

Agile Methods at Delphi Automotive Torslanda

When and how they are appropriate

*Bachelor's Thesis in the Bachelor's Programme
Economics and Manufacturing Technology*

Filippa Svensson
Karolina Lagerstedt

BACHELOR`S THESIS E 2016:080

Agile Methods at Delphi Automotive Torslanda
When and how they are appropriate

Filippa Svensson
Karolina Lagerstedt

Tutor, Chalmers: Per Svensson
Tutor, company: Marcus Hedberg

Department of Technology Management and Economics
Division of Entrepreneurship and Strategy
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2016

Agile Methods at Delphi Automotive Torslanda

When and how they are Appropriate

Filippa Svensson

Karolina Lagerstedt

© FILIPPA SVENSSON AND KAROLINA LAGERSTEDT, 2016.

Bachelor's Thesis E 2016: 080

Department of Technology Management and Economics
Division of Entrepreneurship and Strategy

Chalmers University of Technology

SE-412 96 Gothenburg, Sweden

Telephone: + 46 (0)31-772 1000

Cover:

Authors' illustration of Agile Product Development

Chalmers Reproservice
Gothenburg, Sweden 2016

Agile Methods at Delphi Automotive Torslanda

When and how they are Appropriate

Filippa Svensson, Karolina Lagerstedt

Department of Technology Management and Economics

Chalmers University of Technology

Abstract

Organizations live in an environment of change (Bruzelius and Skärvad, 2011). The recurring and increasing amount of changes are making changes the normal state instead of something that only happens now and then.

One of Delphi's customers will, in a near future, almost reduce their development time by 50 percent. This means that Delphi also have to change their way of working within projects to meet their customer's new demand

These two factors, the increasing amount of changes and the customer's reduction of development time, have forced Delphi to investigate alternative ways of working. One suggestion has been to use Agile methods and they think that it might be the solution to some of the problems they are facing.

Based on this background, the purpose of this report is to determine when and how agile methods are appropriate at Delphi Automotive Torslanda.

Literature studies and interviews have been the foundation to answering the purpose of the report.

It has been found that there are different kinds of uncertainties, which affects what kind of product development style that is appropriate.

One conclusion is that Agile product development is appropriate when the technical and market uncertainties are high. However, when the uncertainties occur because of changes in the specification, planned flexibility is a good way to handle this kind of situation.

The second conclusion is, that one way to apply Agile methods at Delphi can be through a combination between Agile methods and their existing Stage-Gate model. The key to do this is to use Agile methods and the Stage-Gate model at different levels.

Keywords: Agile, Uncertainties, Planned Flexibility, Hybrid, Stage-Gate, Development Styles

Table of contents

1. INTRODUCTION.....	1
1.1 BACKGROUND	1
1.2 PURPOSE.....	2
1.3 RESEARCH QUESTIONS	2
1.4 METHOD.....	2
2. WHEN AND HOW TO USE AGILE METHODS – A LITERATURE REVIEW.....	4
2.1 IMPORTANT BUILDING BLOCKS.....	4
2.1.1 Agile Methods.....	4
2.1.2 Process Model for Projects – The Stage-Gate Model.....	6
2.2 WHEN TO USE AGILE METHODS	8
2.2.1 Different Development Styles for different Situations.....	8
2.2.2 Planned Flexibility	8
2.3 HOW TO IMPLEMENT AGILE METHODS	10
2.3.1 Application of the Hybrid System.....	10
2.3.2 Possible Effects of the Hybrid System	11
2.3.3 The triple A System.....	13
2.4 WHEN AND HOW TO USE AGILE METHODS – A SUMMARY	14
3. THE CURRENT STATE AT DELPHI AUTOMOTIVE TORSLANDA	16
3.1 CONCEPTS IN THE PROJECT ORGANIZATION	16
3.2 INTERVIEWS	18
3.2.1 The Interviewees Description of the current State at Delphi.....	18
3.2.2 The Interviewees View of Problems of the current State.....	20
3.2.3 Agile.....	21
3.3 SUMMARY AND CONCLUSIONS OF THE INTERVIEWS.....	23
4. DISCUSSION AND CONCLUSION.....	25
4.1 DISCUSSION.....	25
4.1.1 When are Agile Methods appropriate to use at Delphi.....	25
4.1.2 How to implement Agile Methods at Delphi, and what Prerequisites they have to do this.....	27
4.2 CONCLUSION.....	30
REFERENCES.....	31
APPENDIX 1 - INTERVIEW QUESTIONS.....	I

1. Introduction

The pace today is higher than it was yesterday, which forces companies to adapt to this new faster environment (Bruzelius and Skärvad, 2011). To be more competitive, the car manufacturers want to reduce their development times, to get new cars with new technology to the market faster.

Delphi Automotive is a company that delivers solutions to the automotive industry (Delphi Automotive, 2016). Delphi Torslanda is a service center for the designing, engineering and manufacturing of the electrical architecture. They work in projects within their product development, closely connected to their customers. Therefore Delphi Automotive is highly affected by the new pace in the automotive industry.

1.1 Background

Organizations live in an environment of change (Bruzelius and Skärvad, 2011). The recurring and increasing amount of changes are making changes the normal state instead of something that only happens now and then.

Increased complexity and competitiveness are forcing companies to reconstruct their organization (Henriksen and Andersen, 2010). Companies strive to be more lean and agile by improving processes, reducing time to market, increasing technological and organizational flexibility, and gaining access to new competence and capabilities.

Product life cycles are getting shorter and products are getting outdated, even if they still work (Bruzelius and Skärvad, 2011). This pushes companies and product development teams to launch their products at a higher pace.

30 years ago the standard for the car manufacturers was to launch a new car model every four to six years (Bruzelius and Skärvad, 2011). Today the launches occur much more often due to the high pace in technology development. Therefore a development time less than two years can be a necessity for some car manufacturers to be competitive.

One of Delphi's customers will, in a near future, almost reduce their development time by 50 percent. This means that Delphi also has to change their way of working within projects to meet their customer's new demand.

These two factors, the increasing amount of changes and the customer's reduction of development time, have forced Delphi to investigate alternative ways of working. They need to be more resource effective and have an effective management of change. One suggestion has been to use Agile methods and they think that it might be the solution to some of the problems they are facing.

The work with Agile methods has already been implemented in the software development at Delphi. However, now they are thinking about implementing Agile methods in their other product development as well.

Experiences of Agile methods shows that using them can lead to shorter development times, flexibility of design and increased resource efficiency (Begel and Nagappan, 2007).

1.2 Purpose

The purpose of this report is to determine when and how Agile methods are appropriate at Delphi Automotive Torslanda.

1.3 Research Questions

The purpose can be divided into two more specific research questions.

1. When are Agile methods appropriate to use at Delphi Torslanda?
2. How can Agile methods be implemented at Delphi Torslanda?
 - How can Agile methods be combined with the Stage-Gate model (PDP) at Delphi?
 - What prerequisites do Delphi Torslanda have, to implement Agile methods?

1.4 Method

The method can be divided into two parts. The first part is a literature study and the second is research of the current situation through interviews.

To find literature to answer the purpose of the report, searches on Agile methods, Stage-Gate systems and the combination of the two have been made.

The literature studies have been done to identify possible solutions. For each of these solutions there are results and a number of prerequisites, which also have to be identified. By identifying possible solutions, prerequisites and results, the relationship between these can be mapped out.

Eight interviews were held. 12 persons of different positions were interviewed: three Program Managers, three Technical Project Leaders, two Functional Managers and four Core Team members. A more thorough explanation of the different positions can be found in section 3.1.

The questions used in the interviews can be found in appendix 1. The interviews should provide information to answer the following questions:

- Get an understanding of the current situation at Delphi.
 - How is the organization around projects?
 - What kind of problems do they experience when working in projects?
- A description of the interviewees' thoughts and experiences of Agile methods.
 - Do they have any experience and knowledge of Agile methods?
 - Do they think that Agile methods will work at Delphi?

To get a deeper understanding, and start a discussion of Agile methods with some of the interviewees, a workshop was held. The workshop was based on theory from the report.

The application of Agile methods in Product Development, outside of Software Development, is still relatively new and scientific reports on this is still scarce.

The knowledge of Agile methods at Delphi Torslanda is low. This is a first attempt to discuss how and when Agile methods can be introduced. This report is therefore of a more explorative nature.

2. When and how to use Agile Methods – A Literature Review

The theory chapter is divided into four subchapters. The first describes the important building blocks of this report. This subchapter contains fundamental theory for the following subchapters and consists of Agile Methods and Process Model for Projects – The Stage-Gate Model. The project model currently used at Delphi is a Stage-Gate model.

The second subchapter is connected to the first research question, when to use Agile methods at Delphi Torslanda. The first part deals with the importance of different development styles when dealing with different kinds of situations. The second part discusses a way of handling uncertainties and the many changes that are becoming more common.

