prone and time consuming tasks.

It is based on graphical models, thus model centric, that can be simulated and used as a single source of many outputs, such as software and different reports. The development is focused on the model and elaboration of it until the desired outputs can be generated (Mahapatra et al., 2008).

Model based design is not a development model but has been successfully implemented in many different models including the waterfall model, the spiral development model and the V-model.

By the help and efficiency gains of automating time consuming tasks, possibilities opens to change the development model to fit the organization’s environment better, i.e. model based design increases flexibility and can work as a change agent. In many cases it is desirable to use model based design to move from a waterfall development model to a more iterative model, to be able to handle change better.
Graphical models as an abstraction of a system also has a number of advantages. For example it is easier to cope with larger systems and it’s easier to communicate between different engineers and different departments with the model as language.

### 3.2.9 Reflection on development process and model based design

In the development models described earlier in this section, model based design can be used to automate certain steps, for example, the coding step or implementation can be automatic with help of code generation. Tests can be continuously performed with simulation. High level algorithmic design can directly be tested with code generation for rapid prototyping.

One of the goals with model based design is thus to make the development more efficient by automating key steps in the development. This does not only shorten development time but it also gives more opportunities to try different solutions and by continuous test and verification, faults are detected earlier in the development making them easier and less expensive to correct.

In the complexity vs. change diagram introduced in section 3.1.6, model based design can expand the area of which a specific development model can operate, Figure 3-9. Also, as stated before, model based design can be a change agent to in the change to a more suitable development model.

By using models the entire organization, as well as the supply chain, get a common language, instead of sending ambiguous documents around. For example in the V-model above, models can be used instead of the documents in the transition between phases. Models can also be used to simulate test cases and be executable specifications.

Methods such as model based design can be viewed as a tool for communication, a common language for people who use and understand it (Löwstedt, 2002). The method can also be used to reduce insecurity; especially for less experienced individuals it provides guidance. More experienced individuals know when to use or adapt the method and when it is ok do deviate from it. For them the method is more of a source of inspiration and support for the memory. When adopted the method builds a culture in the organization. People no longer question the validity of the method but takes it for granted as a part of the organizational culture.
Figure 3-9, waterfall, spiral and V- development models. The red, green and yellow areas show the impact of introducing model based design for the different development models. Model based design makes it possible to cope with a wider range of development projects. The development model can cope with more complexity and change but since it is more efficient a more structured model can be used also for smaller projects. The exact position of the different models can be argued and it is not the aim here to investigate each model in detail. The implementation of each model, the development process, can also differ and make the ability to handle change and complexity better or worse (Own construction).
3.3 Knowledge management

Coordination of tasks requires an exchange of information and doing certain tasks require certain knowledge. Leading organizations also need to create new knowledge to stay ahead. New knowledge can lead to continuous innovation that leads to competitive advantage.

Before discussing more about knowledge creation in organizations let’s take a look at what knowledge and information are. This is of course not easy to define but some relations can be identified.

3.3.1 What is knowledge

Data, information, knowledge and wisdom correspond to different levels of human activity (Bellinger, 2004), see Figure 3-10. Data are just pure elements like 5% and $10,000. When the relation between the data becomes apparent we have information, for example, we have $10,000 on a bank account with 5% interest. The information above together with an understanding of how banking works (the pattern) we get knowledge. We can for example estimate the development of the account in the future. In the last step, wisdom, we have an understanding for bank accounts compared to other forms of savings and how they all develop over longer periods of time, for example.

We learn new knowledge by attaching the new information to an existing knowledge pattern we have. This is why two people that face the same new information can use and interpret it completely different. If we don’t have a pattern developed in the area of the information, the information doesn’t really tell us anything.
Figure 3-10 shows the higher in the chain the less context dependence there are. Knowledge about bank accounts can be used on many different accounts and even between different banks. But the interest and amount is tied to a specific context. And also the higher up in the chain the deeper the understanding of the situation becomes.

Before discussing the architecture of the knowledge creating organization we need to take a look at knowledge conversion and some knowledge creation enablers.

### 3.3.2 Knowledge conversion

There are two types of knowledge tacit and explicit (Nonaka & Takeuchi, 1995). Tacit knowledge is knowledge that can’t easily be expressed. Explicit knowledge is knowledge that is easier to capture and communicate with words. For example, rules of a social game. Converting tacit knowledge to explicit and vice versa is the foundation for organizational knowledge creation. Nonaka and Takeuchi described four concepts for knowledge conversion illustrated in the figure.

**Figure 3-11, knowledge conversion, (Nonaka & Takeuchi, 1995 and Own construction).**

**Socialization: From tacit to tacit**

In socialization individuals learn new knowledge by being close to others with the knowledge. The knowledge can be acquired without any use of language. An apprentice working with his master to learn by observation, imitation and practice is a good example of socialization.

**Internalization: From explicit to tacit**

Internalization occurs when explicit information is used to build a mental model. Explicit results from a *prototyping* experiment can be used to extend an internal mental model or understanding.
**Combination: From explicit to explicit**
Combination of explicit knowledge happens when different types of explicit knowledge are systematically combined to new conclusions. For example, when designing strategy, knowledge about an organization’s internal conditions are combined with knowledge about external environmental factors and the output is new knowledge about suitable ways forward.

**Externalization: From tacit to explicit**
Externalization is when we bring out our internal tacit knowledge or mental models and makes them externally available. This is often done with help of metaphors, analogies, concepts, hypotheses, or models. Externalization is the most important mode of knowledge conversion for organizational knowledge creation.

### 3.3.3 Reflection on knowledge conversion and organizational structure
Connecting back to different structures of organizations, bureaucratic organizations are good at bringing out combination and internalization. Since the organization mainly handles explicit knowledge combination is a natural process and individuals can also build up their own tacit knowledge and internal understandings from the explicit knowledge available in the organization.

The organic organization, built on networks and task-forces, is more suitable for socialization and externalization. Combining people from different disciplines to work together gives a foundation to share tacit knowledge, socialization. When individuals communicate and negotiate in the informal organic organization they need to externalize their tacit knowledge, perhaps acquired by prototyping, to explain their view.

### 3.3.4 Knowledge creation enablers
In some environments knowledge creation thrives more than in other environments. Some of the key factors that enable knowledge creating are intention, autonomy, chaos, redundancy and variety (Nonaka & Takeuchi, 1995).

It is important that there is a long term *intention* to be a knowledge creating company; it has to be included in visions and strategies.

Individuals of an organization should be permitted to work as *autonomously* as possible which may give unexpected results. When taking responsibility for their work motivation increase. This is clearly an organic approach with a decentralized view of the organization.

It has been shown that people become more creative when in a *chaotic* environment and having a slight sense of urgency to accomplish something (Sveby, 1994). When things are too ordered, like in a bureaucratic organization, creativity might suffer. This sense of crisis or urgency can in some cases be introduced by management by setting very challenging goals.

*Redundant* information about what is going on and how things work, enable individuals to get into each others areas and thus boosting the knowledge creating process. The
redundancy can be introduced by defining overlapping roles or another way is to do a strategic job-rotation.

