

IMPROVING PERFORMANCE, QUALITY
AND HAPPINESS OF SOFTWARE
DEVELOPMENT TEAM

Agile and Lean Approach

Heikki-Pekka Noronen

Master's Thesis

5 2014

Master's Degree programme in Information Technology



JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES



Author(s) Noronen, Heikki-Pekka	Type of publication Master's Thesis	Date 28.4.2014
	Pages 139	Language English
		Permission for web publication (X)
Title IMPROVING PERFORMANCE, QUALITY AND HAPPINESS OF SOFTWARE DEVELOPMENT TEAM: AGILE AND LEAN APPROACH		
Degree Programme Master's degree programme in information technology		
Tutor(s) Rintamäki, Marko Huotari, Jouni		
Assigned by Tieto Healthcare & Welfare Oy		
Abstract The main goal for the thesis was to improve the performance and produced quality of two software development teams, while maintaining a high level of employee happiness. The improvement was mainly planned to be done through introduction of some Agile and Lean software development methods and practices. Over the period of two years, a significant number of changes were implemented with the main focus being on Scrum and Kanban method based practices; however, the changes did not limit only to those two methods. Both of the teams ended up having some differences with the practices that they found most beneficial for them, however, they also had many similar practices. Going through these practices offers insight into a possible core set of practices that could be beneficial in the majority of the teams. The research was implemented as action research that was supported by survey research. The results of the research were highly positive and there was a clear improvement in all three factors that were studied : performance, quality and happiness during the time of the research		
Keywords Agile software development, Lean software development, performance, quality, happiness		
Miscellaneous		



Tekijä(t) Noronen, Heikki-Pekka	Julkaisun laji Opinnäytetyö, ylempi ammattikorkeakoulututkinto	Päivämäärä 28.4.2014
	Sivumäärä 139	Julkaisun kieli Englanti
		Verkojulkaisulupa myönnetty (X)
Työn nimi IMPROVING PERFORMANCE, QUALITY AND HAPPINESS OF SOFTWARE DEVELOPMENT TEAM: AGILE AND LEAN APPROACH		
Koulutusohjelma Master's degree programme in Information Technology		
Työn ohjaaja(t) Rintamäki, Marko Huotari, Jouni		
Toimeksiantaja(t) Tieto Healthcare & Welfare Oy		
Tiivistelmä Opinnäytetyön tavoiteena oli parantaa kahden ohjelmistokehitystiimin tehokkuutta ja tuotettua laatua säilyttäen samanaikaisesti korkean onnellisuustason. Parannukset suunniteltiin tehtäväksi pääasiassa ketterien ja Leanin ohjelmistokehityksen mukaisten ohjelmistokehityskäytäntöjen avulla. Kahden vuoden tutkimusjakson aikana vietiin lävitse suuri määrä muutoksia. Muutoksissa pääpainotus oli Scrumiin ja Kanban metodiin pohjautuvilla käytännöillä. Muutokset eivät rajoittuneet kuitenkaan vain kahteen edellä mainittuun metodiin. Tutkimuksen aikana tiimien valitsemiin käytäntöihin tuli jossain määrin eroavaisuuksia, mutta löydettävissä oli myös ydinjoukko käytäntöjä, jotka molemmat tiimit kokivat erittäin hyödyllisiksi. Näiden käytäntöjen tarkempi tutkiminen antaa näkemystä mahdolliseen joukkoon käytäntöjä, jotka voisivat olla hyödyllisiä suurimmalle osalle ohjelmistonkehitystiimeistä. Tutkimus toteutettiin toimintatutkimuksena, jota täydennettiin kyselytutkimuksella. Opinnäytetyön tulokset olivat erittäin positiivisia ja selkeää kehitystä oli työn aikana havaittavissa niin tehokkuudessa, laadussa kuin onnellisuudessaakin.		
Avainsanat (asiasanat) Ketterä ohjelmistonkehitys, tiimit, onnellisuus, tehokkuus, laatu		
Muut tiedot		

Contents

1	Introduction.....	7
1.1	Overview.....	7
1.2	Goals and expectations for the study.....	8
1.3	Thesis outline.....	9
2	Agile and Lean software development	10
2.1	Overview.....	10
2.2	Defining Agile.....	11
2.3	Defining Lean Software Development.....	12
2.4	Agile vs Lean	14
2.5	Scrum	14
2.6	Kanban method	17
2.7	Extreme programming.....	19
2.8	Summary.....	21
3	Agile perspective for quality, performance and fun	22
3.1	Software quality.....	22
3.2	Happiness	24
3.2.1	Happy life.....	24
3.2.2	Maslow and happiness.....	25
3.3	Relationship between performance, quality and happiness	27
4	Continuous improvement as basis for research	28
4.1	Overview.....	28

4.2	Research strategy	29
4.3	Ethical considerations.....	31
4.4	Summary.....	33
5	Implementing the study.....	33
5.1	Overview.....	33
5.2	Initial state	33
5.2.1	Organization and environment	33
5.2.2	Practices and processes	35
5.2.3	Quality, performance and happiness.....	37
5.3	Harvesting research data.....	38
5.3.1	Overview	38
5.3.2	Author's perception	38
5.3.3	Team member survey	38
5.3.4	TPD™.....	39
5.3.5	Voice.....	39
5.4	Story as events and actions.....	40
5.5	Final state	46
5.5.1	Organization and environment	46
5.5.2	Practices and processes	47
5.6	Summary.....	48
6	General results	49
6.1.1	Overview	49

6.1.2	Author's perception	50
6.1.3	Team Practices Survey.....	55
6.1.4	TPD™.....	65
6.1.5	Voice	77
6.1.6	Reliability and generality.....	80
6.1.7	Summary and conclusions.....	81
6.2	Emergent key findings	82
6.2.1	Introduction.....	82
6.2.2	Continuous flow is better than sprints for multi-product owning product development team	82
6.2.3	The most valuable practices are those that improve visibility to the workflow and enable its continuous improvement.....	87
6.2.4	The focus of practices should be on team and not on project	90
6.3	Purpose related findings.....	93
6.3.1	Introduction.....	93
6.3.2	There is no fit-for-all methodology, but team based tailoring is required .	94
6.3.3	Some Agile practices are exceptionally good for improving all three: performance, quality and happiness	95
7	Summary and conclusions.....	96
7.1	Objective and findings of the thesis	96
7.2	Future use of the results.....	97
7.3	In closing	98
	Bibliography.....	100

Appendices 102

Appendix 1: Team survey results 102

Appendix 2: TPD™ results..... 119

Appendix 3: Voice 2011 results..... 130

Appendix 4: Voice 2012 results..... 135

List of figures

Figure 1: Maslow's hierarchy of needs.....	27
Figure 2: Original team structure	34
Figure 3: Action timeline	41
Figure 4: Final team structure	46
Figure 5: Team survey spread for performance.....	59
Figure 6: Team survey spread for quality.....	61
Figure 7: Team survey spread for job satisfaction	63
Figure 8: TPD™, high level trend	67
Figure 9: TPD™ - Openness, trust, confrontation and conflict resolution trend	68
Figure 10: TPD™ - Sound inter-group relations trend.....	70
Figure 11: TPD™ - Output, performance quality and accountability trend	71
Figure 12: TPD™ - Sound team procedures and regular review trend	72
Figure 13: TPD™ - Change, creativity to challenge status quo trend.....	74
Figure 14: TPD™ - Morale trend.....	75
Figure 15: TPD™ - Empowerment trend	76

List of tables

Table 1: Team survey averages	55
Table 2: Team survey, author's perception.....	57
Table 3: Voice results comparison	78
Table 4: Top five of team survey.....	89
Table 5: Team practices survey, average value ≥ 1.0 in all categories	95

Table 6: Tools, practices and actions team survey 102

Table 7: TPD™ survey results 1 and 2 including differences 119

Table 8: Voice 2011 survey results 130

Table 9: Voice 2012 survey results 135

1 Introduction

1.1 Overview

During the last ten years, the software development practices and processes have gone through a major transition. In the end of the 90s, the waterfall approach was the dominating model to run software development projects. The waterfall was originally developed for government projects and was very cumbersome, usually leading to bad results.

The numerous downsides of waterfall model gave birth to Agile movement, which has given birth to many new development methodologies and frameworks. Even though they share several similarities in the basic mindset, there are many differing practices and it might be hard to decide on which would be the most suitable for the team in question.

Both waterfall and Agile practices are also often very much project oriented; however, when developing software as products, some of the realities on usual project based work do not apply. Such special conditions emerge from the fact that in addition to usual development work, the maintenance work plays an integral part of the daily routines of such a team. Having such environmental conditions, as well as many different approaches available, it can be truly hard to try to figure out what practices would truly fit for the situation of a specific team.

This research focuses on two software product development teams, which both are a part of the same group that is working on developing medical software products. The aim for the research was to experiment and take into use different kinds of Agile practices through continuous improvement process. The main goal was to improve the technical quality of the developed software and the performance of the team while maintaining a high level of happiness.

The intention was to find out if there are some specific Agile practices that are especially well suited for this kind of software product development environment and to analyze on what their success is based on in this environment. If such practices were to be found, they could serve as a baseline which other product development teams could use when trying to find ways to improve their practices.

Also, it is expected by the author that there might be some practices that could improve all three: technical quality, team performance and happiness. Special attention is also paid and given to trying to find such practices.

As the research is carried out through continuous improvement and is iterative by nature, it is to be expected that many emergent findings will present themselves through the implementation. Therefore there will be strong bias that the adjustments are made and research is guided by these emergent findings.

1.2 Goals and expectations for the study

The goals of the research project can be divided into an organizational goal and research goals, of which the research goals should support the organizational goal.

The organizational goal is to seek improvement in team performance and produced quality while maintaining a high level of happiness in such a manner that the team can move their bias from maintaining old products to the development of new products

The research goal is to find out the perceived benefit of a set of different Agile and Lean practices that would be perceived to have benefit for one or more of the followed team attributes: performance, produced quality and happiness.

Some of the expected findings for the study are following:

- Some Agile or Lean practices that improve all three: performance, produced quality and happiness are found

- No fit-for-all methodology can be found that would work for both of the teams, but instead teams will end up having differing practices.

It is expected by the author that through continuous improvement process that takes advantage of different Agile and Lean practices, the organizational goal will be met.

1.3 Thesis outline

The outline of the thesis is following.

Chapter two and three focus on literature and theory that the thesis is based on. The attention is given to definition of common Lean and Agile approaches as well as describing and considering the relation of quality, performance and happiness.

Chapter four describes the research approach that is taken for the research.

Chapter five goes through the main events and changes from initial state to the final state. Initial state for both of the teams is described, followed by details of data that was gathered and major events taken. The implementation details are then closed by describing the final state.

Chapter six presents results and findings, first introducing the results on a more general level through different kinds of surveys and observations and later on focusing on some emergent and expected key findings. The data that is gathered in the research is more closely analyzed in this part. In addition, the findings are analyzed more in-depth and the conclusions related to individual findings are gone through.

Chapter seven brings together the whole research, binding together different findings to draw conclusions on the findings together as a whole. In addition, possible suggestions for future research and development actions are gone through.

2 Agile and Lean software development

2.1 Overview

Agile software development and Lean software development are terms that are often used interchangeable and their relationship can be thought quite obscure.

Both of the approaches are quite fresh in software development field as formally described, however, many of their aspects are something that has played a part through common sense and best practices already from the early days of the software development.

Agile and Lean can both be considered as mindsets and guidelines for developing software. Even though for example Lean suggests some more specific tools, in essence they both set a mindset and principles through which one can reach higher customer value.

Under the Agile umbrella, there is a large number of different frameworks and methodologies, of which the best known is Scrum. Scrum has achieved such a well known status that many consider Scrum as being equal to Agile.

In the following chapters, first the principles of the Agile and Lean are discussed in more detail, followed by going through some of the most popular methodologies that are based on Agile or Lean thinking. The selection is based on the listing of most popular Agile methodologies in the VersionOne's "State of Agile" survey (VersionOne, 2014) of the year 2013. The attention is focused on Scrum, Kanban and Extreme Programming.

2.2 Defining Agile

The term Agile software development became largely known in 2001, when a number of well known and respected authorities of software development got together to bring together the core principles behind multiple lightweight software development methods that had been recently rising to challenge the dominant position of heavy and cumbersome waterfall.

As a result, the Agile manifesto was made. The idea of Agile manifesto is to provide a guideline of what to consider as Agile Software Development.

The manifesto can be found at Agilemanifesto.org (Manifesto for Agile software development, 2001) and is quoted as follows below:

“We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.”

As it can be seen, Agile manifesto defines Agile software development very loosely, mostly advocating common sense over bureaucracy. In the scope of this research it is considered that Agile software methodologies and frameworks are something that complies with the mindset of the Agile manifesto, which makes the Agile umbrella quite wide.

2.3 Defining Lean Software Development

Lean software development was introduced by Mary and Tom Poppendieck. The Lean software development has roots on Lean production principles that initially started at Toyota and have gained a large support and success in manufacturing business. In their books about Lean Software Development, Poppendiecks introduce principles and tools for software development as counterparts of what are defined for Lean manufacturing.

Poppendiecks define seven core principles in their book *Implementing Lean Software Development* (2007) including

- Eliminate waste
- Build quality in
- Create knowledge
- Defer commitment
- Deliver fast
- Respect people
- Optimize the whole

Lean and quality

Quality plays an important aspect in Lean software development, as it can be seen that “Build quality in” is one of the core principles of the Lean. Quality also plays an essential part in some other principles, such as “Deliver fast”.

Poppendiecks (2007) summarize the meaning of building quality in as “to build quality into code from the start, not test it in later” going further on defining that such a way of working needs a highly disciplined organization to do it.

Considering delivering fast, Poppendiecks (2007) once again underline the importance of the quality, telling that it is impossible to deliver fast without a very high level of quality.

Lean and happiness

Development of people and better cooperation and mutual respect are also some of the key concepts of Lean software development. These aspects can be connected with happiness to some level. This is especially well shown by the principle “Respect people”.

Respecting people comes all the way from the Toyota Production System that is the origin of the whole Lean thinking, being one of the three corner stones defined in it. Poppendiecks(2007) connect respecting people in providing teams to self-organize to meet the reasonable goals they have been defined, leading to empowerment. In these days, empowerment seems to be a very common part of for example job satisfaction and team performance surveys such as TPD™ and Voice that are used as data in this research.

Lean and performance

The performance aspect also manifests itself in Lean thinking, especially through “Remove waste”, “Optimize the whole” and “Deliver fast” principles.

Removing waste is in the very core of the Lean thinking, as most of the other principles support the goal to reduce waste. Poppendiecks (2007) consider waste to be everything that does not add customer value or delays the delivery of customer value. This in essence is the key for shortening delivery times.