The third subchapter is connected to the second research question, how to implement Agile methods at Delphi, and what prerequisites are needed. This subchapter discusses different ways of handling hybrids between a Stage-Gate model and Agile methods.

Finally, the last subchapter is a summary of the most important parts of the theory, connected to the research questions.

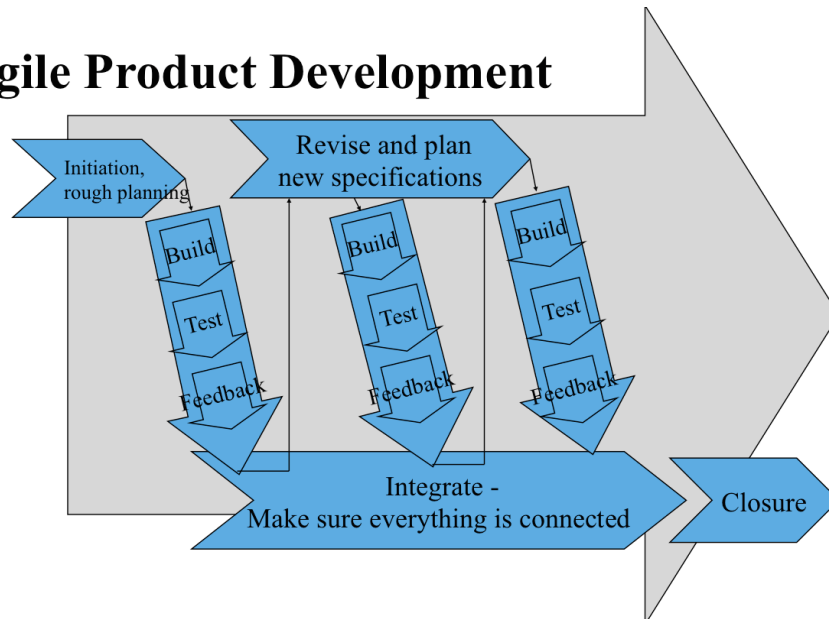
2.1 Important Building Blocks

2.1.1 Agile Methods

In a complex product development environment where specifications can be changed late in the project, iterations are unavoidable and can be necessary for a successful outcome (Sommer, et.al., 2015). In uncertain environments a strict Stage-Gate project model, which try to eliminate iterations, is not suitable. Through the Agile product development approach iterations are managed and the effect of changes is minimized (Karlström and Runeson, 2005).

Agile methodology originates from the software development industry. A basic feature is iterations, sometimes called sprints, which result in testable products (Agile methodology, 2008). This is illustrated in picture 1. The webpage agilemethodology.org can give a more thorough explanation.

Agile Product Development



Picture 1. The basics of Agile methodology are described, iterations and how new specifications can be handled.

The Agile development can help meet additional customer requirements through the stages of build, test, feedback and revise (Cooper, 2009). Time is the fixed variable and the steps are repeated until the project objectives are fulfilled.

The key to handle the iterations is to have an architecture that is both modular and scalable (MacCormack, 2001). A successful modular architecture isolates the changes to specific modules, and therefore minimizes the effect of the change. A scalable system facilitates late changes since it gives structure.

One benefit with using Agile methods is that the development team can move forward even before all product specifications are decided, making an earlier launch of the project possible (MacCormack, 2001). Another benefit is connected with the ambition of Agile methodology, to get technical parts in place early. This gives the project team a chance to detect possible problems with the composition of the product early, before it is too late.

The Agile principles can be summarized through the Agile manifesto that was written by 17 representatives from different areas and with different experience of software development (Agile manifesto, 2001).

The Agile manifesto:

1. Individuals and interactions, over processes and tools.
2. Working software, over comprehensive documentation.
3. Customer collaboration, over contract negotiation.
4. Responding to change, over following a plan.

Begel and Nagappan (2007) sent out a survey to different software developers and through that they came up with a list of benefits and problems with Agile development. Top three benefits are improved communication and coordination, quick releases and

flexibility of design. Mentioned problems is that Agile methodologies does not scale to larger projects and demand too many meetings.

Scrum – A Way of working with Agile Methods

Scrum is an Agile development process that is lightweight, simple to understand but difficult to manage, (Schwaber and Sutherland, 2013). More information about Scrum can be found in The Scrum Guide (Schwaber and Sutherland, 2013).

Within Scrum there are three repetitive stages (Sommer, et.al., 2015):

Product backlog development - The purpose of the product backlog development is to determine the demand specification for a specific sprint. These specifications may or may not be features included in the finished product. Depending on test results, additional customer requirements etc. the product backlog will change over time as the project proceed.

Main sprints - From the product backlog development, specific increments are chosen, that will be fulfilled during the current sprint. The sprint goals are fixed during the time of the sprint. The main sprint can be broken down to sub-activities that is visualized on a board, called a Scrum board.

Daily sprints - In the daily sprint there is an everyday 15 minute meeting which purpose is to update the team members what have been done and what to be done by every team member the last and next 24 hours. If there are any problems, the team can choose to rearrange the priority of sub-activities to accommodate the problem.

There are three roles In the Scrum team (Schwaber and Sutherland, 2013):

The project owner - The person responsible for managing the product backlog and for ensuring that the development team can proceed with their job, both in order to obtain maximum value from the project.

The development team - The people who create a potentially releasable increment of the product at the end of each sprint. They have the empowerment from the organization to manage their own work.

The Scrum master - This person's task is to ensure that the work with Scrum is well understood both by the people in the project and by the rest of the organization.

2.1.2 Process Model for Projects – The Stage-Gate Model

As complexity of projects grow, the project models become increasingly important (Sommer, et.al., 2015). Today most companies have, either their own, or a general project model to help them execute and control their projects. Most of these project models are based on some version of the Stage-Gate model.

The Stage-Gate model divides the project into subsequent stages, and between each stage is a gate which works as a checkpoint (Gunnarson, et.al., 2000). Each gate has a set of deliverables and criteria, which has to be fulfilled before proceeding to the next stage. At each gate there are gatekeepers who review, assess and approve the work at each stage. These gatekeepers are often a group of senior managers. One purpose with

the Stage-Gate model is to predict the project from the beginning, in order to reduce uncertainty and iterations (Sommer, et.al., 2015).

The study made by Sommer et.al. (2015) identifies some of the main advantages of using a Stage-Gate system to control a project. They are: increased speed, better quality, greater discipline and better overall performance. However, one negative aspect of using a Stage-Gate model is that it is often associated with heavy documentation and fixed activities.

In 2011, a study was made by the American Productivity & Quality Center (APQC) and the Product Development Institute (PDI) to identify best practices in the idea-to-launch process (Cooper and Edgett, 2012). The study was made on 211 best-performing business units within different industries, and the purpose was to find information about their idea-to-launch process. After a two-step data collection and a thorough analysis, five key attributes the most successful firms processes share, was identified:

- First, “they are visible and documented at an operational level”. Their idea-to-launch process is clearly mapped out and well documented at all levels in the company. The Stage-Gate model is a tool to get a clearly mapped out and well documented idea-to-launch process.
- Second, “they are really used”. The first step is to map out the process, but that is not enough, the process have to be used in the daily work.
- Third, “they enable project teams to access the resources they need to succeed”. A successful process is one that facilitates the development, in other words, it helps the project team to launch the product.
- Fourth, “they incorporate compliance checks to ensure that the process is followed”. To see how well the process is being followed, is a good way to determine how well, and if, the process is working.
- Last, “they are adaptable and scalable”. One size does not fit all, a low risk project cannot be treated the same as a high risk project. The process should be adaptable and scalable to fit all these different types of projects.

According to Gunnarson et al. (2000) project models are mechanistic and under stable conditions this is the most efficient way to organize projects. If the project is stable, repetitive and possible to presuppose the Stage-Gate model is a good way to handle communication and coordination. When the project environment is both complex and uncertain the response is an organic and decentralized organization, which is not easily controlled with a Stage-Gate model.

Bruzelius and Skärvad (2011) discuss how information and uncertainty affects the development of organizations. Depending on how problematic information processes and information exchanges are, different forms of organizations are developed. Uncertainty can be seen as the difference between the information needed to perform a task, and the already existing information in the organization. In a mechanistic organization dealing with high uncertainty, new problems and tasks are moved up in the hierarchy to be solved, and to establish new rules, programs and processes for the future. An organization like this easily becomes slow and do not always have the capacity to deal with too much information. Organizations that have to deal with high

uncertainty need to have a more organic form of organization. The solution is to have decentralized decision making at the operational level. A demand on professional management and explicit objectives for each division, are necessities for this to work. The question is if it is possible to control these new organic organizations with mechanistic tools and models? (Gunnarsson et. al., 2000)

2.2 When to use Agile Methods

2.2.1 Different Development Styles for different Situations

All projects are different, there are no project that is exactly the same as another (MacCormack, Crandall, Henderson and Toft, 2012). This demands a project model that is flexible and that can be adapted to all different kinds of projects. One size does not fit all, different approaches and models for different situations are needed. Many managers does not even know that it is possible to use different project models in different situations, and even in different phases within the same project, depending on the situation.

