CHALMERS

Learning in Product Development

How Automatics can detect and/or resolve quality issues earlier in the product development process

Master of Science Thesis

Eric Johansson

Salman Eskandari

Department of Quality and Operations Management/Center for Business Innovation

Division of Technology Management and Economics

CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden, 2010

Report No. E2010:065

Abstract

Practitioners and scholars alike have paid tremendous attention to the detection and/or resolution of quality issues early in the product development process (Clark & Wheelwright, 1992; Repenninger, 2001; Bergman & Klefsjö, 2003). The earlier quality issues are detected and resolved in the product development process, the lower the cost for the developing company.

In this thesis, we investigate the conditions for detecting and/or resolving quality issues early within the product development process and give suggestions on how these conditions can be improved for an earlier detection and/or resolution. We do this through investigating a specific product development project at Automatics[1]. The project was called Panther and aimed to replace an outdated product family at Automatics. Automatics is a global provider of Automation solutions, and a business unit within the Nordic Automation group.

By analyzing the formal product development documents at Automatics and conducting semi-structured interviews with various stakeholders in the project, we set out to answer the questions of when quality issues were *detected* during the Panther development project, when they were *resolved* and if they could have been *detected and/or resolved earlier*. Four encountered quality issues were scrutinized and the above questions were asked in relation to these four quality issues.

For some of the quality issues, the answers to the above questions were rather clear-cut. For others, they were not. The fact that the methodology did not suffice to answer the research questions is also one of the central contributions of this thesis, both to Automatics and the academic world.

However, the analysis of the formal documents and the semi-structured interviews unraveled a number of interesting areas for improvement in product development at Automatics. We provide a number of recommendations to Automatics, which hopefully will lead to improved product development efforts generally, and earlier detection and/or resolution of quality issues in the product development process specifically.

---

[1] Automatics is a made-up name. So are the names of the products and their components. Partly, Automatics requested this, and partly it was suggested by the researchers in order for the reader (and other academics) to not get bogged down in details regarding the organization.

Table of contents

1 Introduction

This introductory part of the project plan will begin with a description of the background to this thesis, where we shortly present the company Automatics and the case we are studying. We then continue with a presentation of the purpose and research questions before ending with a description of the delimitations that we have made in this study.

1.1 Background

Automatics is a business unit within the Nordic Automatics Group, providing effective automation solutions globally. The company manages development, production, and sales and distribution of automation products. Automatics manages multiple new product development projects every year, ranging from next generation projects to face-lifts of existing products. A couple of years back, the development projects Panther and Lion where initiated in order to replace two outdated product families. During the progress of Panther and Lion – which were developed within the frames of one development project – various quality issues led to several postponements of deadlines. Some of these quality issues were detected late in the development process, and some detected early, but stayed unresolved for longer periods. In turn, this caused delays to the project and problems at customer sites.

In today's competitive environment, the success of companies is becoming more and more dependent on the capability to consistently develop superior products (Clark & Fujimoto, 1991; Clark & Wheelwright, 1992; Kahn, Castellion & Griffin 2005; Morgan & Liker, 2006). These new products – products that have been on the market for less than two years – account for an increasing percentage of revenues (Kahn, Castellion & Griffin 2005).

Moreover, empirical studies have shown the negative effects on time and cost that accompany poor decisions early in the product development process (Clark & Wheelwright, 1992; Morgan & Liker, 2006). Bergman & Klefsjö (2003) argue that a quality issue that is detected in the marketplace approximately costs 1000 times more than if the same quality issue was detected in the design stage. A number of studies have shown that although this is known by virtually all companies, a majority of them still engage in solving problems as they emerge late in the process (Repenning, 2001). These empirical studies show on the importance of detecting and resolving quality issues early in the product development process. By solving the root causes for quality issues early in the development process, expensive, suboptimal and often inferior late changes can be eliminated (Morgan & Liker, 2006).

As recognized by Repenninger (2001), all firms would like to detect and solve quality issues up-front, but a majority still find themselves in situations where quality issues are detected and resolved late. Thus, detecting and resolving quality issues early in the product development process is an extremely important topic for scholars and practitioners alike. Through a case study at Automatics, this thesis investigates the conditions for such early detection and resolution of quality issues. Although the results of a case-study are context specific and cannot be generalized (Bryman & Bell, 2007), the lessons learned from Automatics may be valuable to other organizations seeking to reduce the negative impacts of late quality issue detection and resolution.

1.2 Purpose

To investigate the conditions for detecting and/or resolving quality issues early within the product development process of Automatics and give suggestions on how these conditions can be improved for an earlier detection and/or resolution.

1.3 Research Questions

To fulfill the purpose of this thesis, we have specified three main research questions.

1. **When were quality issues from the Panther development project detected?**
2. **When were quality issues from the Panther development project resolved?**
3. **Could quality issues from the Panther development project have been detected and/or resolved earlier?**

1.4 Delimitations

In the scope of this thesis, we will not investigate whether or not Automatics made the correct decision to initiate the development of Panther. Moreover, the development project at Automatics included the development of two product families in parallel, Panther and Lion. We will not be directly concerned with the development of Lion. However, the effects of developing two product families within the bounds of one project might be interesting for the development process of Panther. Thus, while the effects of Lion on Panther might be of interest to us, we will not investigate specific quality issues regarding the Lion project. Finally, due to restrictions in resources, we will during this thesis exclude cost as a variable.

2 Theory

In this section, we will begin by presenting three different approaches to viewing error detection and resolution. We then choose to view this thesis in one of the three approaches, and move on by relating it to organizational learning.

2.1 Error detection/resolution within the system

Psychologists and organizational theorists have conceptualized errors and accidents in at least two different ways. Their research has investigated both the causes of errors and the efficacy of preventive strategies (Edmondson, 2004).

According to Edmondson (2004), one of these approaches focuses on the individual and the other on the role of the system within which individuals operate inducing or preventing accidents. Further, Edmondson (2004) proposes a third approach, which integrates the individual- and system level of analysis. She proposes a model that accomplishes this by focusing on the work group and uses this as the unit where "*organizational and cognitive effects meet and play out in enabling or preventing errors* " (Edmondson, 2004, p. 68).

2.1.1 Individual-level approach

Over the years, psychologists have offered both cognitive and affective causes for human error. For example, schema theory proposes that perceivers' expectations, or frames, may steer human attention away from visual data, which enables our perceptual processes to construct images consistent with expectations, i.e. we see what we expect to see (Edmondson, 2004). In a product development project this could mean that one runs a test, which usually runs smoothly, failing to acknowledge when the product deviates from its accepted trajectory.

Moreover, activities that are repeated with a high frequency, which therefore run a risk of being carried out as an unconscious routine, may allow us to make odd slips (Reason, 1990). This kind of slip may be recognized by comments such as "I wasn't thinking" or "I can do it in my sleep" (Edmondson, 2004).

In addition, emotions such as anger or anxiety may cause unconscious effects resulting in distraction from the task at hand (Edmondson, 2004). This might be exemplified by being anxious for a performance review or an upcoming presentation.

Edmondson (2004) notes that commonly suggested preventive strategies for the individual level approach includes individual-forced devices such as computerized order entry to prevent bad handwriting. We argue that this also can be compared to the concept of "poke yoke", which according to Morgan & Liker (2006) is used by Toyota in their product development in the following ways:

- Checklists
- Standard, detailed test plans
- Part quality matrices
- Standard architecture
- Shared components across vehicles
- Standardized manufacturing processes

2.1.2 System-level approach

Edmondson (2004) states that certain sociologists and organizational theorists have focused on the system level in order to understand error. An example is Perrow (1984), who, instead of focusing on why slips are made by individuals, examines the design of systems and its affect on human error. Thus, the assumption is that individual human error is given. According to Perrow (1984), what is critical is instead to understand the conditions that affect errors to escalade into catastrophes. In other words, what conditions, related to the system, that prevent (or allow) simple slips to trigger a series of events that eventually ends up in disaster.

According to Edmondson (2004) the nature of the system both affect the actions of individuals and the consequences of errors. A "normal accident" is a predictable outcome of a system that has both *interactive complexity* and *tight coupling* (Perrow, 1984). Reason (1990) argues that these two characteristics of a system are relatively independent and particularly important. The following general features characterize interactive complex systems (adapted from Perrow, 1984):

- Components that are not linked together in a production sequence are in close proximity
- Many common-mode connections (i.e., components whose failure can have multiple effects 'downstream') are present
- There is only a limited possibility of isolating failed components
- Due to the high degree of specialization, there is little chance of substituting or reassigning personnel. The same lack of interchangeability is also true for supplies and materials
- There are unfamiliar or unintended feedback loops
- There are many control parameters that could potentially interact
- Certain information about the state of the system must be obtained indirectly, or inferred
- There is only a limited understanding of some processes, particularly those involving transformations

Thus, due to irreversible processes and multiple, nonlinear feedback loops, interactive complex systems involve hidden interactions; the consequences of actions cannot be seen directly (Perrow, 1984). Edmondson (2004) argue that the system that administers drugs to hospitalized patients is characterized by considerable interactive complexity. For example, if the physician prescribes the wrong medication, this might go unnoticed until the patient has an unexpected reaction, i.e. the consequences of prescribing the wrong drug cannot be seen immediately. It is not very difficult to argue for that the product development process is characterized by interactive complexity. One could for example use the high uncertainty and lack of process understanding that usually characterizes product development (Clark & Wheelwright, 1992) to conclude that most of the points above are valid for the system. Thus, we argue that product development processes are characterized by interactive complexity.

Further, systems with tight coupling are characterized by (adapted from Perrow, 1984):

- Processing delays are unacceptable
- Production sequences are relatively invariant
- There are few ways of achieving a particular goal

- Little slack is permissible in supplies, equipment and personnel
- Buffers and redundancies are deliberately designed into the system

Tightly coupled systems thus have little slack, which means that actions in one part of the system directly and immediately affect other parts (Perrow, 1984).

Edmondson (2004) further argues that the system that administers drugs to hospitalized patients, in contrary to tight coupling described above, is characterized by loose coupling because a failure caused in one part of the system can be caught and corrected without causing harm (Edmondson, 2004). According to Reason (1990), similar to the system linking medication to patients, research development companies have loose coupling and complex interaction. Further, Edmondson (2004, p. 69) states:

> *"although the modern hospital thus does not fit Perrow's worst case scenario, the interactive complexity of medications does create considerable potential risk for patients"*

However, it is debatable if the product development process can be classified as having tight- or loose coupling. Depending on if the product is highly modular or highly integrative (Ulrich, 1995; Robertson & Ulrich, 1998), and depending on how much the actual process is designed according to the structure of the product, we argue that the product might steer the process into becoming a somewhat more coupled process. However, this is not a thesis about system engineering and its relationship to the product development process. In addition, the organization in which the product development process is run is characterized by a functional structure. Thus, we argue that it is fair to assume that the process is loosely coupled.

Moreover, according to Edmondson (2004), in the same way that the design of a system can cause accidents, design-solutions implemented to prevent errors may trigger other risks and new errors. Thus, according to the system-level approach, solutions implemented to "cure" symptoms of a system might fail to address the complex underlying problem and even aggravate the problem if the "right" chain of cause and effect presents itself (Edmondson, 2004).

This approach further deals with broader social forces that might affect attitudes and behavior related to error. For instance, if the social forces within an organization associate error with incompetence, hierarchical relations might lead to suppression of mistakes and denial of responsibility. In other words, the hierarchical structure might inhibit such a systematic analysis of mistakes needed to prevent them (Edmondson, 2004).

2.1.3 Group-level approach

Edmondson (2004, p. 70) states that: *"individual skill, motivation, and cognition are imperfect. Organizational systems are inevitably flawed"*. Further, this ever-present potential for error might be addressed with teams acting as "self-correcting performance units". Well-functioning teams seem to have a way of coordinating tasks, anticipating and responding to each other's actions (Edmondson, 2004).

There are two research traditions within the social psychology examining the differences in group behavior and performance. The first of these two focuses on social, affective, and unconscious influences on groups and group-members, whereas the second focuses on cognition, goals, and structures. These traditions share the view that a group is more than the sum of its members as well as that group phenomena exist and influence task performance (Edmondson, 2004).

Social, affective and unconscious influences (tradition one)

Edmondson (2004) presents a number of researches within this tradition:

- The 19th century work of Gustave Le Bon; unconscious processes of a crowd or group are manifested in the acts of individual members
- The Hawthorne studies; emotion and tacit group norms can have a greater effect on performance than working conditions or economic incentives
- The impact of having a strong social unit on workers' motivation
- The intergroup perspective; how membership in identity groups (e.g. gender or race) and membership of organizational groups (e.g. rank or function) affect communication and motivation in task groups that could lead to errors due to bad coordination

Furthermore, regarding the last point above, tacit boundaries, imposed by rank or identity group, might disturb communication and thus cause transmission of invalid data (Edmondson, 2004).

Cognition, goals and structure (tradition two)

Within this tradition Edmondson (2004) further present a number of areas. Focus is here on identifying those conditions that help teams work together and solve problems.

- The affect of leadership style on positive group outcomes
- The impact of practice and members' interpersonal skills on team performance
- Structure of team, design of task and supportiveness of organizational context (reward, information, and educational systems)

We argue for that the system perspective is the most appropriate perspective to use while addressing these issues. We agree with the notion of Perrow (1984) that human error is inevitable and will always be present. Edmondson makes the argument that systems are inevitably flawed. However, we argue that taking the group perspective when analyzing quality issues within an organization not is enough in our case. That is because quality issues do not take group-constellations into consideration. In other words, combining factors originating from different groups might cause quality issues. Just as Edmondson (2004) argues, it might be so that some groups, over time, learn how to avoid quality issues because they become self-correcting performance units in which members learn how to recognize each other's common mistakes. However, we argue that groups, over time, develop a certain set of norms that dictate what is allowed and what is not allowed and what is punished and what is not punished. In other words, groups develop a kind of subculture, which dictates behavior within a group (Shani et al., 2009). In addition, different groups naturally have different responsibilities and are thus staffed with people that have acquired different and specific experiences and knowledge. In other words, groups are filled with people with suitable backgrounds. Moreover, groups are naturally involved with different tasks, which means that they over time will develop different bases of knowledge. These three arguments suggest that groups working inside a specific knowledge space are more familiar with potential quality-issue-factors within that knowledge space than other groups operating inside a different knowledge space. Since the product development process at Automatics is run within an organization that is divided by function, we argue that the system perspective more effectively will catch the nature of quality issues. However, if Automatics had used a tiger team or a cross-functional team in their product development we would naturally have had to focus on a single group which means that the group perspective might have been more appropriate. In addition, we would like to point out that our purpose is different from that of Edmondson (2004), who sets out to unravel *differences* between groups and not how a whole constellation of groups together can improve their management of quality issues.

Oxford American Dictionaries define system in two ways that may shed some light into what we mean with system in this thesis. Firstly, a system is:

> *"a set of connected things or parts forming a complex whole,*
>
> *in particular a set of things working together as parts of a mechanism or an interconnecting network".*

Moreover, it could also be defined as:

> *"a set of principles or procedures according to which something is done, an organized scheme or method".*

We argue that the latter of these definitions are most appropriate to use in our case. By studying how quality issues was handled – the procedures, schemes and methods around them – we have unraveled the conditions for detecting and/or resolving quality issues early in the product development process. However, this is not to say that we assumed that Automatics had a formal system. We have also taken the informal parts of the system into consideration. Moreover, taking the group perspective would have meant centering more on human behavior and psychology. Even though we not have neglected such factors entirely, the study is not centered on them.

2.1.4 Quality issues in product development: A systems perspective

We carry with us a number of characteristics of the product development system from Perrow's (1984) framework. First of all, we recognize that both the product and the development process could be seen as systems. We have already argued for the product development process being similar to a system characterized by interactive complexity and loose coupling. We will continue this line of argumentation later as well. Moreover, we argue that the product, like the product development process, can be recognized as a system. The product is further the outcome of the product development process and a system living in parallel to it. We argue that the product, like the product development process, is characterized by interactive complexity. However, we also argue that, unlike the product development process, the product is characterized by tight coupling.

Thus, as the product is built together into a prototype, we argue that the physical coupling between parts become tighter. However, even though we recognize that the characteristics of Perrow's (1984) concept of tight coupling not are directly transferable to the concept of the product, we argue that this comparison brings some clarity into the kind of information that are obtained from the product along the way of the product development process. As the product is built together, events in one part of the product directly affect other parts of the product. However, due to the properties of being a interactive complex system, the effects might not be seen directly, the effects might present themselves in unusual ways, effects from several sources might interact, etc. Thus, the information received from testing the product, or some other activity, might be distorted due to several unknown causes.

One characteristic of the product in this particular case is that testing is relatively time-consuming. Even though we say that the system of the product can be seen as tightly coupled, and events in one part therefore directly affect other parts, it should be noted that the strength of the effect is a matter of degree. The product is worn down during its entire lifetime and therefore, depending on the strength of the effect and the methods and technology used, this might be seen fast or slowly, but in either case be devastating for the project. In fact, it would be preferable to see a failure directly in order to start the work of figuring out the causes as fast as possible.

Moreover, the product- and process systems are interacting during the whole project. The product is the output of the product development process. During this process, the product is planned, constructed, tested, etc. Problems that occur in these activities affect the product development system during the whole product development process. As indicated above, we view the product as one of the information sources within the product development process. The product is further the information source that we focus mostly on in this thesis. However, it should be noted that we focus on the product indirectly, i.e. we focus on the mechanisms used to transform information about the product. Thus, the product development process is viewed as an information handling process, a process that transforms information over time with the purpose of creating a product. This view is consistent with the view of Morgan & Liker (2006) who state that product development essentially is defined by flows of information.

Several information sources have been used during the product development process in order to elicit information about the product and create a foundation for future action. The ones that are most tightly tied to the product are presented in figure 1. We do not argue that this is a complete set of information acquisition mechanisms used during the project. However, these are some of those that have been made available to us by Automatics and used to analyze quality issues. These information sources are presented here for the sake of clarity and will be discussed more thoroughly in section 3.3, *Research methods*. In addition, it is important to note that these information sources do not only supply the project with information regarding the product *per se*, but also with information about the progress of the product development process. E.g. the test results speak about the performance of the product development process in terms of the product and the project meetings more directly address issues regarding project management. Moreover, some of these processes begin at different stages during the project and some follow the project for its entire lifetime. Thus, the figure presented below is not displaying the actual sequence or placement of information acquisition mechanisms over time, but rather the principle.

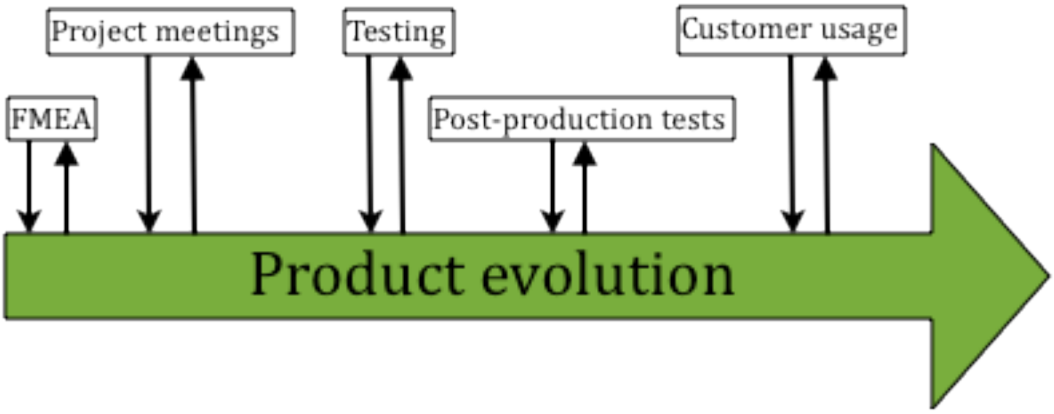


Figure 1. Information sources used for quality issues

In this thesis, we are not directly concerned with the practice of engineering and the mechanisms used to elicit information about the product from various tests and activities, but rather with how this information is managed within the project. However, from the discussion above, we can conclude that data collected from the product as it evolves through the process might be difficult to interpret.