Optimizing the whole focuses on to not micro-optimize, but to concentrate on the whole chain, once again driving performance improvement with a focus on the whole value stream.

Delivering fast focuses on delivering value to customers as soon as possible. This has need for high quality as noted earlier and also one of the key things is that companies that deliver fast “have eliminated a huge amount of waste, and waste costs money”(Poppendieck & Poppendieck, 2007).

Lean software development has quickly gained much support and many followers and is often widely used to extend project based agility to organization level, improving the organization's value output as a whole.

There are also some specific methods that have been developed as based to answer some parts of the Lean approach, such as Kanban method which is described in chapter 2.6.

2.4 Agile vs Lean

The relationship of Agile and Lean is somewhat obscure in a way that there are several different opinions of it and none that could be made as clear cut. Lean is often listed as being one methodology in the family of Agile, however, as often they also seem to be considered as separate philosophies that support each other.

It can be seen that Lean development does fulfill Agile principles; however, it can also be seen that most of the Agile methods also seem to respond well on fulfilling Lean principles, or that Agile principles partially fulfill the Lean principles.

Even though the relationship of the two philosophies is good to know as a background, defining their clear relationship is not in the scope of this thesis. To simplify readability of the thesis, both Agile and Lean methods and practices are mainly referred to as Agile, unless there is some specific need to especially highlight their difference.

2.5 Scrum

Based on State of Agile Survey (VersionOne, 2014), 55% of projects among Agile projects use Scrum and 18% percentage of combination of Scrum and some other method. This clearly shows the position that Scrum has gained as the most widely used and best-known Agile methodology.

The purpose of the Scrum can be summarized as follows: *“Scrum is a management and control process that cuts through complexity to focus on building software that meets business needs”* (Schwaber & Beedle, Agile Software Development with Scrum, 2002).

It is also further noted by Schwaber and Sutherland (2013) that Scrum is especially founded on empirical process control theory.

This empirical approach can be strongly seen through the whole framework, both on iteration level and whole project level.

The Scrum framework was originally authored by Ken Schwaber and Jeff Sutherland, who are still actively maintaining the Scrum rules and providing them as free through their Scrum web pages.

Scrum defines a set of practices and tools as well as specific roles and relationships for software development.

In a nutshell, the Scrum project is run into iterations called sprints. Each sprint starts with planning where a team decides together with a product owner on what can be implemented during the next time boxed sprint. Every sprint should lead to deliverable results.

Scrum defines three concrete roles, around which the responsibilities in processes and practices are to be found:

- Scrum Master
- Product Owner
- Development team

There are also several practices and tools that Scrum provides, including for example

- Sprint planning
- Daily standup-meetings

- Sprint reviews
- Retrospective meetings
- Product backlog
- Sprint burndown chart
- Product burndown chart
- Definition of done

Many of these core practices of Scrum are also used by teams that are using some other Agile approaches.

Scrum and quality

Scrum does not have a very direct focus on quality and none of the artifacts or activities is purely quality focused; there are however, certain aspects that could be considered on driving quality.

The fact that Scrum teams are to produce shippable functionality after each sprint does lead into a situation in which the teams must test early and focus on good quality based practices to ensure that sprint deliverables are on shippable state.

In addition, the definition of done does bring an important tool for ensuring that everybody considers the term 'done' to mean the same thing, which is often very useful in the quality perspective as well, as it turns some quality matters more explicit for all involved.

Scrum and happiness

Scrum does not directly focus on happiness of the people involved in the Scrum activities; however, Scrum aims to empower people to make decisions on the right level and shorten the feedback cycle. Therefore it can be seen that some of the Scrum practices can be expected to have a positive impact on happiness.

Scrum and performance

There are claims that Scrum can offer remarkable improvement on the development performance over the time. The way that constant prioritization and shortened release cycles get team focused on most important things, should lead to reduction of waste.

In addition, Scrum is claimed to speed up the self-organization of the teams and therefore lead to improved performance.

2.6 Kanban method

The Kanban method has its roots on pull-based systems with kanban cards that are common in Lean manufacturing. The method was introduced by David J. Anderson.

Anderson (2010) describes Kanban method *“as evolutionary change method that utilizes kanban (small k) pull system, visualization and other tools to catalyze the introduction of Lean ideas into technology development and IT operations”*.

Kanban method is based on idea of pull-system in controlling work, meaning that the work is only pulled when someone has a time to start working on it, instead of pushing work into a queue for people. This should pinpoint the bottlenecks in team’s work and enable a team to better share the work, and therefore improve performance.

Anderson (2010) defines five properties for implementing Kanban

1. Visualize workflow
2. Limit Work-in-Progress
3. Measure and Manage Flow
4. Make Process Policies Explicit
5. Use Models to Recognize Improvement Opportunities

As can be deduced from the list, Kanban method focuses strongly on the visualizing and optimizing the flow of the work, and pushing continuous improvement through it.

Anderson (2010) also describes his six-step recipe of success for implementing Kanban method, including

- Focus on quality
- Reduce Work-in-Progress
- Deliver Often
- Balance Demand against Throughput
- Prioritize
- Attack Sources of Variability to Improve Predictability

Kanban method is also often combined with some practices of the Scrum. Such a combination is usually referred as Scrumban.

Of the responders of the State of Agile 2013 survey (VersionOne, 2014), 5% have responded using Kanban method and 7% have responded that they are using Scrumban. The number is remarkably smaller than that of Scrum users, however, Kanban method is also a much newer approach.

Kanban and quality

Quality is in a very important role of Kanban method as Anderson (2010) describes it, including it as number one on his six-step recipe of success.

The main reason for that is yet elaborated by stating that *“excessive defects are the biggest waste in software development”*. Anderson also goes on stating that by focusing on quality, the throughput time improvement could be two- to four-times or even ten-times for truly bad teams.

In addition, Anderson also highlights that his experience indicates that reducing work-in progress or shortening the length of iteration will have a large positive impact on quality.

Therefore, it can be seen that it is suggested that to really get most out of Kanban, most of the teams should focus on quality early on.

Kanban and happiness

The Kanban method is built around pull-based work control, which is to eliminate bottlenecks of work piling up for some individuals. Such a way could be considered to lessen the stress and overburden that could otherwise become high on individual level. The side effects of easing the burden have many effects that can be considered as positive for happiness, including for example improved work/life balance.

Kanban and performance

The Kanban focuses heavily on improving performance. Approaches such as pull-based flow, limiting work in-progress and focusing on lead time are all something that aims at improved performance. The focus on optimizing the whole value stream and creating continuous improvement culture are in the core of the Kanban thinking. Therefore Kanban method can be seen essentially as a way of doing continuous improvement, aiming for improved performance, and that as a side effect it takes into account such ways of working that lead to positive results, also in happiness and quality.

2.7 Extreme programming

“Extreme Programming is a discipline of software development based on values of simplicity, communication, feedback, and courage. It works by bringing the whole team together in the presence of simple practices, with enough feedback to enable the team to see where they are and to tune the practices to their unique situation.” (Jeffries, 2014)

Extreme Programming has one characteristic that strongly separates it from for example Scrum and Kanban method. Even though Extreme Programming also has rules for coordinating the work of the team, to which Scrum and Kanban method are mainly

focused on, it also has a strong focus on concrete software engineering practices such as Test-Driven Development and Pair Programming.

Even though the work controlling aspects of Extreme Programming have not gained as widespread usage as for example Scrum, the development practices are widely adopted and used to some extent by most of the Agile teams. This can also be seen for example in State of Agile 2013 survey (VersionOne, 2014), where only 1% has stated of using Extreme Programming as it is, but 11% have stated of using a hybrid of Scrum and Extreme Programming.

There are 13 core practices of the Extreme programming, as defined by Jeffries(2014):

- Whole Team
- Customer Tests
- Planning Game
- Small Releases
- Collective Ownership
- Coding Standard
- Continuous Integration
- Sustainable Pace
- Metaphor
- Test-Driven Development
- Refactoring
- Pair Programming
- Simple Design

As can be seen from the list, of the thirteen core practices nine could be considered to be clearly focused on the development work itself and only four to controlling the work to be done.

Extreme programming and quality

Extreme programming can be seen as very quality focused. It defines practices such as test-driven development, refactoring, pair programming and coding standard which are heavily focused on producing high quality code. Extreme programming can be seen as excellent toolkit for software craftsmanship and is often used as a hybrid with some other Agile methodology or framework to bring value in daily development work as well.

Extreme programming and happiness

As extreme programming is very focused on software craftsmanship, it can be seen as a means to enable happiness as well.

Many developers are motivated by the possibility to carry out their development job with highest quality and to be proud of it. That is something that many practices and a mindset that extreme programming provides, and therefore it can be seen to have a motivating factor.

Extreme programming and performance

Extreme programming is in line with other Agile practices in the sense that it aims for fast deliveries and improved performance of the teams by focusing on developing the real software and minimizing bureaucracy. In addition, many of the quality oriented practices can be seen to have a strong focus on keeping up the performance and performance improvement in long term.

2.8 Summary

Agile software development can be seen as a large umbrella including many different frameworks and methods that are implementing the mindset of Agile Manifesto. Some,

such as Scrum are more focused on Agile work management as others such as Extreme Programming have more focus on engineering practices.

Lean Software Development can be seen as an independent set of principles and a mindset for turning Lean Manufacturing into software development domain; however, Lean Software Development also fulfills the Agile Manifesto and therefore in the scope of this thesis both Lean and Agile methods are mainly referred to as Agile.

3 Agile perspective for quality, performance and fun

3.1 Software quality

Defining software quality is a task that has proven to be very hard for all, as it seems that most often the software quality is not seen as objective but more subjective. In addition, there are many different facets to consider in software quality, varying from technical quality to the fulfillment of the user needs and desires.

In addition of seeing as dependent of perspective, the quality can be seen dependent on the context and environment.

“Quality of software component is not an intrinsic property – the exact same component can be of excellent quality or highly dangerous depending on the environment in which it operates or the intent of the user.” (Capers Jones, 2012)

To be able to define the term in the sense that it is used in this study, it is important to have a perspective of how others have defined it.

Different standards have been defined, including for example ISO/IEC9126, however, they have seemed not to be widely adopted by software development community.

“Many definitions have been suggested over the years, but none have been totally satisfactory or totally adopted by the software industry, including those embodied in international standards.” (Capers Jones, 2012)

A similar trend of difficulty in defining software quality also extends to Agile literature; however, the term technical debt is something that is constantly brought up in Agile literature and could be interpreted as an Agile definition for lack of quality.

The definition itself is quite simple.

“Anything that makes code difficult to change is technical debt.” (Poppendieck & Poppendieck, Leading Lean Software Development, 2010)

Poppendiecks(2007) also give few examples of technical debt, including:

1. We tolerate obscure code.
2. We don't take time for refactoring.
3. We run regression testing instead of continuous testing.
4. We build unnecessary dependencies.
5. We branch code, postponing the system testing until everything is merged again.

There is also a slightly different description for technical debt given in “Economics of Software Quality” which states that *“technical debt is defined as the cost of fixing the structural quality problems in an application that, if left unfixed, puts the business at serious risk. Technical debt includes only those problems that are highly likely to cause business disruption (due to operational problems and/or product/service launch delays) and hence put the business at risk; it does not include all problems, just the serious ones.”* (Capers Jones, 2012)

In practice, both of the descriptions seem to deal with structural quality issues; however, Jones' description seems to solely focus on existing structural quality issues while

Poppendiecks' description also seems to include practices that could lead to structural quality issues.

When we are using the term technical debt in context of this study we will be referring the definition of Poppendiecks'.

In this study we will focus mostly on freedom of defects and lack of technical debt as the aspects of software quality, setting our definition of software quality to be software quality is lack of both defects and technical debt.

From Agile perspective, it is also worth noting that in State of Agile 2013 – survey (VersionOne, 2014), 82% of responders stated that adopting Agile methodologies has enhanced software quality. This is clearly also a positive indication when considering possibilities to improve quality through adopting Agile.

3.2 Happiness

3.2.1 Happy life

Martin Selignam discusses about happiness in his Ted-talk (Selignam, 2004), in which he divides happy life into three kind of happy lives. These include pleasant life, good life and meaningful life.

Pleasant life is considered to bring happiness through pleasures. It is very shallow kind of life as single pleasure doesn't last very long until one gets used to it.

Good life is a life that enables person to use their highest strengths. Fulfillment of it can be seen through reaching flow-state, a highly concentrated state where time feels to stop and person is so focused that they close out everything external for the moment.

Meaningful life is a life that enables one to work for good of some higher cause. It is also a life where one knows his signature strengths.

If we consider mapping the concept of the happy life to the professional environment, there is clearly some connection to be found. If we are to be able to use our highest strengths and know them better in our work, we are able to fulfill both good and meaningful life to some extent. If we also believe that we are working for some greater good than just for ourselves, then the meaningful life is even more fulfilled. Pleasant life can be seen as a crusting of the cake and when both good life and meaningful life exists, it can be extended by social aspect of the work.

3.2.2 Maslow and happiness

Maslow's hierarchy of needs is a very well known theory of human needs and motivation. It was first described by Maslow in his article "A Theory of Human Motivation". Copy of the article is currently also available on internet, to which we are referring in here when we are discussing of the article.

Maslow(1943) describes a hierarchy a needs that humans are trying to fulfill. On the base of the hierarchy there are the basic physical needs related to our survival and on the top of the hierarchy is a need to reach our personal potential. This means that to be motivated to focus on items that are higher on the hierarchy, your lower lever needs have to be satisfied before.

Fulfilling these needs can also be considered as source for happiness.

Categories of the needs that Maslow(1943) describes are physiological, safety, love/belonging, esteem and self-actualization.

Physiological needs are purely related to basic survival and include things such as need for food, water and air. Therefore they are something not well applicable to have effect to our happiness in working environment.

Safety needs considers everything that provides us safety. It includes things such as shelter, but is also applicable to safety of the job as well. Therefore there can be seen connection between the happiness at work and category of safety.

Love/belonging needs are about need to have relationships with other people. Maslow (1943) describes that person *“will hunger for affectionate relations with people in general, namely, for a place in his group”*. In working environment this can be for example belonging to team of people or community of professionals.

Need for **Esteem** is described by Maslow(1943) by stating that *“All people in our society (with a few pathological exceptions) have a need or desire for a stable, firmly based, (usually) high evaluation of themselves, for self-respect, or self-esteem, and for the esteem of others.”*. In the hierarchy of needs, esteem needs to be fulfilled to be able to achieve self-actualization. Esteem can be seen to have high connection to happiness at work, because it strongly comes from the fact that you are successful in what you are doing and are receiving desired appreciation of your achievements. In addition Maslow (1943) considers that by fulfilling need for esteem, one becomes more successful stating that *“Satisfaction of the self-esteem need leads to feelings of self-confidence, worth, strength, capability and adequacy of being useful and necessary in the world. But thwarting of these needs produces feelings of inferiority, of weakness and of helplessness”*.