One company that has found a framework that works for them is HP. Their framework helps the managers to choose the appropriate developing style depending on the business needs. By studying how HP has developed their framework, MacCormack et.al. (2012) have developed four steps to follow when developing this kind of framework.

The first step is to “define different development styles”. The main task in this phase is to define each business objective, depending on how the market is and the company’s position within this market. When the objective is defined, a set of development styles to fulfill each objective and business need has to be chosen.

The second step is to “define the criteria for style selection”. In this phase it is decided how the choice of development style is to be made and what criteria should be conclusive. MacCormack suggest that Agile methods are suitable when either or both the technical and the market risks are high. However, if both are low, a more traditional project model can be used.

Third is to “attack inner criteria when shifting styles”. When shifting styles towards a more emergent or agile style, managers can be a source of resistance since they feel like they lose control. Therefore it is important to really understand why a certain development style is preferred in a specific situation and what makes it a better choice in a certain situation.

The last step is to “manage the style portfolio”. Having many different development styles at the same company can be quite confusing. To find a way to manage these different styles is critical to succeed.

If this is going to work in a company, it is important that everyone agrees on this approach.

2.2.2 Planned Flexibility

In the beginning of a project many decisions are made (Verganti, 1999). Since the early decisions are the ones most unlikely to be changed, they will have the biggest influence

on the project. These decisions are sometimes difficult to make, since the needed information may not be available until further into the project. This is a dilemma that many companies experience, the one between anticipation and reaction.

Anticipation is, according to Verganti, “the capability to anticipate information into the early phase of product development”. The study has identified three categories of anticipation mechanisms. The first one is systemic learning, which is the ability to gain knowledge from earlier projects. The second is teamwork and communication, which is about involving all stakeholders and the importance of having shared objectives. The last is supported proactive thinking. This is about the use of, for example, screening checklists, target costing, failure mode and effect analysis, and early prototyping, in the beginning of the project.

Reaction is, according to Verganti, “the capability to introduce changes late in the process at low cost and time, to cope with unexpected constraints and opportunities”. Verganti lists five categories of reaction mechanisms. The first one is resource flexibility. The second is communication, which is a mechanism that facilitates and accelerates integrated problem solving, for example multi-functional teams. Third, overlapping development activities. The fourth one is redundancies, to have additional resources available to make the impact of late problems smaller. And last flexible solutions, which makes the affect of late product changes as small as possible. One example on a flexible solution is a modular product architecture.

In the study Verganti distinguishes four approaches to handle the dilemma of anticipation in the early phase in a project. To understand these approaches two different kinds of anticipation is discussed: the degree of detailed anticipation and the degree of selective anticipation. The degree of detailed anticipation is how many decisions that are made early in the project are frozen. The degree of selective anticipation is how many decisions that are only selectively anticipated early in the project, and frozen later. The first one is the detailed approach, for example the Stage-Gate model, with a high degree of detailed anticipation and a low degree of selective anticipation. The second is the selective approach, where the degree of detailed anticipations is low but the degree of selective anticipations is high. Third is the comprehensive approach, with a high degree of both detailed and selective anticipation. The fourth is the postponement approach with a low degree of both selective and detailed anticipation, it is in other words fully reactive.

In the study 18 companies with these different approaches were analyzed. The conclusion was that none of the approaches were better than another except for the postponement approach that showed no top performers. This means that it is not the choice of approach that is crucial for successful project, but the ability to handle the chosen approach.

The ability to handle late changes effectively in a project, depend therefore both on the reaction capabilities and the anticipation capabilities. This interplay leads to the concept of planned flexibility. The first step, when using planned flexibility, is to clearly identify critical areas in an early phase of the project. Second, is to plan reactions to handle these identified critical areas. The planned flexibility uses the same anticipation mechanisms that the detailed approach do, but instead of making decisions it is used to anticipate problems and planning reactions.

2.3 How to implement Agile Methods

In recent years, researchers in project management have studied the possibility to integrate Agile methods with the traditional Stage-Gate model (Cooper, 2014), (Karlsström and Runeson, 2005/2006), (Sommer, et.al., 2015).

With arguments like “future project will need both agility and discipline, which can be achieved by containing the Agile development methodology within the gate model” (Cooper, 2014), it does not mean that Agile and Stage-Gate have to be contradictory (Karlsström and Runeson, 2006).

We will summarize two studies of the implementation of the hybrid system, one by Karlström and Runeson (2005/2006) and one by Sommer et.al. (2015). Last, Cooper’s (2014) perspective of the hybrid process will be described.

2.3.1 Application of the Hybrid System

According to Sommer et al. (2015), the product development environment is getting more and more complex. The traditional Stage-Gate model cannot support the increasingly faster pace and the many changes that are typical for today's product development environment. The article argues that a hybrid, between the traditional Stage-Gate model and Agile methods, offers a more flexible approach to today’s organic environment. The study made by Sommer et al. compares seven different technology-intensive companies. Five of them have implemented Stage-Gate at the strategic level and Agile methods at the operational level, the other two have only improved their existing Stage-Gate models.

The results of the study was that those companies that only improved their Stage-Gate models experienced no significant change, while those companies that implemented a Stage-Gate/Agile hybrid experienced a significant change. Since the improvement of the Stage-Gate model did not contribute to any positive changes, the following section will only focus on the hybrids.

All five companies divided their planning and control into different levels. All five had one strategic level and one execution (operational) level, and four out of five companies also had a tactical level to connect the strategic level with the execution level. On the strategic level they used Stage-Gate, as a tool for senior management to monitor the project’s progress and budget changes. On the execution level, where the daily decisions are made, they used Agile methods within the development teams. Those four companies that had a tactical level used this for resource planning and knowledge sharing across project teams.

The tactical level is important because it links the strategic level to the execution level, in other words, it links the Stage-Gate model with the Agile methods. The company that did not have an explicit tactical level had to integrate the Agile methods directly with the Stage-Gate model. This meant that they had team members from their development team at senior management meetings and the other way around. The four companies that did have an explicit tactical level had structured this in different ways.

Two of the companies applied a value chain model on the tactical level. This was a large physical board that showed the companies’ value creation process across all departments. This was a way to support collaboration both within and between projects.

The employees on the two companies found the value chain model helpful in their daily work, it helped them coordinate both projects and tasks on a day-to-day basis. A manager at one of the companies also found the value chain model really helpful, he said, “By noon I know exactly what’s going on in the entire organization”.

The other two companies used work packages from PRINCE2¹. In the beginning of each stage they developed these work packages, which contains the deliverables for each stage. It was a great challenge for both companies to use these work packages as a tool at the tactical level. The work packages are static documents unlike the value chain model, which has a dynamic method to meet the many changed resource needs.

In all of the cases above the resistance for change in the project teams was low because the project managers wanted change, the change started from the bottom and up. After the hybrids were implemented, interviews with employees were made and they all experienced a positive change. From the study and these interviews a cross case analysis was made, to identify what caused these improvements. The analysis identified the following causes: increased knowledge sharing and communication, improved resource coordination, increased visibility and team empowerment.

The project managers involved in the studied companies also identified the main advantages and the main disadvantages with the hybrid system. The advantages were increased flexibility, improved communication and better fit between process and tools. The disadvantages were that projects still got delayed, a lack of fit between reward systems and methods, and a lack of agile culture in their organizations to support the implementation. The last disadvantage, a lack of agile culture in the organization to support the implementation, is an important area that often may be underestimated. It is not only the tools and methods that are important, but also the agile culture and awareness of the agile value set.

In the conclusion of the study, they identified the three most potential areas of improvement with help from the interviews. The first one was to use the agile tools in order to establish the agile values. The second was to realize and make the most of the effect of visualization on communication and knowledge sharing. Last potential area is to support and strengthen the employee motivation and productivity. However, further studies are needed to determine how these factors, together and in different combinations, affect the performance.

One final conclusion from the study is that the combination of the traditional Stage-Gate system and Agile methods creates a healthy tension between fixed planning and iterative problem solving.

2.3.2 Possible Effects of the Hybrid System

Karlström and Runeson (2005) did a study of ABB Automation, Ericsson Microwave Systems and Vodafone, which all are large global companies with a global market. For their product development they use the corporate Stage-Gate model. What differentiate

¹ PRINCE2 is a project management standard that was developed by the British Government and stands for Projects In Controlled Environments version 2. It includes tools for project management and one example is that it helps facilitating when applying work packages.

the three companies are their application domains, products, and markets. Due to higher-priority delivery deadlines, management in all three companies were convinced of the need for a change in their product development process, but they did not know what kind of change or how.