In order to characterize the system we are going to study, we will now develop the characteristics of a complex interactive, loosely coupled system in the context of a product development project. These characteristics are adapted from Perrow (1984).

Firstly, *components that are not linked together in a production sequence are in close proximity*. Of course, this might not be true in every case, but we argue that there are many instances within the process where this is true. For example, a certain component planned in the beginning of the project may affect a component in the production longer down the road. Thus, even though these events not are tied together in sequence, they might have a close connection in the form of a cause-effect relationship.

Secondly, many *common-mode connections (i.e. components whose failure can have multiple effects 'downstream') are present*. In this context, this means that many components of the product development process potentially can affect more than one component down the road, e.g. a decision made today could cause two quality issues in a week. Moreover, due to the high complexity and uncertainty inherent in product development (Clark & Wheelwright, 1992), it is difficult to determine which components visible today will cause quality issues later on. In addition, we have studied a rather complex product which integrative characteristic makes it difficult to see these kind of relationships also in the product.

Thirdly, *there is a limited possibility of isolating failed components*. In other words, in a product that has failed, it might be difficult to isolate one component responsible for the failure. This means that it is difficult to elicit a single root-cause responsible for a single failure. This is exemplified in the following depiction:

Picture a farmer driving his wagon on the road. The wagon hits a tuft of grass on the road. The wheel brakes and the wagon tilts. Now, the farmer asks himself, why did this happen?

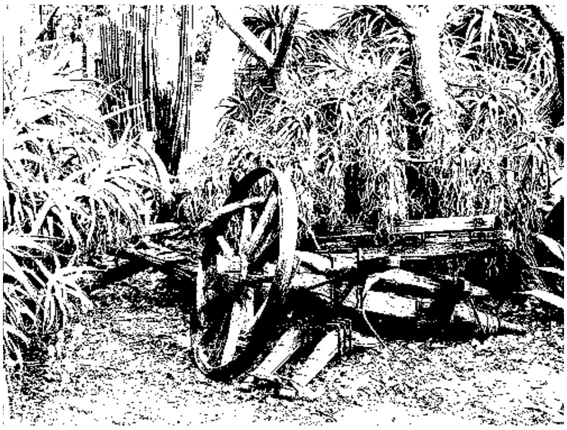


Figure 2. Famer and his wagon

Some possible explanations for why the wagon broke and tilt are:

- The farmer was not monitoring the road while driving
- The wagon was too heavily loaded and the mere pressure on the wheel as it hit the tuft of grass made it break and thereby tilt
- The person responsible for road maintenance had not performed his job correctly and left a tuft of grass on the road, which caused the wheel to break and subsequently the wagon to tilt
- The wagon was not designed with the users' needs in mind
- The farmer did not read the product specification and did thereby not know that the wagon not was designed for usage on roads with tufts of grass

This list could be made longer, but the points it illustrates are that there is no single "true root-cause" and that different causes combine into quality issues (as was mentioned earlier). That there is no single "true root-cause" has implications, both for Automatics and for our study. Through the logic just presented, we argue that there is no way to identify single "true root-causes" for the quality issues we have examined. What we can do is though to study the system used to handle them, compare this system to theory and thereby design strategies aimed at improving the system.

There are unfamiliar and unintended feedback loops, which means that information about the product can surface in unexpected ways and in unexpected contexts. Therefore, important information may be difficult to recognize as it presents itself. Firstly, indications of quality issues in the product might present themselves in unusual ways, e.g. a substance in the product might convert into an unusual characteristic which organizational members may lack the knowledge to make sense of. Secondly, since one lacks knowledge of the characteristic of the substance one might be reluctant to transmit information about it through formal channels and the information might instead surface in the form of a rumor, making it difficult to assess the accuracy of it.

Moreover, *many control parameters could potentially interact*, which we in this context interpret as: many performance indications during the project may interact. Clark & Wheelwright (1992) argue that three variables in particular interact during product development, namely *time, cost* and *quality*. Usually, trade-offs have to be made between these three. In the same way, other performance indications may interact. This could be motivation, working hours and reward or speed, weight and size. Therefore, it might be difficult to connect the dots during the project and foresee subsequent quality issues.

That *certain information about the state of the system must be obtained indirectly or inferred* is true for both the product and the project and is for example tied to the previous point; many control parameters could potentially interact. In order to understand the interaction of several control parameters, their consequences must potentially be inferred.

The last point is highly interesting for this project. That *there is only a limited understanding of some processes, particularly those involving transformations*, is particularly true for a product development process. For example, understanding the process of creativity, how experience is combined, and *transformed*, into new and novel ideas are something that researchers still are wrestling with (McElroy, 2000). In addition, when investigating how different stakeholders within a company viewed their product development process, Clark & Wheelwright (1992) concluded that the images varied significantly across the organization. Thus, we neither expect the project members nor ourselves to fully understand the dynamics of the product development process at Automatics. This is not only because the project were run some years ago, it is also due to the lack of understanding that still exists within this field.

Now, we will develop some theory regarding errors in the kind of system described above. This theory also builds upon the theories of Reason (1990) and will focus on the presence and characteristics of active and latent errors.

2.1.5 Active and latent errors

According to Reason (1990), and regarding his theories about error detection, there are two types of errors in an organization. He calls these active- and latent errors. Active errors are described as:

> *"... effects are felt almost immediately; in general, active errors are associated with the performance of the 'front-line' operators of a complex system (pilots, air traffic controllers, ships' officers)"* (Reason, 1990, p.173)

Further, latent errors are described as:

> *"... adverse consequences may lie dormant within the system for a long time, only becoming evident when they combine with other factors to breach the system's defenses; latent errors are most likely to be spawned by those whose activities are removed in both time and space from the direct control interface (designers, high-level decision makers, construction workers, managers)"* (Reason, 1990, p. 173)

Operators are frequently dealing with out-of-tolerance system states and frequently fail to recover from these. However, the root causes of these states are usually built in to the system long before the active error was committed. Thus, the more removed individuals are from the front-line activities and direct hazards, the greater is their potential danger to the system. Described earlier in the system-level approach, and consistent with Reason (1990), is that efforts to prevent repetition of a specific active error will only have limited impact on the system as a whole. This is because the same mixture of causes is unlikely to reoccur.

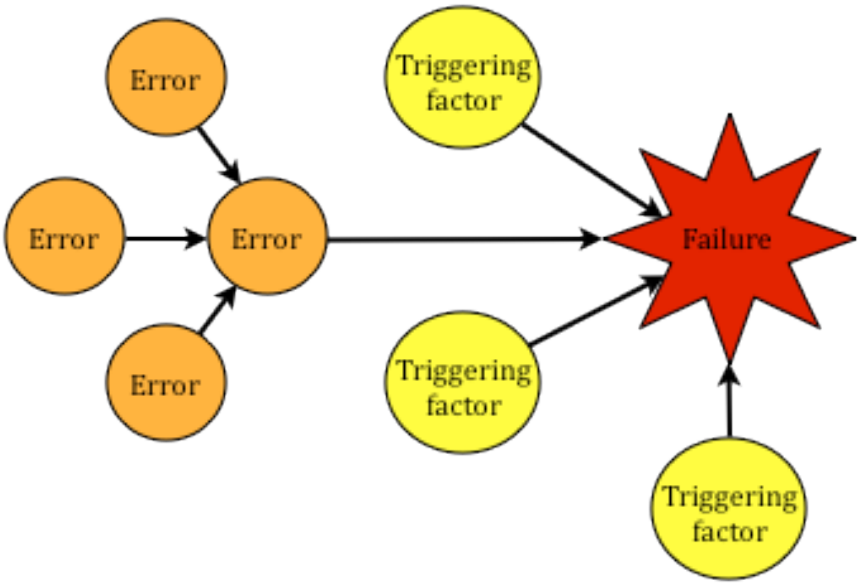


Figure 3. Errors, triggering-factors and failure

Thus, in this particular context, most of the errors built in to the product can be seen as latent errors and hence rather difficult to acknowledge. This is tied to the previously described characteristic of this kind of system, namely that *common-mode connections (i.e. components whose failure can have multiple effects 'downstream' are present*. Since all of the failure-trigging factors (or latent errors), in some cases, unlikely will be active simultaneously, some latent errors will never have any effect on the final product. In addition, as outlined earlier, *there is a limited possibility of isolating failed components*. Thus, several components might cause an error, and only a specific set of triggers might cause a subsequent failure, as depicted in figure 3. If one of these triggers not is active, the failure might never occur and the error might never be discovered. It should be added that these triggering factors not have to be a part of the product, but could just as likely be tied to the environment the product are going to operate within or to the procedures used while operating the product. Thus, one could argue that an infinite amount of errors always are designed in to a system. Unfortunately, this discussion leaves us with an, to our knowledge, unaddressed issue within this body of knowledge. First of all, what is to be considered erroneous in relation to a product is a bit unclear since several factors usually combine causing a quality issue. Presumably, we could assume that one component could be argued to be erroneous in relation to the others. Thus, we should rather address sets of triggering factors than simple errors. However, if that is the case, we have another issue considering the likelihood of the "right" mixture of triggering factors being unlikely to be active at the same time. It might be so that if they are unlikely to be active at the same time, this particular mixture is not to be considered erroneous. Then, we will have a problem concerning calculating probabilities and setting limits for probabilities. Thus, this is a problem to researchers because investigation into this area, as described here, would require technical knowledge within the field investigated.

Thus, when conducting the empirical study and the analysis we have had to rely on what people consider erroneous. The problem with this is that people might not be remembering correctly or for some other reason retell inaccurate stories. However, as explained above, we see no other option.

Above, in the definition of latent errors, Reason (1990) mentions defenses. He visualizes these defenses as layers that latent errors have to slip past in order to go unnoticed through the process. Like Reason (1990), we depict the defenses as layers through which latent errors have to pass before subsequently reaching the product and, in the worst case, customers. Reason (1990) exemplifies these defenses with personal safety equipment and guards on one end on a scale of sophistication to people and engineered automatic safety devices on the other end. Again, Reason (1990) focus on disasters in systems such as nuclear power plants, but we argue that this visualization creates value to our context as well.

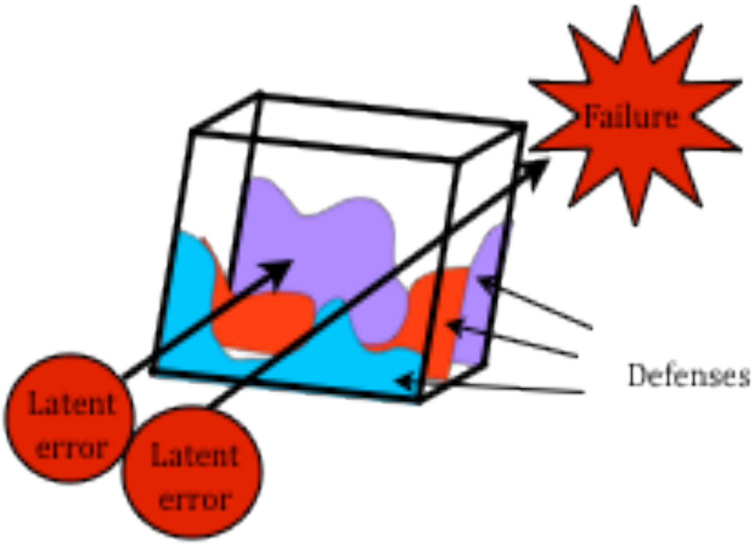


Figure 4. Defenses for latent errors (adapted from Reason, 1990)

As can be seen in the illustration above (figure 4), one latent error manages to escape through the defenses and one is caught by the last defense. As the latent error breaches a defense, it becomes evident to the organization.

The outset of this study was a set of defenses just as the ones described above. However, they are in our study represented in the form of some formal R&D activities, practiced by the organization, with the purpose of detecting and/or resolving quality issues. Thus, the documentation of these defenses have been studied, analyzed and subsequently followed up by semi-structured interviews. The defenses we have studied are:

- Failure mode and effect analysis (FMEA)
- Test reports
- Pro-production test reports
- Databases containing customer data such as warranty issues, quality issue reporting, etc.
- Project meeting protocols
- Gate meeting protocols
- Steering committee meeting protocols

2.1.6 Quality issue detection and/or resolution

Reason (1990, p. 192) states, *"The control of safe operations, like the control of production, is a continuous process"* and argues that prerequisites for safety control are:

1. *"A sensitive multichannel feedback system"*
2. *"The ability to respond rapidly and effectively to actual or anticipated changes in the safety realm"*

In short, these two prerequisites could be called *feedback* and *response*. Reason (1990) calls a system characterized by the above prerequisites a *safety information system*, which is depicted in figure 5 below.

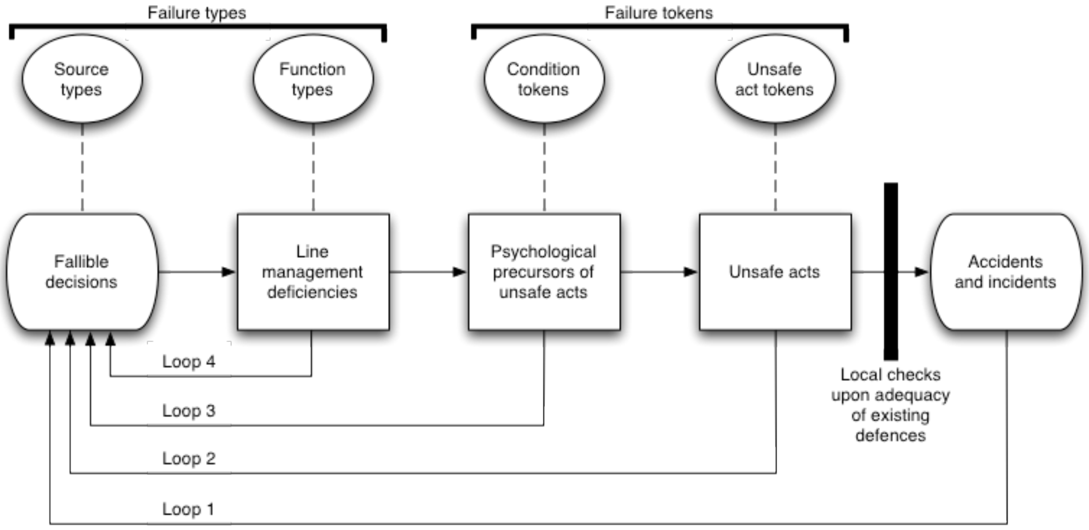


Figure 5. A safety information system (adapted from Reason, 1990)

Since Reason (1990) focuses on accidents in systems such as nuclear power plants and we are concerned with product development, we will not go into the details of this figure. However, it illustrates the point. Reason (1990) argues that there are three sources of latent errors, *fallible decisions*, *line management deficiencies* and *psychological precursors of unsafe acts*. Active errors are committed in relation to unsafe acts. In other words, active errors are committed in relation to operations.

The most fundamental feedback-loop in the system is loop 1, which in our case could constitute learning from customer data and loop 2 could potentially be mistakes committed in manufacturing. However, the aim for an organization should be to operationalize loop 3 and loop 4, which means anticipating errors in the system and changing the system instead of merely reacting to symptoms. Thus, the closer an organization moves towards an effective operationalization of feedback-loops 3 and 4, the more rapidly it will be able to respond to errors and the better it can avoid detecting errors late.

Moreover, in accordance with Reason (1990), we argue that the more interactive complexity the system is characterized by, the more the organization has to win by deliberately designing systems similar to the safety information system. In addition, by being characterized as a loosely coupled system, there is even more to win in these systems. The reason for this is that one part of the system not directly affects other parts of the system, which means that there is some slack and thus time to correct mistakes before they cause too much trouble.

Moreover, we argue that, what Reason (1990) communicates in his theory is that the more and the faster an organization manages to learn, the more errors will be detected and the faster these errors will be resolved. Hence, a *sensitive, multichannel feedback system* and *the ability to respond rapidly and effectively to actual or anticipated quality issues* are of prime importance. However, in order to advance this ability there are a few characteristics from the theory of organizational learning that are important as well.

2.2 Organizational learning

As was stated by Morgan & Liker (2006), product development is essentially defined by flows of information. If information sharing and communication is subtracted from product development, there will be no new products. Purser, Pasmore & Tenkasi (1992, p. 2) further build on this view and claim that product development essentially is

> *"a knowledge development and knowledge synthesizing activity consisting of a stream of routine and non-routine tasks, performed by an array of individuals and groups"*

It is recognized that competitive advantage is a result of organizational learning, i.e. how quickly and effectively individual members can develop, organize and utilize their knowledge base (Meyers & Wilemon, 1989; Purser, Pasmore & Tenkasi, 1992). Moreover, product development teams rely upon learning systems for detecting and correcting errors. An inability to detect errors early in the product development process can have significant impact on time, cost and quality of the project. Similarly, problems that take too long time to fix can significantly impact the product development effort.

> *"Hence, the ability to detect and correct errors in a timely manner is dependent on effective new product development team learning systems"* (Purser, Pasmore & Tenkasi, 1992, p. 4)

This conclusion is also supported by Meyers & Wilemon (1989), who state that learning within teams take place through error detection and correction. Kim (1993, p. 38) states that learning can be thought of as *"increasing one's capacity to take effective action"*. Thus, the ability to detect and correct errors in a timely fashion is dependent on effective new product development team learning systems. In addition, effective learning systems build up the knowledge base, which in the future allow teams to detect and correct errors more effectively.

Kim (1993) argues that all organizations learn, but that only some organizations deliberately advance their capabilities. Other organizations do not take focused action to learn and therefore acquire habits that are counterproductive. We do not fully agree with Kim (1993) because even if an organization takes deliberate action to learn certain capabilities, there will always be some measure of uncertainty about what capabilities that will be fruitful in the future. However, we do argue that taking deliberate measures to learn will enhance the organizations capability to learn fruitful capabilities because a systematic way of learning will make the organization able to systematically monitor and thereby improve its learning systems. This is consistent with two of the cornerstones of *Total Quality Management* (TQM) presented in Bergman & Klefsjö (2003), *base decisions on facts* and *improve continuously*. The underlying meaning of this is that, if you have a systematic way of learning, you will be able to relate performance to the system and thereby improve systematically.

TQM brings us into an interesting comparison, namely the one between quality management in production facilities and quality management in product development. When it comes to production there are several developed methodologies and philosophies for continuous improvement. For example, philosophies like TQM, Six Sigma and the *Plan-Do-Check-Act* – cycle (PDCA-cycle) and methodologies like statistical process control (SPC), 5 whys and Ishikawa diagrams (Bergman & Klefsjö, 2003). However, when it comes to product development, the standardized improvement tools do not seem to be equally developed. The reasons for this could be many. For example, it might be so that, since product development can be characterized as a loosely coupled system with some slack between components, there are more ways of doing one thing than there is in production. In addition, we have learned that there often is a limited understanding of a system characterized by interactive complexity. More ways of doing the same thing and a limited understanding naturally slows down development of improvement since recognizing the better way becomes more difficult. In addition, these factors probably make it more difficult to employ statistical tools in improvement work as well.

Moreover, it could be so that standardization according to the PDCA-cycle not is optimal within product development. According to Nonaka (1994), disruptions of the normal state increase the production of knowledge. E.g. the detection of a quality issue might very well lead to a person discovering a novel opportunity leading to an improvement of some characteristic of the product distant from the direct quality issue. Thus, since the picture of success factors in product development is a bit fuzzy, developing methodologies for improvement is probably a bit difficult. In other words, it is not clear whether a highly systematic or a loosely structured way of developing products is to prefer. However, in order to beat competition in the future, we argue, in accordance with the introductory discussion of this part, that learning within product development is paramount.

Organizational learning cannot be equated with individual learning. Organizational learning is much more complex. The concept of organizational memory is of primary importance in organizational learning. According to Kim (1993), the organizational memory contains everything in an organization that is retrievable, e.g. copies of letters, spread sheet data stored in computers, strategic plans, and what is in the minds of all organizational members. However, according to Kim (1993, p. 44), what is important in the context of organizational learning are those parts of organizational memory that

> "*constitute active memory – those that define what an organization pays attention to, how it chooses to remember its experience – that is, individual and shared mental models*".