Need for **self-actualization** is on the top of the hierarchy of the needs. Maslow(1943) summarizes the self-actualization with words *“What a man can be, he must be. This need we may call self-actualization”*. In other words, self-actualization is connected to following your calling and being able to use your signature strengths, as well as continuously striving for improvement to reach your full potential. Self-actualization can be seen to have strong connection to working environment as it is very much related on if you are able to use your strengths on your work and if you are provided a chance to continuously improve yourself.

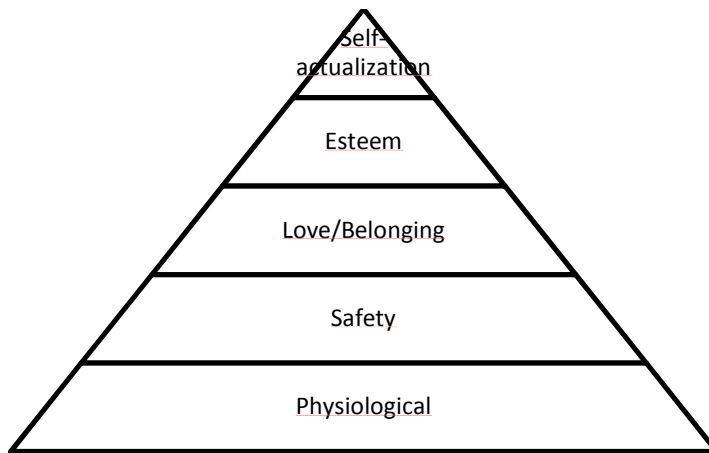


Figure 1: Maslow's hierarchy of needs

There seems to be common point in Maslow's and Selignam's theories. From both of the perspectives it seems quite clear that personal development and possibility to work with highest strengths has high importance for person's happiness. In addition, social aspect of the work and feeling of belonging can be considered meaningful.

3.3 Relationship between performance, quality and happiness

When considering both Selignam's and Maslow's theories related to happiness, we can find some attachment points to both quality and performance.

Both Maslow and Selignam are putting value on doing fulfilling job and being able to use your best strengths. In addition it is noted to be important to be appreciated due to that. Therefore it can be considered that if your work lacks the quality that you consider necessary or you provide quality that others don't feel satisfactory, you will not be satisfied and feel happy. On the other hand, providing superior quality should have high positive impact on happiness.

The happiness also can be considered to have correlation with the performance. As the performance is higher, you reach your goals better and faster. This should have positive effect to your self-esteem as well as increase the feeling of appreciation. Therefore high performance could be seen positive for happiness.

In addition, we can see connection between quality and performance. As described earlier, the poor quality and technical debt is considered to leading into slowing down development efforts. This would be especially true in longer term. Therefore it can be seen that poor quality would have negative impact on performance.

4 Continuous improvement as basis for research

4.1 Overview

This research has a clear focus on continuous improvement of the teams' ways of working as joint action. Research strategy, as refined for this research, is therefore built around such choices that would be very supportive for such a continuous improvement.

The following list summarizes the strategy chosen and is then followed by more detailed information of reasons for choosing them.

The main aspects of the chosen research strategy are:

- Pragmatic research philosophy
- Research method is action research, mixed with survey method
- Mixed research approach, transition from deductive to inductive
- Longitudinal time horizon

In the research, the researcher actively participates in the research as a member of the teams, which will position him as participant-researcher.

The exact role of the author inside the team did vary during the duration of the research and the changes in roles are defined more in-detail in chapter 5.4.

4.2 Research strategy

Continuous improvement of the team can be considered to be continuous loop of iterations of taking actions and reflecting them to consequences. Many Agile practices support such an approach, with the prime example being the very popular Scrum framework that defines retrospective meetings, which are held at the end of iterations and that focus on team development. This kind of continuous improvement is very much in line with how action research can be implemented, as *“action research involves learning in and through action and reflection”* (McNiff, 2013, s. 24).

There are actually so many similarities between continuous improvement through Scrum and action research that Lahti(2008) came up with the conclusion that Scrum process is actually a form of action research and therefore the action research action research cycles would naturally materialize through Scrum practices.

Even though from the perspective of author there are meaningful parts of Scrum that really do not fit under the action research, having more bias on production work, the author agrees that there are parts of Scrum framework that can act as base for action research implementation if they are given enough attention. Good mapping with some of the initial team practices can also be seen there that are already in use or will be taken into use early on. These include for example mapping of retrospective meetings for reflection and action planning.

Considering all this, it is very natural to choose action research as the main research method for the study.

One of the vulnerabilities of action research is the fact that the quantitative data produced through it can often be very subjective. This is due to the fact that researcher is also a participant.

This is realized by the author and to increase the validity of results, action research will be mixed with survey research method. The survey method is used to get together anonymous and more structured data from the perception of the team members about the effect of the changes taken through continuous improvement. This is also reflected onto the author's perception and major differences are evaluated to bring more objectiveness for the evaluation of the results as a whole. Such a method level triangulation should provide valuable mix of quantitative and qualitative data for proper analysis.

When focusing on improvement over the time, it is important to focus on trends over the time which will be given attention when gathering data for the study. In addition, the teams under the study are expected to mainly have the same persons for the duration of the study, allowing a possibility to track real trends for the changes of the practices as the members of teams perceive instead of perceived effect of changes to the team memberships. Therefore the focus in data shall be on longitudinal time horizon.

The continuous improvement is by its very nature expected to lead into new theories either through emergent findings or through a new combination of existing theories. Also, continuous improvement will have need for a possibility to adjust the research emphasis due to the emergent findings. This all supports inductive research approach, which is even further supported by the fact that team development is very much people oriented and a great deal of the research is related to considering human relation to the events and decisions taken by the team. However, it is important to take a note of the fact that there is plenty of theory and literature that will be taken into account early on when the first changes are to be made. Most of the initial actions will be very likely to

follow some specific methodology such as Scrum, being very much in line with literature suggestions. Therefore it can be considered that during the early phases of the study, a deductive approach is strongly present.

Due to these different aspects in the study, a mixed approach of induction and deduction is used in the study. Their balance can be considered so that in the beginning of the study the deductive approach is dominating, however, as the changes keep on steering practices further from the predefined vanilla frameworks or methods the inductive approach will take over and become dominating.

Based on the need to adapt research emphasis as well as implementation details by emergent needs of the team, the pragmatism can be considered as overall research philosophy.

4.3 Ethical considerations

“Ethics is not only about taking action; it is also about doing research. In action research both are related.” (McNiff, 2013)

Due to the nature of the action research it is therefore in place to consider ethics from both of the perspectives.

Ethics in action

When considering the perspective of action in the context of the study, it is the fact that teams that are studied consist of individual persons. It is important that no harm is to be done to these persons through the research. As the number of persons in the teams is small, describing actions or reactions by a person’s role or anything else that clearly separates them from others in such way that they can be easily identified, special attention must be given to anonymity. Anonymity is ensured by structuring data gathering in such a way where individual responders cannot be separated and by

ensuring that on personal observations of the author, no persons are described by their role.

Confidentiality and anonymity must be considered to a certain level company-wise as well. Especially when using confidential company data, such as internal surveys, proper permission for usage needs to be negotiated. In addition, no confidential information from products should be revealed. To further ensure this, products are not referred by their real names.

Ethics in research

McNiff (2013) enlists plagiarism, name-dropping and pedantry as common ethical issues in research. Of those ethical issues, plagiarism and name-dropping are clearly some issues that are taken into account through honest approach for the research and carefulness in reporting.

Pedantry, however, is something that needs to be given special attention. It is especially important in the case of action research to clearly state the position of the researcher, which the author has done in chapter 4.1.

It is also especially important to have courage to come out and tell the truthful opinion, even if it could conflict with authorities of the field and literature. This is something that can sometimes be very hard in a qualitative research as a great deal of the findings are based on the author's own observations and therefore proving them is not as straight forward as with quantitative data. That is something that the author has to remind himself of when making conclusions and he needs to encourage himself to out of box thinking.

4.4 Summary

The continuous improvement, such as in focus of research, has natural mapping to action research, which is therefore taken into use. Action research is further supported with survey research. The general approach in research is to choose a way of working that supports responding to emergent findings, as they are in the core of the research.

Ethical aspects are mostly to be considered from individual and organizational perspective, as well as courage to be honest in case of controversial findings.

5 Implementing the study

5.1 Overview

In the following chapters the main aspects of implementing the study is reported with the main “story” of the study on a high level .

An overview of initial state is given, followed up by the main change actions and research data gathering, ending up with an overview of the final state.

5.2 Initial state

5.2.1 Organization and environment

Figure 2 describes the initial organizational structure that was concretely involved into this study.

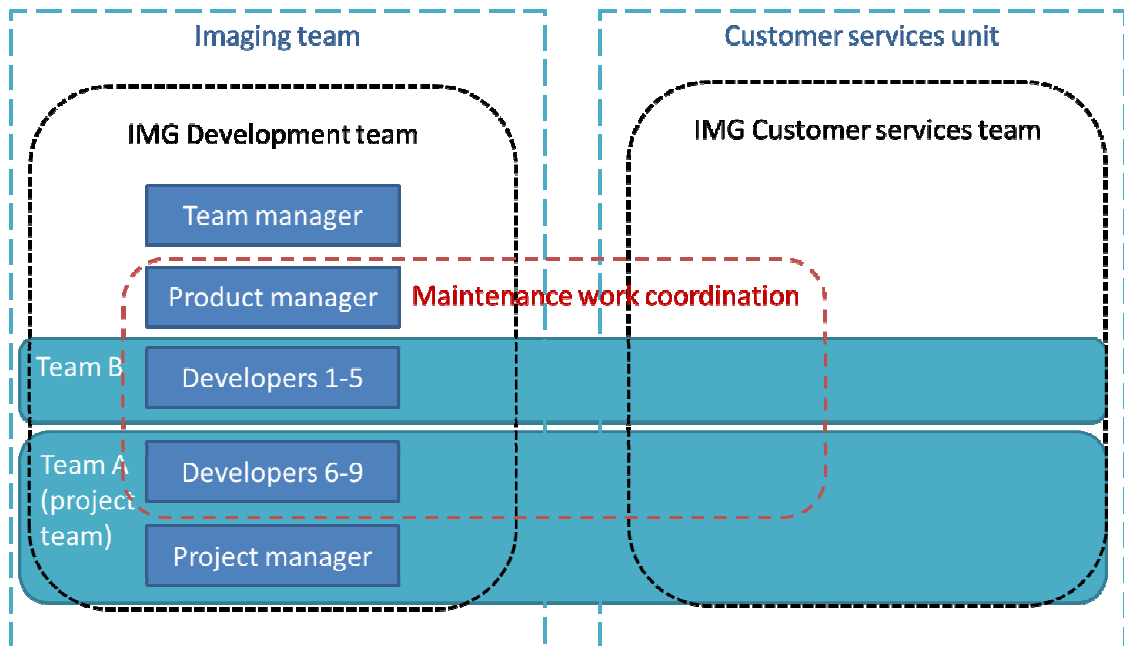


Figure 2: Original team structure

On the highest level, two different concrete and quite separated organizations were included, the one being the development organization in which imaging development team resided and the other the customer services unit, in which imaging customer services team resided.

There was tight cooperation between these specific teams even though they were located in different units.

The development team, which is the main focus of the research, was in practice divided into two sub-teams, however, that was not formal and the stability was more decided by project needs.

The maintenance responsibilities for both teams were more carried out by individual persons responsible for specific products. The situation was clearly unbalanced in a way that most of the products were under maintenance of only one or two persons. The maintenance work was coordinated to a great extent by the product manager.

The author acted in both teams, for the project team of team A as a project manager and for the team B as a developer.

Team A had a new development project starting and it also had several products to maintain by some individual members of the team.

Team B had only maintenance responsibilities for a single product which had been taken into production a year before. In addition to that, one member of team B had maintenance responsibilities towards one older product.

In addition to this base organization, some team members had responsibilities to some other teams in development unit, thus also having their commitment divided into other parts of a wider organization.

5.2.2 Practices and processes

In the beginning of the research there was great variation in practices and processes used inside the team. These can be divided into whole imaging domain team level practices and processes and sub-team related processes and practices. In addition, these can be furthermore divided into development and maintenance.

Team A had grown their processes very organically, not paying much attention to any coordinated actions in developing their ways of working and only following organizational processes on very minimal level. This was something that the developers of the team A liked, however, it caused trouble when trying to reach any predictability for the team or when the team had to coordinate work with other teams. There was also no real visibility to the work for members outside the team, and even inside the team the visibility seemed to be clear mostly on personal level as work was very much divided into individual tasks and responsibilities.

Team B had taken Scrum as their main framework for development during the development of their last project with activities such as sprints, daily meetings, sprint

demos, sprint planning and retrospectives on place. The same was now continued mainly also during the maintenance work, even though sprints were something that often could not be followed rigorously due to the nature of maintenance work and possibility for urgent service requests. The team also had issues with backlog as the earlier product was started as badly documented and Scrum practices including backlog were taken into use during the final parts of the project.

Neither of the teams had initially really taken much use of any Agile methodology while starting new projects and in addition, also earlier approach to use waterfall was taken very lightly in the sense that the project coordination and management had been lacking a decent approach. It seems for the author that the projects had been pulled through mostly due to the competence of individual developers.

Maintenance work was usually organized by a product manager who gave maintenance tasks for individual developers. These tasks could be related to specific bigger maintenance release of a whole system that imaging team products integrated into, or in urgent cases immediately. These maintenance tasks were rarely coordinated with project management and the needs were not synchronized with other additional work.

In addition to that, there were some prototyping and research activities that could come from a product manager or team manager that did not really map into current development work very well.

Wider organization had project management practices set up, however, they were mainly considered cumbersome and something that the team followed only to fulfill minimal possible criteria and often missing even them.

However, the team as a whole managed to do its work and, due to having skilled members, was able to deliver decent results.

5.2.3 Quality, performance and happiness

In the beginning of the research, the team seemed to be above average in all three: quality, performance and happiness when compared to other teams in the organization.

The impression of the other teams was that imaging team is a good team with skilled individuals. The team was able to deliver products, however, there was also concern from other parts of organization considering visibility of what the team is producing.

The team was considered to be delivering good quality, but the author's personal view differs from that. The best example of quality issues that the team was facing was their latest product that had been taken into production use a year before the start of the research. After release the product's maintenance had taken full effort from team B to maintain it and to fix quality issues reported by customers. Most of the rest of the products of the team had already stabilized due to being in production for some time longer, so the author does not have that good a view on their initial state immediately after their release. There was, however, regular need for maintenance also for them, even though the amount was remarkably smaller.