In ABB and Ericsson they already tried to use Agile methodology in limited parts of a software development project. The primary aim of the study was to collect experience data, to come up with suggestions for improvement of the project models. In the Vodafone case the aim was to identify problem areas within the existing Stage-Gate model where an Agile methodology could be a solution to the problem.

The finding of Agile features, positive effects and effects that might require attention, from the study, are described in table 1.

Findings summary for the effects of agile methods in three industry cases

Area	Agile feature	Effect*
Planning and prioritization	Most important feature first	+ Early feedback on features + No delays of important features
	Micro planning	+ Avoidance of requirements' cramming + Fixed plans avoided ! Little support for long-term plans
Communication and follow-up	Coherent teams	+ Good internal communication ! Potential isolation of agile team
	Automatic testing	+ Means for communication of change + Higher quality
	Small, manageable tasks	+ Feeling of being in control
	Continuous integration	+ Higher quality + Progress measure for management
Process model and roles	Customer involvement	+ Continuous feedback + Relevant features + Technical product manager is a good candidate as customer representative
	Documentation as tasks	+ Priorities resolved between documentation and code ! Conflicts visible between different amounts of documentation
Project management	Engineering-level empowerment	+ Engineers feel motivated ! Managers afraid initially ! Management training needed
	Focus	+ Engineers focus on past and current release, managers on current and future release + (!) Technical issues raised (too) early for management
	Engineering-level initiative	+ Little resistance to change

*+ Indicates positive effect, and ! indicates effects that might need attention.

Table 1. Findings from Karlström and Runeson's study at ABB, Ericsson and Vodafone (2005).

Karlström and Runeson conclusion is that the benefits with a hybrid between Agile methodology and the Stage-Gate model are that Agile methods are useful for micro planning while the Stage-Gate model is more useful when coordinating with other development teams and divisions.

The integration of Stage-Gate and Agile is entirely feasible according to Karlström and Runeson (2005). They also mention three guidelines that will facilitate the integration. First, the Agile development team has to interface with requirements from the gate model. Second, to avoid management's fear of the Agile methodology, the organization must handle this attitude up front. Last and third, the change has to be accepted within

the development team. Either the changes are initiated by the team, or they are made in small steps over a long period time.

At Ericsson, management felt a lack of control and they did not have the ability to add new features without sacrifice something else in a specific sprint. The development team on the other hand felt like they had more control and the quality of the project was significantly higher than in the rest of the organization. The same result of a higher quality was also achieved at ABB.

Findings show that in the Ericsson case, the gate model is less detailed than ABB's case. Therefore Ericsson's process is more flexible for the underlying technical process model.

One factor that made the integration between Agile and Stage-Gate possible was the isolation of the development teams. Karlsström and Runeson also lists some factors that are important for an effective integration:

- Involve developers early in the product development.
- To take both the micro planning and the macro project planning into consideration, when planning the project.
- Identify and make critical feedback loops as short and fast as possible.
- Strive towards having an early version of the product as quickly as possible, this will make exemplifying and communication simple.
- Make customer roles and interactions clear.
- Handle management attitudes towards the Agile methodology.

2.3.3 The triple A System

Cooper's idea for this new idea-to-launch system is called "The triple A system". This system include three wanted features:

Adaptive and flexible - With spiral or iterative product development the operative team can present something, not necessarily the final product, for the customer early. This makes it possible for the development process to be adaptive to new information and requirements. The activities in each stage are unique, deliverables for each gate are unique and processes for each project are unique, which makes the project flexible for each situation.

Agile - Elements of Agile development are included in the way Cooper describes the next-generation of product development. The Agile methodology ambition is less bureaucracy and more value adding activities, which allows the project to move faster with less waste.

Accelerated - The last A is to accelerate the project by ensuring sufficient resources and having dedicated team members for each project. It is more important to identify objectives, risks and uncertainties than to dictate on activities and stages.

2.4 When and how to use Agile Methods – A Summary

When the project environment is complex the need of a project model is necessary and the Stage-Gate model is one alternative. The benefits of using the Stage-Gate model as a project model is that it facilitates communication and coordination which leads to increased speed, better quality, greater discipline, and better overall performance for the project. It should not be forgotten that the Stage-Gate model is often associated with heavy documentation and fixed activities.

The Stage-Gate model suits a project environment that is stable, possible to presuppose and repetitive. When the project environment is uncertain, a different approach is necessary. The response to this kind of environment is an organic and decentralized organization.

MacCormack argue that it is important that the project model can adapt to different kinds of projects. He mentions four steps that will help to determine a framework to handle these different kinds of projects:

1. Define different development styles.
2. Define the criteria for style selection.
3. Attack inner criteria when shifting styles.
4. Manage the style portfolio.

It is normal, that project teams have to face the difficult task to make important decisions in an early phase of the project. This is because needed information may not be available in this early phase. The team can handle this difficulty by different degrees of anticipation and reaction. The study by Verganti (1999) showed that there is no specific degree of anticipation or reaction that makes a project successful. What is important is that the project team uses some degree of anticipation and reaction, and their ability to handle the chosen approach.

Verganti, lists three anticipation capabilities, which are:

- Systemic learning
- Teamwork and communication
- Support proactive thinking

- And five reaction capabilities:
- Resource flexibility
- Communication
- Overlapped development activities
- Redundancies
- Flexible solutions

A good way to interface anticipation and reaction is to use planned flexibility. The idea is to, in an early phase, identify critical areas and then make a reaction plan to handle these critical areas.

Therefore, when the needed information is not available in the beginning of the project, planned flexibility is a good approach to handle these uncertainties.

According to MacCormack (2012), when both the market- and/or the technical risks are high, an Agile approach is suitable for the project. Agile methodology can be used within product development. When using Agile methods, the product development is made through iterations of build, test, feedback and revise. These iteration leads to a product development that can handle late changes and new specifications, without affecting the overall project too much.

According to MacCormack (2001), a modular and scalable architecture is the key to handle these late changes when using Agile methods. This is something that Verganti also finds important when he discusses planned flexibility.

There are different authors that discusses the benefits of a hybrid between a Stage-Gate model and Agile methods. Cooper, the ardent author of many Stage-Gate articles and books, have in one of his latest articles argued that in future projects, there is a need of integrating Agile methods with the Stage-Gate model.

Sommer et. al. (2015) describes how the hybrid of Stage-Gate and Agile have been applied at different companies and in their development teams. Four of the companies that experienced a successful outcome of the hybrid project model introduced three different levels. These were one strategic level, one tactical level, and one execution level. The key to linking the Stage-Gate model, that was used at the strategic level, with Agile methods, that were used at the execution (operational) level, was the tactical level. One example on how the companies managed the tactical level, was with a Value chain model. This model showed the value creation process across all departments and helped coordination of projects and tasks on a daily basis.

Common between the project teams that were studied, and especially between ABB and Ericsson, was that resistance for change was low. The project manager that wanted the change was also satisfied by the result. The study also showed that there is a healthy tension between fixed planning and iterative problem solving.

The idea with the hybrid is that the Stage-Gate model is used for macro planning, while Agile methods are used for micro planning. Karlström and Runeson (2005) mention three guidelines that will facilitate the integration of a hybrid:

1. Agile development team has to interface with requirements from the gate model.
2. Handle management attitudes up front.
3. Change has to be accepted within the development teams.

A fourth attribute that Karlström and Runeson (2006) observed was an important factor for a successful hybrid was the isolation of the development teams since this allows the team members to work undisturbed. However, a negative aspect is that since the development team works isolated, they also appears isolated from the rest of the organization.

3. The current State at Delphi Automotive Torslanda

Delphi delivers safer, greener, and more connected solutions for the automotive industry. It is a high-technology company with technical centers, manufacturing sites, and customer support services in 44 countries. In Sweden, Delphi has two offices, both located in Gothenburg to be near to their customers. Information about Delphi and how they work in projects has been collected through interviews.

Delphi Torslanda designs and manufactures the electrical architecture for the automotive industry. The organization is a matrix organization where work is divided by both functions and projects. This thesis focuses on the work within projects and therefore the focus will be on the organization of projects. The projects at Delphi Torslanda are global, and both manufacturing and development are located in more than one country.

3.1 Concepts in the Project Organization

In the following section many contractions will occur. They are listed in table 2.

PM	Program Manager
TPL	Technical Project Leader
PE	Product Engineer
DE	Design Engineer
PDP	Product Development Process
KPM	Kaizen Project Management
KPI	Key Performance Indicators
FLM	Flawless Launch Metrics

The Program Manager (PM) is in charge of the project from the beginning of the product development process, until 90 days after the launch. They are therefore responsible for R&D, logistics, manufacturing and more. They describe their role as coaching and leading with a helicopter-view, and they try to avoid a detail focus.

For each project there are around 8-10 level 1 owners. These are people responsible for different areas, for example sales, finance, manufacturing and development. In this study the Technical Project Leader (TPL) is the most important level 1 owner. The TPL is in charge of the development part of the project.