**Variety** or diversity among members of the organization is important to get different inputs and perspectives.

### 3.3.5 Five-phase model of knowledge creation

Nonaka and Takeuchi (Nonaka & Takeuchi, 1995) introduced a model of how knowledge is created in an organization. This is a continuous spiral of knowledge creation and output come in form of innovation, Figure 3-12.

In the first phase individuals are sharing tacit knowledge, socialization, which is a great source of information. However, tacit knowledge is not easily communicated. Methods might be to have meetings over a beer at the end of the day or put together self organizing teams with a slightly ambiguous task to solve. It is important to note that knowledge can not be created by an organization, only inside individuals.

In the second phase the shared tacit knowledge is externalized to explicit knowledge. By using metaphors and models on a high level the concepts emerge.

In the third phase the concepts are scrutinized and justified. Are the new concepts worth pursuing? Do customers like it? Is it technically feasible?

In the fourth phase the justified concept is elaborated into something concrete, a prototype for example.

In the fifth phase the newly created knowledge is spread inside and outside the organization where it can be internalized by other individuals (they build up their tacit knowledge) and used for new innovations.

![Five-phase model of knowledge creation](image)

*Figure 3-12, the five phase knowledge creating process (Nonaka & Takeuchi, 1995 and Own construction).*

### 3.4 Reflection

Previously we have looked at the extreme cases of organizational structure and development models.

In short we can say that Bureaucratic organizations work well when there is no change in the environment. Construction as development model fits into the bureaucratic organization and combination and internalization as knowledge transformation works well. Disadvantages are inability to adapt to change, lower motivation among employees
and suppressed knowledge transforms socialization and externalization, the most important ones for creating new knowledge.

Karl-Erik Sveiby confirms this in “Kreativitet och makt” (Sveiby, 1994), where he states that the traditional organization with its hierarchy, planning and production, isn’t good for creativity.

The organic organization fits well with the evolution development model. Socialization and externalization as knowledge transfers are emphasized in the organic organization. The disadvantages are inability to cope with size; coordination becomes difficult when the organization grows. The knowledge transformations combination and internalization are also suppressed in the organic organization.

Taking the above insight and putting that together with the knowledge creating enablers and the five-phase model of knowledge creation also show the two organizations inability to cope with the entire knowledge creating process.

Karl-Erik Sveiby states that innovation and creativity can’t be controlled or managed but instead employees should have time that is not allocated to specific tasks (creating redundancy) that give them opportunity to create. A certain amount of positive anxiety is also necessary to create motivation for innovation. A capable person with time and positive anxiety will innovate (Sveiby, 1994).

So what do we do about this? The answer is of course obvious, we need to combine the two models to get the benefits from both and deal with the drawbacks. This is a work filled with trade-offs.

A common way is to put the organization somewhere between the two extreme poles as suitable for the environment, Figure 3-13. For example, having a bureaucratic organization as a foundation but some people also work in cross-functional teams that are put together to solve a problem in a more efficient way (matrix organization).

![Figure 3-13, position for the organization in a specific environment (Own construction).](image_url)

Using a linear approach like this can give some benefits but it doesn’t give many degrees of freedom in the development of the organization.

A way to gain more freedom can be to use different perspectives of the organization. Different parts can be organized differently, adapted to its specific purpose. Some parts of an organization can benefit from a more bureaucratic situation and others from a more organic. The hypertext organization is an example of this. For instance, production and maintenance might benefit from a stricter hierarchy to allow for a quality improvement processes. Customer reported issues could be treated in a similar way. Documented properly and in the same way every time to ensure quality and customer impressions. This part might benefit from a more bureaucratic control. On the other hand, when developing
a new product a new concept never been done before, creative forces are needed and knowledge creation becomes extremely important. In this case a more organic approach can be suitable. As stated before it is also dependant on many other things like the culture of the company, the environment and the nature of the products.

3.5 Change Management

In this chapter John P. Kotter’s ten steps for successful change (Kotter, 1996) are presented to work as guidance for the plan developed from options generated from the analysis of the organization.

The steps described below are not strictly sequential but can work simultaneously and overlap in time. Any mistake in any of the points will, according to Kotter, have severe impact on the result. It is also important to note that even very successful change efforts are perceived messy and are full of surprises for the involved.

1. Establish a sense of Urgency

Determine reason for change (potential crisis, major opportunity…), through an external and internal analysis.

People don’t change without a good reason. If they don’t think they need to change they won’t. In this phase management has to show the obvious urgency or create one. This is absolutely critical to any change. Leadership is important here; typically companies have too many managers and too few leaders.

Because of the tendency to shoot the bearer of bad news sometimes it can be useful to bring in external people to explain a bad situation.

2. Form a Powerful Coalition

Establish a team with power to lead the change (Transition management team, TMT). If the ones introducing change are not powerful enough it will lead to nothing.

3. Create a Vision

What would it be like after the change?
Develop strategies to guide the change work

The vision has to be easy to communicate, if it is not it will fail no matter how brilliant it is. It shouldn’t take more than 5 minutes to communicate the main points. The vision should help clarify the direction in which an organization needs to move.

A vision includes a core ideology and an envisioned future. Core ideology is core values and core purpose. Envisioned future is a BHAG (Big Hairy audacious goal with 50-70% probability of success).

4. Communicate the Vision

Use any means to communicate the vision and strategies.
5. Empower others to act on the Vision
Encourage risk taking and new thinking.
There are always a few positive individuals, give them a chance to act.

6. Reduce resistance and obstacles
Change structures and systems that undermine the vision.

7. Plan for and create short-term wins
Make improvements visible.
Reward people acting on the vision and creating improvements.
Start pilot projects.
It is important to really plan and create short-term wins, not hope for them.
Don’t declare victory because of a short-term win, instead use the credibility created to continue the change process.

8. Increase efforts for change
Use capital of trust generated from the short term wins to produce even more change.
Develop employees to implement the change.
New change projects to change systems and structures.

9. Institutionalizing new approaches
Articulate connection between success and the new behaviours.
Dissolve TMT.

10. Continuous improvement
Follow-up change effort and improve pieces.
4 The Organization

In this chapter the research and development department of the organization is described with an overview of the systems they develop.

4.1 Background

The organization studied here has a long history of development and manufacturing of large machines. The past fifteen years they have used computer technology as an embedded part of their machinery for more efficient control and additional capabilities. The parts in their machinery are interrelated and needs to be designed to work as a coherent entity, a system. Their machinery, the systems, includes actuators, sensors, electronics, mechanical structures and software for control and supervision.

The last few years the complexity of the systems has grown and requirements of functionality and quality have gotten harder and harder to meet. Previously this has been dealt with by organic growth by adding more people. However, now a point has been reached when adding more people doesn’t seem to affect the result as expected. Instead some of the managers believe that the focus has to be on the coordination of tasks and how to execute the tasks more efficiently instead of adding resources, see section 3.1. Being more efficient also gives time over for employees to experiment and create more innovative solutions which is important for competitive advantage.