The performance of the team was also considered to be on a decent level, and when considering the whole organization the level seemed to be above average.

Happiness of the team was very high. The team had a long history of being the top team of the whole organization when considering job satisfaction. The team had become a closely knit bunch with much power over how to do their work and most of the members of the team had already worked together for many years.

5.3 Harvesting research data

5.3.1 Overview

Several different sources of data were used to analyze the effects of the changes taken during the research period. They include both quantitative and qualitative data, from the author's perception to different surveys.

The following chapters go through the different research data that was collected during the research and details of the way they were collected.

The data is analyzed more in detail in chapter 6.

5.3.2 Author's perception

As one of the main aspects of doing the research is through action research, one important way of gathering information is by evaluating the author's personal perception and informal discussion with other team members. The author's perception is the backbone of the qualitative analysis, and it is further supported by quantitative results provided by different surveys.

The perception of the author originates from active participation in the daily work of the team and observing the team.

5.3.3 Team member survey

Four months after the end of the implementation phase of the research, a survey was made for the team by the author. In the survey, the author had listed the most major changes in practices and environment that had occurred during the research period and he asked the team members to evaluate their effect from the perspectives of performance, quality and jobs satisfaction.

The purpose of the survey was to get anonymous information from the team members to get quantitative data to highlight the most positive and negative changes as experienced by the team members.

This was then used to reflect against the perception of the author and the earlier informal discussions between author and team members.

The survey was implemented as an electronic survey.

Full results of the survey are included in Appendix 1: Team survey results.

The results of the survey are discussed more in detail in chapter 6.1.3.

5.3.4 TPD™

Team performance diagnostic (TPD™) is a survey that was performed by SIA Group as a part of leadership training in which the author participated.

TPD™ was run twice in the middle of the research period, with a six-month span between the initial run and rerun, in spring and autumn of year 2012.

TPD™ focuses on finding out the perception of team and manager about the current status of the team from many different perspectives. It is used to find possible pain points in the team and to suggest some focus areas that could lead to an improved team performance.

TPD™ survey results are included in Appendix 2: TPD™ results.

The results of TPD™ are gone through more in detail in chapter 6.1.4.

5.3.5 Voice

Voice survey is a company's employee survey with focus on employee satisfaction, however, also including different parts with focus on for example perceived level of team practices.

Voice survey is run on a yearly basis and it is comparable between different years in most of its parts. In this study, the Voice survey results are used to compare the change between the surveys of year 2011 and 2012.

Data of full Voice surveys of the team for years 2011 and 2012 are included in Appendix 3: Voice 2011 and Appendix 4: Voice 2012.

The results of the Voice surveys are gone through more in detail in chapter 6.1.5.

5.4 Story as events and actions

There were several major events that took place during the research, some of which were triggered by the research itself and some of which were triggered by some other sources such as organizational changes.

Figure 3 shows a simplified timeline for the most major changes. The list of the major events has been chosen by the author by his perception of what should be considered most major.

In practice, all the organizational changes that had effect on team structures are included and in addition, changes in practices that seemed to have a major impact on ways that team worked or caused plenty of debate inside the team. Some of the major events include many changes that are also evaluated separately when going through the findings. A good example of such is introduction to the Scrum, as it includes multiple Agile practices that bring also value separately from each others.

In addition, different surveys are mapped on the same timeline to give a better impression of possible effect of such events on the results of the surveys.

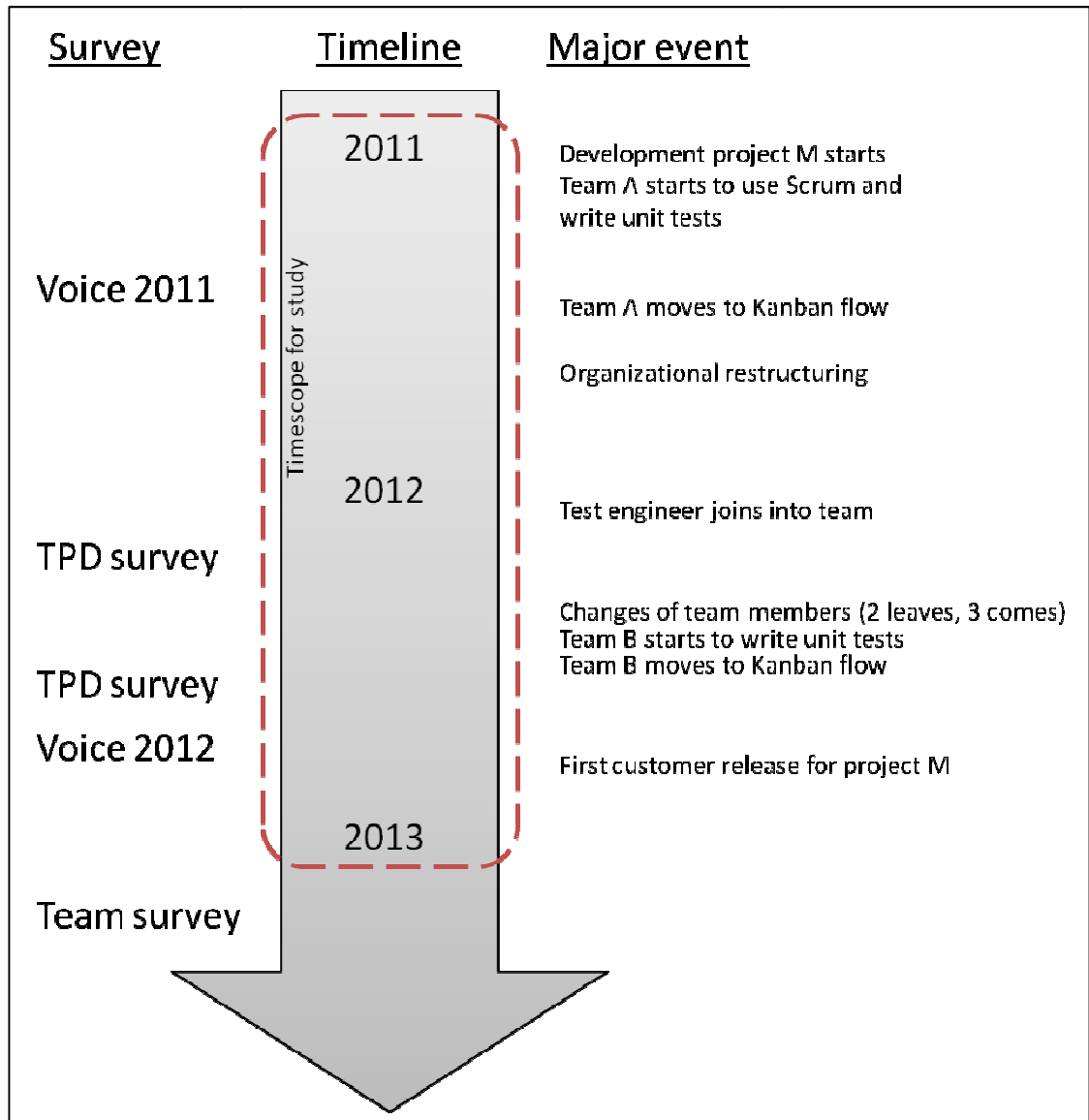


Figure 3: Action timeline

More detailed description for each of the major event is given in the following paragraphs.

Development project M starts

Development of product M was the most major product development project for the whole imaging team during the period of the study. The start of project M acts as a starting point for the whole research.

Product M was developed by team A. The author's initial role in the project was to act as the project manager.

In the beginning of the project, the earlier development practices of Team A were carefully gone through and major changes were made to them.

Some other early major events, such as Team A starting to use Scrum, starting to write unit tests or taking continuous integration into use are triggered by the starting of project M.

Team A starts to use Scrum

In the beginning of the project M, Team A first took Scrum into use. In practice this included all the basic Scrum artifacts such as:

- Daily meetings
- Sprint planning
- Sprint reviews
- Sprint retrospectives
- Roles of Scrum Master and Product Owner

In addition, to better follow up the work and progress during sprints, the team started to use a virtual task board.

This was quite a major change compared to earlier practices and processes that this team had been using. The earlier ways of working had been very ad-hoc and had minimal project level control, being instead very Agile in the sense that work was simply just done without really any coordination. However, it caused issues with predictability and the team did not seem to have any concrete way to take care of continuous improvement.

Team A starts to write unit tests

When starting the project M, a decision was made by author with the team that to ensure good technical quality for the project, the team would start writing unit tests for the new product.

This decision was something new in the whole organization, as unit tests had not really been written by any teams of the organization. In addition, none of the team members had much experience of writing unit tests, thus there was much to be learnt with only a small amount of internal guidance available.

The author, acting as a change agent in this action, tried to increase his own knowledge and experience as much as possible early on to be able to help the team with any obstacles they might be facing.

Team A moves to use Kanban flow

Team A really seemed to struggle with adopting Scrum, and especially the sprint based approach seemed to be a major issue. All of these issues clearly had a great impact on the team in many levels.

As a solution, the team decided to switch from sprint based development into flow based, in a very much similar fashion as Kanban method describes.

This change led into one of the key findings of this study considering that flow based approach would be superior to sprint based in this kind of environment and is discussed more in detail in chapter 6.2.2.

Organizational restructuring

There was a major restructuring in the imaging team level, including changes in management responsibilities. In the changes, a customer service unit for imaging products was moved as part of the same imaging team as the developers were in. In

addition, the earlier manager of the development team moved to become the director of the imaging team, when new team managers were named for both development and customer services teams.

At this point, the author was promoted to team manager of the development team, including the duties of the first-line manager for developers and a product manager. Some other of the author's new duties and responsibilities included for example the organizing of the development activities, processes and practices.

Test engineer joins into the team

A test engineer joined to be a part of the development team.

Earlier all test engineers had been part of an organization wide QA-team, which was responsible for planning and implementing major system level regression tests and in addition provide separate projects resources and guidance in testing. An organization wide change was made that most of the test engineers left the QA-team and became members of specific imaging teams. This was to give continuity in product knowledge and domain understanding for the test engineers, and in addition, to enable teams to better develop their own testing practices.

Changes to the members of the team

There were few changes in the ranks of the development team that did take place in the time span of three months. First of all, two of the members left the team. One person, who had been in the team for only a year did not feel like home and wanted new challenges by joining another team in internal transfer. The other one had to be fired due to person related reasons. Both who left were members of Team B.

To fulfill the gap left by the two team members and to respond to a large amount of work in the development queue, three new developers were hired into the team. Two of them were assigned into Team B and one of them was assigned into Team A.

A special focus in hiring was given on earlier experience in writing unit tests, as it had become clear that such an experience would benefit the team's transition to produce products with improved technical quality.

The author's perception is that all the new members that joined the team outperformed their predecessors in the level of their competence as developers.

Also, the team members were clearly good matches to the team by their personality and were quickly accepted as part of the team by existing team members.

Team B starts to write unit tests

As the author was encouraged by unit testing experiences with Team A, he suggested unit testing to be taken into use by Team B as well. There was not any clear resistance in writing unit tests in Team B, which was most likely partially due to the fact that the new team members brought to the team before had existing unit testing experience.

Team B also set higher coverage targets from the beginning and boldly focused on finding improving the testing and tools used for it quite independently.

Team B moves to use Kanban-flow

Team B had been using Scrum for quite a while and was quite happy how it worked for them. The idea of getting rid of sprints and moving for a more flow-controlled approach was brought to them by the author as he had perceived Team A becoming much more productive with it than it had been with sprint-based approach.

There was some initial resistance in the team on taking the flow-based approach into use, and the author had to actually suggest it multiple times in different situations before the team decided to give it a try.

After the initial try, also Team B found flow-based approach to suit them much better than sprint-based, and they decided to keep it.

First customer release for product M

The product M was released into production use for the first customer. The technical quality of the product seemed to be extremely good and in period of first five months of production use no customer reports were filed that would be due to technical defects. This was a major change with the experiences from earlier product releases.

5.5 Final state

5.5.1 Organization and environment

The organization as whole as well as the team's internal structure experienced some major changes during the period of the study. Some of these changes were initiated by actions taken in the scope of the research and some were external.

Figure 4 shows the main organizational structure of the team in the end of the research period.

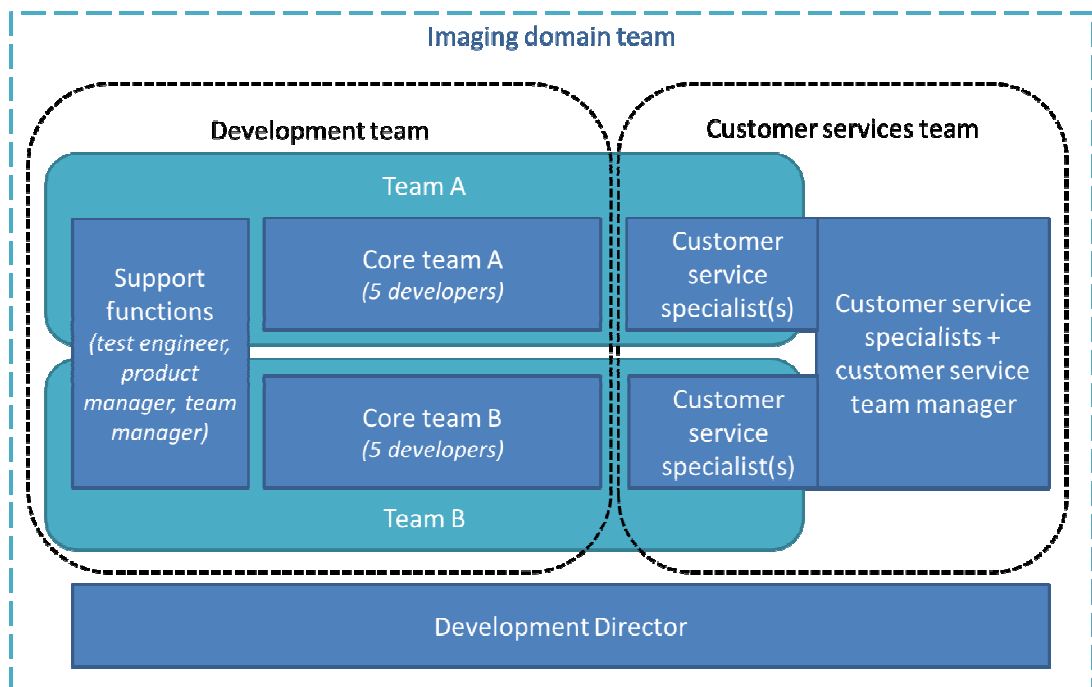


Figure 4: Final team structure

One of the major external changes was that imaging domain team was formed, in which case both development team and customer services team were assigned under same director. The newly appointed development director was a previous team manager of the development team and the author took the position of team manager of the development team.