It is important to understand that these mental models can be both explicit and tacit, that is, both outspoken and represented mostly by inner feelings and preconceptions. Thus, according to Kim (1993) organizations are dependent on individuals improving their mental models and making them explicit (in e.g. writing or talking) in order to be able to share them with co-workers and thereby develop new, shared mental models. In addition, this process is what makes the organization independent of any single individual. Moreover, we argue that this active memory is of primary importance to detecting quality issues in a timely manner. In other words, in order to detect quality issues in the future, an organization needs to incorporate events the events around past quality issues into its active memory.

In order to relate organizational mental models to standardized routines, Kim (1993) mention *Standard Operating Procedures* (SOPs). SOPs can be both good and bad for the organization. On the one hand, SOPs are an important repository of organizational memory, and on the other hand SOPs can inhibit the organization to search for new routines. However, this is an on-going discussion in literature and some argue that SOPs do not inhibit the search for new routines unless they become too strongly rooted in the organization. Strong in this sense means applied in a wide range of situations.

After this theoretical discussion we can conclude three things that are of primary importance for detecting and correcting quality issues. Firstly, a *sensitive, multichannel feedback system* that provides project members and managers with information about quality issues. Secondly, *the ability to respond rapidly and effectively to actual or anticipated quality issues*, which avoids issues in a timely manner, reducing the possibility of quality issues spreading through the system affecting other parts. Thirdly, *a systematic way of storing and evaluating learning in the organization*, so that the way quality issues are handled within the organization is improved over time and so that this learning is remembered.

3 Methodology

This section will elaborate on three concepts considering design of empirical research: (1) *research strategy*, (2) *research design* and (3) *research methods*. These concepts are neither equivalent nor mutually exclusive; choices in each will affect the others. The choice of what research strategy, design and methods utilized will to a great extent influence the collection and analysis of data, hence it will affect the final results of the study (Bryman & Bell, 2007).

3.1 Research strategy

When it comes to research strategy, Bryman & Bell (2007) recognize that there are three major factors that affect the choice of a particular research strategy. The three factors influencing the choice of research strategy are: (1) *the link between theory and research*, (2) *epistemological considerations*, and (3) *ontological considerations*.

The purpose of this thesis and the research questions developed to fulfill that purpose implies an inductive stance to theory and research, meaning that theory will emerge from the study rather than being tested by it. Since the causes – for why quality issues are not detected and resolved early in the product development process – were unknown, and our aim was to provide suggestions on how this could be done, we argue that an inductive stance was more appropriate.

Further, our role in this project can be compared to what Schein (1999) calls academic research consultants, since we are to provide suggestions to a particular problem, namely the late detection and/or resolution of quality issues. Schein (1999) claims that a fundamental criterion for being successful with implementing such suggestions is that the client organization must own the solution, since it is the client that has to live with that solution. This claim, we argue, justifies the view that we cannot see Automatics reality from an objective point-of-view; rather, we have to view it from the perspective of the employees, which is in line with a qualitative research strategy (Bryman & Bell, 2007). Further, taking an objective stance means coming up with suggestions spanning over the whole project. Different departments, and even different project members within each department, have different preferences regarding how to work. This is natural since different departments usually have different assignments and therefore different interests. Since some proposals for improvement will have some elements of constraint on employee behavior, this phenomenon will impact the success of our improvement suggestions. Therefore, we argue that taking constructionism ontological position will benefit the outcome of this study mostly. This position challenges the view that social phenomena and their meanings exist independently of the social actors (Bryman & Bell, 2007). Thus, during the study, we have taken different group's interests into consideration.

An interesting point regarding the ontological position is that although the data generated have been from the individuals' point-of-view, the purpose was to investigate the conditions for detecting and/or resolving quality issues early and give suggestions on how these conditions can be improved for an earlier detection and/or resolution. However, these suggestions will probably apply to all employees involved in product development. Thus, the suggestions must be generalized to fit the majority of the employees, which maybe implies an objective ontological position. This stresses an important point to bear in mind: the epistemological and ontological positions taken were not meant to dictate the entire research process and exclude other positions; they were rather seen as more or less dominant in this study and hence served to guide the researchers.

The view of theory as inductive together with the interpretivism and constructionism position would, according to Bryman & Bell (2007), be in line with the qualitative research strategy.

3.2 Research design

The research design of our study has naturally taken the form of a case study, since we were studying a specific development project at Automatics. The case study design is commonly used to study problems in the organization's specific context (Bryman & Bell, 2007). The utilization of a case study design will have implications for the research methods utilized. Case study researchers tend to collect data by means of participant observation and unstructured interviews, since these methods of data collection are viewed as helpful in the generation of an intensive detailed examination of a case (Bryman & Bell, 2007). Since we have studied a product development project that already had lasted for over 2 years when we started, participating observations were not possible to conduct. Hence, unstructured interviews have been the main tool for data collection.

Due to the intensive examination of a particular case, this type of research design has one main advantage: it has proven to be particularly strong in eliciting the unique features of the area under investigation. However, this also leads to the main disadvantage of case studies: the investigation of one case automatically eliminates the possibility to generalize the results. Since the data and subsequent analysis only reflects the specific context of the organization under investigation, it is not possible to transfer results to other organizations in other contexts. For our study, we were neither interested nor obliged to generalize our results. Since our purpose was to provide a set of suggestions to Automatics, the case study design was preferred over other research designs.

3.3 Research methods

The research methods that we have used to answer our research questions were analysis of Automatics' documentation of the Panther development project and semi-structured interviews.

The documentation available for the Panther development project ranged from the pre-project market study (project plan, risk assessment, intended performance, budget, etc.) – through design, testing, verification and production – to customer complaints received from the field. Automatics gave us access to the project folder, which included all types of documentation during the project. We have in this thesis read and analyzed various documents, partly to gain knowledge about the project and partly to validate what has been said during the interviews. Some of the documents have however served as our main source of information for finding quality issues. These are:

- *Failure Mode and Effect Analysis (FMEA) documents*
- *Test reports*
- *Pro-production test reports*
- *Databases containing customer data such as warranty issues, quality issue reporting, etc.*

- *Project meeting protocols*
- *Gate meeting protocols*
- *Steering committee meeting protocols*

FMEA is a tool commonly used for managing quality issues in product development. How the tool is used can vary from company to company, but in its basic form, potential quality issues are listed and assigned points with regard to the probability that they will occur and how severe the effect is upon occurrence. One of the difficulties of using the FMEA work at Automatics for our purpose has been that Automatics utilizes FMEA in a way that makes quality issues untraceable. The FMEA documents are modified and quality issues are added/removed continuously. Therefore, it was impossible to extract specific time-points for when a quality issue was anticipated or discovered. In those cases where we have faced this problem, we have relied on the memory of our interviewees. For a more detailed explanation of FMEA, we refer the reader to Slack, Chambers & Johnston (2007). Project meeting documents are the power-point slides presented at each weekly project meeting, where the project manager meets with the partial project managers from the different engineering disciplines and functions to discuss issues such as project status, quality issues, important coming activities, etc.

The test reports are reports written by the test-engineering department after they have completed a series of tests on the product and its components. For example, the test reports provide information about the assembly, testing, disassembly and evaluation of the prototypes, and the quality issues discovered. As was the case with FMEA, one potential problem with these documents is that it is difficult to ascertain the exact point in time when a quality issues was detected. The only point of time documented is the date when the reports were finished and approved. Since testing spans over practically the entire project (~ 100 weeks), it has been difficult to attribute all detections with a date. For those quality issues where we have encountered this problem, we have relied on the memory of our interviewees. After the production of each product, some tests are conducted in the production facilities, where quality issues may be detected. These tests result in documents which we have called post-production test reports. Databases containing customer data such as warranty issues, quality issue reporting, etc., are basically data coming in from customer usage. Finally, project meeting-, gate meeting- and steering committee meeting protocols are documents stating project status, issues, decisions, important activities, etc., that were discussed in the three different forums: project meetings, gate meetings and steering committee meetings.

The data inherent in these documents and databases served as a basis for answering our first two research questions:

1. When were quality issues from the Panther development project detected?
2. When were quality issues from the Panther development project resolved?

The facts that we have used to answer our first two questions were then, together with our theoretical framework, used as a foundation for the semi-structured interviews.

The semi-structured interview lies on a continuum between the anonymous, neutral quantitative approach and the personal, emotional therapist interview. It is useful for eliciting rich data on issues that are rather ambiguous (Kvalé, 1997). The number of interviews conducted will depend on both issues of theoretical saturation and time. Theoretical saturation is a feature of grounded theory, which we describe more thoroughly below, meaning that data is collected until

> *"(a) no new or relevant data seem to be emerging regarding a category, (b) the category is well developed in terms of its properties and dimensions demonstrating variation, and (c) the relationships among categories are well established and validated"* (Bryman & Bell, 2007, p. 460).

A category is a concept emerging from the data that can be regarded to represent real-world phenomena (Bryman & Bell, 2007). However, as also stated by Bryman & Bell (2007), it is difficult to define exactly when theoretical saturation is achieved. Research in general is restricted by time and budget constraints, which can further make it difficult to reach theoretical saturation. This was also the case for us, and thus we followed Bryman & Bell's (2007) advice to continue the data collection until we felt satisfied with the data collected and had enough time left for the subsequent analysis.

For the qualitative part of the study, we have used the seven steps proposed by Kvalé (1997): *constructing the themes*, *planning*, *the interview*, *print-out*, *analysis*, *verification* and *report.* The rest of this section is devoted to describing how we handled the specific steps.

Constructing the themes for the interview guide includes formulation of the purpose of the investigation and a description of the area of investigation. The questions *why* and *what* should be cleared out before the question *how* – the method – is asked (Kvalé, 1997). *What* we have done is stated in our purpose:

> *To investigate the conditions for detecting and/or resolving quality issues early within the product development process of Automatics and give suggestions on how these conditions can be improved for an earlier detection and/or resolution.*

As recommended by Kvalé (1997), we *planned* the study before conducting the interviews. However, we did not have a standardized interview guide, but rather altered questions as the study progressed. Since case studies are very context specific, the answers from one interview may affect the questions asked in subsequent interviews (Bryman & Bell, 2007). In this sense, answers from some interviews served to plan for the subsequent interviews.

> *"The interview is a scene on which knowledge is built through interaction between the interviewer and the interviewee"* (Kvalé, 1997, p. 120)

The interviews were fairly structured in the sense that we used an interview guide. In total, we conducted 18 formal interviews, each lasting approximately for one hour. The interviewees represented different functions and departments that had been involved in the Panther development project, such as mechanical engineering, purchasing, production, marketing, etc.

Regarding the *printout*, every interview has be recorded and later transcribed exactly according to the audiotape in order to correctly retell the stories told by the interviewees (Wallén, 1996). For the sake of avoiding misinterpretation, we took advantage of the human memory and transcribed the interviews as fast as possible after finishing them (within 1-2 days).

To *analyze* our qualitative data, we have utilized the concept of grounded theory, which is widely used. In essence, grounded theory posits two central features. First, it has an inductive stance when it comes to the relation between theory and research, meaning that theory emerges from the data. Second, it is an iterative process where the researchers constantly collect and analyze data simultaneously, with constant referencing between theory and data (Bryman & Bell, 2007). This iterative feature of grounded theory means that a literature review cannot be viewed as an isolated event in the beginning of the study, but literature needs to be reviewed throughout the research process. By constantly moving between data and theory, we sought to reveal the root causes for late quality issue detection and/or resolution at Automatics. Hence, the qualitative data has worked as the main source of information for answering our third research question; why are quality issues not detected and/or resolved early in the product development process?

Regarding the process of *verification*, Kvalé (1997) claims that there is no absolute way to validate qualitative research. Thus, we have followed his advice and rather tried to dismiss possible sources of failure. The method of having *several interpreters* (Kvalé, 1997) has been used, and the analysis has been conducted with both interviewers present in order to avoid misinterpretations. When disagreements of the interpretations have occurred we have debated the issue until an agreement has been reached.

To some extent, we have relied on our interviews to answer our third research questions, namely if quality issues could have been detected and or resolved earlier. The answers given to this question have in turn been crosschecked against other interviewees' answers, and also against the formal documents. This was done in order to triangulate responses in cases where this was possible. The answer to our final research question, together with the theoretical framework, has aided to answer our final research question and provide suggestions for how quality issues can be detected and/or resolved earlier in the product development process of Automatics.

4 Empirical study

We will start this section by describing the Panther development project and the resulting product, which has been central to our study of quality issues. The stories are built upon the information gained from interviews and the analysis of formal documents that we have conducted. After we have presented the basics of the project and the resulting product, we will continue this section by presenting our empirical findings for the four quality issues we have encountered.

Finding dates for when quality issues came up for the first time have been a bit difficult. However, we have tried to calculate dates from the documentation we have had access to. Due to confidentiality we will though not present how these dates were calculated. Moreover, the exact dates are not that important for the results of this thesis. It is though more interesting that we had a hard time calculating them.

4.1 The Panther development project

In short, two of Automatics' product platforms had been outdated and two new were going to be developed: Panther and Lion. As we have stated earlier, we have not investigated anything directly related to Lion.

The aim of the Panther development project was to replace a previous model, but also to significantly reduce the weight and cost of the product. This was regarded necessary to gain market share in high-volume segments, where competitors have more market share. The predecessor of Panther was the company cash cow and, when it comes to some performance characteristics, considered by the customers as the best in its class. In general, the products of Automatics are regarded as very good in terms of performance. They have therefore been able to take positions on the market where their competitors cannot compete with them in terms of performance. However, this world-class performance was to some extent the result of the product robustness. Panther, on the other hand, weighs less than 50 % of its predecessor, and is hence not as robust. This loss of performance was going to lead to some loss of market share in particular niche segments of the market (~10 %), while gaining market share in other, high-volume, segments (~80 %). The project was planned to run for approximately 100 weeks, but due to problems with one component, market launch was delayed by roughly 30 weeks. In fact, because of these problems, the project is still not terminated as of today. The problems with this component are described more thoroughly in a later part of this section (4.3, *Quality issues*).

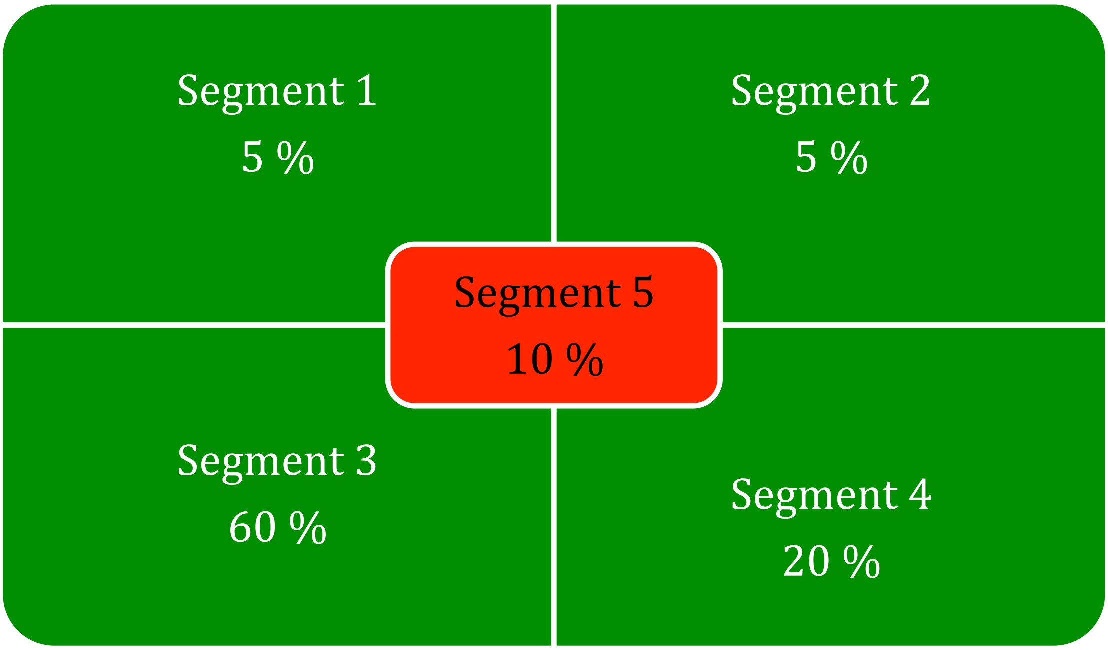


Figure 6. Market segments for Panther (percentage of total volume)

The market for Automatics' products can roughly be divided into five segments (see figure 6). In the project description, it is stated that four of these segments (green) were included in the Panther development projects, whereas the fifth segment (red) was not included due to the fact that achieving the required performance level for that segment with such a lightweight product would not be possible. However, it was also stated early in the project that the low volume/high performance segments that were not included in the Panther project would not go lost. Rather, Panther's predecessor would remain as an alternative for those segments until a variant of Panther was developed which could cope with the demanding performance level.

One important point that should be made is that the performance of Panther is not solely depending on the weight. Through the utilization of advanced optimization, some performance characteristics of the product can be improved. Thus, depending on how well the product can be optimized, the product may reach higher levels of performance. To determine the performance of the product, tests were conducted where Panther was compared to its predecessor on some performance characteristics. This has been a topic coming to the surface during the interviews we have conducted, i.e. exactly what were the expectations on the performance of Panther, as compared to the product it was going to replace. As put by one of our interviewees:

> *"… We had got relatively good performance in many tests. So it was marketed very strongly then, which probably also was correct, as a fantastic product; cheaper, lighter… and with better performance. And it may have been a slight exaggeration, but in most cases we had the same performance* [as its predecessor], *sometimes better, sometimes worse. And in some way I feel that there was a mismatch in the images of what Panther should accomplish. People thought they had gotten a superior product, but were not aware of what was said in the beginning; that you must have an alternative to keep the most demanding segments* [the 10 % market in the figure]. *Thus, in pure joy over some preliminary test performance results, the product was marketed as the best ever made, which perhaps was not true in all aspects*"

The general impressions of some interviewees were that the performance tests conducted to assess Panther's performance in relation to its predecessor were conducted on few prototypes that were not fully representative of the finalized product. The concerns spoken to us about this matter regard:

1. The number of units that the tests were conducted on, and
2. The representativeness of the test product to the final product

The first point above is basically a matter of how much statistical certainty the tests provided. The second point on the other hand has to do with the fact that several changes were made to the product after these tests, such as changes to Component A; a component that affects product performance. Our review of Automatics' performance test reports show that due to time constraints, testing on more than one unit of each model was not feasible. Furthermore, the tests were conducted with a substance for Component A that was changed for serial production. The details of all changes made to Component A are presented in another part of this section (4.3, *Quality issues*).

Other interviews we have conducted indicate that the plan all along has been to keep Panther's predecessor until a variant of Panther is developed that can satisfy more demanding customers (in terms of performance). Our review of Automatics' documents also show that the phase-in/phase-out plan, presented to the steering committee approximately 100 weeks after project start, specifies that Panther's predecessor should be kept until a new variant of Panther is developed. It should be noted that the image of what Panther could and could not accomplish in terms of performance is however not purely determined by performance tests. Rather, iterations between market demands and technical solutions result in a product specification that aims to satisfy the needs of the market. This has important implications, namely; marketing representatives are dependent on clear communication from the project regarding what can and can't be done, whereas the project is dependent on marketing representatives to communicate what is desirable and not.

For the Panther project, the product and project manager conducted the market pre-study by visiting a number of customers and gathering information on what product the customers were requesting from Automatics. This comprehensive pre-study later resulted in a specification of the market requirements, which serves as a foundation for the technical specification. Through iterations with the project, the market requirements are transformed into a technical specification of the product. One interviewee described this transformation process:

> "*First it is the market specification, and it is pretty rough… where we specify 'these are the segments, these are the requirements on performance, etc.' Then when you want the product specification… there is no magic formula for that… there is an iteration between market requirements and the project* [team]. *The project specifies what is feasible and we revise the market specification, and so on… the technical specification is later signed off by the different functions and technical departments* [the functional organization]… *We were not told that this couldn't be done. The project said that they could compensate the loss of performance through optimization and signed off on the specification*"

An interesting point to bear in mind is that three different people served as project managers in the Panther development project. The first project manager was active during the market pre-study and left the project when the project officially started. The second project manager almost led the entire development effort, from project start to commercialization. However, due to technical problems with Component A, the project was not terminated after commercialization. Hence, a third project manager led the project after it had been released to the market.