Then there are the engineering teams that actually develop the product. They are arranged in three different core teams: Floor, IP and Underbody. The floor team is in charge of the electrical architecture in the floor area. The IP team is in charge of the electrical architecture around the dashboard. The underbody team is in charge of the electrical architecture in various parts remaining in the car, for example the ceiling. In each core-team there is a mixture of Product Engineers (PE) and Design Engineers (DE) that works close together in the R&D process.

There are also two functional managers, one for the PEs and one for the DEs. They are in charge of the engineers in the core teams and their main task is to make sure that the right resources, with the right competence, are available for the projects.

Delphi has its own version of the Stage-Gate model called Product Development Process (PDP). Their PDP follows a standard type of a Stage-Gate process. However, it is relatively detailed and specific for Delphi since it contains many years of their aggregated experience (Hedberg, 2016). This has resulted in a best practice for Delphi.

This model is mainly used by the PMs and the TPLs since they have the main responsibility towards Delphi. However, the TPLs do not use it as frequently as the PMs. The interviewees are of the opinion that the PDP tells the PMs what to do, but not how.

The purpose of the gate reviews in the PDP is to show management the progress and status of the project. The PMs do not have suitable information to give management an overview without focusing on details. Therefore the PM makes a new document especially for the gate review more suited for the purpose. This document shows the finance, progress, the next step etc.

The company has a way of categorizing their projects. In the beginning the projects are given a categorization which can be either AA, A, B, C or X. The PDP is only officially used for AA, A and B projects. In the smaller projects they are not bound to use the PDP and if it is used, it is only evaluated on a local level. When deciding which categorization a project is given, different aspects are taken into consideration such as turnover, investments, product and process technology, and if it is a new customer.

Delphi has many different types of projects. It can be platform projects, new model projects, facelifts or yearly model updates. Lately they have considered another way of dividing their projects, and this by high voltage projects, low voltage projects, commercial vehicles or autonomous cars. The high voltage projects, which mainly concerns electric vehicles, operates on a new market and therefore consists of many uncertainties. The same applies to the autonomous cars, they operate on a new market with many uncertainties involved. Delphi has more experience with the low voltage and commercial vehicles projects, and therefore they do not contain the same kind of uncertainties as the high voltage and autonomous cars projects.

A tool that has helped Delphi to create structure is the Kaizen Project Management (KPM). This tells the PM how something should be done (Hedberg, 2016). The KPM is an Excel file that consists of Key Performance Indicators (KPI). These KPIs visualises the progress of the project when being reported. This is a way to keep track of the project and detect possible problems early on. Responsible of reporting the progresses in the KPM file are the PM and the TPL. The file has lines with tasks, and to each task there is a box where the status of the task is reported. The status is described in percent and depending on the status, the box is either red or green.

To determine how well the project has been launched, Delphi has something they call Flawless Launch Metrics (FLM). Everything they do somehow ends up in these metrics

that are measured in the end of the project. However, there is no clear connection between the FLM and the KPM.

The normal state at Delphi is characterized by many late changes. Almost all parts in the car are connected to the electrical architecture, which means that most changes in the car affects Delphi's product.

3.2 Interviews

Below is a summary of the information collected through the interviews. The questionnaire can be found in appendix 1. This chapter is divided into three parts. First a general summary, second a summary of experienced problems, third, and last, the interviewees' thoughts about Agile methods.

3.2.1 The Interviewees Description of the current State at Delphi

A general summary of the interviews divided by position.

3.2.1.1 Program Manager

The Program Manager is in charge of collecting the data needed for each gate, and present this at the gate review. The Program Manager's role is also to do a follow-up on the KPIs and, when needed, make sure that there is an action plan to get back on track. According to the Program Managers the PDP model is being followed and they do not deviate from it.

What kind of categorization a project has, does not affect how the Program Manager works with the PDP. The only difference is that in C projects or smaller projects the PDP is not officially used. It is also more of a global focus and extra reviews in the AA projects, than in the others. In some projects that are based on previous projects, not all stages have to be passed. This because the first stages are the same as in the previous project, therefore only the stages that differ from the previous project has to be passed.

Each project has project meetings where they discuss problems, progress and status of the project. The goal is to have these meetings once a week, but usually they only have them every second week. The participants on these meetings are the Program Manager and the level 1 owners. It is difficult to get everyone to attend the meetings due to shared resources between projects, but the attendance rate is usually around 80 percent.

The Program Manager usually has the authority to decide and prioritize without interventions from a higher level in the organization. However, sometimes they need to consult with the functional managers in order to get the right prioritization of resources. The team usually has the authority to prioritize by themselves and they also have the knowledge to do so in most cases. But the ability to prioritize is often related to experience. The less experienced can easily get caught in the trap where the loudest problem gets the highest prioritization.

3.2.1.2 Technical Project Leader

The Technical Project Leaders do not work close to the PDP and therefore do not know much about it more than the basics. The Program Manager tells them when they need data or other input for a gate review, and they make sure the Program Managers get what they need.

Depending on what categorization the project has been given, the number of reviews are different. An AA project has more reviews and follow-ups than a B project.

The Technical Project Leaders feel that they have a good overview of what has to be done, but not what has to be done in the PDP. They only get in contact with the PDP through the Program Managers.

They also feel that they have a good overview of what happens in their team. They have project and customer meetings once a week and daily contact with their team members if any problems would arise. Since the different projects share resources it can be difficult to get perfect attendance at the project and customer meetings.

The Technical Project Leaders prioritize resources between each other in the projects and they prioritize their own tasks. The teams also prioritize their own tasks with control from the Technical Project Leaders who tells them what has to be done and when.

3.2.1.3 Functional Manager

The functional managers do not have experience of the PDP model. Their job is to make sure that the needed resources are available for the projects. They have to be aware of the PDP, but it is mainly the Technical Project Leaders and Program Managers who are responsible for the projects. The Functional Managers have their own processes and checklist to handle the resource allocation. They keep more track of the customer's project model than Delphis project model, PDP.

The planning of resources, that is one of the tasks of the design engineers' Functional Manager, is sometimes based on planning that was made two or three years ago. The Functional Managers have weekly meetings with the design and product engineers to coordinate resource allocation and if there is need to relocate resources between the core teams. If there is a need of extra resources they carefully consider their options to get the best compromise. Is it worth to relocate staff and spend time on teaching, or can the team solve the task in a shorter time themselves.

3.2.1.4 Core Team

As mentioned before, the core teams consist of product engineers and design engineers. Product engineers are responsible for construction, structure of part numbers, and communication towards the factories etcetera. The design engineers are the ones working in CAD actually drawing the wire harnesses. Product and design engineers are working close together and according to the core teams, it is not unusual that their assignments overlap each other.

Delphi's project model (PDP) is unfamiliar to the engineers in the core teams, however they are more familiar with the customer's project model. PDP is something that they have heard of but they know very little about it. They are told by the Technical Project Leaders when data or other input for gate reviews are needed.

Categorization of projects does not affect the way team members in the core-team work. Each task is equally important as another, and they cannot tell the customer that something else is more important and therefore should be prioritized higher than them.

Core team members know the connection between activities and in what project phase it should be done. Program Managers and Technical Project Leaders make sure that the project is proceeding as planned.

Both core teams that were interviewed have core meetings every, or every other week, where the team members are updated about the other team members' work and problems they have encountered. This, combined with the fact that the team members sit together in the same room, makes it possible to get an overview of all team members' progress problems. The core team responsible for the wire harnesses in the floor area, have recently started with everyday meetings in front of a whiteboard.

Feedback is mostly received through customers, but also through engineering manager, program manager, technical project leader and functional manager.

The core teams generally have the authority to prioritize their own work as long as they manage the tasks on time. The core team has the knowledge to prioritize their work and the individual get the knowledge to do prioritizations as the experience grows.

3.2.2 The Interviewees View of Problems of the current State

During the interviews some problems experienced by the interviewees came up. Those are listed below, divided by position.

3.2.2.1 Program Manager

- In general only the PMs and TPLs have knowledge and understanding of the PDP.
- The categorization is made before the project starts and it keeps this categorization throughout the whole project. If the conditions are changed another categorization might be more effective.
- Almost only the PM:s take the categorization into consideration in their daily work.
- The PDP and KPM are reactive instead of proactive. Problems often surface at the project meetings so the PDP and KPM becomes more of a summary, and do not help to identify the current problems. Therefore they are both more reactive than proactive.
- Something they feel is missing is a real consequence if they pass the gate or not. Today they often just get to do a re-review at the next gate, but no real consequence.
- The PDP does not give a good overview, in order to get that they have to focus on the details.
- KPI and KPM do not give the project members feedback or other input, so it is difficult to get them to work actively with it.
- The delivery of different tasks, especially between locations. The roles are not always clear for everyone and everywhere. Sometimes someone think they have handed over a task and given the needed information, but the receiver has not apprehended the situation in the same way.
- Unclear roles.
- When it comes to the PDP, it is all or nothing. They do not have a version of the PDP with less stages for smaller projects.