4.2 People and what they do

The organizational structure is well documented by the company, in Figure 4-1 the relevant parts for this work is described. The two main groups in software development are a research group and a software group. The software group is divided in an application team and a platform team. It is a matrix organization where each employee has one line-manager and one or more project leaders at the same time.
Figure 4-1, organizational chart (Own construction).
Figure 4-2 shows a schematic picture of the software. The platform team is responsible for the software platform, marked in yellow. The application team is responsible for implementing the application software marked in green and the system architect is responsible for fitting the application units together. The research group is responsible for the contents of the application units, the algorithms implemented there.

There are also some other people involved that are not in the chart (Figure 4-1), for example two project managers that belong to another part of the organization. For larger projects there are also steering groups with different stakeholders of the company.

4.3 The development process

This is a generic project process used in all kinds of projects within the company. It is well documented and very detailed. Figure 4-3 is a summary of the most important parts of the process.

In the **pre-study** a project group with people from sales, marketing, research and development discuss new ideas and technologies with consideration of customer and market needs. The timeframe should be quite limited and the work performed in a free and creative environment. The output from this phase is a project specification document.

The **concept study** is driven by the research group. Here several difference solutions or concepts are discussed and evaluated. Both technological factors and market factors are
considered. Needs and requirements are documented. The output is a recommended concept with high-level requirements.

In the feasibility study more detailed requirements are set up together with a risk-analysis. The outputs are more detailed requirements and a clear motivation why this project will have a high probability to be possible to complete technically and be a financial success long term.

In the product definition phase research together with key development people (such as system architects in the software team) created a product specification document that describes the key implementation concerns. A test strategy is also set up and documented in the test specification document.

In the construction phase the main development work is performed. Developers take the design specification and develop the product based on it.

In the test and verification phase the implementation is verified and then tested, with help of the test specification, to make sure it meets the requirements.

4.4 Workflow

This part describes the work and workflows that different groups use in the organization. Figure 4-4 summarizes this graphically.

The information here was collected during discussions and meetings in small groups, sometimes only with one person. In some cases people had a somewhat different view of how things work but the result presented emerged after a few iterations.

4.4.1 Research

The research group is a small group that works in a very informal way. Together with marketing they have set up a few research areas and within those they work relatively freely. In one area there is also cooperation with a local university with a PhD student working part-time for the company. Except for the PhD, cooperation with external organizations is very limited. They do read some articles and books to gain new knowledge in key areas.

Internal discussion groups in a web collaboration tool are used to discuss different topics and it also serves as storage for ideas and development of new ideas.

MATLAB and Simulink are used quite heavily for calculations and simulations of new ideas. They have a common model library to store simulation models for re-use.

In combination with marketing the outputs from the research are requirements specifications to the application developers. Documents like requirement specifications are stored in a document handling system that is quite primitive and lacks advanced search capabilities.

In some rare cases the research needs to be tested in the real system. In such cases taskforces have been put together. This team includes researchers, application developers and integration engineers. Together they can in a timely manner implement a prototype for research, test, and experiments. The task forces have been quite popular and provided some status for people included, especially among the application developers.
4.4.2 Application development and test

The application group includes system architects, application developers and test engineers. Even though they have different official titles the roles can sometimes be a little bit mixed. The normal workflow for this group is:

1) System architect receives requirements from the research group.
2) The variant and interface database is updated as necessary
3) System architect uses the database and system requirements to develop component requirements and system architecture. The system architecture is delivered to the platform group and the component requirements to software designers.
4) Labels for the configuration management systems are received from the platform group.
5) Components are developed in parallel.
6) Software designers write a design description of the components.
7) Software designer check in code in configuration management system and notifies platform group.
8) They wait for build and integration.
9) When the system is up and running test engineers do tests and writes test documentation. This is the best case. In most cases the developer of the component just checks a few things, no formal tests are performed.

4.4.3 Platform development and integration

The platform and integration group develops a small real-time operating system including hardware device drivers and scheduling for the different applications to be executed on the processor.

They also have the responsibility to integrate and build all the target software.

The operating system is actually quite stable and doesn’t require much work or further development.

This group gets a system architecture description from the application group and from that they schedule and set up execution order for the different applications. Later when the software developers have finished the implementation of the applications they build and integrate all the target code. The output is an executable that is delivered to the test engineers.

The main tool for this group is a configuration management system and a change management system that logs all changes done. These tools are shared with the application development group.
1) Decision of a new release for the project
2) Send labels for the project to application group (so they will work from the right baseline)
3) Receive system architecture from application group
4) Implement scheduling and execution order
5) Wait
6) Receive application software from application group
7) Build and integrate all code
8) Label the release
9) Send release to test engineers
10) When the code is tested and verified a new baseline for the code is set.

Not all projects include just new application code. They also build different variants of the same system that differs slightly in what application components that are include. The only real difference in building a variant from a new system is that the wait, step 5 above, doesn’t exist.
Figure 4-4, the development workflow (Own construction).
5 Analysis, objectives and recommendations

In this chapter an internal and external analysis of the organization is presented. In the analysis a number of recommendations are formed that lead up to a number of objectives for the change work.

<table>
<thead>
<tr>
<th>1 Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>External Environment</strong></td>
</tr>
<tr>
<td>- Rate of change</td>
</tr>
<tr>
<td>- Complexity of systems</td>
</tr>
<tr>
<td>- New technology adoption</td>
</tr>
<tr>
<td><strong>Internal Environment</strong></td>
</tr>
<tr>
<td>- Organizational structure</td>
</tr>
<tr>
<td>- Development models</td>
</tr>
<tr>
<td>- Knowledge management</td>
</tr>
</tbody>
</table>

2 Objectives and Recommendations

<table>
<thead>
<tr>
<th>Objectives</th>
<th>Recommendations</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMART</td>
<td></td>
</tr>
</tbody>
</table>

Figure 5-1, the first two steps of the strategy development included in this chapter (Own construction).

5.1 Analysis of the external environment

Much of the external environment analysis was given as an input and is not investigated further here. The main inputs about the environment were:

- The complexity of the systems grow, complexity is a result of increased demands of efficiency and legal requirements that is pushing for more complex solution to limit environmental impact.
- New technology creates possibilities for creative innovations in functionality, usability and efficiency.
- The fundamental base for competitive advantage remains the same.
- Market requirements for quality increases.

From this the organization gave the high level requirements as pre-requisite to this work: be able to cope better with the increase in complexity of their systems and to become more innovative.
5.2 Analysis of the internal environment

The internal analysis is segmented in three categories;

- organizational structure,
- development model and
- knowledge management.

In each section of the analysis the current situation is compared with the theories in chapter 0. From the analysis a number of findings are listed together with a recommendation aiming for the high level goals.

5.2.1 Organizational structure

The organization is of bureaucratic nature with some projects, making it lean towards a matrix organization. Officially they say it is a matrix organization.

The projects are popular but they also create an ambiguous situation since they create status for the members but the line managers in the bureaucratic organization are deciding about people’s careers and the status and opinions don’t always match.

Productivity seems very low in the line organization, the boosts are the projects or task forces but they are not frequent. Some people have built islands of power over areas they are in control of. In reality there is an informal power structure which is hard to go around and it paralyzes information sharing.