What comes out from the data we have collected is that there exist different views on what the expectations regarding Panther's performance were, and whether or not it was possible to reach those expectations. Based on the interviews we have conducted, we recognize two parties with different views. On one hand, there is a view that the expectations were set too high for such a light product. On the other, there is a view that a promise has been made to reach those expectations. In more general terms, there seems to be confusion in the organization regarding what type of producer Automatics wants to be, the low-cost producer who does exactly what the competitors do or the high-performance producer who beats competitors on performance. Consider the following statements made from various interviewees:

> "*The guidelines* [for what kind of producer Automatics want to be] *are not clear... Some consider that performance is core, and we should be the best at it while other think we should produce 'good enough' products and just lower our costs ... We must become more clear on where we want to be in 5-10 years*"
>
> Interviewee 1

> "*Do we want to be a low-cost producer who is 'good enough' and can do the same things as our competitors, or do we want to continue being Automatics, who produces products that can also satisfy the most demanding segments* [in terms of performance]? *... I don't really think that people agree in the organization, on what we really want to be*"
>
> Interviewee 2

> "*We need a more clear strategy that is anchored in all parts* [of the organization] ... *Individual projects are left to take strategic decisions, which should not be the case*"
>
> Interviewee 3

These statements show that the differing images are not isolated to Panther. Rather, there is a general confusion in the organization regarding what type of producer Automatics should be.

4.2 Project organization

Since Panther was considered a large project in the organization, including over 30 people, an organizational model with a project manager and partial project managers was utilized. The partial project managers represent the various engineering disciplines from R&D and other functions, such as production, marketing, supply, etc. The number of people, and the degree to which they were directly involved in the project varied over the lifetime of the project. A graphical representation of the project organization is provided in figure 7.

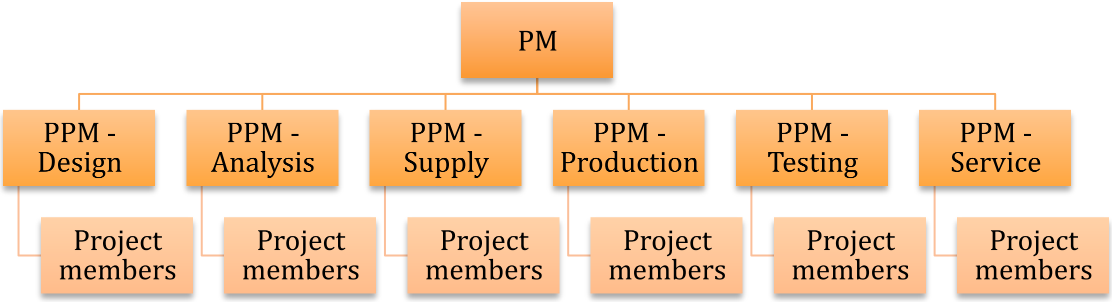


Figure 7. Panther project organization

In addition to this project group is also the product manager, who is the person who orders the product from R&D and is responsible for the product during its life cycle. This person thus belongs to the marketing department of Automatics. The product manager also funds the project during its course of action, and is a member of the steering committee, which will be presented later.

The project team is not fully co-located but the majority of the functions are located in the same industrial complex. Various engineering disciplines are however co-located and close to the project manager.

Also present in the Panther development project organization was the steering committee, a cross-functional constellation of managers that review and control the project. Generally, the functions represented in the project are also represented in the steering committee. Meetings between the committee and the project manager are kept continuously during the entire project, where important issues are discussed, such as project status, time-plan, resources, critical activities, etc.

4.3 Quality issues

In this section we will present results regarding the quality issues we have investigated in this master thesis. These results come from data collection from sources previously mentioned. These are interviews with project members and data from project meeting protocols, steering committee meeting protocols, gate meeting protocols, test data, post-production test data, customer usage data and other relevant documents. The quality issues are:

- QI_1 Discolored Substance, Component A
- QI_2 Exceeding Critical Parameter, Component A
- QI_3 Customer Representative Testing
- QI_4 Malfunctioning Component, Component B

The results from the empirical investigation will first be presented according to our research questions:

- When were quality issues from the Panther development project detected?
- When were quality issues from the Panther development project resolved?
- Could quality issues from the Panther development project have been discovered and/or resolved earlier?

Subsequently we will present our analysis of these quality issues, followed by recommendations. However, before entering into the presentation of our quality issues, we will describe a critical component of Panther, which has been the source of many quality issues.

Component A

For the type of products that Automatics develop, one component is of significant importance when it comes to cost and performance. Moreover, some critical product development activities, such as product optimization, are dependent on having a stable and functioning model of this component. We will in the rest of this report refer to this component as Component A. Many of the quality issues we have discovered during this project, which also are the most severe ones, are related to this component. In particular, the quality issues related to Component A have caused several delays to the project.

In Panther, there are 6 types of Component A. The types of Component A are located at different places in the product, and do not need to be identical, but can be of different brands and models. Further, the 4th-6th types are technologically different from the 1st-3rd, and are developed in-house. These types of Component A (4th-6th) have not been the source of any major quality issues, and our interest lies mainly in the first three types. Table 1 describes the offerings from various suppliers for the different types, where we have excluded types 4-6. Since the component is critical in the product and bought externally from suppliers, extensive testing and verification of the different types and models are necessary before the product can be released to the market. These tests take considerably long time to conduct (~30-50 weeks) and any changes to the component that require tests to restart risk delaying the project.

Type	**Supplier**	**Model**
1	A_2	A_2 200
2	A_1	A_1 100
3	A_1	A_1 101

Table 1. Component A offerings from suppliers

The market situation for Component A was rather special prior to the development of Panther. There were few suppliers on the market, where one, supplier A_1, dominated the market in terms of sold units. Thus, Automatics was in a position with few choices and had recently experienced quality issues with supplier A_1.

In the past, Automatics had not been testing Component A extensively prior to development projects. The lessons learned from various quality issues over time with Component A had thus created an awareness in the organization about the risk of not having tested and verified component A prior to development projects. Experiencing these quality issues, and being uncomfortable with the sourcing situation, Automatics wanted to introduce a new supplier, and had therefore prior to the Panther development project started to test Component A of various brands and models. The test results showed that one European supplier, Supplier A_2, did well and it was decided to bring in Supplier A_2 for the 1st type of Component A in Panther. Moreover, Automatics knew that one of its competitors had been sourcing Component A from Supplier A_2 for a couple of years. Thus, Supplier A_2 was new to Automatics, but not to the industry. The project started with types of component A according to table 1 above.

As the Panther project started and proceeded, more problems were experienced with Supplier A_1's products, both in Panther and in other products. For this reason, a plan B was developed 50 weeks after project start, where it was decided to start testing another model of Supplier A_2's products, as a possible replacement for Supplier A_1 on the 3rd type. Table 2 summarizes the various suppliers and the number of units they had in Panther over time. The parentheses represent the decision to start evaluating another type of Component A from Supplier A_2 as a replacement for Supplier A_1. Table 3 describes the new constellation of models and suppliers of Component A, after the decision was made to replace the 3rd type of Component A from Supplier A_1 to supplier A_2. The bold font represents the type of Component A changed when executing plan B.

Weeks after project start	Total units of Component A in Panther	Unit(s) of Supplier A_1's products in Panther	Unit(s) of Supplier A_2's products in Panther
0	6	2	1
~50	6	2 (-1)	1 (+1)
~80	6	1	2

Table 2. The evolution of Component A over time

Type	Supplier	Model
1	A_2	A_2 200
2	A_1	A_1 100
3	A_2	A_2 201

Table 3. New Component A offerings from suppliers

Although this was the biggest change when it comes to component A, several other changes were made as well. Firstly, some confidence in Supplier A_2 had been gained due to the fact that they had been supplying a competitor with Component A for a couple of years. However, the interviews we conducted revealed that the model of Component A that Automatics was using differed from the model that their competitor was using. For Automatics, Supplier A_2 had developed a new generation of Component A, where some improvement had been made. As put by one of our interviewees:

> "*One thing that we are experts on here at Automatics is that if a supplier has a number of Component As to offer, we have a tendency to say 'ah, we'll take that one, but we would actually like to change that* [a part of the component]*'. We always make small changes, so we didn't use the components* [Component A] *that they* [Supplier A_2] *had experience of. We eventually used a modified, Automatics variant, of this component*"

Secondly, quality problems with the 2nd type, delivered by Supplier A_1, eventually led to a change in that model. The A_1 100 model was then upgraded to A_1 100+. Thirdly, the introduction of A_2 201 as a replacement for A_1 101 on the 3rd type did not function as expected. This resulted in a modification also for the 3rd type. Finally, the functionality of component A is highly dependent on a specific substance. Changes in this substance lead to significant changes in the quality of the component. As stated by one of our interviewees:

> "*What we have learned now about these components* [Component A]*... you have to find the right substance for them... This is something we have learned with both Supplier A_1 and Supplier A_2. If you have the right substance, you can get them*

> [Component A] *to work. But if you don't have the right substance, you can get various kinds of problem. They are incredibly sensitive*"

The perfect combination of substance and component was not present prior to project start, and three different substances have been evaluated during the Panther project, each change requiring new tests. In essence, all these changes required a restart of the tests that had been made on the components. Figure 8 visualizes the major changes made on the different types of component A during the lifetime of the project.

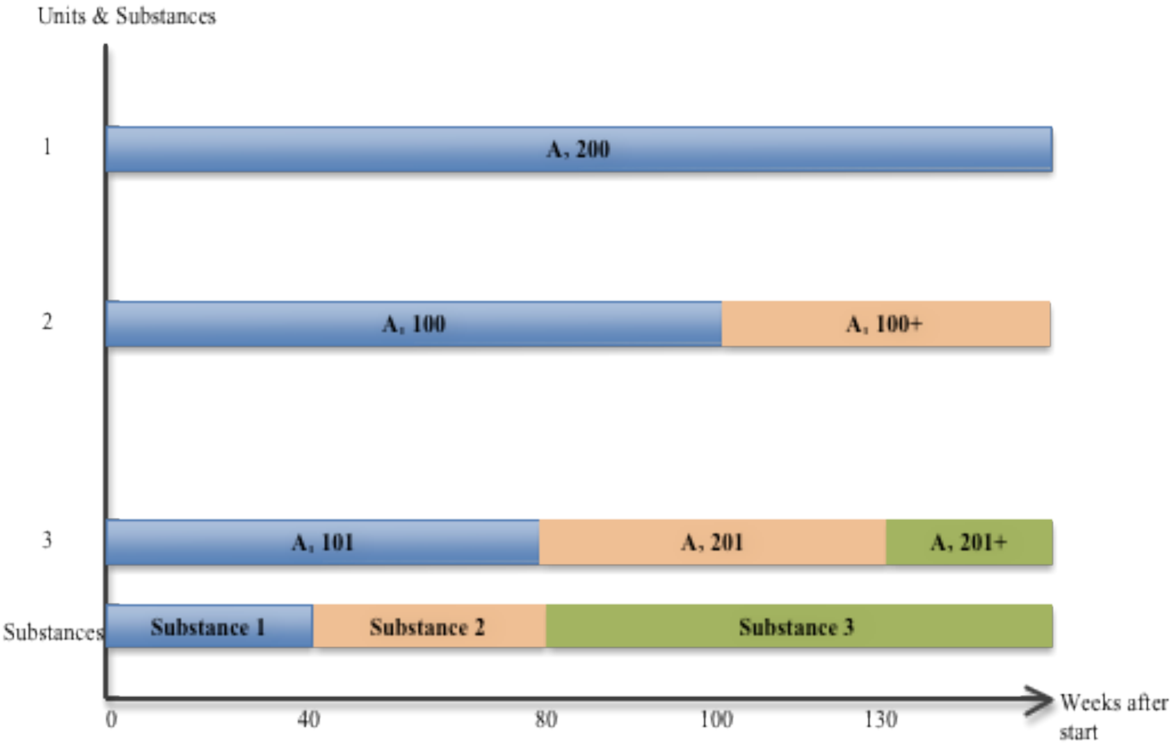


Figure 8. Changes made to Component A over time

4.3.1 QI_1 Discolored Substance, Component A

As we saw in figure 9 above, some changes were made to the substance used in Component A during the lifetime of the project. This substance is vital for the performance of Component A, and when it is discolored, it may be an indication that there exist quality problems with Component A. As can be seen in table 4, this quality issue regards Component A, type 3.

Source of information	Detection (~ weeks after project start)	Comment
Test report (type 3)	121	Test of prototype
Test report (type 3)	131	Test of component
Customer usage data (type 3)	160	The detection was a result of field service activities
Customer usage data (type 3)	160	The detection was a result of field service activities
Interviews (type 3)	80-140	

Table 4. Discolored Substance, Component A

When were quality issues from the Panther development project detected?

First of all, some of the possible causes for the discolored substance, mentioned to us by interviewees are noted in the FMEA. However, the FMEA does not show when entries into the document were made. Hence, we cannot draw any conclusion regarding when Automatics noted this quality issue for the first time based on the FMEA. It should though be noted that the actor making these FMEAs is the supplier of respective components A, and not Automatics.

From the information we have been able to elicit from interviews and other types of data collection it is not entirely clear when this quality issue was detected for the first time. Four interviewees confirm that the detection was made in production. In addition, one interviewee state that the detection in production was made in relation to a repair conducted due to some other matter. The same interviewee says that this detection was made sometime within an interval between 80 and 140 weeks after project start.

Another interviewee has said that it was detected after an adjustment made to the component, to which the quality issue is connected. This adjustment was decided on 130 weeks after project start, but it is unclear when it was implemented. However, the same interviewee also states that the detection was potentially made on an earlier occasion, but on the product Lion, which shares this particular component with Panther. However, it is unclear when this potential detection was made on Lion.

Moreover, another interviewee states that this detection later on was connected to a later detection (QI_2), which led to a change in the substance (substance 2 to substance 3) mentioned earlier. According to our calculations, testing of the component with this changed substance (substance 3) was commenced 131 weeks after project start.

Hence, the detection of this quality issue, in production, should have been made earlier than 131 weeks after project start. One of the interviewees that confirmed that it was first seen in production says that, as it also was detected in testing, the issue was investigated more thoroughly.

As can be seen in table 4 the test reports indicate that the detection was made around 121 weeks after project start. Thus, if it is so that this quality issue was detected in production before it was detected in testing, it should have been detected earlier than 121 weeks after project start. Moreover, the same kind of quality issue surfaced again in a subsequent set of tests. This is the same test referred to above that was commenced 131 weeks after project start.

In addition, as we can be seen in the table, the quality issue also hit two of Automatics customers approximately 160 weeks after project start.

Thus, what we can conclude from our findings is that it probably was detected in production before it was detected in testing. However, this detection must have been made in relation to some prototype or pre-series since gate 5 was passed and production commenced in week 132. Moreover, we can conclude that it was detected in testing twice in relation to two different substances (substance 2 and substance 3) and that two customers detected it.

When were quality issues from the Panther development project resolved?

As we can see in table 4, this quality issue also hit the customer and was thus not resolved during the product development process we are studying.

Could quality issues from the Panther development project have been detected and/or resolved earlier?

Three interviewees state that the experience with or knowledge about this quality issue not existed within the organization. One interviewee said:

> "*The guys at mechanics could not find anything specific and you also see this on other products. Now I think that we would have acted on it faster. However, I can confirm that this is a phenomenon that we haven't really figured out yet. There was nothing scientific that could confirm that this quality issue was a real problem.*"

Another interviewee says:

> *"The problem is that we don't know if it is a problem… … in other products we also see this indication, but there it is no problem... … I suppose that is why we not have focused on it."*

Five interviewees state that it took some time between the detection in production and the point when this was regarded as a "real" quality issue. However, the stories differ a bit when it comes to the reasons for why this quality issue became regarded as a "real" quality issue.

Two interviewees say that this quality issue was regarded as more serious when the organization saw a relationship between it and another quality issue (QI_2, type 3). How this other quality issue was detected is though also a bit unclear. One interviewee believes that it was a coincidence in pro-production testing. The other interviewee does not mention how it was detected.

One interviewee states that, as the two quality issues (QI_1 and QI_2) were connected, analysis was made of the substance that led to the decision of changing the substance (change to substance 3).

The third interviewee recalls that the point when the project began looking upon the quality issue as more serious was when it also was detected in testing and analyzed more thoroughly.

One interviewee commented on the connection between the two quality issues:

“This kind of quality issue does not have such a high caliber. You only see signs, but there is no evidence. Then you can’t be really sure. No one knew at the time that these two quality issues were connected, especially since we not tested for them. Then such an indication does not have such a large impact.”

In addition, two interviewees confirm that the communication regarding quality issues could be improved:

“We mentioned it to project management, but did not receive much feedback. It could be so that the person communicated to took it seriously, but we did not see any feedback... ... It's a bit worrying that they maybe not took it seriously.”

"I think that there is a point with the communication-flows back to people. They experience that they not were heard. That could be improved."

Moreover, we have heard opinions stating that individuals should take responsibility and request information if they want it.

Regarding communication channels interviewees have also mentioned the following:

"It could be so that we just take it over the telephone, or maybe you stumble on someone in the corridor and mention it. Sometimes the procedure is that sloppy."

"I don't think we have any formal communication channels"

In addition, it seems to be unclear where this kind of detection was supposed to be documented. It was mentioned during interviews that since this kind of quality issue should not have been detected during this activity, it probably was not documented.

4.3.2 QI_2 Exceeding Critical Parameter, Component A

A specific mechanical parameter of Component A should not exceed a maximum of 3 mm. Preferably, this parameter should be 0 mm. The performance of Panther is dependent on this mechanical parameter, and the more it increases, the more difficult it becomes to achieve a given performance level. There are two sides to this quality issue. One is a higher parameter *per se* and one is this parameter increasing over time as the product is used. A higher parameter affects the accuracy of the product negatively and an increase over time destroys the calibration, which of course also affects the accuracy negatively.

The review of test reports show that during some time of the project, these measures for the Components delivered by Supplier A_2, 1st and 3rd type, were between 3-4 times the accepted limit. Post-production test reports and customer usage data show that this has been a quality issue after commercialization as well.

Source of information	Detection (~ weeks after project start)	Comment
Test report (type 1)	104-120	Test of prototype
Test report (type 3)	104-120	Test of prototype
Test report (type 3)	116	Test of component
Test report (type 3)	121	Test of component
Customer usage data (type 1)	152	
Post-production test data (type 3)	153	Test of product
Customer usage data (type 1)	160	
Post-production test data (type 1)	167	Test of product Resulted in an exchange of component A, 1st type

Table 5. Exceeding Critical Parameter, Component A

When were quality issues from the Panther development project detected?

We have not found any entry regarding this quality issue in the FMEA. However, this quality issue might be connected to QI_1, for which we have found some potential entries. Since the time of entry into the document not is noted, we can though not say much regarding when the quality issue was discovered based on the FMEA.

When asking questions about this quality issue it is sometimes difficult to elicit exactly what happened. The reason for this is that this issue was detected on two types of component A, type 1 and type 3 and it is a bit unclear which detection came first.

From data (see table 5), we can see that the issue under consideration was detected in testing somewhere between 104 and 120 weeks after project start on both type 1 and type 3. In addition, it was also found in another kind of test 116 and 121 weeks after project start, on type 3. The quality issue has also been found at two different customer sites 152 and 160 weeks after project start. In addition, it was also detected in production 153 and 167 weeks after project start.