One other clear change was that as a team manager, the author decided to form two core teams to give continuity for the team between projects. Both of the teams were assigned responsible for a certain set of the products as a team, instead of just individual responsibilities. These core teams were taken straight forward from the earlier team separation that developers had from recent project teams.

It was also decided that the rest of the development team members would support both teams as they were needed by their special role by both teams. These were team manager, product manager and the newly appointed test engineer.

It was also decided that a specific customer service specialist would join the daily activities for given periods of time depending on need. In case of a new development project they would participate in daily activities for the period of projects and for maintenance work they would join for the period that they were needed for that specific case.

In addition, there were some changes in the memberships of the development teams. From core team B, two members left during the period and two new persons joined the team. One new developer also joined core team A. In addition, the test engineer joined the development team and took a supporting role for both teams.

5.5.2 Practices and processes

In the end of the study period the practices and processes of the teams had gone through a major evolution.

A large amount of both major and minor changes were gone through in practices and processes during the research period. In this chapter some changes that seemed to have the most remarkable effect on the teams as a whole are described.

Both teams had moved to team based approach, now having team responsibility for a set of products instead of having individual person level maintenance responsibilities. Both the development work and maintenance work were tracked on teams' Kanban boards. Both teams had their own Kanban boards with some variation on the flow phases. This was a very significant change to improve the visibility to the work and decrease the amount of conflicts between development and maintenance work.

Both teams had replaced sprint based approach with Kanban flow based approach. The work in progress was limited as part of controlling the flow.

The daily meetings of the both teams had also gone through a change from traditional Scrum approach of three questions to going through all the items in the flow.

The teams had replaced sprint demos and sprint planning with user story level variants, which matched better with the flow based approach.

Both teams still had regular retrospectives.

Both teams were actively writing unit tests as part of their development activities.

Both teams also had a role similar to Scrum Master; a facilitator to take care of maintaining Kanban board and responsible for facilitating daily meetings.

5.6 Summary

The research was implemented in a two-year period consisting of two software development teams working closely in the same organization.

The teams went through both structural changes and changes in the practices. Some of the changes can be considered as more major, bringing some remarkably big change to the practices of the team. In addition, nearly hundred smaller changes were gone through to fine tune the teams' processes and practices.

During the time, multiple separate surveys were taken the two of which were run twice and are used to provide trend data of the teams' progress. Some of the surveys were not particularly run only due to research, but were part of teams' usual work and related to management trainings of the author. This data, however, provided valuable backbone for the research. In addition, a more specific survey was run to focus solely on how the team felt the effect of the changes made during the research period.

Cross-referencing these surveys provides the results from different approaches, and therefore increases the validity of the research by eliminating possible survey specific issues.

6 General results

6.1.1 Overview

The main goal of the research project from the organizational perspective was to improve teams' performance and produced quality while maintaining a high level of happiness.

It is the author's perception that the goal was clearly met, which is elaborated more in the details of the different sources of information as it is gone through in the following chapters and in the conclusion.

In addition to the author's perception, the results of the surveys initially described in chapter 5.3 are analyzed. These surveys also support the author's perception that the goal was met.

6.1.2 Author's perception

6.1.2.1 Overview

As stated in earlier chapters, the author was an active participant inside the team in which role he also formed his perception of the gained advantages.

The author was actively acting as initiator of introducing many of the practices and had earlier experience, and bias, on using Agile practices. This is something to take into account when evaluating the author's perception as part of the research.

The perception of the author can also be seen in his answers to the team survey, however, it is also gone through on a more general level instead of just focusing on individual practices.

6.1.2.2 Quality

From the author's perspective, quality was to be one of the essential cornerstones that also happiness and higher performance required. A large focus was put on the quality part.

One of the most important steps to take, in the field of quality, was to take automated unit tests into use. If considering one single practice on the quality side, the unit testing seemed to have the most impact on improvement of the overall technical quality of the products, providing improvement also in structural matters of the code in addition to providing improved test coverage for regression testing.

It also seemed that the management and team structure based changes were crucial for the improvement of the quality. The management commitment and focus for improving quality did clearly enable the team to produce better quality.

Structural changes in the team did also have a remarkable effect on quality. There were some changes in the personnel of the team as two people left the team and three new

joined. Strong focus on the new recruitments was given on their capability to produce quality code as well as preference and professional attitude towards aiming at it. This seemed to have a great impact on improving produced quality of the team. The changes were mainly focused on Team B, in which the author could perceive a dramatically higher improvement in quality than with Team A.

The author also rates highly the value of the code analysis tool as improvement of the code. When the team took an analysis tool (ReSharper) into use, it seemed to boost the produced quality by helping the team to find possible issues more quickly. In addition to expected improvements, it also steered the team towards better common coding guidelines as well as amplified learning of individuals by providing them more information on which of their coding practices and habits could be harmful.

From the perspective of the author, both teams seemed to have remarkable improvement on the quality that they produced. When looking at the most important perceived individual practices, it can be seen that most of them are not really considered solely as Agile practices. However, from the author's perspective, it seems that tool and practices related quality improvements would not have occurred without the continuous improvement flow enabled by Agile practices such as workflow visualization and retrospective meetings. In addition, many Agile practices advocate taking advantage of test automation and tools to enable shorter cycle teams for providing small increments and often tools such as unit testing are strongly suggested by them.

6.1.2.3 Happiness

From the author's point of view, the level of happiness was high already when the research started. This had been a case for many years already and the team had stabilized quite a great deal to be very tight and closely knit bunch.

When starting to take different practices into use, both teams were eager to adopt Scrum; however, it was clear that Scrum did not work well for team A, in which case it

was very noticeable that the team's happiness was lowered temporarily. This seemed to stem from the fact that the team did not feel that they could really commit even for two week sprints due to other responsibilities. Team B was more satisfied with Scrum, and such decrement was not seen in their happiness due to the Scrum. Also, the movement of team A from Scrum to Kanban flow did seem to counter the negative impact on happiness.

It was also noticeable that after almost two years of intensive and continuous improvement of their own practices and processes, the team members started to get tired of the changes. This could have had an impact on the happiness, however, when the members of team A brought this matter up, a concrete decision to let things stabilize for a while was made, to give the team a chance to catch a breath. That does not mean that no changes were made after that, however, any major changes were put aside and focus was only on minor fine tuning of the current way of working.

It is also something to note that the author feels that departures of two team members were at least partially triggered by strong focus on Agile practices. The Agile practices, especially the ones that focus clear on the visibility of a team's work, tend to bring issues on the surface. In some cases those issues can be about the performance or commitment of individual persons, in which case the fact that those become visible might lead into a situation where their happiness is dramatically decreased and leads to the changes in the team. In this case it may be that it was a case with the two members that left, however, also that change did seem to boost the morale of the rest of the team. This indicates that the issues that were not visible before did have a negative effect for other members of team and the tables were turned when enough visibility was brought in. So therefore the author considers that such practices could lead to dramatic decrement of happiness for few individuals, however, that can contribute to the improvement of happiness for whole team level.

In the end, the author rationalizes that the high level of happiness was maintained during the research even though there were some periods of decrement during the time as well. It is also the author's impression that in the end of the period the happiness was higher than when the research was started.

6.1.2.4 Performance

From the author's perspective, there was improvement in performance of the team during the time of the research.

The performance improvement could be seen to occur related to many of the practices taken into use, some of which seemed to have an effect very quickly and some took a longer time before they started to have a real effect.

Practices such as workflow visualization and daily standup meetings clearly started to provide performance improvement after a very short adoption period when the team got familiar with them. The same can be said about some of the tools that were taken into use, such as ReSharper tool that integrates into development IDE and extends its functionality.

On the other hand, for example adoption of unit testing seemed to have quite a strong negative impact on the team performance at first and it took several months of learning before the team seemed to gain nearly the same daily performance that they had before. This was especially visible with team A that did not have any earlier experience of unit testing. Also, the performance of producing totally new code did never achieve same performance level for them; however, the author has the impression that the team's overall performance did gain increment as even though the team lost some time in writing unit tests as part of development, they also gained long term performance increment in, for example decreased need for manual regression testing as well as due to improved maintainability of developed code.

The author's impression is that in the end it was essential that the team had both quick performance improvements but also improvements that would enable high performance in the long term. If only issues that had quick effect, such as daily meetings and workflow visualization, would have been taken into use the initial performance increment would have been even bigger than it was now. However, the author's opinion is that it would have been dramatically decreased during time if enough attention had not been given to long term quality based improvements, as the increased effort added to maintenance work of new product would have swallowed the majority of the improvement gained from other sources.

6.1.2.5 Summary

The author's perception is that the majority of the Agile practices seemed to have some amount of positive effect on all quality, performance and happiness. The main business goal of improving quality and performance while maintaining a high level of happiness was met.

Some specific practices also seemed to have a negative effect on some of the three but could have a high positive impact on the rest.

It would also seem that some of the practices would focus more on long term improvement and have an initial negative impact. One of such examples is the starting of writing unit tests, which initially decreased performance due to the time developers needed to spend on them while learning to write them; however, turning towards positive over the time as writing them became more routine and they started to provide true safety network and improved maintainability.

6.1.3 Team Practices Survey

6.1.3.1 Overview

The team practices survey was conducted about four months after the research period itself had finished, to give members of the team some more time to get experience with the latest changes as well to evaluate them better.

In the survey, some of the changes during the period were picked by the author for the team to rate. The rating was carried out by valuing each given change in terms of effect to performance, quality and job satisfaction. There were five possible values for rating: major negative effect, minor negative effect, no effect, minor positive effect and major positive effect.

Table 1 presents the calculated averages based on each of the change. The values were calculated so that the numeric values are given in scale -2 to 2, from major negative to major positive.

8 out of 11 team members responded to the survey. The author also responded to the survey; however, those values are separated from the team's survey and are used to compare the author's perception to the perception of the team members.

The number of answers by respondents to a single question varied between 4 and 8, due to the fact that some of the changes were such that happened when not all the team members were in the team yet. In those cases it was encouraged to leave that question unanswered.

Table 1: Team survey averages

Change	Perf.	Qual.	Job sat.	Total
1. Moving from sprint based approach to Kanban-flow	1.80	1.33	1.83	1.65
2. Introduction of more extensive build automation	1.50	1.38	1.25	1.38
3. Introduction of daily meetings	1.57	1.29	1.14	1.33
4. Introduction of virtual taskboard for projects	1.42	1.14	1.43	1.33
5. Changes to team's members	1.33	1.33	1.00	1.22

6. Introduction of ReSharper as a tool	0.75	1.63	0.75	1.04
7. Changing daily meeting structure from Scrum-style to Kanban-style	1.00	0.71	1.42	1.04
8. Replacing sprint demos with user story based demos	1.29	0.58	1.14	1.00
9. Introduction of developer cross-testing	0.88	1.50	0.50	0.96
10. Organizing team into two core teams and supporting functions	1.33	0.83	0.60	0.92
11. Organizational change: Customer services as part of imaging team	1.00	0.67	1.00	0.89
12. Introduction on product backlog	0.71	0.86	0.71	0.76
13. Changes in management positions	0.67	1.00	0.33	0.67
14. Introducing monthly meetings	0.50	0.75	0.63	0.63
15. Defect root cause analysis meetings	0.57	1.14	0.14	0.62
16. Transition from using project task board to team task board	0.50	0.50	0.83	0.61
17. Introduction of labday	0.38	0.38	1.00	0.59
18. Introduction of user stories	0.57	0.43	0.67	0.56
19. Introduction of WIP limits on the task board	0.71	0.43	0.43	0.52
20. Introduction of workflow phase based definition of dones	0.14	0.86	0.33	0.44
21. Changing retrospective meetings more structural	0.29	0.43	0.58	0.43
22. Introducing definition of done	0.00	1.13	0.13	0.42
23. Mapping team values	0.33	0.29	0.29	0.30
24. Defining the state of the team and setting common goals	0.13	0.36	0.36	0.28
25. Defining negative testing as part of developer testing	-0.25	0.75	0.25	0.25
26. Introduction of sprint demos	0.14	0.44	-0.14	0.15
27. Introduction of unit testing	-0.88	1.50	-0.25	0.12
28. Introduction of retrospective meetings	0.43	0.43	-0.58	0.09
29. Introduction of Scrum sprints	0.25	0.50	-1.00	-0.08

As it can be seen from the table, most of the remarkable changes that the team faced during the time of the research period, be they initiated by team itself or by wider organization, did have a positive effect to the performance, quality and job satisfaction.

It is also worth taking a note that there seems to be clear top five of the changes that were considered most important. Of those five, three can clearly be considered as suggested Agile practices. In addition, extensive use of build automation is also

something strongly advocated by Agile approaches as it is essential in implementing continuous integration.

Therefore, on the general level, the survey supports the fact that the team perceived that Agile practices had a remarkable positive influence on the performance, quality and job satisfaction.

There are also some important findings that can be seen from the given table and are discussed more in-depth in their own chapters.

One of those findings is the fact that there indeed seem to be some Agile practices that are exceptionally good in improving all three: performance, quality and job satisfaction. This finding is gone through more in-depth in chapter 6.3.3.

It is also worth to take a note that many of the highly rated changes were related to practices with a strong bias on improving visibility and communication considering work and workflow, which is discussed more thoroughly in chapter 6.2.3.

Table 2 presents the author's responds to the survey.

Table 2: Team survey, author's perception

Change	Performance	Quality	Job satisfaction	Total
1. Moving from sprint based approach to Kanban-flow	1	1	2	1.33
2. Introduction of more extensive build automation	1	0	1	0.67
3. Introduction of daily meetings	1	1	2	1.33
4. Introduction of virtual taskboard for projects	2	1	2	1.67
5. Changes to team's members	2	2	2	2.00
6. Introduction of ReSharper as a tool	1	2	2	1.67
7. Changing daily meeting structure from Scrum-style to Kanban-style	1	0	1	0.67
8. Replacing sprint demos with user story based demos	0	0	-1	-0.33
9. Introduction of developer cross-testing	1	1	0	0.67

10. Organizing team into two core teams and supporting functions	0	0	0	0.00
11. Organizational change: Customer services as part of imaging team	0	1	0	0.33
12. Introduction on product backlog	1	0	1	0.67
13. Changes in management positions	1	2	1	1.33
14. Introducing monthly meetings	0	0	1	0.33
15. Defect root cause analysis meetings	0	1	0	0.33
16. Transition from using project task board to team task board	1	0	1	0.67
17. Introduction of labday	0	0	0	0.00
18. Introduction of user stories	1	0	1	0.67
19. Introduction of WIP limits on the task board	1	0	0	0.33
20. Introduction of workflow phase based definition of dones	1	1	1	1.00
21. Changing retrospective meetings more structural	0	0	1	0.33
22. Introducing definition of done	1	1	1	1.00
23. Mapping team values	0	0	0	0.00
24. Defining the state of the team and setting common goals	0	1	0	0.33
25. Defining negative testing as part of developer testing	0	0	0	0.00
26. Introduction of sprint demos	1	0	0	0.33
27. Introduction of unit testing	0	2	0	0.67
28. Introduction of retrospective meetings	0	1	-1	0.00
29. Introduction of Scrum sprints	1	2	-2	0.33

As it can be seen from the table, the author had very similar view with team. This can be seen especially clearly when taking a look at the author's top five list, in which four of them are shared with a similar list from the responses of the team members.