As described in section 3.1, development is divided into tasks and the coordination of the tasks. In this organization work is divided by the nature of the work, software, research and so on. People with the same background and responsibilities are organized in groups in a bureaucratic way. Some managers are officially coordinating the work, while an unofficial structure is also affecting the coordination. In the projects the project managers coordinate the work and the work is performed by individual contributors in cross-functional teams.

The organization obviously suffers from drawbacks of the bureaucratic organization as well as the organic organization. The official organization is quite bureaucratic and that has an impact on motivation, see section 3.1.1. However, the informal organization is very organic and this is the structure that really matters. Because of the organic nature it suffers when the complexity of the product and development grows, coordination becomes a problem, see section 3.1.2. Since one of the main problems the organization experiences is their limited ability to cope with complexity (rather than change in the environment), the formal structure needs work properly.

To get away from the unofficial power structure information needs to be open and shared. There should be some redundancy in the positions so nobody becomes a limited resource to unbalance power. This will give the true hierarchy more power and thus the organization will become more bureaucratic which is necessary to cope with growth in complexity and size. Note that it is the true organization that should be a little bit more bureaucratic, not the current formal one. Also note that this is not the same thing as making people completely replaceable. Having an organization with replaceable people is inherently wasteful since everybody needs to work on the lowest common denominator.
Common decisions about general things should be handled by the bureaucratic hierarchy but the individual contributors should make decisions that affect their part, so decentralize these decisions as much as possible, according to section 3.1.5.

To cope with change a blend of the organic organization is necessary. To keep up efficiency in communication and information sharing, socialization is necessary by social events and informal connections between groups, section 3.3.2.

Task forces are popular and have shown to create motivation so they could be used in a strategic way to increase motivation. Individuals in the task forces should also belong more to the task force than the hierarchy in the organization to limit the ambiguity in loyalty; this is one of the advantages with the hypertext organization as presented by Nonaka, section 3.1.4.

With other words, the formal organizational hierarchy should be bureaucratic of its nature but decisions decentralized to the right level, section 3.1.5. The task forces stands for the organic part of the organization and that allows a flow of people back and fourth between the layers and spreading knowledge and learning different parts.

Another important thing is to build up the information layer of the organization. Formally express the intent of knowledge creation and put systems in place to store and share knowledge, more about this in section 5.3.

### Summary

<table>
<thead>
<tr>
<th>Findings</th>
<th>ID</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organizational structure is not clear. Struggle between official and unofficial structures. Hard to cope with size and complexity of the products.</td>
<td>[R_OS_1]</td>
<td>Information must become more open and shared to reduce the dependency of key people. There should also be some redundancy in positions. &lt;br&gt;The true organization should be a little bit more bureaucratic than today to handle complexity, section 3.1.1.</td>
</tr>
<tr>
<td>Cope with change and efficiency.</td>
<td>[R_OS_2]</td>
<td>A blend of organic organization is needed that facilitates socialization where people naturally can network and share information, section 3.4. &lt;br&gt;Also decentralize decisions to the right level, so people involved in something can make the decision. This will provide better decisions and people who take responsibility for the decisions, section 3.1.5.</td>
</tr>
<tr>
<td>Motivation is low.</td>
<td>[R_OS_3]</td>
<td>The task forces have proven to increase motivation so that should be formally institutionalized. Also the status of project manager should increase and people in a task force should not be part of the formal organization, section 3.1.4.</td>
</tr>
<tr>
<td>There is no common place to store information and there is no common understanding about how things work.</td>
<td>[R_OS_4]</td>
<td>A knowledge layer of the organization needs to be built to store and share knowledge and create a common understanding, section 3.1.4.</td>
</tr>
</tbody>
</table>

Table 5-1, summary of findings and recommendations for the organizational structure. R in the ID field stands for recommendation and OS for organizational structure.
5.2.2 Development model
The well documented official development model is a waterfall model, see section 3.2.4. It isn’t used in this part of the organization both because it doesn’t fit the informal organic structure and because it doesn’t fit software development. In reality semi-formal practices are used depending on who is involved.

The workflow in section 4.4.3 shows a waterfall flow. Even though different projects do things a little bit differently the parts of the flow described are the common denominators. While complexity of the products is the major concern they also struggle with the fact that it is not really possible to estimate up-front what the final implementation will be. They need a way to test and experiment with different solutions before the final implementation.

Introducing tools that work seamlessly together throughout the entire flow could facilitate a faster path from idea to implementation and a more iterative development model.

The research group sends requirements to the application developers, without knowing how they can be realized, through a document. There can be ambiguous and incomplete requirements in it and in most cases that isn’t detected until the test phase at the end when it usually is too late to fix the errors. Instead the requirements are updated.

The requirements are then used by system architects and engineers to split the work and create component requirements. They do that with help of a variant and interface handling tool. The variant part is because they might create several variants of the same product and the tool keeps track of the individual differences that customers have ordered. The component requirements inherit the possible ambiguities and incompleteness from the system requirements. On top of this, there is no way to guarantee that the different components will dynamically work together except for the skills of the system architect.

The architecture description is then passed to the platform group but it only includes the dependencies of the component so scheduling is not yet possible. Scheduling is done when the components are ready and handed to the integrators together with the platform code. When integrated it is the first time when it is possible to see if something works and if things are reasonable. Perhaps the algorithm is too complex or requires too much computational power for the target. In such cases major redesign is necessary but as stated above generally they change requirements instead, leaving a system that doesn’t match the market requirements and what was expected. This also leads to internal friction between marketing, sales and so on. They experience that they don’t get what they ask for and thus looses trust in engineering. On the other hand engineering looses respect for marketing and sales because they think they get unreasonable requirements.

Perhaps requirements are unreasonable and if that was detected in an early phase of the project a discussion with sales and marketing could sort out reasonable requirements and get control over their development.

Finally the test engineers do system tests when everything is put together but now errors are very expensive to correct and only the worst absolutely necessary problems are fixed. A note here is also that they don’t do any formal component tests, it is up to the individual developer to do his/her own tests and quite some angry faces can be seen when there is more or less simple errors in individual components detected during system tests.
### Summary

<table>
<thead>
<tr>
<th>Finding</th>
<th>ID</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ambiguous and unrealistic requirements are detected too late.</td>
<td>[R_DM_1]</td>
<td>Introduce a way to verify requirements early in the development, an iterative model based on evolution, section 3.2.2. Model based design can facilitate fast iterations to make it more efficient, section 3.2.8.</td>
</tr>
<tr>
<td>Errors detected too late in development.</td>
<td>[R_DM_2]</td>
<td>Introduce continuous test and verification to detect errors right after they are introduced. This is achieved with an iterative model and model based design, sections 3.2.8 and 3.2.9.</td>
</tr>
<tr>
<td>Problems with handling the complexity of the system.</td>
<td>[R_DM_3]</td>
<td>Introduce early system simulations to support the development. By refining parts of the system and still be able to do a full system simulation control can be maintained. Model based design grows an iterative model to handle complexity better, section 3.2.9.</td>
</tr>
<tr>
<td>No component tests are performed that makes it hard and time consuming to search for faults on a system level.</td>
<td>[R_DM_4]</td>
<td>Introduce formal component tests based on component requirements, this is facilitated by the V-development model, section 3.2.6.</td>
</tr>
<tr>
<td>Tools don’t work together, which is a possible source of errors and makes development less efficient.</td>
<td>[R_DM_5]</td>
<td>Minimize the number of tools and tool vendors used. Necessary tools should be integrated to work seamlessly together. Automate and integrate steps between tools, the idea with model based design, section 3.2.8.</td>
</tr>
</tbody>
</table>

Table 5-2, summary of findings and recommendations for the development model. R in the ID field stands for recommendation and DM for development model.