- The KPM file is difficult to understand and to update.
- The communication within the team is insufficient, especially between different locations.
- A lot of documentation that is not always necessary for the proceeding of the project.

3.2.2.2 Technical Project Leader

- PDP and KPM are insufficiently explained and general. They do not have anyone to ask easily, if they are unsure or something is unclear. The person they need to ask is often someone a lot further up in the hierarchy. Therefore they only ask if the question is absolute necessary.
- The handover of work is insufficient, it happens that tasks and information falls between the functions. However, this depends on how experienced the team members are. The technical project leader do not focus on the details, they trust their team in that matter.
- The teams are fixed within their areas, floor, underbody and IP. If someone is sick, it could delay everything. The technical project leaders believe that if they were more Agile, they would be able to be more flexible and move resources when needed.

3.2.2.3 Functional Manager

- The project is ended 90 days after the launch. Due to this there is no additional focus or budget, for the continuing delivery, if needed.
- There is a lack of transparency between project and functional work. Trying to prevent this, they have jointly meetings between the functions and the projects.
- Tasks are sometimes forgotten and they need to remind the core teams close to deadlines.

3.2.2.4 Core Team

- New tasks can sometimes be forgotten and lost, and they think that they need more structure.
- Lack of processes increases the need of firefighting. If they want to change a process or create a new one, it is slow and heavy.
- The Product Change Request (PCR-flow) takes longer than what has been agreed on with the customer.

3.2.3 Agile

Questions about Agile methods were asked during the interviews. The questions were to answer whether the interviewees had any experience of Agile methods, and their thoughts about it.

3.2.3.1 Program Manager

The Program Managers know about the Agile methods in theory, but they do not know how they can apply them in their daily work. One of them thinks that Agile methods can be great in the development teams where the team members meet face to face, but not virtually across the whole project. They also think that it can work in the factories

where people meet face to face, but not on the more overall level where the Program Managers have their focus.

3.2.3.2 Technical Project Leader

The technical project leaders have heard of Agile methods but they do not really know what they mean. They have heard more of Scrum, which is a specific Agile method, but do not have any experience of it. They all understand that in order to survive in today's increasingly faster pace they have to find a new way to work and Agile methods might be one answer. Every decision cannot be sent up to a higher level in the organization, some decisions and prioritizations have to be made in the development teams. They all believe that they might work more Agile than they know, but they do not have an explicit Agile method of working.

They also believe that Agile methods, and scrum in particular, are applicable at Delphi. They think that the methods will work in the development teams, their core teams. As mentioned before, the core teams consist of a mixture between Product Engineers and Design Engineers. The Technical Project Leaders do not know if it is possible for the product engineers to work with Agile methods since they are responsible of a specific car model and very conversant in that specific project. But they believe that it will be easier for the design engineers to work in a more Agile way because they are more flexible and it is easier for them to do tasks from different projects.

They think that they could work completely Agile within the investigations. This would also broaden their range of skills, and make them even more flexible in the future.

Another positive aspect of the Agile methods according to the technical project leaders, scrum in particular, is the visualization. If they were to visualize everything it would be easier to get a good overview, both over the resources, and the finished and waiting tasks.

Finally, the technical project leaders feel that they do not have the knowledge they might need about Agile methods. Agile is a modern word they should know the meaning of, but they have not gotten any education about it. They feel that they need more knowledge and tangible proposals on how they can use Agile methods at Delphi.

3.2.3.3 Functional Manager

Both the Functional Managers have limited knowledge of Agile, which they mainly have gotten through workshops and educations. One of them has also tried to figure out how to assimilate Agile to their work. They have the opinion that the core teams work in a more Agile way now than a few years ago, since they do have more visual planning.

Both Functional Managers think that Agile methods would help with resource allocation and integration between teams. This would help the team to solve the task the in best way possible. If someone is sick or the workload is uneven, Agile methods would help them be more flexible. By visualizing tasks and resources problems can be prevented.

Since the core teams are so dependent on other divisions in the company, there might be a problem with working Agile all the way. However, they want to strive towards the mindset of Agile methods.

They have seen successful cases when working with Agile methods. Problems have been solved efficiently by isolating the people involved until they have found a solution to the problem. It would take much longer time if the problem had been to be solved in the normal way, by passing the problem between people.

3.2.3.4 Core Team

In the core teams they have heard of Agile methods, but they do not have any experience of working with it. They have the opinion that, since they partially can choose how to work, Agile methods can be a way of working. There is still a problem with the fact that work cannot be divided between everyone in the team, because the need of specific competence. They think that it might be easier to work with Agile methods for the design engineers.

Resource management has previously been a problem. The floor team has recently, as mentioned before, started with everyday meetings and to visualize tasks and resources on a whiteboard. The purpose is to easily get an overview of the resources and tasks. The assignments on the whiteboard are broken down into smaller work packages, because they do not have the flexibility to divide all work into too small tasks.

3.3 Summary and Conclusions of the Interviews

Delphi's core teams, which are the development teams, consist of a mixture of different kinds of engineers. In each team there are both Design Engineers and Product Engineers.

Delphi's project model is based on a Stage-Gate model and is mainly used by the Program Manager (PM) and the Technical Project Leader (TPL). The core teams do not have large knowledge about the PDP. They know some basics, but are more familiar with the customer's project model since they work closer with it. According to the PMs and TPLs, a problem with the PDP is that it only gives input to reactive actions and not proactive actions.

According to many of the interviews, on all different levels, what type of project it is does not affect the development style or how they do their work. However, something they do notice is that what kind of categorization affects the amount of documentation and reports.

Today Delphi categorizes their projects based on turnover, investments, product and process technology, and if it is a new customer. Depending on these factors, the project will be given a categorization. The category can be AA, A, B, C or X where AA is the project with the most focus and X the project with the least focus. However, right now they are thinking about another kind of categorization because they see that it might be needed. One thought is to divide the projects into the categories of High Voltage, Low Voltage, Commercial Vehicles and Autonomous Cars. The high voltage and autonomous cars both operates on a new market with new technologies, which brings many uncertainties. The low voltage and commercial vehicles do not face the same

kind of uncertainties since they operate on an established market with more familiar technologies. However, these two kinds of projects also face uncertainties. These uncertainties are more connected to changes in the specification from the customer than uncertainties in market and technology.

During the interviews two problems connected to the categorization was mentioned. First, the categorization is not changed even if the circumstances do. The second problem is that Delphi does not, explicitly, have different versions of the PDP to suit different kinds of projects.

It is the PMs and TPLs responsibility to make sure that the project is proceeding as planned. The functional managers' responsibility is to coordinate the resources, and when needed, relocation of resources. It does not seem like it is often that resources are relocated for a shorter time, since it often takes more time to teach a new person the job than doing the job themselves. The core team has the authority to prioritize their own work and tasks, as long as they are finished in time.

A problem that was mentioned during the interview was that the teams are relatively fixed within their areas. As mentioned before, it is not often resources are relocated for a shorter time. Another problem that was mentioned is that there is a lot of firefighting, which takes time from other tasks. This is often caused by the many late changes. One more problem that was mentioned was the amount of documentation that is not directly connected to creating value for the customer.

Generally the interviewees had limited knowledge and experience of Agile methods. However, the interviewees were positive to Agile methods. They think Agile methods might be a solution to some of the problems they experience, but they do not know how and when to use Agile methods.

4. Discussion and Conclusion

The following chapter will contain a discussion based on the theory and empirics in the report. After the discussion there will be a conclusion of the report.

4.1 Discussion

The discussion is divided by the research questions. The first part discusses research question 1, when are Agile methods appropriate to use at Delphi. The second part discusses research question 2, how to implement Agile methods at Delphi.

4.1.1 When are Agile Methods appropriate to use at Delphi

Today Delphi have their own version of a Stage-Gate model, the PDP. The Stage-Gate model is suitable when the project environment is stable, possible to presuppose and repetitive. A stable environment is not something that describes Delphi's product development environment. Their daily work is affected of the many changes from their customers. To handle these changes, a lot of firefighting has to be done to solve the problems that arise. In this kind of environment, with high complexity and uncertainty, a different approach is necessary.

Uncertainty is the difference of what you need to know and what you do know. MacCormack (2012) discusses uncertainty in markets and risks. These uncertainties can be connected to how new a product is. If the market is uncertain it can be because it is a new market, you do not know how the product you are developing will be received at the market. If the technology is uncertain it could be because it is a new technology, you do not know how the finished product is going to turn out because it is something completely new.

MacCormack (2012) states that when both the market uncertainties and the technical uncertainties are high, Agile methods are a good way of handling the product development in this situation. Since Agile methods can handle late changes in a good way, it is also a good approach to handle these uncertain situations. The key to handling late changes, without affecting the overall project too much, is a scalable and modular architecture when using Agile methods.