6.1.3.2 Performance

From the perspective of performance, it was clear that there were lots of changes that teams considered to having had positive impact for the quality and few practices with seemingly negative impact.

Figure 5: Team survey spread for performance presents the survey results from the perspective of the performance.

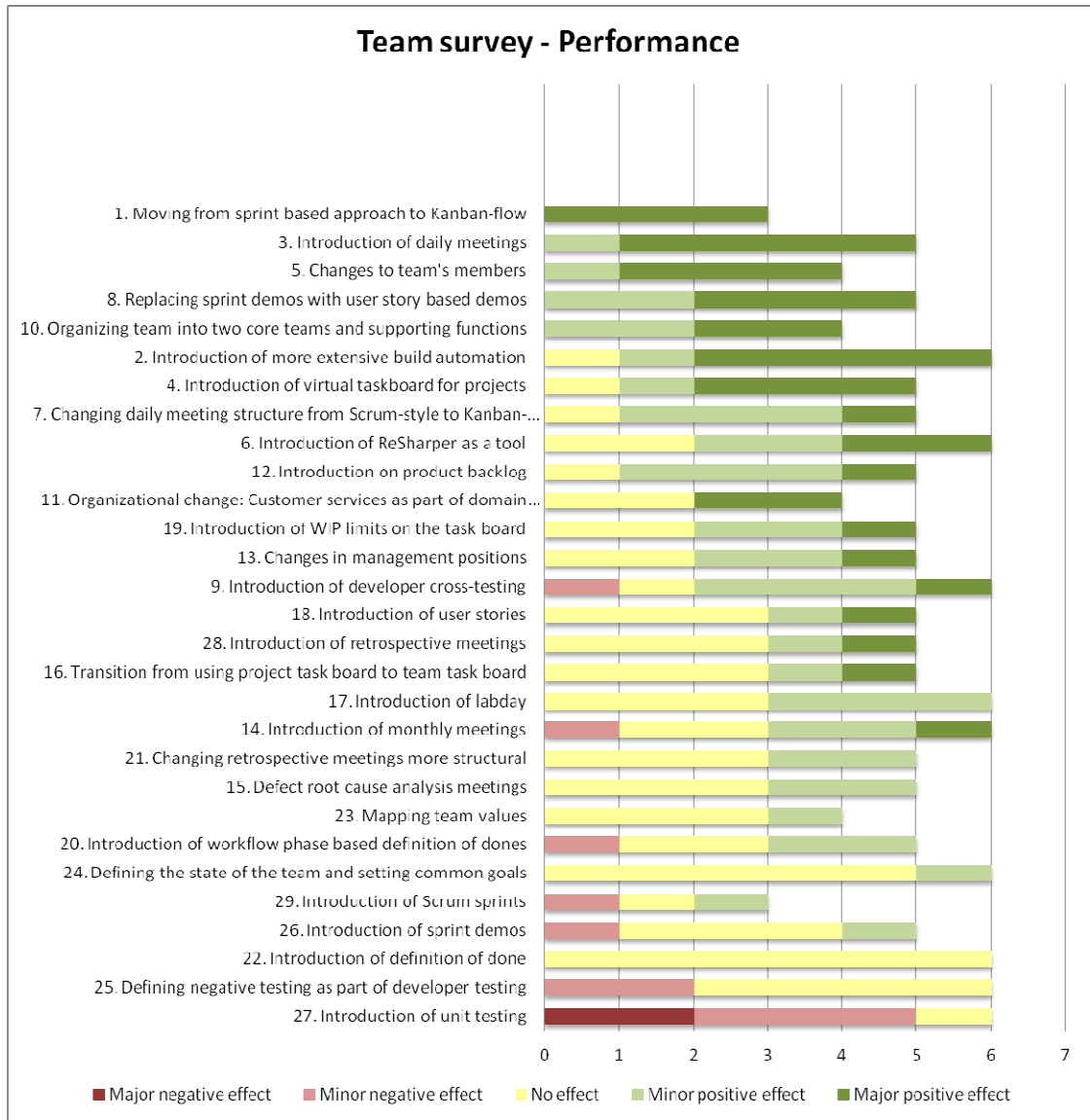


Figure 5: Team survey spread for performance

There are five changes that all of the responders have considered to have a positive impact, of which the top three are something that all, or nearly all, have considered to have a major positive impact.

If the top five from Agile perspective are observed it can be seen that actions related to switch to use Kanban flow, such as 1 and 8, are strongly involved. In addition, the daily meeting is seen extremely important for performance.

Another group that is highly present in the top five of the list is about organizational matters, such as changes to team structures. This is very much in line with the traditional understanding of what in general are the requirements for forming high performance teams, getting the right persons in the bus and setting clear borders inside of which a team can act freely.

On the other hand, when looking at the bottom rows of the list some changes are visible that the team seemed to consider harmful for the performance. Especially introduction of the unit testing was considered to have a negative impact on the performance. The author disagrees on this with the team's perception if considering long term gain for performance, however he agrees on that initial effect when only short term effect is considered. Therefore, it would seem that the team's perspective on answering this survey could be mostly from short-term perspective.

Also, including negative tests as part of developer work, instead of being done by test engineer as before, was considered to have some negative impact. This most likely reflects the fact that it was considered as new work to be done for the team as it was not earlier done on such a level that would be required to reach good quality.

One interesting matter to consider as well is that many Scrum-related issues were considered to have no impact or even a negative impact on performance. Matters such as Scrum sprints, sprint demos and definition of done were considered as a waste from the perspective of performance. This is quite controversial with the promises that Scrum makes considering performance.

6.1.3.3 Quality

When looking the survey results from perspective of the quality, there is big number of changes that were considered to have strong positive impact by both of the teams.

Figure 6: Team survey spread for quality represents team survey from perspective of quality.

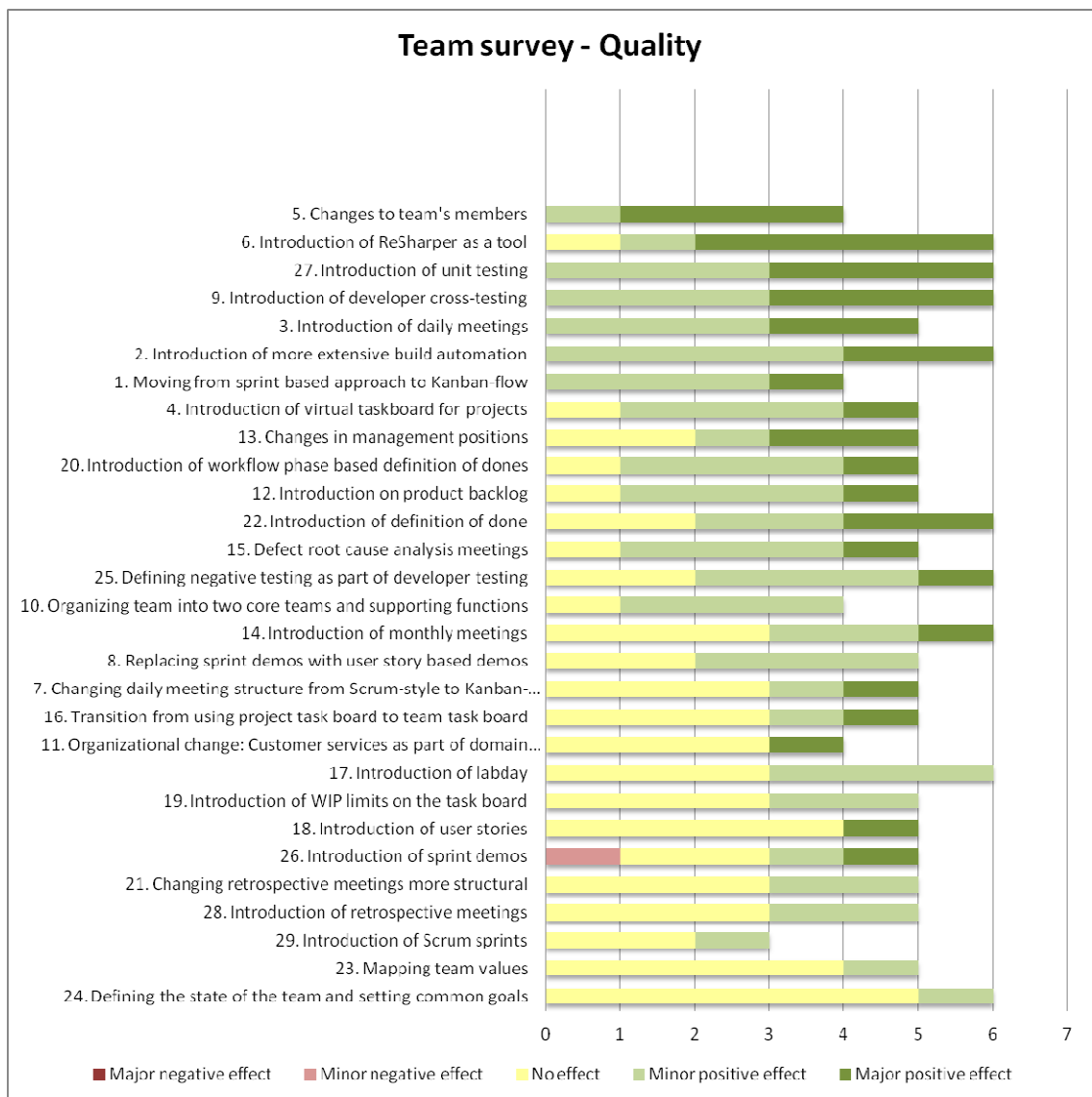


Figure 6: Team survey spread for quality

When taking a look at the changes that all or most the team considered of having had positive impact, a concrete set of Agile and Lean practices can be found such as:

- Introduction of unit testing
- Introduction of daily meetings
- Introduction of more extensive build automation (in practice moving towards continuous integration due to that)
- Moving from sprint based approach to Kanban-flow

It is also worth considering that many other highly rated matters were triggered by the usage of Agile practices. Some examples reflecting this are that

- Developer cross-testing was introduced through Kanban-flow as it visualized bottlenecks in testing and turned the team to consider options in improvement of that.
- Author's impression is that changes in teams' structure were also partially triggered by additional visibility that was provided by Agile practices, leading non-dedicated team members out of their comfort zone and having a strong impact on their departure.

6.1.3.4 Job satisfaction

Job satisfaction perspective of the survey can be clearly considered to be connected for the happiness part of the research. When looking at the results from perspective of job satisfaction, a big number of positive changes and also some negatives can be seen.

Figure 7: Team survey spread for job satisfaction represents team survey from perspective of job satisfaction which can be directly connected to happiness.

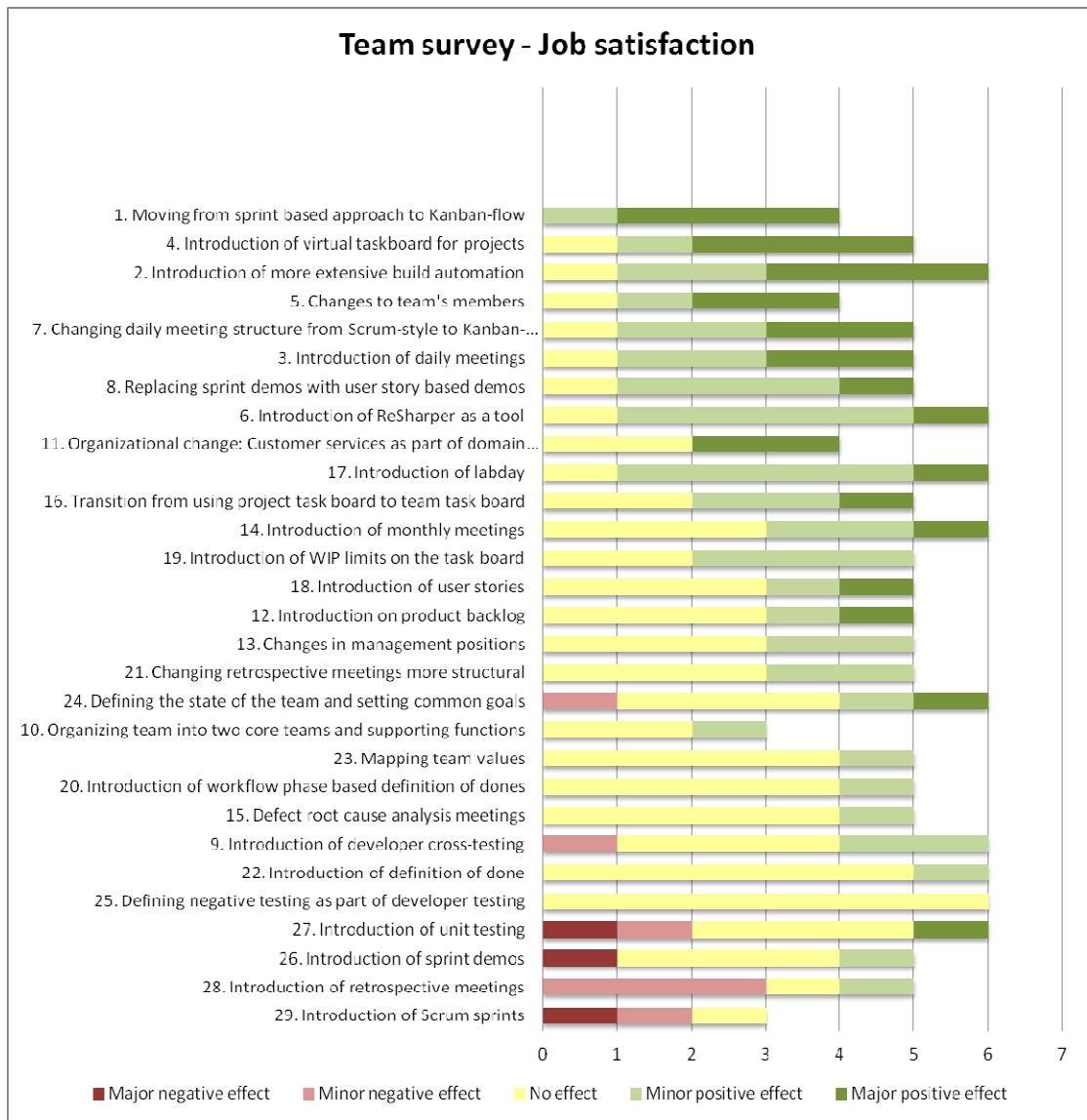


Figure 7: Team survey spread for job satisfaction

When looking at the list of changes, a large number of different actions can be seen that most of the team members have considered to have had a positive impact on their job satisfaction.