One interviewee believes that this quality issue has been known from the start, but that focus has been on other areas. Another interviewee guesses that problems with this quality issue first began 108 weeks after project start. A third interviewee say that this quality issue was detected after the decision to change from substance 1 to substance 2 was made, which means somewhere between week 40 and 80.

Another interviewee say that these quality issues first was detected on type 3 and later on type 1. According to the interviewee, the project members suspected, since type 3 comes from the same supplier as type 1 (supplier A_2), that the same quality issue also would be found on type 1. Another interviewee states that it was most serious on type 1. However, this interviewee also states that type 3 was a problem. In relation to this, it should further be noted, which was mentioned under QI_1, that this quality issue, potentially, first was detected in the production department. If our calculations are correct, this should have been done before week 121. It is though not possible to pinpoint an exact time of detection. In addition, QI_1 deals with type 3 of Component A and we have not found any recording of a quality issue similar to QI_1 in relation to type 1.

When were quality issues from the Panther development project resolved?

According to one interviewee, Automatics still does not know if this quality issue is resolved or not. The product have never lasted more than 25 % of an accelerated test. The interviewee states that it is possible that the parameter under consideration in the future will increase in products on the market and thereby cause problems for customers. However, the same interviewee states that, depending on what the customer uses the product for, there might or might not be a problem because different applications require different measures of this parameter.

Another interviewee say that 75 % of the Panther products that are tested today measure up to the performance requirements, while 25 % do not. However, the same interviewee also states that it is difficult to test a product like Panther. Component A, for example, is tested both built in to the product and separated from it. When component A is tested separated from the rest of the product, it is difficult to say if the test represents reality or not. In addition, all tests are accelerated, which means that there is a possibility of the test procedure itself being the cause of the quality issue.

Thus, we can conclude that the organization still does not know if the problem is resolved. That customer complaints came in after the commercialization would though suggest that the quality issue not was resolved during the development project.

Could quality issues from the Panther development project have been detected and/or resolved earlier?

Regarding if this quality issue could have been detected earlier, one interviewee say that it is very difficult to foresee this kind of problem during the construction stage due to the difficulty in applying any mathematical methods in analyzing it and that the quality issue thereby not could have been detected earlier.

Another interviewee says, regarding the new supplier:

> *"The thing is that this supplier is new to us. It was the first time we used them. And we have never had any issues of this kind on this type* [technological type] *of component A with the other supplier. Thus it has not been a very large focus on this particular quality issue. This could have caused us to detect this quality issue a bit later than we should have..."*

On the question if this quality issue should have been detected earlier, another interviewee states:

> *"Yes, if you are after-wise, you should have had... Maybe more, so to speak... follow-up on this quality issue... Really have focused on it from the beginning... However, this is not something I know for sure and you will have to ask ___ about it"*

In addition, the same interviewee states that it was not clear from the beginning how the quality issue really was caused. Automatics worked with fault tree analyses together with the supplier in order to pinpoint the exact location of this increase and the reasons for it. An external organization, experienced within this field, was also contacted in order to help with the analysis. One interviewee states that this external organization could have been involved a bit earlier than it was.

Another interviewee says that testing missed a bit in the beginning:

> *"A big part of the development is testing. The test department has been good. What they missed a bit was that they had to test new properties. Since we choose a new type of component A, we were not sure on how to test them. From the time when testing was commenced it might have taken around 78 months before we knew how to test. This is particularly the case for component A. The reason for why they not detected an increase in this measure earlier was that there were other issues with component A as well, which focused attention elsewhere"*

Another interviewee also mentioned that Automatics not knew how to test this component from the beginning. This one states that in the former types of component A, sourced from another supplier, the condition of component A could always be checked by measuring the amount of a specific material in the substance. However, with the new type of component A, this could not be done in the same way.

That other issues kept the project members' attention away from the quality issue under consideration here is confirmed by two other interviewees, who more specifically pinpoint a quality issue that was detected earlier than this one. However, it is difficult to say if it was attention that was taken away since the earlier quality issue resulted in breakdown in testing. Thus, it might have been impossible to detect this quality issue because testing was aborted before there was a chance to spot it. However, on this last point we cannot be sure because we do not know exactly when testing was aborted or if the quality issue under consideration here was possible to detect at that time.

In addition, the changes in substances presented in figure 9 have also affected the work on this quality issue. Testing was begun with substance 1 in component A.

However, the supplier had previous experience of substance 2 and managed to convince Automatics to use this substance instead. Sometime after this substance was introduced, increases in this parameter were detected. The same interviewee that provided the above information states that somewhere between 26 and 52 weeks may have passed, after the change to substance 2, before this quality issue was detected, analyzed and connected to this substance as the reason for the quality issue. However, according to figure 9, substance 2 was introduced week 40 and abandoned week 80. Thus, it is likely that the interviewee remembers correctly regarding 52 weeks after project start. This quality issue must though have been detected in relation to substance 2 after week 40.

An expert from the external organization, mentioned above, was brought in and agreed with the analysis saying that substance 2 was the root-cause. Because of this, the substance was changed again. At this point, the project wanted to go back to substance 1 since the quality issues had begun after the switch to substance 2. However, substance 1 was no longer available. Then it was decided to use substance 3.

One interviewee gives two reasons for why Automatics choose to change to substance 3. The first is that there was a belief that component A would have worked with substance 1, and substance 3 was not very different from substance 1. The second was that this substance had been developed for and seemed to work in another type of component A (from supplier A_1, type 2). In addition, an underlying reason is of course that Automatics believed that substance 2 was the root-cause of the increase in this parameter at this time.

In addition, one interviewee has told us about a number of other issues that were resolved during the use of substance 2. These quality issues were also believed to be part of the factors causing an increase in this parameter. As already mentioned above, Automatics can still not say for certain if this quality issue has been resolved or not. However, one interviewee states that the substance is the key to solving this issue, which they also have learnt from experience with another supplier.

> *"These types of component A, they are so sensitive. You have to find the right substance for them. That is the key to make them work. That is something we have learnt with both supplier 1 and supplier 2. If you find the right substance you make it work. However, if you do not have the right substance you can have different types of problems. They are incredibly sensitive"*

As stated above, Automatics is still not certain whether they have found the right substance or not, and thus not certain whether they have resolved the quality issue or not.

However, another interviewee retells a slightly different story. According to this interviewee, the specification on the parameter under consideration here was increased after the quality issue, mentioned above, that was detected before the issue with the parameter. This interviewee said that this parameter was increased in order to cope with the first quality issue detected. Moreover, in relation to this deliberate increase of the parameter, there are opinions of too little integration between departments in taking the decision. In addition, this interviewee spoke about an investigation done some time ago regarding this matter, but been forgotten about when this decision was taken. According to this interviewee, this investigation had shown that this decision would decrease the performance of the product to a degree below the acceptable limit.

Moreover, interviewees have expressed opinions about the supplier's ability to foresee some of the quality issues with component A. They think that the supplier should have been able to foresee some of the issues in advance. This is an important point because we have other interviewees stating that information from the supplier regarding quality issues with component A was a substantial part of the decision basis for how to proceed with it. In addition, the same interviewees stated that this information was more or less confirmed by in-house knowledge.

4.3.3 QI_3 Customer Representative Testing

This quality issue regards testing the product in a way that represents customer usage. In particular, Panther was during the development project tested at Automatics facilities in an environment that was not totally representative of customer environments. The product is adjusted according to the behavior it shows in the environment it is tested. Therefore, adjusting the product based on test results that are not fully representative of customer usage affects some performance characteristics of the product. Conducting tests in non-representative environment might therefore result in some loss of performance during customer usage.

Source of information	Detection (~ weeks after project start)	Comment
Project meeting	50	
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance
Customer usage data	130+	Loss of performance

Table 6. Customer Representative Testing

When were quality issues from the Panther development project detected?

Regarding the question when this quality issue first was detected, the answer depends on how one defines detection. The issue here is that the project never tested for this quality issue, which the organization sometimes do in other projects. Hence, we can conclude that the knowledge about this being a potential quality issue existed in the organization. How it was handled is though a bit unclear.

As we can see in table 6 the issue was brought up during a project meeting 50 weeks into the project. However, it was not detected on the product until the product was commercialized, which happened 130 week after project start. The exact dates of these detections are though not possible to read from the data. Table 6 shows that these constitute nine cases up until we received the data. The number could though be greater today.

Regarding why this issue not was tested for, we have talked to one interviewee who says that there was no time:

"There has been no time for this in this project because changes to component A has taken all our time"

Another interviewee says that it was forgotten about:

"Interviewer: Was there no time for it?

Interviewee: Yes, there was just no one that thought about it"

A third interviewee seems to have forgotten about how it was handled, but believes that the discussion was brought up and then lost somewhere on the road.

"I believe that we had those discussions, but it was lost somewhere"

Another interviewee says that he believes that the tests actually was performed, but not performed thoroughly enough.

"I know that they performed some kinds of tests in that environment. However, I do not believe that they were performed thoroughly enough"

What we can conclude from this section is that the quality issue was known in the organization, but not detected on the product within the product development process, but at customer sites after it was commercialized.

When were quality issues from the Panther development project resolved?

This quality issue has been handled at the customer sites. Some were resolved the day we received the data and some were not. We can thus conclude that this quality issue not was resolved during the product development process.

However, testing within this environment has commenced today and the project has configured the product to work also under these conditions.

Could quality issues from the Panther development project have been detected and/or resolved earlier?

We have already seen that the organization did detect this quality issue at some time. However, it was forgotten about along the road. We have heard during interviews that if it not had been forgotten, it would have been addressed. Thus, if the project had tested for the quality issue, it is possible that it could have been detected and thereby resolved earlier. That the product was tested after detection at customer sites and afterwards configured for this environment speaks for that it would have been possible to detect and/or resolve it earlier.

One interviewee describe handling of this quality issue in the following way:

> *"Aside from application relevant knowledge and application relevant testing there is also robustness in testing, testing the products robustness, that is also one thing that there is no time for in projects like Panther. In some projects we have had time for it. With this I mean that you test the product, you do not only test the product on one sample, you test it on several samples since samples have variation, component A has variation, component B has variation, parameters have variation, stiffness has variation, friction have variation, etc... ... There has been no time for that in this project because changes to component A have taken all our time... ... The question is, should someone in our department have the power to say STOP, we can not guarantee quality on this product from a performance point of view, leading to a delivery stop. That seldom happens"*

A second interviewee say the following regarding this quality issue:

> *"This is also one thing that we need to establish in our culture. That you reprimand. Yes we have done these tests, and I don't think it's good or I think it's enough... ... Because it's so easy to just let someone else that actually may have less knowledge to make the decision. You have to take that responsibility yourself"*

4.3.4 QI$_4$ Malfunctioning Component, Component B

Similar to Component A, Component B is critical to the performance of Panther. Automatics has for a long time been working with a supplier of Component B, and it was also this supplier that delivered Component B for Panther. The delivery of Component B for the Panther prototypes worked as intended and there were no issues present. However, between the first prototype series and the second, the supplier had done some improvements in their production process to reduce their costs. These improvements affected the functionality of Panther, and during the second prototype series it was evident that the component was not functioning according to specification. This resulted in a series of interactions with the supplier, eventually leading the supplier to undo the changes.

When were quality issues from the Panther development project detected?

Our data show that approximately 90 weeks after project start, during the second prototype series, Automatics noticed that Component B was not functioning as specified.

Source of information	Detection (~ weeks after project start)	Comment
Project meeting	90	Test of prototypes
Interview	90	Supplier had made changes to their manufacturing process

Table 7. Malfunctioning Component, Component B

When were quality issues from the Panther development project resolved?

For the problem with Component B, the supplier had made some changes to their manufacturing process in order to reduce the cost of Component B. This resulted in the fact that Component B malfunctioned during the second prototype series. However, it did not take long to identify and eliminate the root-cause. As put by one of our interviewees, when asked about the resolution of QI$_4$:

> "*You could say that according to their* [the supplier] *calculations, it should have been much more robust than it was. We tested to pull to see how much force it needed before it broke, and it showed that it was much less than their calculations. Then we told them to go back to the old product* [Component B]. *Now they are using that one again, and this maybe took us one day before we identified it*"

Could quality issues from the Panther development project have been detected and/or resolved earlier?

As was noted by one of our interviewees, the problem did not lie in resolving QI$_4$, since when detected, it was immediately resolved. Rather, the problem came from the fact that the supplier had made changes to Component B late in the development process (second prototype series). This was the response to the question of whether it could have been detected earlier:

> *"Well, you could say that this is a problem that should not emerge in the first place. They* [the supplier] *made a change to Component B without… before we had a chance to test it… In the ideal case, maybe you could evaluate this change in parallel… perhaps not on this prototype, but in another product outside the project"*

Another interviewee discusses this problem with us, and agrees with the interviewee we cited above. During a more general discussion regarding QI$_4$, he said that:

> "They [the supplier] *had obviously not understood how we want to work… that they are not allowed to make changes between prototype series, and definitely not allowed to make changes without communicating it very clearly with us. Thus, they have not understood how we want to work, and how we want them to work"*

What we can say regarding QI$_4$ is that the detection and resolution of it did not take long time, it was detected and resolved within one day. However, as also stated by our interviewees, QI$_4$ was a quality issue that should not have been present in the first place. In addition, it is interesting to note that the organization itself investigated this matter and found that the same supplier earlier, in another project, had made changes to components without consent form Automatics, resulting in problems.

4.4 Cross-functional cooperation

We concluded that detecting and/or resolving quality issues is linked to organizational learning systems, and that product development is essentially about flows of information. Therefore, we specifically investigated the management of quality issues. That is, we have investigated how flows of information were managed generally, and how information about quality issues was managed specifically.

For QI$_1$ and QI$_3$, we saw that formal communication channels were lacking and the organization failed to remember a quality issue respectively. In essence, the management of these QIs was related to the interaction of upstream- and downstream groups. The consequences of having discolored substance in Component A, for example, were not totally known by the person who found it. Therefore, a consultation had to be made with other groups in the project. For QI_3 – customer representative testing – the severity of not performing customer representative tests did not become clear until the product had been released to the marketplace.

The above examples justified the investigation of how communication flows between upstream- and downstream groups during development projects in general, and for managing quality issues specifically. Our interviews showed that for general quality issues that are known for the company; formal, cross-functional routines and processes exist for managing them, e.g. FMEA. However, for previously unknown quality issues that emerge during a project, the routines and processes are not as formal. Using the FMEA as an example, one of our interviewees expressed it in the following terms, when asked about how quality issues that emerge are managed:

> *"When you have done your FMEA, it is filled with a lot of problems. But when problems show up later, it is not automatically put into the FMEA. Rather, you deal with them as they come. You use FMEA to find errors, not to document those errors that emerge"*

Moreover, during our stay at Automatics we discovered that, as they face quality issues, different groups have no standardized or formal communication channels to use. It should be noted that different formal systems exist for documenting quality issues in different functional departments. So the problem does not lie in the existence of tools and routines. Rather, these tools, documents and routines are not integrated across functions. In those cases where they are integrated, people still tend to not use them. Consider the following two statements from two interviewees when asked about the processes for gaining information across departments about various quality issues:

> *"There exist a lot of systems and a lot of different routines. For example, we have this 8D report, but I think production has something called a 4Q report. So then it is documented in their system, which we don't have insight in. In some way, it is like an 8D report, but it is documented at different places. So it is pretty poor integration… If you face a problem, then you ask people that you know instead* [in the department at question]"

> *"Ever since I started here I have thought that we are lousy at documentation… it is just recently that we have begun to document electronically… If I want to know about the problems of other departments, I have to talk to them. Okay, I CAN look in other peoples' reports, but I seldom do. And they seldom look in mine. Rather, you just ask the people you know* [in the department at question]"

Thus, it is evident that collaboration processes between different groups, regarding the detection and resolution of quality issues, are rather informal. If members of a department wish to get an update on quality issues or other issues that other departments are struggling with, they go through informal channels.

5 Analysis

We will start this analysis by showing that the product development process can be seen as a system characterized by interactive complexity. We will do this by providing examples from our research that point in this direction. However, we do not argue that the entire system and its constituent parts undisputable are systems of this kind, but rather that the product development system can be seen as it from some perspectives. If one zooms in on a single part of the system, there are undoubtedly instances that not are similar to a system characterized by interactive complexity. However, with these examples and the fact that Reason (1990) argues for it, we also argue that it can be seen as a system characterized by interactive complexity.

First of all, *components that are not linked together in a production sequence are in close proximity*. For example, as we wright in the beginning of our empirical results section (4.3, *Quality issues*), Automatics choose to involve supplier A_2, first on type 1 but later also on type 3. Later on, as we write under QI_1 and QI_2 Automatics did not manage to find the right substance during the product development process. As we write under QI_1 they did though manage to find the right substance for component A from supplier A_1. In retrospect, making the decision to use components A from supplier A_1 would probably have resulted in less problems and a shorter development process. In fact, one of our interviewees has stated that choosing supplier A_1, would, in retrospect, have been better. However, even though the question of why Automatics not managed to foresee these inconveniences is an interesting one, it is not the purpose of this thesis to analyze this decision. From our point of view, taking the information available to Automatics at that time and thus the principle of *bounded rationality* into consideration, we have not found any indications of this being a faulty decision. What we can see here is though that the decision made early on in the development process still affects the development process even though these components of the system not are linked together in a production sequence.

Moreover, choosing supplier A_2 on type 1 and type 3 created a whole chain of quality issues, which affected other activities in the product development process. These quality issues took time from other activities. Together with a time pressure, some activities had to be prioritized and some were not completed as thoroughly as wanted. For example, see QI_3, where someone has expressed a need to stop the process because quality not can be guaranteed as a result of insufficient time to complete testing. Thus, the decision to source from supplier A_2 again affected activities down the road, activities that not were linked in a production sequence.

Secondly, many *common-mode connections (i.e. components whose failure can have multiple effects 'downstream') are present*. Again, component A is a good example of this. Choosing component A from supplier A_2 resulted in several quality issues down the road:

- QI_1
- QI_2
- The supplier's production process were evaluated and changed because of quality issues
- Component A got overheated
- Etc.

As we can see, a single decision may be connected to a multitude of quality issues down the road, making it difficult to foresee inconveniences.

Thirdly, *there is a limited possibility of isolating failed components*. For example, the question of why component A caused so many quality issues to the process is an interesting one. As presented in the results, some interviewees have argued that there is a lack of an overarching development strategy. Others argue that there is a tendency to make changes to out-sourced components, leading to quality issues. We have also heard that having verified components before the project started was the issue. Moreover, some say that the supplier should have had better knowledge about the out-sourced components and thereby been able to anticipate some of the quality issues in advance. Thus, it is difficult to pinpoint a single factor responsible for component A.

Moreover, the same phenomenon can be seen in the product. The relationship between QI_1 and QI_2 is here an interesting one. QI_1 seems to have been detected first in production. However, it was not until it was connected to QI_2 that it really was seen as a "real" quality issue. Experts were brought in and confirmed Automatics suspicion of substance 2 being the cause of QI_2. In addition, as we mention under QI_2, several other quality issues, supposedly connected to QI_2 were also resolved up until the change to substance 3. Moreover, someone has commented on the possibility of that testing too aggressively may be the cause of the quality issue. Now, Automatics is still not certain of whether this issue has been entirely resolved or not. What we can see from this is that the difficulties in isolating failed components from each other obstruct the process of resolving the quality issue. In addition, as shown in the previous paragraph, the issue is further connected to factors in the development process, making it even more difficult to find a root-cause, even in retrospect.

A good example of that *there are unfamiliar and unintended feedback loops* is QI_1. As stated under QI_1, this quality issue was supposedly found in production for the first time. This is further said to have occurred during a repair conducted due to some other matter. In addition, we have not found any record of this in data and interviewees have commented that the finding probably not was documented because it was detected in a place where it was not supposed to be detected. This shows that the quality issue surfaced in an unfamiliar place, which caused problems in handling it.