Uncertainties do not have to be connected to risks and markets. There can also be uncertainties because of changes in the specification from the customer. These uncertainties are not the same kind as the ones connected to risks and markets, these are changes that the development team are used to handle.

Verganti discusses that the uncertainties that arises because of changes in the specification can be handled through something he calls planned flexibility. Planned flexibility can be divided into anticipation and reaction. He argues that changes in specification can to some degree be anticipated, and to help anticipate these changes there are three mechanisms. One of these, teamwork and communication, we can see at Delphi. This mechanism is about involving stakeholders early and to share the same objectives. Since Delphi works close to, and are dependent on, their customer, they have to involve that stakeholder early and make sure that they share the same objectives.

Verganti also argues that a company can have reaction capabilities, to handle changes. There are five different reaction mechanisms. One of them, communication, which is

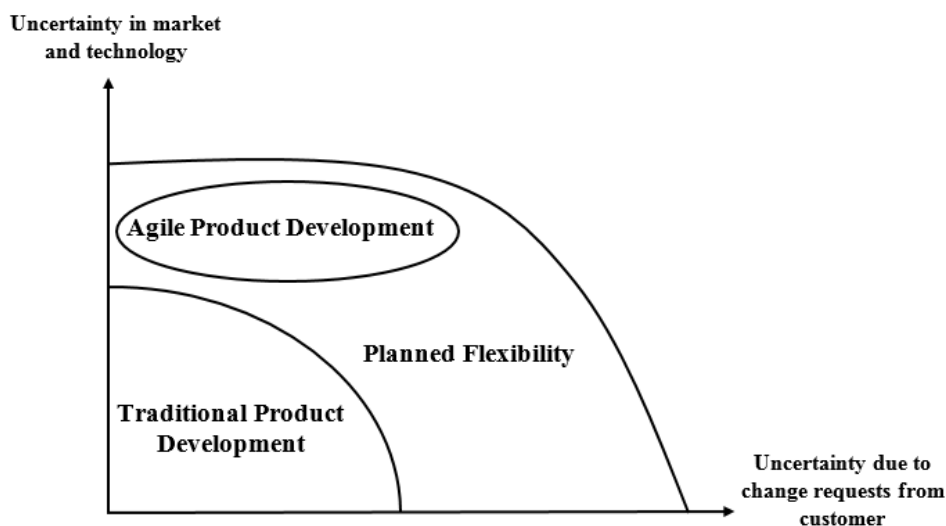
about working in multifunctional teams to improve the communication, Delphi do have. Their core teams are multifunctional and within the core team the communication is good. It can be speculated that the communication within the core team is better than in other parts of the project since they are physically in the same place. However, one important mechanism they do not have is flexible solutions. This mechanism advocates a modular product architecture to minimize the effect of late changes.

The combination of these anticipation and reaction capabilities, Verganti calls planned flexibility. It is not enough to only have the reaction capabilities, it is also important to have the anticipation capabilities. First you anticipate problem areas, and then you plan for a reaction in these areas.

We think that Agile methods might be a good way for Delphi to handle their increasing amount of changes, but also planned flexibility could be a solution. However, as MacCormack states, one size does not fit all. No project is exactly the same, and therefore one way of working will not fit all different kinds of projects.

Today Delphi categorizes their projects based on investments, turnover etc. But we think they might need one more kind of categorization, based on what kind of uncertainties there is. We believe so, because it is the uncertainties that affect what kind of development style that is suited. One example of that is that MacCormack says, if both the market and the technical uncertainties are high, Agile product development is a good way to handle this.

MacCormack also mentions four steps to find a framework to handle different kinds of development styles. The first step is to define different development styles. So far we have defined two different styles, Agile methods and planned flexibility. However, more development styles can be defined. The second step he mentions is to define the criteria for style selection. For this we have summarized our conclusions in a model we think might be a good way to do this.

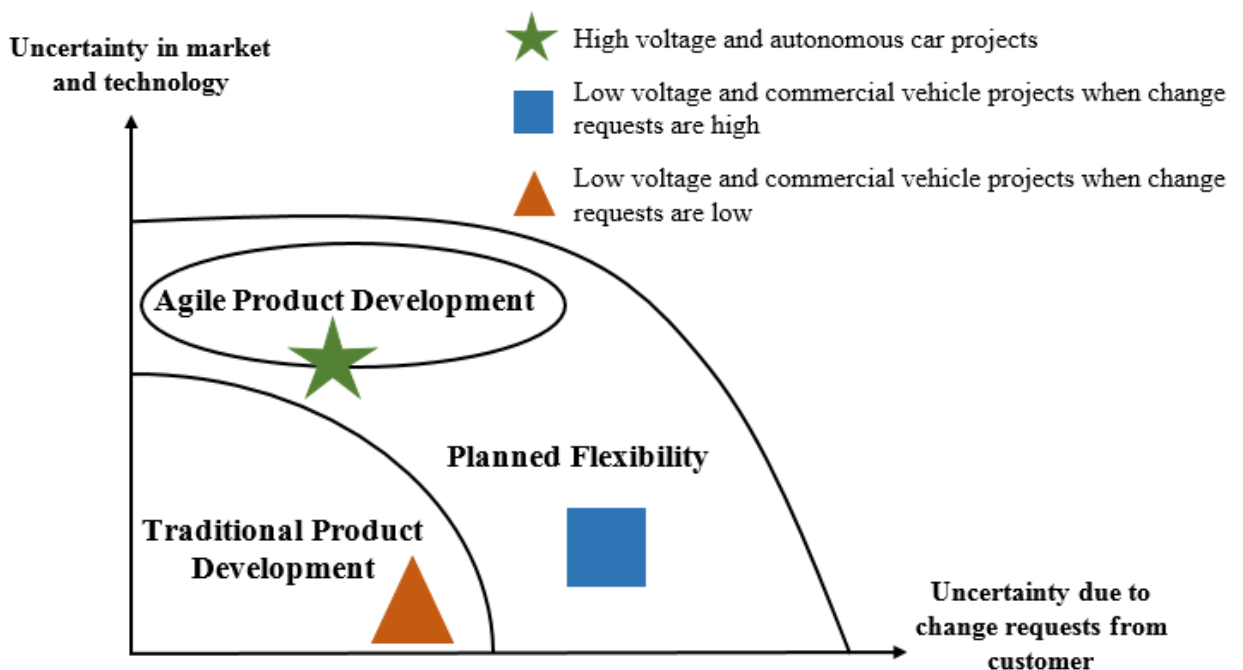


As you can see Y-axis defines the first kind of uncertainty, the one connected to the market and technical uncertainty. If this is high, Agile is a good way of handling those kinds of projects. The X-axis defines the other kind of uncertainty, which is, changes in

specification because of customer requests but with known solutions. If this is high, planned flexibility can be a good way to handle those kinds of projects. If both if the uncertainties are low, a more traditional product development is appropriate, such as a Stage-Gate model.

We can also see that many of the capabilities that is connected to planned flexibility, can facilitate the iterative work in Agile product development. One example of this is the modular architecture, which is connected to one of the reaction capabilities when using planned flexibility. Therefore we think that planned flexibility can facilitate the handling of any kind of uncertainties in a project. Even in those situations when Agile methods are most appropriate, it can help to use some degree of planned flexibility.

As mentioned before, one way of categorizing Delphi's projects can be in high voltage, autonomous cars, low voltage and commercial vehicles projects. These projects could, based on their different kinds of uncertainties, be placed in the model.



This is only a start and we believe that this model can be improved with more development styles and a more precise way of choosing development styles.

4.1.2 How to implement Agile Methods at Delphi, and what Prerequisites they have to do this.

As shown above, we suggest the use of Agile methods when the market and risk uncertainties are relatively high.

ABB and Ericsson implemented Agile methods in their work with projects because they saw a need of change. They had also used Agile methods in limited parts in the

organizations. This same situation can be found at Delphi. The development teams are positive to Agile methods and think it could be an alternative way of working.

We think that the core teams can work with Agile methods since their work is not bound to the PDP. They also have the authority to prioritize and organize their own work as long as they manage their tasks in time.

Delphi has to be able to handle a complex product development environment. They therefore cannot abandon their Stage-Gate model, the PDP. They also need the PDP to give higher management an overview and control of the project.

According to Sommer, a hybrid between the Stage-Gate model and Agile methods is a good solution to handle today's organic environment. A hybrid can both give management the control they need, and allow the development team to work more Agile.

So how should the hybrid be applied? Karlström and Runeson, and Sommer et.al. have come to the same conclusion: the Stage-Gate model and Agile methods should operate at different levels.

Karlström and Runeson suggest that the Stage-Gate model should be used for macro planning and Agile methods for micro planning. Sommer et.al. mention that the hybrid should distinguish between three different levels, the strategic level, the tactical level and the execution level. In Sommer et.al.'s description, the key is the tactical level, which purpose is to link the Stage-Gate model with Agile product development.