In the top five of the list, four Agile changes can be clearly seen, and in addition, the changes in the memberships of the team can also be considered at least partially to be triggered by the focus on Agile practices.

It can also be seen that team members considered that some of the actions with highest impact were those that involved the transition to use Kanban method.

Also, some practices that were provided by Scrum, such as daily meetings and usage of task board are rated very high.

When looking at the actions that seemed to have most negative effect, Scrum practices are also strongly present.

Sprint-based workflow was seen dramatically bad for job satisfaction, which was also the reason to try out the flow based approach from Kanban method.

Retrospective meetings are also ranked very low. The author is somewhat surprised by this, even though it was clear that team members often felt that retrospectives add little value. The author's opinion differs in this, since most of the major changes considered having a high positive impact were taken into use through retrospective meetings. However, it is hard to say why team member perspective differs so much and if there was be a good replacement for retrospectives to provide a chance to focus on more major continuous improvement.

6.1.3.5 Summary

Many actions were taken that had positive effect on one or more of the three main focus areas: performance, quality and happiness.

It seems that especially flow based work control such as described in Kanban method and related activities had a remarkable positive effect on all measurements.

In addition, some of the Scrum practices such as daily meetings were highly rated.

The Scrum sprints seemed to have a strong negative impact on job satisfaction, which also seemed to affect performance and quality as well.

One emergent finding is based on experience that in this kind of environment Kanban flow is superior to Scrum sprints and is discussed more in detail in chapter 6.2.2.

Also, the set of practices that are mainly focused on visualizing and controlling flow of work were found to have high impact in performance, quality and happiness. This finding is gone through more in detail in chapter 6.2.3.

In addition to the positive impact gained from Agile practices, structural changes to the team and management seemed to have had a remarkable positive impact.

In addition, there were some specific practices that seemed to have a negative effect on one of the measures while having a remarkably positive effect on others. A prime example of this would be unit testing, which the team considered to have a negative impact on performance but a highly positive impact on produced quality. In this case, the author's impression is that teams were giving bias on short term performance, as the performance gains from unit testing would be more of long term as they would have more effect on maintainability than performance during the initial development project.

6.1.4 TPD™

6.1.4.1 Overview

Team Performance Diagnostics (TPD™) survey is a survey that was carried out by an external company, SIA Group Inc.

TPD™ was performed as an electronic survey responded to by team member and manager through a web interface.

The survey is an anonymous on the part of the team members. The manager's answers are separated from the team's answers and used to compare the manager's view to the view of the team.

The survey is made through negative questions, consisting of statements that are negative for team performance. Then the percentage of “no”-answer is calculated to get a percentage based rating.

The TPD™ survey was run twice, with half a year break in between. The survey was taken in spring and autumn of 2012.

The different categories in the survey are following:

- A) Balanced roles
- B) Clear objectives and purpose
- C) Openness, trust, confrontation and conflict resolution
- D) Co-operation, support, interpersonal communication and relationships
- E) Individual and team learning and development
- F) Sound inter-group relations and communication
- G) Appropriate management / leadership
- H) Sound team procedures and regular review
- J) Output, performance, quality and accountability
- K) Morale
- L) Empowerment
- M) Change, creativity challenge the status quo
- N) Decision-making and problem solving

The letter ‘I’ is intentionally left out from indexing in the survey to ensure that it does not get mixed with the letter ‘L’.

There was increment in almost all of the categories, between the two surveys. Only category “Change, creativity to challenge the status quo” seemed to have had small decrement during the period of the half a year.

Figure 8: TPD™, high level trend presents the high level comparison of trends between the two surveys.

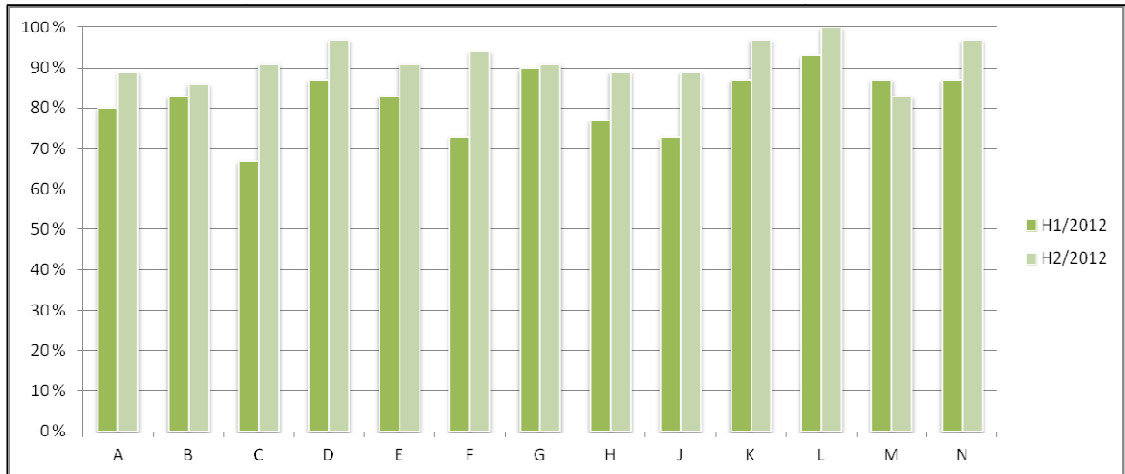


Figure 8: TPD™, high level trend

The most major increments, from highest to lowest, are on

- Openness, trust, confrontation and conflict resolution
- Sound inter-group relations and communication
- Output, performance, quality and accountability
- Sound team procedures and regular review

A closer look into details of those major improvement areas, as well as the one category where there was decrement is taken. In addition, the Morale category is discussed in more detail, as it is very closely tied to the goals of maintaining high level of happiness. The Empowerment category is also discussed, as empowerment is one of the core concepts in the most Agile approaches and has reached the maximum score in the survey.

6.1.4.2 Openness, trust, confrontation and conflict resolution

Openness, trust, confrontation and conflict resolution can be considered as essential for Agile teams, but also well functioning teams in general. Ability to continuously change and work together as a team instead towards individual goals need a high level of trust.

Figure 9: TPD™ - Openness, trust, confrontation and conflict resolution trend presents the results and changes in this category.

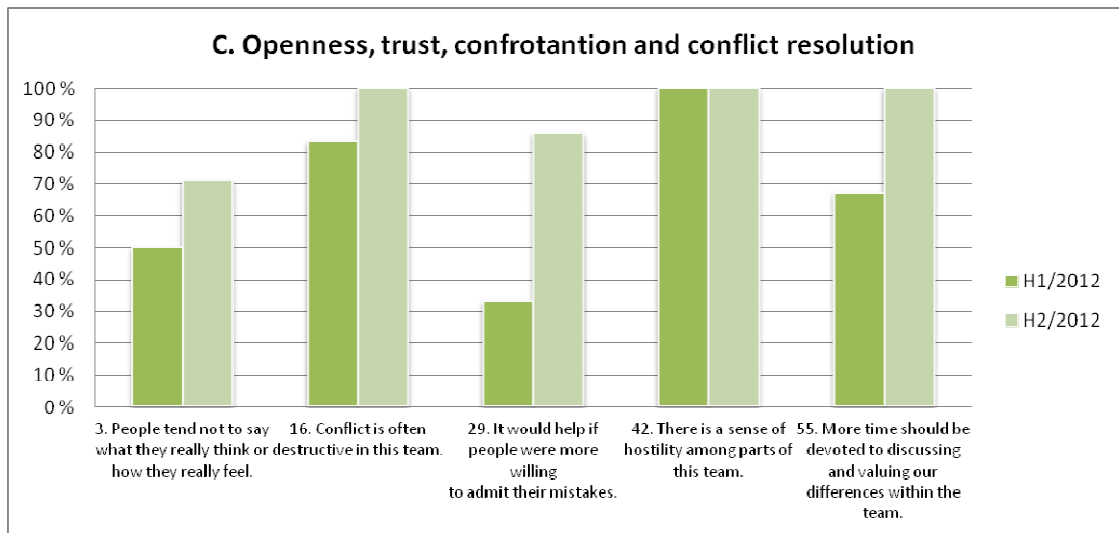


Figure 9: TPD™ - Openness, trust, confrontation and conflict resolution trend

As it can be seen, there is increment in all except in question 42, which was already on maximum value in the first survey.

When considering the possible reasons for the increments, there are few main reasons that the author expects them to be due.

First of all, the change in the membership structure of the teams had a positive effect. This would be mostly reflected in questions 3, 16 and 29. Earlier there was one team member who was not happy with his role in the team and one team member whose personal issues had also effect on the trust inside the team. The fact that they were replaced with team members who were excited about the new opportunities and also

had personalities preferring more open communication, are some reasons that had a healing impact for the whole team even in short term.

The second reason perceived by the author is the setting up of practices for continuous improvement and improved visibility. These practices include such as regular retrospective meetings and visualizing and controlling workflow in a way that makes issues visible to be solved early on.

Clearly, the visibility is one of the core topics in all of these changes, and it can be even suggested that the changes in memberships of the team were initially triggered by the improved visibility which brought the issues with two earlier team members to the surface and made them, manager and other team members to act on the matter.

6.1.4.3 Sound inter-group relations and communication

This category is not tightly tied to the exact focus of this study, as it studies more the interaction with a wider organization; however, it is worth to note that acting in a wider organization can have an impact on the team performance in the sense that bad communication can leave the team in lack of information or with false information that leads them to do additional work.

Figure 10: TPD™ - Sound inter-group relations trend presents the results and changes in this category.

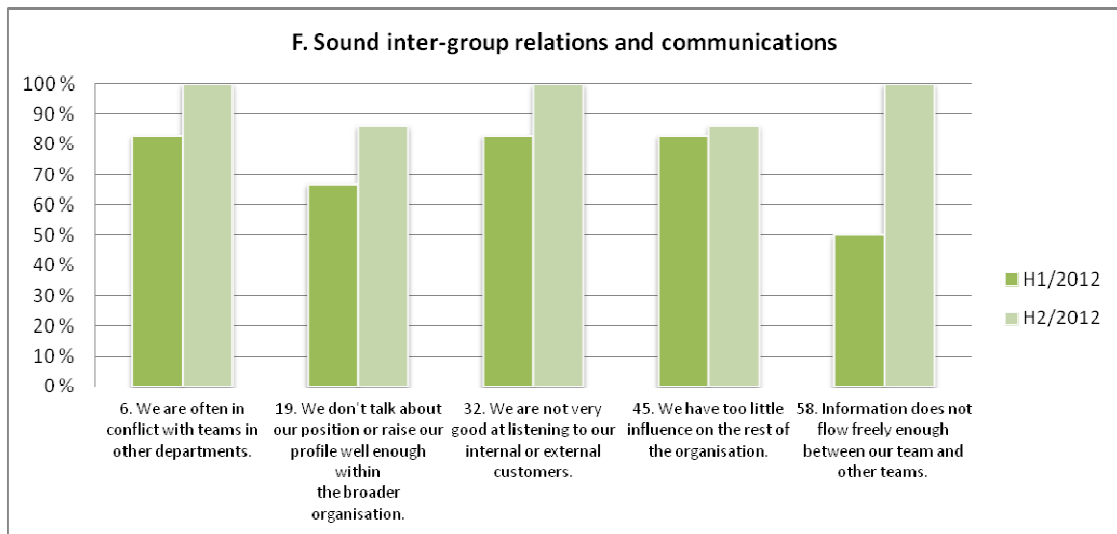


Figure 10: TPD™ - Sound inter-group relations trend

As it can be seen, there is increment in each of the questions related to this category, with most major change in question 58.

In general it can be evaluated by these results that the team perceived improved communication and information flow with the rest of the organization, which in some cases seems to be remarkable.

The author's impression is that most of these changes are not mostly due to the new internal practices taken into use, however, more due to the fact that this was one of the focus points that the team decided to improve when they discussed their improvement goals in the team meeting. In addition, the author as a manager, consistently encouraged team members to act as part of organization wide virtual teams and also spend their time in sharing their knowledge wider in organization. This caused a minor increment to the work the team did outside of their own team, however, this seems to have had a major effect in inter-team communication.

6.1.4.4 Output, performance, quality and accountability

Performance and quality are one of the main focuses of this research and therefore this category has very some important aspects to focus onto.

Figure 11: TPD™ - Output, performance quality and accountability trend presents the results and changes in this category.

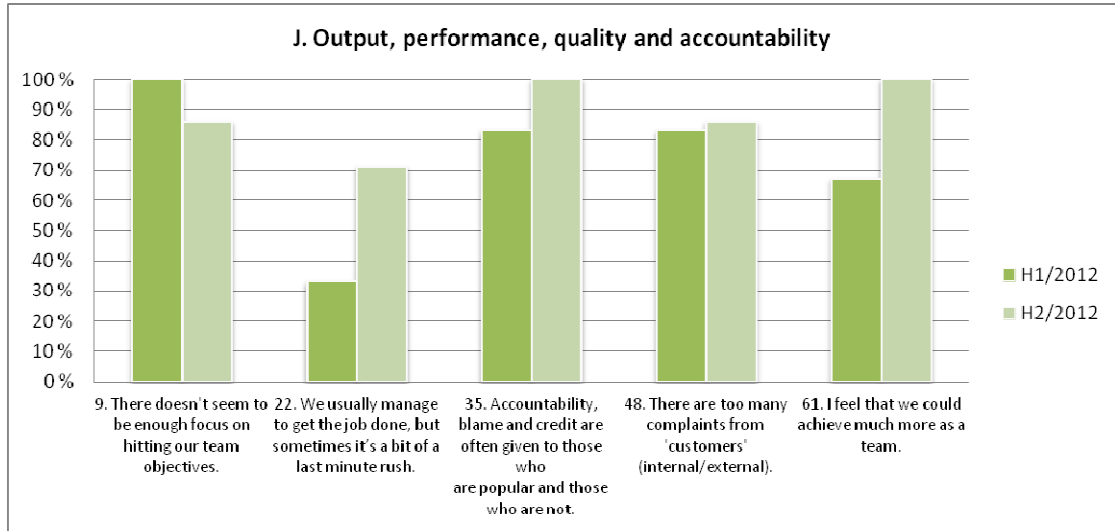


Figure 11: TPD™ - Output, performance quality and accountability trend

When looking at the level of the topic, this category could be considered one of the major categories from the perspective of the scope of the research. In the author's opinion, the questions 9, 22 and 61 are clearly tied to the perception of the team's performance. The question 48 is tied to the general product quality. The question 35 would seem to be more tied to the internal cooperation of the team members and management.