In addition, as presented in results, on other types of component A, it has been possible to elicit information regarding the condition of it by measuring the extent of a specific material in the substance. On components A used on type 1 and 3 this was though not possible. In addition, interviewees have commented on the difficulties of testing components A from supplier A_2 due to the new characteristics of it. Moreover, as presented under QI_1, knowledge of the quality issue discussed was not held within the organization, which slowed down the reaction to it. In addition, interviewees have stated that Automatics, later in the process learned how to test component A from supplier A_2 more effectively. This shows that both the product and the process have unfamiliar and unintended feedback loops.

Many control parameters could potentially interact is a characteristic very similar to *there is a limited possibility to isolate failed components*. However, they may not be as similar in the case of Reason (1990), whose primary interest lies in larger catastrophes in systems like nuclear power plants. We argue that the control parameters in our case not are as studious and hence consist more of what is stumbled upon than a controlled system of control signals. However, as we write in theory there are trade-offs between time, cost and quality. Time, cost and quality are key performance indicators also in the case of Automatics. Moreover, we refer back to the second paragraph under *there is a limited possibility to isolate failed components* for an example of interacting control parameters in the product.

C*ertain states of the system must be obtained indirectly or inferred*, is certainly true for the development process as well as the product it produces. For example, in the product, it was believed that component A from supplier A_2 could be tested as component A from supplier A_1. However, later in the process it was realized that the condition of component A from supplier A_2 not could be inferred in the same way. Regarding the development process, our continuing analysis of the process will in itself show that certain states of the system must be obtained indirectly of inferred.

The last characteristic concerns that *there only is a limited understanding of some processes, particularly those involving transformations*. There are many examples of this characteristic that have come up. Firstly, as we write under QI_1, one interviewee has expressed that he/she does not know if there are any formal communication channels that are to be utilized if a quality issue is discovered. In addition, another interviewee has expressed that testing itself might cause QI_2, but that it is unclear if it does. We could go on bringing out more examples of this, but consider this enough to make the point.

We will not provide examples speaking for the product development process being a loosely coupled system. We have already argued for this, and whether it can be seen as it or not, the properties we outlined in the end of section 2, *Theory*, are important in either way.

5.1 QI_1 Discolored Substance, Component A

Regarding QI_1, the factors that stood out during our study were (1) *differing stories about quality issues*, (2) *lack of memory* regarding what happened around quality issues, (3) *insufficient knowledge* about component A and (4) an *internal communication problem*.

First of all, people have retold different stories regarding how this quality issue was detected. In addition, there are different stories regarding why this quality issue over time became regarded as more serious. Moreover, some interviewees have difficulties remembering clearly, and we have experienced that it is difficult to navigate through documents and reports trying to elicit what really happened. This indicates that Automatics not have taken time to unravel what really happened. Now, it seems to be difficult to do this due to lack of memory. This means that Automatics now have lost the opportunity to learn and improve the handling of this quality issue. When a similar issue surfaces in the future, learning from this experience will not be available to project members.

However, as presented in results (4.3.1, *QI_1 Discolored Substance, Component A*), we have been able to elicit some information regarding the quality issue. One factor, mentioned in two different ways during interviews, hindered Automatics from responding rapidly and effectively to this anticipated quality issue. According to

Reason (1990), an ability to respond rapidly and effectively, also to anticipated quality issues, is paramount for detecting and resolving them.

Firstly, regarding the time of detection, interviewees have expressed that Automatics lacked sufficient knowledge regarding this quality issue, which potentially hindered a reaction. Not having the sufficient knowledge is not a problem in itself because, in the context of product development, an organization will sooner or later always face novel situations. However, in this case, this factor seems to have caused severe problems. Thus, it might be so that Automatics knew too little about this component when they first started working on it. Further, as we stated in results, it seems like this is part of the reason for why the organization not responded faster.

Secondly, as we present under results, interviewees stated an opinion regarding that the symptom of a discolored substance had presented itself on components A from another supplier. At the components A from this other supplier this issue had never had any other effects. Interviewees have said that, since this symptom never had any effects on the other components A, Automatics did not react as fast. Obviously, Automatics here worked under an assumption stating that components A from the different suppliers were similar enough to connect the two different components A and draw conclusions about one based on experience from the other. As we have seen, this assumption was faulty. Thus, we can again confirm that there was a lack of knowledge regarding the technology of components A.

That this issue was regarded as more serious when it later was connected to QI_2 is only natural. That is because the organization at that point detected that QI_1 had an effect on the performance of the product.

What further has been brought to our attention regarding this quality issue is the feedback mentioned in results (4.3.1, *QI_1 Discolored Substance, Component A*). According to some interviewees, feedback regarding the detection was not given and could be improved. In addition, we have heard opinions stating that people maybe should request feedback if they want it. The problem with this is that people who detect quality issues and report them, but not receive any feedback, might start believing that management not cares. In the future people are thus less likely to report quality issues. If people stop reporting quality issues, the organization goes away from a *sensitive multichannel feedback system*, which according to Reason (1990) is important for detecting and resolving errors. Further, this shows that project members not were in agreement regarding how communication should be handled during the process.

As we can see, there are some areas that could be improved. However, the stories retold in this analysis are not valid enough to draw any definite conclusions. Thus, apart from a rapid response and a feedback system, we can conclude that there is a need for incorporating learning into some kind of organizational memory. Either, and preferably, the issue should be investigated and the process changed directly in order to store learning in the work structure, or the issue should be investigated and documented clearly for a later analysis and change in the system. Individual mental models regarding this issue should be let out and discussed. Later, agreement should be reached regarding how quality issues should be handled. This could be done through some kind of SOPs. This is important because, when this kind of quality issue surfaces again in the future, people will have learned from this experience and thus have a better starting position to deal with the problem.

5.2 QI_2 Exceeding Critical Parameter, Component A

When it comes to QI_2 we have found three factors prominent factors affecting detection and/or resolution of quality issues. These are (1) *differing stories* about the quality issue, (2) *insufficient knowledge* about the quality issue, (2) *Automatics-supplier communication problems* and (3) *lack of memory*.

As in QI_1, there are different stories retold regarding the detection of this quality issue, people have a hard time remembering clearly and reports and documents are difficult to navigate.

Moreover, interviewees again stated that more focus would have been directed toward this quality issue if the organization had sufficient knowledge. For example, we have heard stories stating that Automatics not tested the right characteristics from the beginning, that more follow-up regarding this quality issue would have been good and that conclusions not could be drawn regarding the condition of component A from the new supplier, based on the same information that could be used for component A from the old supplier. In addition, an interviewee stated that an external expert was brought in. However, he/she also states that this could have been done earlier. Again, we see that lack of knowledge hindered the organization from reacting.

We have also presented results stating that this quality issue was thought to be, and maybe was, connected to a number of other quality issues. The organization has worked with the belief that QI_1 was the root-cause, but not managed to entirely find a substance resolving this quality issue. In addition, a number of other issues have been taken care of along the road. These issues naturally required attention, and were, according to Automatics, factors that obstructed the detection and resolution of this quality issue. As shown earlier, in a system of this kind, this problem is though expected to present itself.

Moreover, Automatics has expressed the opinion that the supplier should have been able to foresee some of the quality issues along the road. Thus, we can conclude that Automatics not had sufficient knowledge to resolve this quality issue in a satisfying manner. In addition, it is likely that the supplier also lacked sufficient knowledge to resolve this quality issue in a satisfying manner. This is important, especially since we have understood that the supplier's assurances have played an important role in the decision making around component A. Even if the supplier had sufficient knowledge to resolve the quality issue, this indicates a communication problem between Automatics and its supplier. Several interviewees have expressed that the supplier again and again has assured Automatics of that they had found the root-cause. However, this has often turned out to be a false assurance and the issue has come up again. Thus, the communication between Automatics and its supplier is in need of improvement.

We have not been able to find any problem with the communication within the organization regarding this quality issue. Interviewees have stated that this issue was thought of as more serious than the previous one and therefore treated with greater seriousness. Since it was connected to a previous quality issue and also discovered in testing, the indication of a potential quality issue was stronger and it was probably clearer that there really was an issue.

It is though still clear that Automatics not have taken time to discuss the handling of this matter since different stories regarding how it plays out have presented themselves. We have presented opinions stating that the quality issue could have been handled faster. Thus, we have another indication of that learning regarding handling of this quality issue might have gone lost along the way and will thus not be available for use to Automatics in the future. In addition, we have learned that an investigation into this issue had been done a couple of years ago. According to interviewees, the resulting report had been forgotten about as this issue presented itself. This is another indication of that the organization has a problem with remembering past experiences.

5.3 QI$_3$ Customer Representative Testing, Component A

Three things stand out in particular when it comes to QI$_3$, namely (1) *inconsistent stories*, (2) *lack of memory* and (3) *communication problems*.

One of the difficulties with tracking the story of QI$_3$ has been that there exist several different views on why this quality issue was not dealt with. Some claim that there was not time to test for it, while others claim that tests were conducted, but not thoroughly enough. The fact that people have different explanations has, as in the cases of QI$_1$ and QI$_2$, implications for the early detection and/or resolution of quality issues. Since we have covered that already, we will not repeat it here again.

The second point that stands out is the lack of a system to document and remember quality issues that emerge during the course of the project. Testing the product in a customer representative environment is something that Automatics has done in other development projects, and is thus not a new phenomenon for them. Yet, in the Panther development project, this was brought up to discussion but either not done at all, or done but in an insufficient way. No matter the reason, we argue that in terms of Reason's (1990) defenses, the project meeting protocol does not function properly as a systematic way of documenting and remembering quality issues that emerge during the project. It should be noted that the purpose of project meeting protocols might not be to remember quality issues, but nonetheless, it was the only documented forum where this issue was brought up.

The final point has to do with the two citations we provided in the empirical part of this thesis. One of our interviewees explicitly stated that nobody in their department had the power to say, "stop", when they believe that there exist uncertainties regarding product performance. However, another interviewee explicitly said that it is the responsibility of every individual to take decisions if they are most knowledgeable about a particular matter. Clearly, these two statements indicate that there not existed an agreement regarding responsibilities for QI$_3$ in the project, which in turn is a result of poor communication.

If one department perceives that they do not have the mandate to say *stop*, whereas another department perceives that it is the responsibility of each individual with knowledge in a particular area to say *stop*, it is not surprising that this quality issue went unresolved for such a long time. As with QI$_1$, if people report quality issues and do not receive feedback about it, the organization will move away from a sensitive multichannel feedback system, which Reason (1990) argues is important for detecting and resolving errors. The reason is simple, if no feedback is provided, people will in the future tend to not report quality issues since they do not believe that the organization cares.

5.4 QI_4 Malfunctioning Component, Component B

Regarding QI_4, we have found two interesting factors, (1) an *Automatics-supplier communication problem* and (2) *lack of memory*.

QI_4 differed from the other quality issues in the sense that it was discovered and resolved almost immediately. However, the reason behind the quality issue carries similarities with the other quality issues. Clearly, Automatics and their supplier of Component B have not reached a mutual understanding of what is, and is not, allowed to change during the development project. If a mutual understanding existed, the supplier would not have changed a working component during the project.

Also for QI_2, we questioned how well the communication works between Automatics and its suppliers. Although this quality issue did not cause severe harm to the project, it is valid to assume that the communication between Automatics and its suppliers is in need of improvement.

Moreover, we have seen that this supplier earlier, in another project, had taken similar action, changing components. This again indicates a failure to remember past experiences.

6 Conclusions

6.1 When were quality issues from the Panther development project detected

In his theory about hazardous accidents, Reason (1990) described some layers of defense that aimed to detect issues that later could lead to accidents. At Automatics, some product development activities and stakeholders functioned as error detection layers as well, such as FMEA, testing, customer usage data, project members, etc. These activities occurred at various points of time during the project (which lasted for approximately 130 weeks). For the four quality issues we encountered, we saw that these activities and stakeholders detected them all during the project.

For some quality issues, we could identify specific dates when they were detected. For others, this was not possible (see table 8). In those instances, we relied on our interviewees to help us figure out when they were detected. It turned out that this was difficult to achieve.

Quality issue	Description	Time of detection (weeks after project start)
1	Discolored Substance, Component A	Not certain
2	Exceeding Critical Parameter, Component A	Not certain
3	Customer Representative Testing	50
4	Malfunctioning Component, Component B	90

Table 8. Time of quality issue detection

6.2 When were quality issues from the Panther development project resolved?

Utilizing the same methodology, review of formal documents and interviews, we attempted to answer the question of when the four quality issues were resolved. The results are shown in table 9.

Quality issue	Description	Time of resolution (weeks after project start)
1	Discolored Substance, Component A	Unresolved
2	Exceeding Critical Parameter, Component A	Unresolved
3	Customer Representative Testing	130+
4	Malfunctioning Component, Component B	90

Table 9. Time of quality issue resolution

The first two quality issues caused severe delays to the Panther development project, and were not resolved with absolute certainty when we finished this thesis. As for the third quality issue, Automatics resolved it after Panther had been commercialized, as a result of customers complaining about loss of performance. This loss of performance could have been avoided if QI_3 would have been resolved during the course of the project. Instead, it went unresolved for more than 80 weeks. The fourth quality issue was resolved basically as soon at it was detected. For QI_4, the supplier had made a change to component B, which caused it to malfunction. However, this was sorted out immediately with the supplier. Thus, as with the detection of quality issues, we conclude that some can be answered in a clear-cut way, and others can not.

6.3 Could quality issues from the Panther development project have been detected and/or resolved earlier?

As we have seen in our analysis, some factors have stood out more than others when it comes to this question.

First of all, we have seen that there are different stories retold about the quality issues. In addition, several interviewees have expressed a lack of memory regarding what really happened. These two factors together indicate that it today will be difficult to learn from the experiences with these quality issues. That is because the organization has failed in incorporating this experience into some kind of organizational memory.

Not remembering past experiences means that the organization in the future not will have maximized its probability of anticipating similar quality issues.

Moreover, especially regarding Component A, we have seen that insufficient knowledge slowed down the response from the organization and caused severe disturbances to the product development process.

We have also found an internal communication problem regarding quality issues that we have argued will discourage people from contributing to finding quality issues. In addition, we have also seen that communication with suppliers could be improved.

However, we are also questioning, partly our methodology and partly our choice of research questions. This issue will be taken up below in a discussion.

7 Discussion

As we have shown in our analysis, the quality issues encountered are difficult to follow during the course of the project. Hence, it is difficult to state with certainty the earliest time when they could have been detected and/or resolved. During our study, we have however gained insight in other areas of Automatics' product development efforts. As this discussion will show, these areas may have been contributing factors for why quality issues have been detected late or gone unresolved for a lengthy period of time. We have identified 3 areas that might have contributed to the late detection and/or resolution of quality issues, and will devote the major part of this section to these areas.

However, before going into those areas we would like to discuss the efficacy of our research methodology. That is, whether our methodology has generated suitable answers to our research questions or not. The fact that quality issues have been difficult to follow could be a result of us utilizing an inappropriate methodology.

As stated in section 1, *Introduction*, the purpose of this thesis was to investigate the conditions for detecting and/or resolving quality issues early within the product development process of Automatics and give suggestions on how these conditions can be improved for an earlier detection and/or resolution.

We aimed to accomplish that by answering the following research questions.

1. **When were quality issues from the Panther development project detected?**
2. **When were quality issues from the Panther development project resolved?**
3. **Could quality issues from the Panther development project have been detected and/or resolved earlier?**

Thus, the two first research questions are important for answering the third. However, there are certain problems with this methodology. As we have stated in section 2.1.5, *Active and latent errors*, a great amount of errors will always lie dormant in the system and may go unnoticed during the whole product life cycle. Thus, according to Reason (1990) we did not discover all quality issues within the system because a lot of them were latent and not even known by the project members. This has implications for the efficacy of our methodology because it is dependent on what people actually know and what is visible in the system. If a quality issue never was detected during the product development process, it is impossible for us to acknowledge it. Thus, since we have missed a number of quality issues, we have certainly also missed a number of processes working within this system. According to system theory, by implementing an improvement into a system you are likely to have positive as well as negative effects. That is because the system is characterized by such a high degree of complexity, which makes it nearly impossible to predict all outcomes of a specific change. In other words, in a system as complex as the one we have studied, it is impossible to be sure of having investigated all relevant processes. Reason (1990) confirms this with the characteristic *there is only a limited understanding of some processes, particularly those involving transformations* of a system characterized by interactive complexity.

Moreover, regarding research question number one, we have seen that people retold inconsistent stories when asked about the point in time when a quality issue was detected. This may be due to several different factors.

People may have experienced events in different ways because they were involved in them with different interests and/or different roles. As we have seen in section 2.2 Organizational Learning, people see things differently depending on their individual mental models. Project members from different departments may naturally have different mental models because they are connected to the project with different interests and different responsibilities. In addition, it could be so that people from different departments has seen only their side of the story. Thus, that we got different stories retold regarding the quality issues may be a sign of that the organization not learn as an organization. In other words, it lacks an integrated system that remembers and learns about quality issues.

As we have presented in section 2.2 *Organizational Learning*, speed is paramount for successful product development. Moreover, we have argued that organizational learning plays an important role in achieving rapid product development. More specifically the competitive advantage by organizational learning has been defined as the degree to which individual members are able to quickly and effectively develop, organize and utilize their knowledge base. Moreover, in this specific context *"the ability to detect and correct errors in a timely manner is dependent on effective new product development team learning systems"* (Purser, Pasmore & Tenkasi, 1992, p. 4). We argue that these inconsistent stories are a sign of lack of integration between departments. More specifically, it is a sign of an inability to share information. According to Huber (1991), organizational units constantly piece together information they have obtained from other units into new information. However, the problem is that organizational members often are unaware of what the organization actually knows. Moreover, organizational members are often unaware of what information the organization would need in order to advance successfully. At Automatics, and regarding the quality issues studied, we have seen that people retell different stories. In addition, the following conversation took place during one of our interviews:

*"**Interviewer:** When you solve that type of problems, do you usually document the problem, the cause, the intervention?*

***Interviewee:** well, I have written a report on the problem… … it is documented… … if I quit here, it will be available.*

***Interviewer:** How easy is it to reach that information?*

***Interviewee:** It is in our ERP-system, so it is reachable, if you search my name or some other suitable word…*

***Interviewer:** We have seen that you document problems on different locations. Customer problems in J5, testing problems in test-reports, production has their system, etc.*

***Interviewee:** And I write reports in some other system*

***Interviewer:** How do you integrate that knowledge?*

***Interviewee:** ... Since I began here, I have thought that we are really bad at handling documents. In fact, I am fascinated. We started putting our reports into the ERP-system only one and a half years ago. Previously, they have been collected in binders without any records of revisions… … no crawlability or anything like that. I think it is like the Stone Age here.*

***Interviewer:** How do individuals update themselves about knowledge existing in other departments?*

***Interviewee:** Well, that is not so easy… … you talk to people*

***Interviewer:** Informally then?*

***Interviewee:** I would say so… of course, I could go into other reports and look… in the test-reports for example… I can say that I seldom look in their reports, and I am sure that they never look into the reports we write either… I don't know if they still… where they put their reports… Earlier, it was in folders… now they maybe put them in the ERP-system.*

***Interviewer:** In five years, will the reports we have discussed be available?*

***Interviewee:** Well, I would say that that's a problem… After this project we will have to write a report… … We will have to write a report with all the experience we have collected… … However, handling documents is not easy… … A lot of reports from suppliers… documents that we have gotten from them… and it is thousands of pages. They are under a catalogue on a disk. And I believe that under the main folder, in which these reports are stored, there are over 75 000 unsorted documents"*

As we can see, Automatics seems to be in the process of building an integrated system. It is possible that this lack of integration explain some of the inconsistencies in the stories told to us. However, this learning system seems to be in no way optimized and we argue that there is a pretty large potential for improvement. Knowledge between departments seems to be available to people. However, there seems to be a problem with availability. As we can see in the above conversation, it is not natural for this person to acquire knowledge existing within other departments. In addition, he experiences that people from other departments not will acquire the knowledge he makes available to the organization either. Moreover, he says that he do not even know where other departments store their information. According to Huber (1991), before information search takes place within an organization, there is some kind of threshold that needs to be exceeded. That is, if perceived costs for searching for information exceeds perceived benefits, information search will not be commenced. We argue that the above raised issues increase the perceived costs for information search and that they should be addressed in order to increase the probability of organizational learning taking place. In addition, according to Huber (1991), the choice of information sources used is highly influenced by source accessibility. Thus, at the least, important sources should be easily accessible, but we argue that all sources should be accessible because people often are unaware of the knowledge residing within the organization.