#### 5.2.3 Knowledge management

In this part, the organization will be analyzed with help of the knowledge creating enablers, section 3.3.4 and in terms of knowledge transfer, section 3.3.2. Together they determine how well the five-phase knowledge creating model is working, section 3.3.5. Also the knowledge layer of the hyper-text organization is analyzed here, section 3.1.4.

**Analysis of knowledge creating enablers**

The organization suffers from no explicit goal of creating new knowledge and being creative. It is up to the individual to create ideas and sell them internally. Also there is no plan of how to share knowledge internally nor externally. Some informal knowledge sharing clusters (power coalitions) exist.

Parts of the development are quite chaotic and rely on some strong individuals to take responsibility. This means that in regard of knowledge creating enablers autonomy and chaos is present but not intention, redundancy or variety.

**Analysis of knowledge transfer**

It is clear that there is no knowledge creating process in the organization. Socialization is present in the popular task forces they have. However, since those are a very small part of the overall activity socialization is not very developed. But the experiences that exist are important and can be used as momentum for change and further development.
Externalization comes in the form that each researcher documents requirements for systems based on their findings and knowledge. Developers only externalize by their implementations in code. There are no tools that can help them with externalization.

Internalization is hard since there is no knowledge layer where explicit knowledge can be studied and used for internalization. Once again it is up to the individual to gather information relevant for him or her and internalize. Also there are no good tools that can help with the internalization process such as simulation programs where experiments can be performed to increase understanding and build up the tacit knowledge. In the research department more internalization takes place, but still quite unorganized. Some computer programs are used for calculations and to present results. Also prototypes and physical experiments are used.

Combination of explicit knowledge takes place in the researches internal web collaboration tool where researches can share and combine explicit knowledge freely. This system also stores the explicit knowledge in a way that is searchable. However, it is only available for researches.

**Analysis of the knowledge layer**

There is no developed knowledge layer. Researchers have their web collaboration tool and documents are stored in a primitive system with limited search capability. The rest of the knowledge layer is based on informal structures.

<table>
<thead>
<tr>
<th>Finding</th>
<th>ID</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>All knowledge creating enablers are not managed.</td>
<td>[R_KM_1]</td>
<td>The knowledge creating enablers; intention, autonomy, chaos, redundancy and variety needs to be managed with a plan matching the organization and process.</td>
</tr>
<tr>
<td>The knowledge creating process does not work.</td>
<td>[R_KM_2]</td>
<td>The organization, process and tools need to facilitate the knowledge conversions to get the knowledge spiral going.</td>
</tr>
<tr>
<td>Insufficient knowledge layer.</td>
<td>[R_KM_3]</td>
<td>The knowledge layer needs to be improved and expanded with help of tools, databases and other means of storage. It also needs to be made easily available for everybody.</td>
</tr>
</tbody>
</table>

Table 5-3, summary of findings and recommendations for knowledge management. R in the ID field stands for recommendation and KM for knowledge management.

### 5.3 Objectives

The purpose with the objectives is to summarize a few things that based on the analysis will move the organization towards the high level goals. The objectives should be in the SMART format (Chaffey, 2002), specific, measurable, achievable, relevant and time-bound.

#### 5.3.1 Organization

The analysis in section 5.2.1 discovered that the organization officially is a matrix organization but in reality it was more organic in its nature because of its informal
To reach the high level goals the organization should move towards a hyper text organization. The hyper text organization was created to improve knowledge creation (Nonaka & Takeuchi, 1995) and by separating the task forces from the business layer allows the business layer to be more bureaucratic and thus be able to cope better with complexity. The knowledge layer is also included as part of the organization.

**Figure 5-2, the organizational shift captured by objective 1 and 2 (Own construction).**

**Objective 1 – Task force layer**

Recommendations: [R_OS_2, R_OS_3]

From three to nine months at least three pilot task forces should be executed. The reason for starting after three months is to give initial time to Objective 2 (below). The task forces should operate independently after getting goals and report directly to R&D manager. Within 18 months the pilots should be evaluated and the processes and work descriptions should be formalized, reviewed and approved by senior management outside of the R&D organization.

**Objective 2 – Business system layer**

Recommendation: [R_OS_1]

A plan for the transition of people from the business system layer to a task force should be developed within three months. It should include how to select people and how to fill the gap while they are gone. A next step should be completed within nine months when the first task forces are dissolved (Objective 1). This step should include how to bring people back in the business system layer once the task force is dissolved and how to spread their gained knowledge within the business system and knowledge base layer.
With the experience from the gain from the first two steps the plans should be formalized with processes, checklists and so on within 18 months.

**Objective 3 – Knowledge layer**

Recommendations: [R_OS_4, R_KM_3]

To improve the knowledge layer it should be clearly stated that it is under development, within three months. Two major tools for knowledge storage and cooperation should be up and running within nine months. It is important that high status people are utilizing these tools to get other employees to join in.

### 5.3.2 Development model

The analysis in section 5.2.2 showed that there is a difference in the actual development model and the official one. The official development model is similar to the waterfall model and should cope well with complexity. However, a lot of things are missing in the official model and all things and knowledge needed to develop the system are not known up front when the project starts. The actual model should thus be changed to one similar to the V-model, section 3.2.6 that together with model based design copes well with change and complexity, see Figure 5-3.

![Figure 5-3](image)

*Figure 5-3, Overview of the analysis of the development model. The suggested model is close to the V-model, see section 3.2.6. The green frame is the contribution of model based design to the suggested model, making it more capable to handle change and complexity (Own construction).*

The transformation to the suggested model is made by first identifying problems and bottlenecks in the current model and then by improving them with help of tools and model based design while keeping the steps in the V-model in mind.
Objective 4 – Standardize the simulation environment
Recommendation: [R_DM_5]
The foundation for sharing knowledge with models should be created by using Simulink as the only Simulation tool. This should be clearly communicated within three months. Then model libraries for utility models and blocks should be set up within nine months. Also the process for updating and verifying the models as part of libraries should be created. Using the processes all major blocks and models should be added to the libraries. The last part should be completed within 15 months.

Objective 5 – Process improvement
Recommendations: [R_DM_1, R_DM_2, R_DM_3, R_DM_4]
Within three to nine months one bottleneck in the development model should be improved by model based design. The specific improvement should be carefully selected to make a difference short term. Within 15 months two more areas should be improved, rule of two, (Smith et al., 2007). The findings/recommendations in section 5.2.2 should be the starting point for selecting the specific areas.