It has been found that Delphi already uses the PDP at a strategic level. It is also shown that they have a clear execution level, where the development team operates. However, Delphi does not seem to have a tactical level. A defined strategy of the tactical level could facilitate the linkage between the PDP and Agile methods, when they are used by the core teams.

In other organizations a Value chain model have been used as a way to establish a tactical level. The purpose is to, on a physical board, show the value creation process across all departments. This helps to coordinate projects and tasks. Something similar to this we think might be a good way to establish a tactical level at Delphi. However, exactly how this tactical level will be applied, is best decided by the company itself.

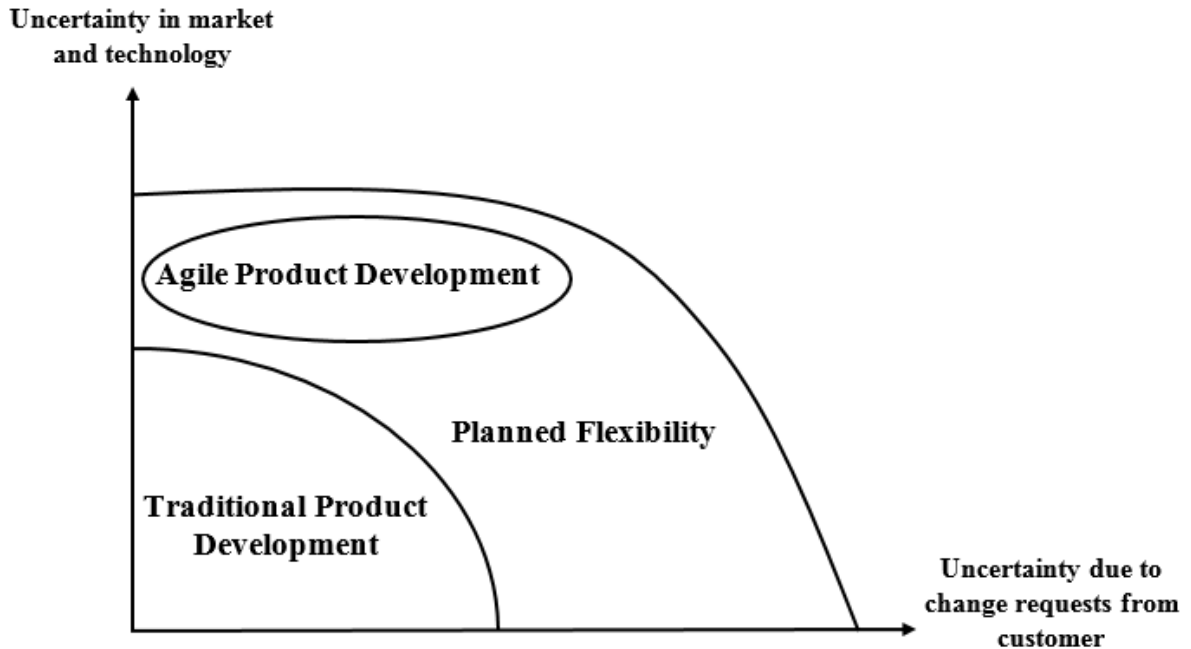
Our suggestion to a tactical level at Delphi is that it should contain elements from both the strategic level and the operative level. At Delphi, the TPLs can represent the strategic level and the core teams can represent the operative level. However, a few members from the core teams can represent the core teams since it might not be possible for everyone to be involved at the tactical level. One way of handling this tactical level can be to have weekly or daily meetings between these representatives to link the strategic decisions to the operative work. This could help with the prioritization of both resources and tasks.

We think that Delphi has the right conditions to integrate Agile methods into their current product development. Karlström and Runeson (2005) mention three guidelines to facilitate the integration of a hybrid. Of these three we can see that Delphi already

fulfills one of them, which is that change has to be accepted within the development teams. In this report we have tried to give them a start with another one, to interface Agile development with the Stage-Gate model. That only leaves to handle management attitudes up front, which is something that, if needed, has to be done.

4.2 Conclusion

To answer the first research question of the report, when Agile methods are appropriate at Delphi, we have summarized our conclusions in a model. This model gives a hint on when Agile methods are suitable, and when they may not have to be used.



To the second research question of the report, how Agile methods can be used at Delphi, we think that the answer is to use the Stage-Gate model and Agile methods on different levels where the tactical level is the key to success. In the discussion chapter it has been speculated how the tactical level can be applied at Delphi, however exactly how this should be applied is still unknown.

At Delphi the majority of the interviewees felt that did not have enough knowledge about Agile methods. We think that knowledge is a necessity when integrating something new into a company. Finally, we believe that a pilot project is a good way to start when making a change. Therefore we think that Delphi should start to integrate this alternative way of working in one development team first, before doing so throughout the whole product development process.

We believe that our conclusions are a step in the right direction to handle the future environment, with an even higher pace and an increasing amount of changes. We hope that this report will be a helpful guideline for Delphi to manage this future environment they are facing.

References

Agile manifesto. (2001). Manifesto for Agile Software Development. Retrieved 2016-04-20, from <http://www.agilemanifesto.org/history.html>

Agile methodology. (2008). The Agile Movement. Retrieved 2016-05-26 from <http://agilemethodology.org>

Begel, A. and Nagappan, N. (2007). Usage and perceptions of agile software development in an industrial context: An exploratory study. *First International Symposium on Empirical Software Engineering and Measurement*.

Bruzelius, L.H. and Skärvad, P-H. (2011). Integrerad organisationslära. Studentlitteratur.

Cooper, R. G. (2009). How companies are reinventing their idea – to – launch methodologies. *Research Technology Management, volume 52, page 47-57*.

Cooper, R. G. and Edgett, S. J. (2012). Best practice in the idea-to-launch process and its governance. *Research Technology Management, volume 55, page 43-54*.

Cooper, R.G. (2014). What´s next? After Stage-Gate: progressive companies are developing a new generation of idea-to-launch processes. *Research Technology Management, volume 57, page 20-31*.

Delphi Automotive. (2016). Automotive Industry. Retrieved 16-05-24, from <http://www.delphi.com/manufacturers/auto/>

Gunnarson, S., Linde, A., and Loid, D. (2000). Is standardization applicable to project managers of multi-project companies? *Proceedings of IRNOP IV Conference "Paradoxes of Project Collaboration in the Global Economy: Interdependence, Complexity and Ambiguity", 9-12 January 2000, Sydney, Australia*

Hedberg, M. (2016). PM at Delphi Automotive Torslanda, interview.

Henriksen, B and Andersen, B. (2010). Is there a tactical level of business processes?. *The TQM Journal, volume 22, page 516-528*.

Karlström, D. and Runeson, P. (2005). Combining agile methods with stage-gate project management. *IEEE Software, volume 22, page 43-49*.

Karlström, D. and Runeson, P. (2006). Integrating agile software development into stage-gate managed product development. *Empirical Software Engineering: an International Journal, volume 11, page 203*.

MacCormack, A. (2001). Product-Development Practices That Work: How Internet Companies Build Software. *MIT Sloan Management Review, volume 42, page 75-84*.

MacCormack, A., Crandall, W., Henderson, P., and Toft, P., (2012). Do You Need a New Product-Development Strategy? Aligning Process With Context. *Research Technology Management, volume 55, page 34-43*.

Räisänen, C. and Linde, A. (2004). Technologizing discourse to standardize projects in multi-project organizations: Hegemony by consensus?. *Organization*, volume 11, page 101-121.

Schwaber, and Sutherland, (2013). The Scrum Guide.
<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100>

Sommer, A. F., Hedegaard, C., Dukovska-Popovska, I., and Steger-Jensen, K., (2015). Improved Product Development Performance through Agile/Stage-Gate Hybrids: The Next-Generation Stage-Gate Process?. *Research Technology management*, volume 58, page 34-44.

Appendix 1 - Interview Questions

The purpose of the interviews is to get an overview of the following three areas:

1. The current situation.
2. Thoughts and ideas from employees about the future.
3. Experience and thoughts about Agile methods.

Questions:

- What is your position and what are your work tasks?
- What is your relationship to the project model (PDP)?
 - How well do you follow it?
 - How often do you use the PDP?
 - What is your general opinion about the PDP?
- Do the categorization of the project affect your work with the project?
- Do you feel that you have an overview of what should be done? Can you see the connection between the different activities?
- Do you have an overview of what your team members are working on, and if they have any problems?
- How does the handover of tasks work? For example, between design- and product engineers?
- Do you feel that you and/or the team have the mandate to prioritize and make decisions by yourselves?
- Do you feel that you and/or the team have enough knowledge to prioritize and make decisions by yourselves?
- Have you heard of Agile methods? Do you have any experience of it?
- Do you think it is possible to use Agile methods at Delphi?
- Do you have anything else that you want to share?