Moreover, according to Huber (1991), information distribution is an important component to improve organizational learning. Huber (1991) argues that, aside from systems that routinely store information, organizations tend to have weak systems when it comes to finding the location where a certain piece of information is known to the system. He argues that the way around this is to increase the probability of learning by having the information more widely distributed so that information retrieval efforts are more likely to succeed.

We argue that this system, where different departments store information regarding quality issues in separate systems in separate formats can be improved by integration and standardization. Making information available, transparent and easily accessible to people will improve the likelihood of learning. Further on, as previously argued in section 2.2 *Organizational Learning,* having a standardized system is what will allow the organization to improve and thereby achieve an even better system in the future.

However, it could also be so that *people not remember exactly what happened*. Forgetting what happened before being able to codify learning is another threat to the effectiveness of organizational learning (Huber, 1991; Kim, 1993). That people do not entirely remember exactly what happened around the quality issues is a major threat to improving management of quality issues because of uncertain information. However, that people entirely would make up stories around the quality issues is highly unlikely and we can therefore not attribute the inconsistencies in stories around quality issues.

In addition, it may also be so that people actually do remember events the same way, but retell them in a way that favors their own position within the organization. This may be done consciously or unconsciously, but does in either way sub-optimize the organization. If this were true, which we by no way can prove it to be, it would mean that it lies in the culture of Automatics to withhold and/or distort information for the sake of benefiting oneself. The problem with such a culture is that information, and especially information about mistakes, are less likely to be shared within the organization. Not sharing information about mistakes would directly affect the main point of this thesis, namely to detect and/or resolve quality issues. The reason for this is that people would be less knowledgeable about quality issues and thereby worse equipped to handle them.

Due to the fact that we have had difficulties in eliciting credible information it is likely that the methodology used conducting this thesis not were optimal. It seems like people have forgot exactly what happened, and it would therefore be better to conduct a study of this sort in the form of an ethnography or participating observation (Bryman & Bell, 2007) of some sort. Ethnography would provide the researcher with more direct information and a chance to live the situations rather than only having them retold by different stakeholders. This would also give the researcher a better chance to assess interests within the organization in relation to action and thereby give a more accurate assessment of the dynamics within the product development system.

Moreover, regarding our third research question, we have gotten different stories retold to us. We will not repeat the discussion about research question one, but instead refer back to it. However, after conducting the study the researchers have come to believe that research question number three might not be optimal to the study we have conducted. Reason (1990) state that in a system like the one studied *there is only a limited understanding of some processes, particularly those involving transformations* and many *common-mode connections (i.e. components whose failure can have multiple effects 'downstream') are present*. This means that, from an objective perspective, any quality issue could have been avoided by a change in an activity not known to actually affect that particular quality issue. In other words, anything is possible and therefore the answer to our third research question, *could quality issues from the Panther development project have been detected and/or resolved earlier*, always be *yes*. However, according to Reason (1990), in this kind of system it is even more difficult to neglect this fact.

As we have seen in the above discussion, and also to some extent in 5, *Analysis*, asking questions with the aim of finding answers to our research questions seems to be a good way to investigating into the organizations preconditions for learning about quality issues. In addition, we have seen that several researchers have stated that effective organizational learning is one of the keys for detecting and correcting quality issues in a timely manner. That is, to proactively avoiding late detection and/or resolution of quality issues. In the light of this, and even though we have had problems addressing our research questions, we argue that we have managed to contribute to our purpose even though not in the way anticipated from the beginning.

Regarding our two first research questions, when were quality issues from the Panther development project detected and when were quality issues from the Panther development project resolved, we argue that our methodology was suitable to find the answers. That is because of the nature of the questions. Asking direct questions and investigating databases should naturally be the way to address them. As we have seen, specific dates or clear plots have though been difficult to elicit due to inconsistent stories and inadequate documentation.

However, with our methodology, we did not get clear-cut answers to our final research question regarding the various quality issues. As the discussion above has shown, the reasons may be many. The inability to answer the third question was perhaps a result of the characteristics of a complex system, which product development essentially is. Perhaps it was a result of poorly designed information systems at Automatics, making it difficult to trace quality issues. And perhaps it was a result of people choosing not to share all information with us. Whatever the reasons, the fact remains that the methodology we utilized to answer our research questions did not work entirely in the case of Automatics.

We now turn to the three areas of improvement that we identified during this research, before moving to the final section of this thesis, where we conclude our study and provide some recommendations for Automatics.

7.1 Project definition

In the beginning of the empirical results (4.1, *The Panther development project*), we presented the aims of the Panther development project, and the segments for which it was intended. We also showed that there exist different views on the expectations regarding the performance of Panther. Some groups regard the expectations as being set too high, e.g. as a result of performance tests that were neither statistically valid nor representative of the final product. Others recognize that no indications have been given that the project cannot deliver on those expectations. Moreover, there is a general question pending in the organization regarding what type of producer Automatics should be. Are they aiming to become a low-cost, “good-enough” producer, or are they aiming to continue being a producer of high-performance products?

Regarding the performance tests conducted on Panther, two issues stand out. Firstly, only one unit of each model was tested. Secondly, the products tested differed from the final product when it comes to the critical Component A. For the former – testing one unit of each model – we will not enter into a lengthy discussion. The reason for that is that Automatics knew about this risk and stated it in the test-report. If it is so that everyone not knew about it, we can simply state that there was an issue with the communication regarding this matter. However, for the second issue – not testing representative products – we argue that this should be avoided. Of course, at the time the test was conducted, Automatics might not have known that there was going to be changes made to Component A. Thus, the problem rather lies in not having fully tested and verified components prior to project start; a topic covered in the next part of this discussion section (7.2, *Project type*).

Another issue that we recognized during our study was the impression of some members that different stakeholders in the project had different images of what type of product Panther was. More specifically, the different images are related to what the expectations were on Panther's performance. Although not directly related to our quality issues, we argue that the existence of different images regarding the product may in fact lead to a late detection and/or resolution of quality issues. A number of authors have recognized the importance of having a shared understanding of the project's intent throughout the organization (Cooper, 1986; Cooper, 1990; Clark & Wheelwright, 1992; Griffin & Hauser, 1996; Morgan & Liker, 2006).

It should be noted that the divergent images of the Panther development project is not related to absence of a product specification. Indeed, a comprehensive product specification exists. We argue that the problem rather lies in the process of transforming market requirements to a product specification, and the integration of marketing and R&D to create shared images of the development project. According to Morgan & Liker (2006), Toyota has managed particularly well in creating a shared image and understanding among all project members when it comes to product development projects. At Toyota, the project manager is what they call "*the voice of the customer*", i.e. he/she is the person most knowledgeable about the market needs and thus the person responsible for developing a product that satisfies those needs. At Automatics, two persons gathered information about market requirements, the product- and first project manager. The responsibility of developing a product satisfying those requirements was the second project manager's. Hence, due to the fact that the first project manager left the Panther development project, *the voice of the customer* had to be transferred to the second project manager, and also transformed into a product specification.

Our interviews indicated that no standardized process is utilized at Automatics for the transformation of customer requirements to a product specification. Rather, it is an iterative process between marketing and R&D, based on experience and expertise. We argue that the risk of loosing insight and information, e.g. when changing project manager or transferring knowledge about the market to the project manager, can and should be minimized through the use of a formal process. This argument is in line with several scholars of product development, such as Clark & Wheelwright (1992), Griffin & Hauser (1996) and Morgan & Liker (2006). The reasons are simple; clear definitions and a shared understanding of the project's intent is one of the fundamental elements of effective product development (Cooper, 1990; Clark & Wheelwright, 1992).

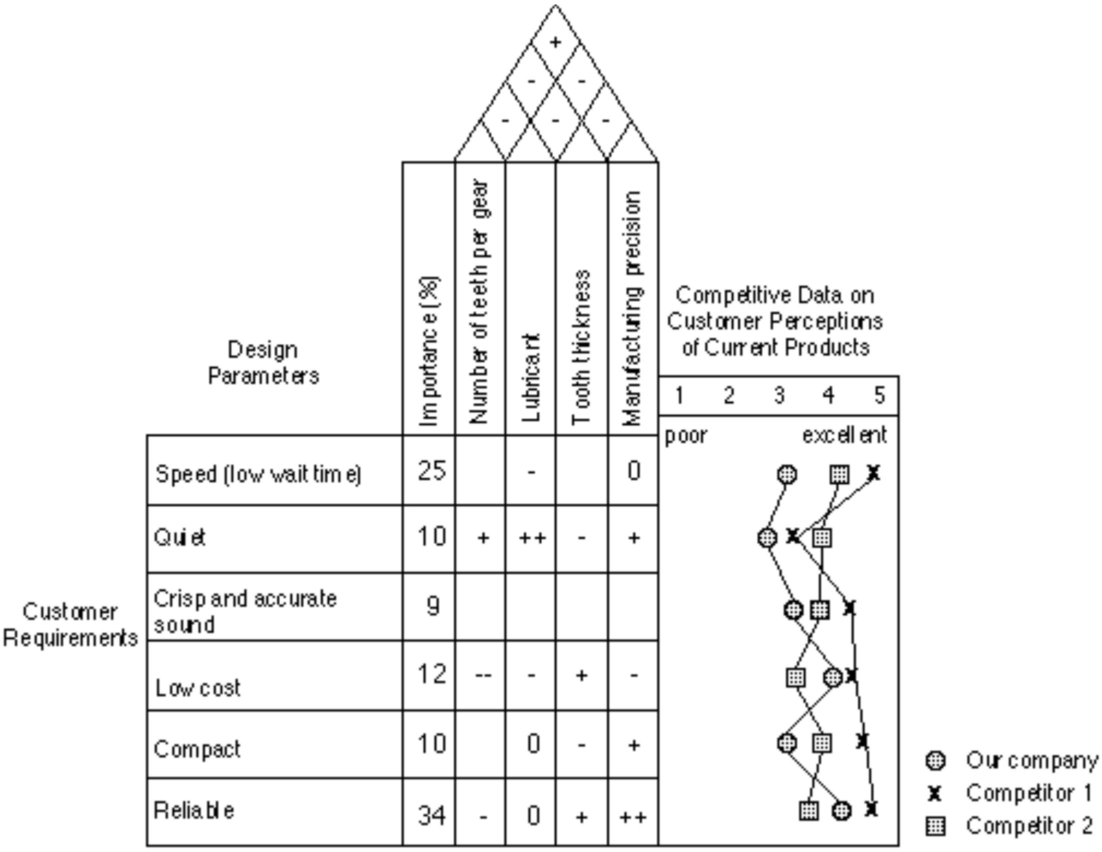
As we have stated, a standardized process for transforming customer requirements to design parameters was not utilized during the Panther development project. Instead, this transformation was done through an iterative process between marketing and R&D, mainly based on experience and expertise. We do not argue in any case that Automatics should not rely on the expertise and experience of employees. Rather, we argue that a formalized process can serve as a forum where this experience and expertise can be utilized in an effective manner. One widely used process for transforming market requirements to a product specification is *Quality Function Deployment* (QFD) (Hauser & Clausing, 1988; Clark & Wheelwright, 1992; Griffin & Hauser, 1996; Bergman & Klefsjö, 2003; Forsberg, Mooz & Cotterman, 2005; Lager, 2005). Since a cross-functional team conducts the QFD work, it will aid in creating shared images of both the market requirements for the product, and the technical parameters that will satisfy those requirements. In the case of Automatics, the use of QFD perhaps could have reduced the gap between different groups' images of Panther's performance.

Using a cross-functional team to perform QFD work has another advantage as well. One of the most frequently cited benefits, as perceived by organizations, is that the actual work done to build the "*House of Quality*" (HoQ) serves as a strong integrating mechanism for functional co-operation (Griffin & Hauser, 1996; Cristiano et al., 2000; Cristiano et al., 2001; Martins & Aspinwall, 2001; Lager, 2005). In his review of nine different studies conducted about the industrial usability of QFD, Lager (2005) concluded that the most reported benefit by organizations was that QFD had increased the amount and frequency of communication between functional departments. We therefore argue that utilizing QFD serves a two-folded purpose.

Firstly, it aids and visualizes the process of transforming *the voice of the customer* to specific design parameters that will satisfy customer needs. By formalizing and visualizing this task in a cross-functional team, we argue that a more coherent view of the project will be gained by the various functions. If people from different functions are part of specifying the customer requirements and transforming these to specific design parameters, the risk of getting different expectations on the product may be reduced. We also believe that the use of QFD might give some protection against people joining and leaving projects. In the case of Panther, one of the persons involved in gathering and transforming customer requirements left the project just before it officially started. Although we recognize that interchanging people indisputably will lead to loss of experience and expertise – and that QFD by no means mitigates that risk – the use of a formal, visual tool utilized by a cross-functional team will spread knowledge about *the voice of the customer* among more individuals, and perhaps retain that knowledge inside the organization for new people to acquire.

Secondly, it serves as a mechanism for integrating members of different functional departments (Clark & Wheelwright, 1992; Griffin & Hauser, 1996). Griffin & Hauser (1996) claim that marketers and engineers might have difficulties to co-operate due to some barriers, such as personality, cultural thought worlds, language, organizational responsibilities and physical barriers (e.g. physical distance). QFD is one means utilized to break down those barriers (Griffin & Hauser, 1996), and integrate people from different functional departments by increasing the amount and frequency of communication (Clark & Wheelwright, 1992). It is also recognized by Clark & Wheelwright (1992) that frequent and high amounts of communication between upstream and downstream groups in product development are the fundamental characteristics of the type of co-operation necessary to detect and resolve problems early in the product development process.

In essence, QFD is "a method used to identify critical customer attributes and to create a specific link between customer attributes and design parameters" (Clark & Wheelwright, 1992, p. 229). In our terminology, QFD is a formal process utilized to transform customer requirements to a product specification that will satisfy those requirements. For sake of simplicity, we will not enter into a theoretical elaboration on QFD. For such an elaboration, we advise the reader to Hauser & Clausing (1988). One of the main parts of QFD is the HoQ. Each step of the QFD process involves building the HoQ. Clark & Wheelwright provide an illustrative example of a HoQ for a gear design problem (see figure 9). There are multiple ways to use QFD, ranging from a simple HoQ (Clark & Wheelwright, 1992) to advanced QFD (Raharjo, Brombacher & Xie, 2008). Therefore, the example provided by Clark & Wheelwright (1992) only represents one way of utilizing QFD. They have divided the QFD work into a five-step process.



	Importance (%)	Number of teeth per gear	Lubricant	Tooth thickness	Manufacturing precision
Speed (low wait time)	25		-		0
Quiet	10	+	++	-	+
Crisp and accurate sound	9				
Low cost	12	--	-	+	-
Compact	10		0	-	+
Reliable	34	-	0	+	++

Figure 9. HoQ for gear design problem (adapted from Clark & Wheelwright, 1992)

The *first* step entails gathering market information and determining the critical customer requirements for the product. A team consisting of both marketers and engineers usually does this first step. As seen in the figure, the customer requirements build the rows of HoQ's central matrix. Once the customer requirements have been identified, the *second* step is to establish the critical design parameters, which describe the product in measurable terms. The design parameters are directly linked to the customer requirements and form the columns of HoQ's central matrix. The *third* step is to depict the relationships between customer requirements and design parameters, in terms of their existence and strength. For example, increasing manufacturing precision will strongly affect the reliability customer attribute, but also negatively affect the cost of the product (Clark & Wheelwright, 1992).

The *fourth* step in building the HoQ is to depict how the company performs on each customer requirement as compared to its competitors. This builds an additional block to the HoQ, where the different shapes represent different companies. Finally, the *fifth* step of building the HoQ is to identify the interrelationship or interaction between design parameters. This is the roof, and final block, of the HoQ. The negative signs indicate that increasing the value of one design parameter reduces the value of another, and vice versa (Clark & Wheelwright, 1992).

7.2 Project type

In the empirical section (4.3, *Quality issues*) of this thesis, we emphasized the importance of Component A & B for the cost and performance of the product. These two components accounted for over half the cost of the product and are critical for Panther's performance. Moreover, Component A in particular, takes extremely long time to fully test and verify (~ 30-50 weeks).

We argue, in line with our interviewees, that a development project with a time-plan of 100 weeks from concept development to commercialization leaves no room for iterations of changes, tests and verifications on critical components that take 30-50 weeks to test and verify. If the component does not function as intended, the budget and time-plan of the project is seriously jeopardized, which also happened in this case. However, the changes made to Component A have not solely affected the project time-plan. Since Component A is vital for the product, any quality issues related to it required much attention. Some of the optimization adjustments done on the product were dependent on Component A, and changes to it have required re-work to be done.

The interviews we conducted clearly show that the organization is aware of the risk involved in not completely testing and verifying components prior to project start. For the Panther project, this awareness also resulted in action. Several models of Component A were tested before the project started. Yet, the results were not optimal. It is neither in our purpose nor our interest to discuss whether or not specific changes to the component during the project were justified or not. We do not have that technical knowledge. What we argue to be the case is that incorporating an unproven component and making changes to it, particularly if it is a critical component, adds additional risk that seriously can damage the project in terms of time, cost and quality.

This phenomenon of having proven critical components and technology is not specific for Automatics. Two scholars who have recognized this are Clark & Wheelwright (1992) and Morgan & Liker (2006). Clark & Wheelwright (1992) categorize product development projects in four broad categories (see figure 10): (1) *Advanced R&D*, (2) *Breakthrough projects*, (3) *Platform projects*, and (4) *Derivative projects*. These four categories differ significantly in the sense that they require different mixes of expertise, different amounts of development time and resources, and different ways of organizing. Thus, differentiating projects in this way "*helps an organization to allocate its efforts in proportion to the need for and benefits from projects of each* *type*". Moreover, it helps to make an explicit connection between research projects and development projects (Clark & Wheelwright, 1992, p. 50).

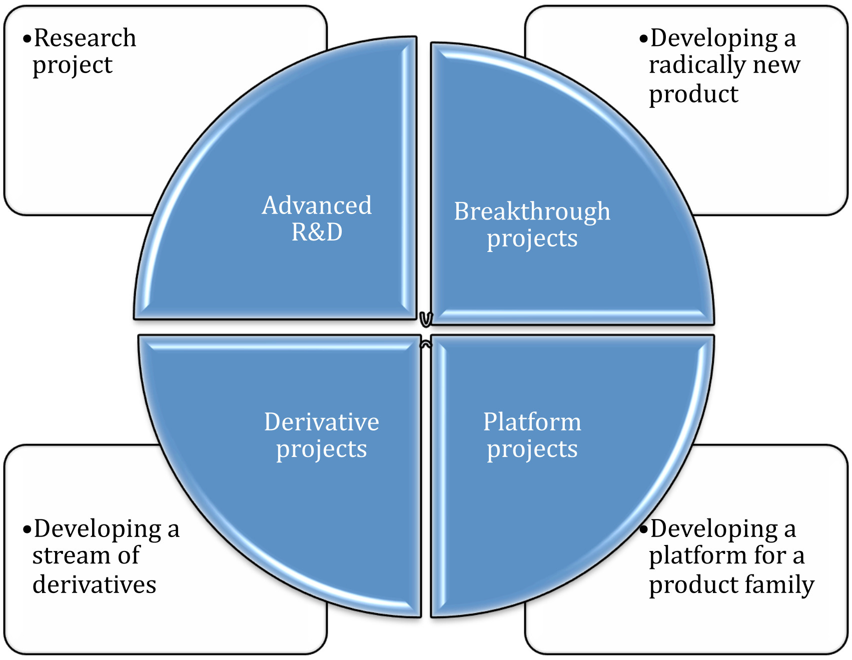


Figure 10. Project types (adapted from Clark & Wheelwright, 1992)

Breakthrough projects are often radically new, both to the company and the industry, and if successful they will most likely create an entirely new product family for the organization. Platform projects on the other hand provide the common from which a set of derivative products can emerge. Since Panther aimed at replacing an existing product family, and was not radically new to the industry, we view it as a platform project. The individuals we have spoken to at Automatics also confirmed this. We argue that the Panther development project, being a platform project, also had elements of advanced R&D. For example, the efforts to find the right combination of Component A and substance is very much a research project, rather than the utilization of research in a development project.