5.3.3 Knowledge management
The knowledge layer is handled as part of the organization. In this section the focus is on getting the knowledge creating spiral, knowledge transfer, to work with help of knowledge creating enablers, section 3.3.4. The current situation, as analyzed in section 5.2.3, of the spiral is illustrated in Figure 5-4. In Figure 5-5 the spiral is shown with improvements.

Figure 5-4, The current situation of the knowledge creating spiral. As it is it is not working properly (Own construction).
Objective 6 – Knowledge creating enablers
Recommendation: [R_KM_1]
Three areas should be identified and improved in regards of the knowledge creating enablers. This should be done between nine and 12 months. These improvements should give a long term effect on knowledge creation and are thus hard to measure. Employee surveys should be used to measure the short-term impact.

Objective 7 – Knowledge transfer
Recommendation: [R_KM_2]
At least two areas should be identified and improved in regards of knowledge transfer. This should be done between nine and 12 months. These improvements should give a long term effect on knowledge creation and are thus hard to measure. Employee surveys should be used in a task force to measure the short-term impact.

5.3.4 Timeline of objectives
The timeline in Figure 5-6 is a summary of the time bound objectives in the previous sections. These are suggested bounds that based on experience are a little bit provocative to introduce a sense of urgency for the change. However it is recognized that the organization needs to cope with current activities and deliver during the change work. Because of that these bounds might have to be adapted to the situation.
Figure 5-6, timeline of time bounded objectives (Own construction).
6 Option generation, evaluation and selection

In this chapter we will take the results in the previous chapter, the objectives, and generate a number of options of how to reach each objective. The options will then be evaluated and used in the next chapter to set up a plan for the change.

The options here are based on the analysis and recommendations. It is recognized that the organization might have different priorities and other options based on the situation. This chapter can serve as a guide and starting point for a more thorough analysis.

<table>
<thead>
<tr>
<th>Options</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Option Generation</td>
<td></td>
</tr>
</tbody>
</table>

Figure 6-1, the option generation and evaluation parts of the strategic development (Own construction).

6.1 Options related Objective 1 - Task force layer

6.1.1 Options for how to set up the task force structure

[O_TFL_1] Since there is a history of task forces in the organization it should be continued and used more frequently. Small changes should be made to match the hyper text organization.

[O_TFL_2] Create a new concept for task forces using external knowledge and new concepts.

6.1.2 Options for what goals the task forces should have

[O_TFL_3] Use tasks involving the change work as goals for the task forces.

[O_TFL_4] Use the task forces to implement functionality on the system.

6.1.3 Evaluation and selection

The first options are if the new task forces should be based on the old structure and just adapted or a completely new concept should be created. It might be tricky to change the old structure but in this case it is recommended to use it since it is proven and have shown an increase of motivation. Recommended option to use: [O_TFL_1]

For the options of what goals to use for the initial task forces the recommendation is to use tasks involving the change work. This will give increased momentum to the change work but there are drawbacks such as the work will be limited to the task force and...
doesn’t involve other people which might make the acceptance of the change harder. In the future when everything is up and running the task forces will work will functionality in the system, the second option, and it might require some changes to make this work. A third consideration is that work with the real system and releases needs to continue throughout the change work. In the final decision all this have to be taken into consideration but it is still recommended to use option [O_TFL_3] to implement specific parts in the new process to get things going and to test the task force concept.

6.2 Options related Objective 2 – Business system layer

<table>
<thead>
<tr>
<th>Recommendation: [R_OS_1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>A plan for the transition of people from the business system layer to a task force should be developed within three months. It should include how to select people and how to fill the gap while they are gone. A next step should be completed within nine months when the first task forces are dissolved (Objective 1). This step should include how to bring people back in the business system layer once the task force is dissolved and how to spread their gained knowledge within the business system and knowledge base layer. With the experience from the gain from the first two steps the plans should be formalized with processes, checklists and so on within 18 months.</td>
</tr>
</tbody>
</table>

6.2.1 Options for the selection

- [O_BL_1] Interest, volunteers
- [O_BL_2] Competence and knowledge
- [O_BL_3] Skills, other than technical such as problem solving, social, etc.
- [O_BL_4] Diversity, in background, experience etc.
- [O_BL_5] Availability
- [O_BL_6] People that can easily be replaced in the business system
- [O_BL_7] Status
- [O_BL_8] Use consultants

6.2.2 Options for filling gaps in the business system

- [O_BL_9] Using stricter and better defined tasks make people more replaceable.
- [O_BL_10] Define roles in a clear way to make it easier to get into the role.
- [O_BL_11] Develop specific training for roles.
- [O_BL_12] Have redundancy in roles.
- [O_BL_13] Use consultants to fill gaps.

6.2.3 Options for bringing people back in the business system layer and knowledge sharing

- [O_BL_15] Redundancy, extra time, for all to be able to learn
6.2.4 Evaluation and selection

For the first phase a small management team should set up a plan for how to select people for the task forces and how to fill the gaps in the business system layer.

The selection process is very situation dependant so all options are valid. It is, however, important that the task forces get the higher priority than the business system when selecting members.

To fill gaps in the business system layer options should be considered initially by the management team. Later in the formalization process should also be considered together with the experience from the initial task forces.

When people come back from a task force it is important that the knowledge they gained is shared and stored in the knowledge layer. Redundancy is an important part of the role descriptions and culture as is a reward system for knowledge sharing. The first task forces will be done before the culture and roles can change so it is recommended here to use presentations and Documentation.

Long term, however, all options can be valid depending on the situation and nature of knowledge and results.

6.3 Options related Objective 3 – The knowledge layer

[O_KL_1] Expand the usage of the web collaboration tool to include everybody, see section 4.4.1. One person should be responsible for each discussion group and there should also be someone who is responsible for adding new groups. It is important that high status members and managers are active in the discussions to give the system credibility.

[O_KL_2] Set up a document handling system with advanced search functionality. Appoint a person who is responsible for the document handling system.
Develop a style-guide for models and code and continuously improved to include more experience of development. This will make development faster and with higher quality in the long term. Appoint a person with excellent knowledge and credibility of all parts of the development to be responsible for this. This will create a common language and make communication of knowledge easier.

Set up a system that keeps track of different metrics to measure performance of processes. Feedback from customers should be included and bugs/enhancements suggestions from other departments.

### 6.3.2 Evaluation and selection

Long term all the options should be considered and probably also other ones. For this objective two should be chosen to start up the knowledge layer. It is important that the foundation is visible and can be used by everyone. The selected options are to continue the development of the web collaboration tool [O_KL_1] and set up a document handling system [O_KL_2].

### 6.4 Options related Objective 4 – Standardize the simulation environment

**Recommendation: [R_DM_5]**
The foundation for sharing knowledge with models should be created by using Simulink as the only Simulation tool. This should be clearly communicated within three months. Then model libraries for utility models and blocks should be set up within nine months. Also the process for updating and verifying the models as part of libraries should be created. Using the processes all major blocks and models should be added to the libraries. The last part should be completed within 15 months.

#### 6.4.1 Options setting up libraries and models

- [O_SE_1] Implement a common library for Simulink utilities. It should include two parts, one with tested and qualified utilities that can be used and trusted and another with more experimental utilities. It should be possible for everybody to access this. To add and change parts of it must go through a person who is responsible for the maintenance, who can check dependencies and so on. Develop a formal process for tested and qualified parts to be accredited.

- [O_SE_2] Build a Simulink library with plant models. This library will hold the collected knowledge about how the systems work; their understanding of it. The library will be the collective intellectual property of the mechanical structures controlled by software.


- [O_SE_4] Set up a configuration management tool for models and related artifacts. Develop a strategy for how to use the tool in projects.

#### 6.4.2 Evaluation and selection

To be able to handle models and libraries properly it is recommended to use the configuration management tool and strategy as the starting point [O_SE_4]. The configuration management tool is also the foundation for making models available for
everyone [O_SE_3]. Then it is recommended to do an inventory of existing parts and then set up a library for utilities [O_SE_1].

6.5 Options related Objective 5 – Process improvement

**6.5.1 Options for development model improvements**

[O_DM_1] By doing system level simulations on a high level in the beginning of a project requirements can be verified and problem detected right in the beginning. By continuous verification by simulation errors are detected a lot earlier and are much easier to correct, see [R_DM_1]. In the same way early system simulations verifies requirements to catch ambiguous and unrealistic requirements early, see [R_DM_2]. System level simulations can also help understanding the full system and thus handle the complexity of the system, see [R_DM_3].

[O_DM_2] To avoid component error to be present in system simulations and final implementation, introduce formal component tests, see [R_DM_4].

[O_DM_3] Integrate the variant and interface tool seamlessly with Simulink to be able to automate interface checks and automatically create components model templates.

[O_DM_4] Use automatic code generation from the simulation model for rapid prototyping of the software.

[O_DM_5] Use automatic code generation from the simulation models for the final implementation.

**6.5.2 Evaluation and selection**

Doing formal component tests have a direct impact on the quality. It is also possible to start without being dependant on other things, it is therefore recommended to start with [O_DM_2]. Then with components in order is it possible to put them together to perform system level simulations [O_DM_1] to get the benefits of testing the full system early in the simulation environment to detect system level errors.

It is recommended to use automatic code generation from the components as the third improvement. When the components are formally tested it is possible prepare the model for code generation [O_DM_5].

6.6 Options related Objective 6 – Knowledge creating enablers

**Recommendation: [R_KM_1]**

Three areas should be identified and improved in regards of the knowledge creating enablers. This should be done within 12 months. These improvements should give a long term effect on knowledge creation and are thus hard to measure. Employee surveys should be used to measure the short-term impact.
6.6.1 Options for improvement of the knowledge creating enablers

[O_KC_1] Management needs to articulate and include knowledge creating goals in the corporate vision, intention.

[O_KC_2] Introduce redundancy by giving employees specific time that is not allocated to direct work. This “free” time can be used to investigate ideas, learn things from the knowledge layer and add or structure things in the knowledge layer.

[O_KC_3] Put people together with overlapping responsibilities, this will give them an opportunity to try different approaches and get into each others work. When people have overlapping areas it will introduce some healthy competition and increase knowledge creation.

[O_KC_4] To increase variety, put people together with different backgrounds or views about how things should be. Also in task forces include people from other parts of the company to give different perspectives on the project. Other parts can be sales, marketing, production and management.

[O_KC_5] In research and in the task forces a controlled chaos can be created to increase creativity. In development less chaos is need to be able to cope with the complexity which is a more important requirement. Chaos can be introduced by giving a task force somewhat unclear tasks or ambiguous goals.

6.6.2 Evaluation and selection

It is recommended to start with the intention, [O_KC_1], to make the effort known to everyone and to show that knowledge creation is valued.

Redundancy, [O_KC_2-3], is important both in the business system layer and in the task force layer, Objective 1 and 2. It is recommended to start with redundancy in the business system layer but focus on the long term instead of the current improvement as in Objective 2. It is recommended to work with people responsible for Objective 2.

When the business system layer have been improved and the first task forces have been completed the focus can be on redundancy in task forces, also thinking about the longer term use lessons learned from the task forces.

6.7 Options related Objective 7 – Knowledge transfer

Recommendation: [R_KM_2]

At least two areas should be identified and improved in regards of knowledge transfer. This should be done within 12 months. These improvements should give a long-term effect on knowledge creation and are thus hard to measure. Employee surveys should be used in a task force to measure the short-term impact.

[O_KT_1] Socialization increases with more task forces where individuals work closely together to share tacit knowledge. But also when individuals come back to their groups they share knowledge. Allocate time for an individual who returns to the business layer for spreading knowledge in the new role.
Introduce more social events like off-site workgroups to set up goals or discuss specific topics. Socialization takes place both in the discussion and in spare time on these events.

Create creative corners and places where people meet naturally to socialize.

Externalization will be greatly improved with simulation models. The model can be a facilitator for the researcher or developer to experiment and test their ideas. The models bring out their tacit knowledge and can then be used by others to build up their tacit knowledge or understanding, internalization. Models should be completed with documentation of parts that is not easily understandable from the model.

6.7.2 Evaluation and selection

Improvements of knowledge transfer are also covered by other objectives as a side effect, task forces [O_KT_1] and simulation models [O_KT_4] fall into that category. But many other changes from other objectives will also improve the knowledge creating spiral. It therefore recommended here to focus on socialization in the form of social events [O_KT_2] and improve the working environment [O_KT_3].
7 Planning

In this part a plan of how to implement the changes is presented. The plan is based on Kotters ten steps (Kotter, 1996) for successful change in organizations, see chapter 3.5. The plan includes the options presented in chapter 6 and should be treated as a recommendation or a guide for setting up the real plan. There might be cultural obstacles or other things not caught during the analysis that needs to be taken into consideration. The transition management team, TMT, should take the recommendations and adapt it to the situation before starting the change work.

![Figure 7-1, The planning part of the strategic development (Own construction).]

7.1 Kotters ten steps and recommendations

In this section Kotters ten steps are described together with specific recommendations for the organization.

1. Establish a sense of Urgency
Since there have been some major failures due to complexity of the system there should be enough background to create the sense of urgency and communicate this to the organization.

2. Form a Powerful Coalition
Identify key people and create a transition management team, TMT. It should include 5-6 people. It is recommended to have the R&D manager, research manager and system development manager in the team. If possible having the CEO involved in the work and make that visible would be very good. If someone is identified from the beginning to be against with change, it might be good to include him or her in the TMT but with some care.

3. Create a Vision
A vision about a knowledge creating company and the efficiency of model based design should be created. In particular the vision should show why the new way will help develop more complex systems.

4. Communicate the Vision
Ask the CEO or a person as high as possible in the organization to communicate the vision. A short presentation at a company meeting followed by some more details in an email. Internal web-page could also be used. This top person should also explain that the TMT has his full support in the efforts.
5. Empower others to act on the Vision
Take a few positive individuals and make a first task force to make a pilot project. Make their work visible.