In their extensive study of Toyota, Morgan & Liker (2006) recognized this issue also at Toyota and identified the way that Toyota deals with incorporating new technology. Morgan & Liker (2006) found that Toyota, unlike their Western competitors, is rather conservative when it comes to incorporating new technology. One of their development principles is that, if not fully proven, they insist to wait and incorporate a new technology in future vehicles, rather than incorporating it in a current development project (Morgan & Liker, 2006). Moreover, they are extremely demanding when suppliers promote new technologies, where the supplier must provide data to prove that the new technology is tested and verified.

The fact that an unproven component was incorporated into Panther, together with the fact that Automatics has a tendency to request changes to already proven components indicate that such a principle has not been fully present at Automatics. As was noted by our interviewees, there is a tendency in the company to ask the supplier to make some changes. In that particular case, Automatics requested a change. The story of

QI_4 showed that there also had been instances where the supplier had made changes to Component B, without a formal request from Automatics.

With respect to the discussion above, we come to two conclusions. The first is that there was no clear distinction between research project and development project in Panther. Both of these efforts were present during the course of developing Panther. The second conclusion is that a separation of these would not have been enough. Even if the components had been tested and verified, any changes to them would jeopardize the validity of those tests. If components could be tested and verified in separate advanced R&D projects, and if no changes were made to these verified components during the project, a great amount of uncertainty would be reduced. This would also move the detection and/or resolution of quality issues related to Component A & B upstream, eliminating that they risk to delay the project or require re-work to be done.

7.3 War room

During our stay at Automatics, and during our search for quality issues, we came to recognize that there was no shared, formal system for managing quality issues that spanned across all functional areas. Our search showed that production indeed had their FMEA; design had their formal processes and tools for quality management, etc. However, these systems were not identical, different functions had different systems, and they were neither visible nor understood by all.

One example of this phenomenon is QI_3 – customer representative testing – where a discussion actually took place that if tests were not conducted in customer representative environments; the product might not reach the desired performance level. But as stated by one of our interviewees:

> *"I believe that we had those discussions, but it was lost somewhere"*

Thus, the project forgot about this and customer representative tests were not conducted. This caused quality problems after the product had been released to the marketplace. We also provided an example with QI_1 – discolored substance in Component A – where an early warning signal was not taken seriously. The empirical presentation regarding cross-function cooperation (4.4), together with the empirical presentations of QI_1 and QI_3, point out one interesting issue when it comes to managing quality issues.

A formal and transparent system for dealing with previously unknown quality issues that emerge during the project is lacking. In the Panther development project, each department had its own way of documenting quality issues that surfaced, and the various documents, tools and routines were not visible to the entire project. Moreover, the lacking of such a system, we argue, leads to difficulties when wanting to gain information about a particular problem. Naturally, if you wish to gain information about a particular problem from another department, you contact the person you know best, which does not need to be the person most knowledgeable about the particular problem in question. This was also confirmed by one of our interviewees.

We also argue that the missing of a formal and transparent system to deal with emerging quality issues at Automatics lead to quality issues going unresolved. If emerging quality issues are occasionally dealt with in an informal way, there is a risk that information about it will not reach the right person, or that it will be forgotten later on. One good example is QI_3, where it was recognized that not testing the product in a particular environment might lead to quality issues later on, which also eventually became the case since this was “forgotten” in the organization.

To deal with this issue, we argue that a formal and transparent system should be introduced, where emerging issues can be brought up to surface and made visible to all. One commonly used forum for this is the *war room* (Clark & Wheelwright, 1992; Forsberg, Mooz & Cotterman, 2005; Morgan & Liker, 2006). The *war room* (see figure 11) is a room or shared area entirely dedicated to a project, where project members, managers and other stakeholders can meet to discuss ongoing issues or review project status (Morgan & Liker, 2006). The various descriptions and utilizations of *war rooms* differ, but some key elements are common, and it is also those elements that we believe would be of value to Automatics for the early detection and/or resolution of quality issues.

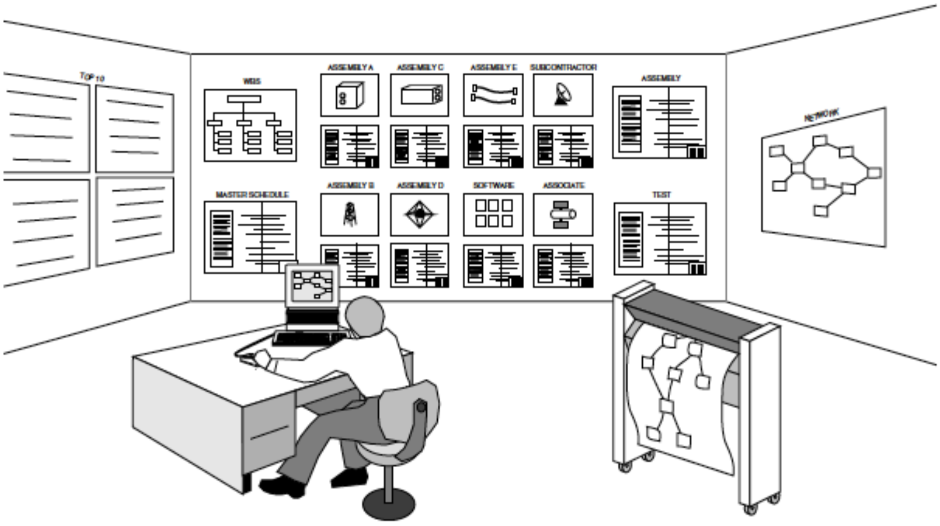


Figure 11. War room (adapted from Forsberg, Mooz & Cotterman, 2005)

Firstly, it provides an organizational memory for quality issues during specific development projects. For example, each functional department or engineering discipline can have its own area in the room. If a project member recognizes a risk or potential quality issue, he/she can post it on the wall, and mark it as “unresolved” until the risk has been mitigated or the problem solved. Making the problem visible in this way significantly increases the probability that it will be solved (Slack, Chambers & Johnston, 2007), rather than if it only is discussed informally. Moreover, it creates a forum where the people who care about the detection and/or resolution of quality issues have access to information about those quality issues. Perhaps, if the question regarding customer representative testing was visible during a long period of time, actions might have been taken to ensure that those tests were conducted. Of course, making quality issues visible is not a guarantee for early resolution. However, it aids in remembering the existence of that quality issue, which will increase the likelihood that it will not be forgotten along the way.

Secondly, we argue that more knowledge can be utilized in the attempt to resolve quality issues early. As we heard several of our interviewees say during our study; when searching for information about particular problems, they contact the people they know best. Because of that, the people who have the best knowledge about a particular quality issue might not be the people involved in solving it. We do not in any way argue that a war room will replace the need for people from various departments to collaborate informally. Several authors have recognized the importance of informal collaboration between functional groups for success (Takeuchi & Nonaka, 1986; Pinto & Pinto, 1990; Clark & Wheelwright, 1992; Griffin & Hauser, 1996; Morgan & Liker, 2006). Rather, we argue that the war room will complement this informal collaboration. If potential issues are made visible for all, people that in other cases would not even know that a problem existed, would now have visible information about the problem and, if they are competent enough, can contribute to solve it.

Last but not least, the use of a war room has been reported as a particularly effective way to facilitate cross-functional communication (Morgan & Liker, 2006). Morgan & Liker (2006) argue that the war room (*Obeya room* in their book) facilitate cross-functional communication without the need to co-locate people from different departments. By being located in the respective functional departments, a deeper functional knowledge can be developed (Clark & Wheelwright, 1992). By having a war room where you can meet and collaborate with project members from other departments, that knowledge can be utilized in an effective way (Morgan & Liker, 2006). Both Clark & Wheelwright (1992) and Morgan & Liker (2006) emphasize that the single best way to solve problems early in the product development process is through intensive, integrated problem solving, which is characterized by rich and frequent cross-functional communication. As with QFD, it is not the primary purpose of the war room to facilitate such problem solving. However, it is a frequently cited benefit that the existence of such a room will intensify the communication between project members, and perhaps lead to early resolution of quality issues.

As we pointed out earlier, war rooms take different forms and are used for different purposes. We will here just shortly describe the essentials of a war room, and advice the reader to Forsberg, Mooz & Cotterman (2005) for a more comprehensive description. The war room is basically just a room designated for a particular project, where information of various kinds is posted on walls and visible to all. For example, different walls (fix/mobile) in the room can be designated for different functions or engineering disciplines, where they can post the status of their work, eventual problems they are experiencing, etc. Each function or discipline hence assigns a person who is responsible for keeping “the wall” up to date. Thus, a key element of the war room is visual communication. Another key element of the war room is that it should be open for all project members and a place people naturally go to when working on the project. One of the most frequently cited benefits of the room is that it facilitates cross-functional cooperation between project members. Therefore, the room should be seen as a room for project members (Forsberg, Mooz & Cotterman, 2005; Morgan & Liker, 2006).

8 Recommendations

In this thesis, we have attempted to provide suggestions on how quality issues can be detected and/or resolved earlier in Automatics' product development process. To do so, we specified three research questions in the beginning of this study. This section will provide the answers to those questions and provide that set of suggestions for Automatics.

Although it became evident that our methodology was not appropriate for the purpose of this thesis, we recognized several areas of improvement during the time we spent at Automatics. These areas were not directly linked to our quality issues, but might affect the early detection and/or resolution of quality issues at Automatics.

Firstly, different views existed regarding the expectations of Panthers' performance. Some argued that the expectations were set too high, while others argued that no indication had been given that those expectations could not be met. Having a shared understanding of the project's intent throughout the organization is vital for successful product development (Cooper, 1986; Cooper, 1990; Clark & Wheelwright, 1992; Griffin & Hauser, 1996; Morgan & Liker, 2006). Not having a shared understanding could lead to a late detection and/or resolution of quality issues.

We recognized that the differing images of Panther existed mainly between R&D and marketing. Hence, we argued that there could be potential problems in the integration of marketing and R&D. Using Toyota as a yardstick, we propose a widely used formal process for transforming customer requirements to design parameters, namely *Quality Function Deployment* (QFD). By conducting QFD with a cross-functional team, Automatics can gain several advantages. Firstly, a shared understanding among various departments will be created since QFD is a cross-functional activity. Secondly, QFD can serve as an integrating mechanism, which enhances the amount and frequency of communication between functions, which is one fundamental element in early problem resolution (Clark & Wheelwright, 1992). Finally, utilizing a formal and visual tool, such as QFD, can reduce the risk of loosing information and knowledge when people are transferred to and from projects.

The second area we identified was related to specification of project types. We have several times in this thesis described the importance of Component A, in terms of cost and performance for the type of product that Automatics develops. Numerous changes to this component were made during the course of the Panther development project. In a way, this component was developed parallel to the actual product in which it was going to be used. This caused several delays to the project and component A is the main reason for why the project is not terminated yet. The optimal situation would be to have models of Component A tested and verified prior to development project.

Several authors have recognized this phenomenon, and propose that development project should be categorized. Clark & Wheelwright (1992) provide four categories of development projects: (1) *Advanced R&D*, (2) *Breakthrough projects*, (3) *Platform projects*, and (4) *Derivative projects*. These four categories differ significantly in the sense that they require different mixes of expertise, different amounts of development time and resources, and different ways of organizing. Thus, differentiating projects in this way "*helps an organization to allocate its efforts in proportion to the need for* *and benefits from projects of each type*". Moreover, it helps to make an explicit connection and distinction between research projects and development projects (Clark & Wheelwright, 1992, p. 50). By clearly separating the different types of development projects, the risks taken in Panther that ultimately lead to several delays can be avoided in future development projects.

The third and final area of improvement that we identified was related to the organizational memory. During this study, we came to recognize that no cross-functional system existed for the documentation and storage of quality issues that were previously unknown. When quality issues that had not been documented in the FMEA activity emerged, there was no process in place to manage them. The case of QI_3 exemplifies this, where the organization forgot to resolve this detected quality issue. In essence, there is a lack of a formal, cross-functional and transparent system for managing quality issues.

We suggest that Automatics should consider using war rooms during their development projects. Firstly, it provides an organization memory for quality issues during specific development projects. In this way, the organization can document and remember the issues that emerge during the project, reducing the risk of forgetting to solve emerging quality issues. Secondly, formalizing and visualizing the quality issues can increase the possibility that more people become aware of them, and thus more knowledge can be utilized in the attempt to resolve quality issues early. Finally, similar to QFD, the use of a war room has been reported as a particularly effective way to facilitate cross-functional communication (Morgan & Liker, 2006). As we have stated earlier, cross-functional communication is essential for early problem resolution.

9 Suggestions for future research

One very interesting conclusion from this thesis was that the methodology we utilized did not suffice to answer the third research question. Perhaps it is more interesting as a contribution rather than a suggestion to the academic world that this methodology cannot provide a clear-cut answer to research question three. We also concluded that no generalizations could be made to this field. Therefore, it is in our belief that studies in this field with similar and other methodologies would be of interest to both scholars and practitioners.

One of the main problems we encountered during this study was to find appropriate literature in the field of early quality issue detection and/or resolution in product development. Numerous of authors recognize that the later quality issues are detected and/or resolved, the higher the cost for the organization (Clark & Wheelwright, 1992; Repenning, 2001; Bergman & Klefsjö, 2003). Despite this wide recognition, little is written about the means by which organizations can detect and/or resolved quality issues early. Our research is an attempt to develop that theory for the organization under study. More generalizable research is hence needed in this field.

Moreover, it seems that no clear definition of a quality issue exists. Reason (1990) presented the notions of active and latent errors, and how different triggering factors could lead to a failure (see section 2.1.5, *Active and latent errors*). Reason's (1990) theory was related to a characteristic of complex system, namely that c*ommon-mode connections (i.e. components whose failure can have multiple effects 'downstream' are present*. Thus, is it the root cause that is the quality issue? Or is it the triggering factors that are the quality issues? Are the effects that the quality issues have "downstream" the actual quality issues? Having an explicit definition of quality issues will help future researchers to focus their efforts when conducting research in this field.

Finally, much of what is written today about quality issue detection and/or resolution use manufacturing as point of reference. For example, the theories within *Total Quality Management* (TQM) are clustered around the various learning cycles, such as the *Plan-Do-Check-Act* (PDCA) cycle (Bergman & Klefsjö, 2003). However, in complex systems, such as product development, the learning from error detection and resolution is not as clear-cut as in manufacturing. The characteristics of complex systems, as described by Perrow (1984), indicate that error detection and/or resolution within product development, as compared to manufacturing, is a much more difficult task. Perhaps researchers within the field of quality management will address this issue in future research.

List of references

Bergman, B. & Klefsjö, B. (2003). *Quality from customer needs to customer Satisfaction.* 3rd ed. Lund. Studentlitteratur

Bryman, A. & Bell, E. (2007). *business research methods*. 2nd ed. New York. Oxford University Press

Clark, B. K. & Fujimoto, T. (1991). *Product Development Performance: Strategy, Organization and Management in the World Auto Industry*. Boston. Harvard Business School Press

Clark, B. K. & Wheelwright, C. S. (1992). *Revolutionizing Product Development*. New York. The Free Press

Cooper, R. G. (1986). *Winning at new products*. New York. Perseus Books

Cooper, R. G. (1990). State-Gate Systems: A New Tool for Managing New Products. *Business Horizons.* Vol. 33. No. 3. pp. 44-54

Cristiano, J. J., Liker, J. K. & White, C. C. (2000). Customer-driven product development through quality function deployment in the U.S. and Japan. *Journal of Product Innovation Management*. Vol. 17. No. 4. pp. 286–308

Cristiano, J. J., Liker, J. K. & White, C. C. (2001). Key factors in the successful application of quality function deployment (QFD). *IEEE Transaction on Engineering Management*. Vol. 48. No. 1. pp. 81–95

Edmondson, A. C. (2004). Learning From Mistakes Is Easier Said Than Done: Group and Organizational Influences on the Detection and Correction of Human Error. *The Journal of Applied Behavioural Science*. Vol. 40. No. 1. pp. 66-90

Forsberg, K., Mooz, H. & Cotterman, H. (2005). *Visualizing Project Management: Models and Frameworks for Mastering Complex Systems*. 3rd ed. New Jersey. John Wiley & Sons

Griffin, A. & Hauser, J. R. (1996). Integrating R&D and Marketing: A Review and Analysis of the Literature. *The Journal of Product Innovation Management*. Vol. 13. No. 3. pp. 191-215

Hauser, J. R. & Clausing, D. (1988). The House of Quality. *Harvard Business Review*. May/June. pp. 63-73

Kahn, K. B., Castellion, G. & Griffin, A. (2005). *The PDMA Handbook of New Product Development*. 2nd ed. New Jersey. John Wiley & Sons

Kim, D. H. (1993). The Link between Individual and Organizational Learning. *Sloan Management Review*. Vol. 35. No. 1. pp. 37-50

Kvalé, S. (1997). *Den kvalitativa forskningsintervjun*. Lund. Studentlitteratur

Lager, T. (2005). The industrial usability of quality function deployment: a literature review and synthesis on a meta-level. *R&D Management.* Vol. 35. No. 4. pp. 409-426

Martins, A. & Aspinwall, E.M. (2001). Quality function deployment: an empirical study in the UK. *Total Quality Management*. Vol. 12. No. 5. pp. 575–588

McElroy, M. W. (2000). Integrating complexity, theory, knowledge management and organizational learning. *Journal of Knowledge Management*. Vol. 4. No. 3. pp. 195-203

Meyers, P. W. & Wilemon, D. (1989). Learning in New Technology Development Teams. *Journal of Product Innovation Management*. Vol. 6. pp. 79-88

Morgan, J. M. & Liker, J. K. (2006). *The Toyota Product Development System – Integrating People, Process, and Technology*. New York. Productivity Press

Nonaka, I. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organization Science*. Vol. 5, No. 1, pp. 14-37

Perrow, C. (1984). *Normal accidents*. New York. Basic Books

Pinto, M. B. & Pinto, J. K. (1990). Project team communication and cross functional cooperation in new program development. *The Journal of Product Innovation Management*. Vol. 7. No. 3. pp. 200-212

Purser, R. E., Pasmose, W. A. & Tenkasi, R. V. (1992). The influence of deliberations on learning in new product development teams. *Journal of Engineering and* Technology Management. Vol. 9. pp. 1-28

Reason, J. (1990). *Human Error*. Cambridge. Cambridge University Press

Repenning, N. P. (2001). Understanding fire fighting in new product development. *The Journal of Product Innovation Management*. Vol. 18. No. 5. pp. 285-300

Robertson, D. & Ulrich, K. (1998). Planning for Product Platforms. *Sloan Management Review*. Vol. 39. No. 4. pp. 19-31

Raharjo, H., Brombacher, A. C. and Xie, M. (2008). Dealing with subjectivity in early product design phase: A systematic approach to exploit Quality Function Deployment potentials. *Computers & Industrial Engineering.* Vol. 55. No. 1. pp. 253-278

Schein, E. H. (1999). *Process Consultation Revisited – Building the Helping Relationship*. Reading. Addision Wesley

Shani, A. B., Chandler, D., Coget, J-F. & Lau, J., B. (2009). *Behavior in Organizations: An Experiential Approach*. New York. McGraw-Hill/Irwin

Slack, N., Chambers, S. & Johnston, R. (2007). *Operations Management.* 5th ed. New York. Pearson Education

Takeuchi, H. & Nonaka, I. (1986). The New New Product Development Game. *Harvard Business Review.* Vol. 66. No. 1. pp. 137-146

Ulrich, K.T. (1995). The role of product architecture in the manufacturing firm. *Research Policy*. Vol. 24. No. 3. pp. 419-440