6. Reduce resistance and obstacles
If individuals or power coalitions resist the change, highlight this and use different tactics to convince them to share knowledge and change behaviour. Identify if there are some old structures or processes that hinders the change.

7. Plan for and create short-term wins
It is important to really plan for and create short-term wins, not hope for them. Don’t declare victory because of a short-term win, instead use the credibility created to continue the change process.

8. Increase efforts for change
Train and inform more employees to implement the change. New change projects to change systems and structures.

9. Institutionalizing new approaches
Articulate connection between success and the new behaviours. Dissolve TMT. Document new workflows and processes, create internal training classes.

10. Continuous improvement
Follow-up change effort and improve pieces.

7.2 The plan
During the implementation of the change it is important to have support from someone with experience from model based design and knowledge management. This is important both from a technical point, help with integrations and tools, and from a cultural view. The knowledge creation is more a cultural thing than something that happens with tools that is why it is important to have someone who can support the cultural change.

The intention with this plan is to have it as an input to the real planning process that takes more consideration to the actual situation in the organization with current requirements of delivery, culture and so on.

The plan is divided in five phases going from zero to eighteen months. This might be a little bit aggressive for the change but it is important to have a tight schedule to create a sense of urgency. In general change takes longer than expected but when successfully managed it has a greater impact than expected long term.

Phase three can optionally be implemented in parallel with the rest of the work. It could be beneficial to have some parts of it going earlier, for example socialization in the form of a kick-off for the entire project or a comfortable creative corner for discussions.
7.2.1 Phase 1 – Preparation (0-3 months)

The first phase of the change handles some necessary preparations, both process wise and to prepare employees for what to come. Steps 1-4 above in section 7.1 should be the start of the change effort. Form the transition management team (TMT) to supervise and plan for the change. They should also set up and communicate the vision clearly to the organization. The vision should include the intention to be a knowledge creating company, first part of objective three and objective six, and that a knowledge layer is to be created. The intention of introducing Simulink as a standardized modeling and simulation tool should also be communicated, first part of objective four.

When the vision is created and communicated the TMT should adapt the current task force system to the new way according to objective 1. They should also look at the process for how to select people from the business system layer to participate in different task forces, objective two. Since this work is about change it is important to select people who are willing to do the change at this stage. Thus the selection process must be flexible enough to handle different situations, see [O_BL_1-8] for different options.

Finally the TMT needs to put together a plan of how to fill the gaps in the business system while people are working in the task forces, objective two. It is important to make sure that the business system still is functional and can deliver as required. Defining tasks and roles in the business system in a stricter and clearer way will help to fill the gaps, options [O_BL_9] and [O_BL_10]. With clear descriptions it is easier to hand over the work to someone else.

7.2.2 Phase 2 – Pilots (3-9 months)

In this phase three task forces should be put together, objective one. Each task force should implement a part of the change work according to objectives.

The first task force should focus on developing the knowledge layer by setting up the web collaboration tool and the document handling system according to objective three.

The second task force, working with objective 4, should set up the configuration management tool with a file structure suitable for models. They should also make an inventory of existing components and utility blocks. Libraries for utilities should be set up for the most common blocks to make them easily available for everybody.

The third task force should set up a process for formal component tests, objective five.

During phase two the TMT should work on a process of how to bring people back in the business system layer when a task force is dissolved, objective two. They should also plan for how to spread and store knowledge that has been created in the task forces.

Problems and obstacles during this phase should be resolved by the TMT to avoid them in future projects. View every step as a learning opportunity.

7.2.3 Phase 3 – Knowledge Management (9-12 months)

This phase can optionally be implemented by the TMT in parallel with other activities.
The TMT should look plan for how to introduce redundancy in the work. Objective 2 handled this for the short term, for the first task forces but here the focus should be the longer term, objective six.

Introducing natural places and venues for socialization is another job for the TMT. It should be in the form of kick-offs and other social activities as well as changing the working environment by creating creative corners and possibly rearrange seating, objective seven.

In this phase knowledge should be spread more widely by formal training of the new tools and processes. Training should not only include people from the R&D department but also interfacing groups such as marketing and sales. By training everybody communication and knowledge transfer will be greatly improved. Perhaps there should be specific light weight training for interfacing groups.

7.2.4 Phase 4 – The process implementation (12-15 months)

In this phase the focus is on the development process and improvements of it. It is recommended to continue to use task forces for the work here.

The libraries and inventory created by the second task force in phase 2 should now be further developed by a formal process for how to change and add models and libraries. All available utilities should then be added to the libraries and models should be check into the configuration management system, objective four.

With formal component tests as developed in phase two and all component models available in the configuration management system work should continue to support system level simulation, objective five. System level simulation is a great tool to handle the complexity and to detect system level problems early in the development rather when all parts have been put together.

To further gain advantages with model based design models should be used for one more purpose, objective five. With formal component tests the model have assured quality and are thus suitable for code generation which will further save development time and reduce errors.

7.2.5 Phase 5 – Institutionalize the new way (15-18 months)

In previous phases some major changes towards a more hyper text based organization and processes more fit to the organization and its situation have been made. It is time institutionalize the new way with approved and documented processes, workflows, checklists and so on.

In particular the task forces and the processes around it should be documented by the TMT and approved by senior manager. It includes documenting the purpose of task forces and how they should operate, objective 1. It also includes how to select people for the task forces, fill the gaps in the business layer and how to dissolve the task forces and spread knowledge, objective 2.

In this phase it is also necessary to take a look at the big picture, what have changed and what has not? What are the next steps?
8 Conclusions

Based on the findings in the organization and theory it was clear that the organization wasn’t well adapted to its internal, external environment or goals. A number of specific improvements could be identified based on the theory. The improvements were expressed as specific objectives and include process, organizational and knowledge management change.

To handle both innovation and complexity of a system it is not possible to make significant improvement by focusing on one piece of an organization such the processes. Process improvement by itself could improve small parts but for a major change and significant improvement the processes must be put into context with the organizational structure and knowledge management. Internal environment such as culture and resources together with the external environment should shape the development of the organization to allow it to work towards the high level goal, see Figure 8-1.

It was also shown that Model-Based Design can improve the development processes and knowledge management significantly, section 5.3.

![Figure 8-1, important parts in organizational development (Own construction).]

8.1 Further work

Metrics have not been considered in this study but measuring value of knowledge and other assets is necessary to track improvements. Metrics that give a balanced view of the different parts of an organization and how well they work together is also an area for further work.
The improvements of processes when Model-Based Design is introduced are well understood and obvious but the effects on organizational structure is not well understood and thus an area for further research.

In chapter 3.2.7 some agile development models are listed but left out of this work since there are no studies in this area. How model-based design can be used with agile development would be a very interesting investigation.
References

Suwatthikul, J. (2009). Retrieved April 5, 2009, from Warwick School of Engineering: http://www2.warwick.ac.uk/fac/sci/eng/pg/students/esrebe/research/

**Related literature, not directly referenced**