There is increment in almost all the questions, with exception of question 9 where there is small decrement.

The author's perception is that the decrement of question 9 would be connected to the fact that team B had changed to use Kanban flow instead of sprint based approach, which still was causing some worries for some team members considering the feeling of needing tightly set schedules instead of aiming to do the best that team can. On the other hand, this change would also seem the most obvious reason for increment in the question of 22. The constant rushes to finish the sprints was replaced with more

constant flow and therefore decreased the feeling of constant rush and possible feelings of failures due to missing some sprint goals.

Question 61 supports the fact that the team considers that they have made major improvement as none of them considered anymore that they could achieve much more as a team. The setting of this question can be somewhat problematic, however, as the term “much” could be interpreted quite differently between different persons.

6.1.4.5 Sound team procedures and regular review

Sound team procedures and regular review category is focused much on the team’s perception of continuous improvement, usefulness of the processes and information sharing. All of these are important from the perspective of this study, as they are directly related to many of the advantages to be gained through Agile practices.

Figure 12: TPD™ - Sound team procedures and regular review trend presents the results and changes in this category.

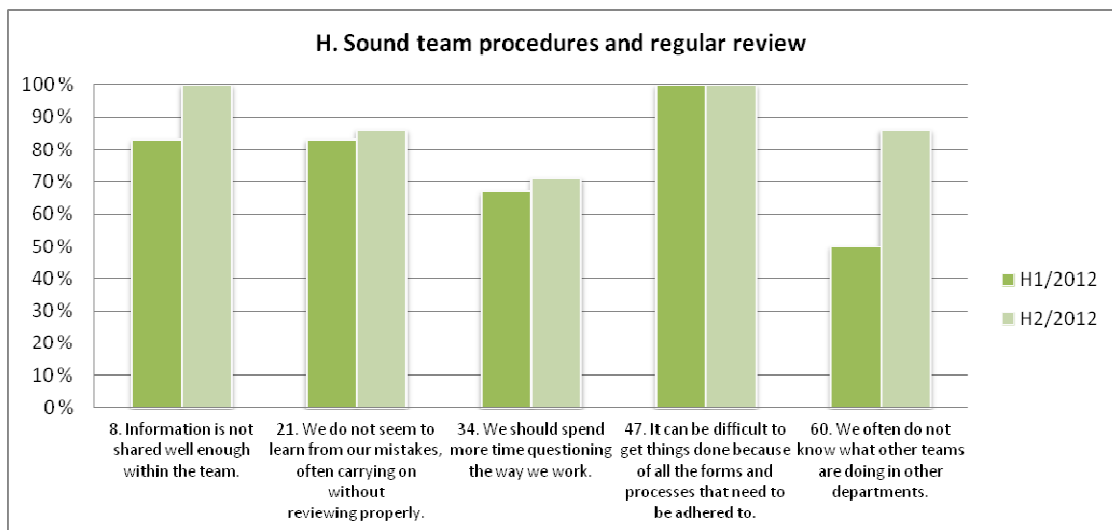


Figure 12: TPD™ - Sound team procedures and regular review trend

When looking at the results, it can be seen that there are no decrements on the values. The increment in 21 and 34 has, however, been smaller than the author expected them to be.

The question 34 in particular, which would indicate that many of the team members still seem to think that more time should be put on questioning the way that team works. This is in conflict with the discussions that the author has had with members of the teams, which were indicating to author that the team would be more likely willing to decrease than increase the amount of time used to that.

The most remarkable changes are on questions 8 and 60, both of which clearly indicate that communication and visibility, for both internal and external parties, have improved dramatically. On the internal part, the author would give most of the credit to such practices as daily meetings and teams' task board. The external part could be considered as increased focus on being more active in organization activities.

6.1.4.6 Change, creativity to challenge status quo

This category is chosen as important by the author due to the fact that the focus has been strongly on creating an environment that enables continuous improvement. In addition, most of the Agile practices are essentially focusing on enabling the improvement through continuous improvement.

Figure 13: TPD™ - Change, creativity to challenge status quo trend presents the results and changes in this category.

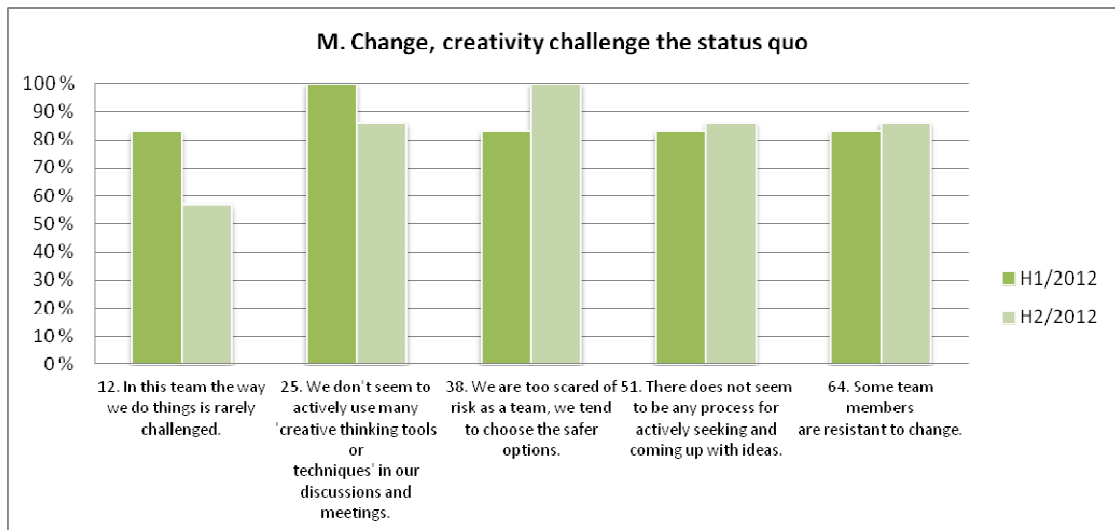


Figure 13: TPD™ - Change, creativity to challenge status quo trend

In this perspective, it comes as a surprise for the author that questions 12 and 25 have seen noticeable decrement. Especially the question 12 did raise concerns, and the author wanted to discuss that with the team more in detail. The team was slightly amazed of the result as well, and their opinion was that it was a bad setting of question that had been hard to understand as English is not their native language. Therefore, it will not be given a great deal of attention.

The decrement of 25 can mostly likely be explained by the fact that there had been a period of many major changes before the first survey, after which the changes had become smaller. It is to be noted though, that there were few major changes on that period as well, however, in general they were more related to workflow than to new development practices. During earlier changes, the author had really taken plenty of more time in using different techniques to enable the change and now it seemed that the team was more ready for change and less effort was put on selling the changes for the team by author.

6.1.4.7 Morale

One of the important aspects to research is happiness which is very clearly tied to morale category. It can be seen that there is clear increment in this category, and most of the questions are getting positive responses from all the respondents.

Figure 14: TPD™ - Morale trend presents the results and changes in this category.

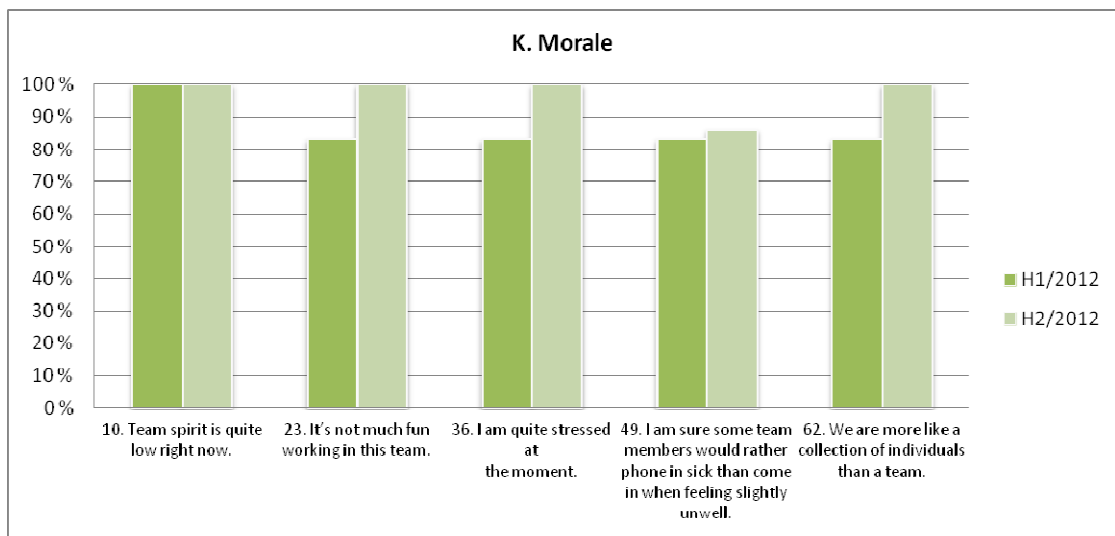


Figure 14: TPD™ - Morale trend

It can be seen that in overall the happiness seems to have increased, which is a major issue from the perspective of the research. Also, the stress levels of the team seem to have been lowered, of which the author expects that moving to flow-based approach has played an important part. Earlier, the sprint based approach clearly was creating non-healthy and artificial feeling of busyness. In addition, it can be seen that all the respondents now feel strongly that they are working as a team.

These results clearly indicate that the changes on the period have had positive effect for the happiness of individual team members and of the teams as a whole.

6.1.4.8 Empowerment

Empowerment is one of the core aspects of the most Agile approaches, and therefore it is essential to evaluate it in the sense that it gives an impression of how well that perspective of agility has been adopted by the team and author.

Figure 15: TPD™ - Empowerment trend presents the results and changes in this category.

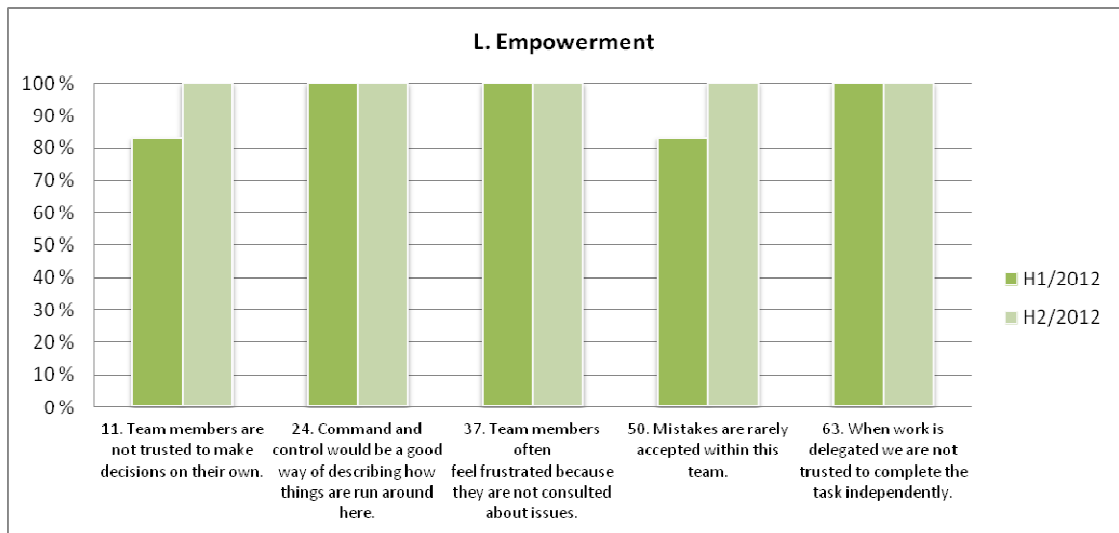


Figure 15: TPD™ - Empowerment trend

As it can be seen, the level of empowerment has been high for the team all the time, but it has even increased from the earlier. The increment has happened on questions 11 and 50, which both clearly indicate an increased level of trust and accountability inside the team.

This increment of trust and accountability can be considered to have many sources including for example changes of the members in the teams and improved visibility to work inside the team.

6.1.4.9 Summary

A clear trend could be seen between TPD™ surveys indicating that the teams' perception was that their performance had improved. Also quality and happiness related increment could be seen to some extent.

The results were very much in line also with the specific areas which go through characteristics that are often considered to be natural characteristics of Agile teams.

There was especially high increase in the category of openness, trust, confrontation and conflict-resolution. Partially the changes in this category can also be explained by changes to the teams' structures, however, increased visibility and cooperation are likely to have had an effect on this.

This general trend that is seen indicates clearly that there has been positive development inside the team during the period between the surveys.

6.1.5 Voice

6.1.5.1 Overview

Voice survey is Tieto's internal yearly employee survey that has quite a strong focus on job satisfaction, however, also on some aspects of learning and competence.

The survey was run twice during the period of the research and in these chapters a look is taken at the latter survey's results and change between the two years.

Generally, the survey categorizes the questions into different categories with internal comparisons. In this research, however, the focus will be more on individual questions than different categories.

6.1.5.2 Analysis

Table 3 presents the average of yearly voice results for year 2012 as well as change between 2011 and 2012 voice surveys, including items that have yearly change of 0.25

or higher. The rating of the survey is between 1-5, which is entitled in survey as to be between “Strongly Disagree” and “Strongly Agree”. The selection to include only items that have either positive or negative change of being 0.25 or more, is to focus on items which have had clear change and not just daily fluctuation. The list only includes high positives, as there was not a single question that would have had meaningful decrement of -0.25 or more.

Table 3: Voice results comparison

Question	2012	Change
... I constantly benefit from the knowledge and experience of others	4.67	0.97
How satisfied are you with your overall situation in Tieto?	4.38	0.68
How satisfied are you with your current job?	4.46	0.66
... we often try out new ways of working	4.83	0.63
In my team everyone takes responsibility for problems that arise in their work	4.58	0.58
Tieto is making a voyage I really would like to follow	4.33	0.53
In my team we make sure that new ideas are evaluated irrespective of who suggests them	4.59	0.39
... we encourage and support new ideas	4.58	0.38
I would gladly recommend a good friend to apply for a job at Tieto	4.26	0.36
In our team we always do our best to find solutions that would add value to the customers' business.	4.64	0.34
... we learn from our mistakes and continuously improve the way we do things	4.34	0.34
I would stay on at Tieto even if I were offered a similar job at approximately the same pay and benefits in another company	4.42	0.32
... we have an atmosphere of trust where we can openly talk about mistakes and disagreements	4.59	0.29
I feel that I develop and expand my competence at work	4.29	0.29
I feel I have good possibilities to make a career at Tieto	3.68	0.28
I feel content with my overall situation in Tieto	4.18	0.28
I believe Tieto will become one of the winners within its field	4.16	0.26
How satisfied are you with the processes available?	4.16	0.26
In my team we always try practical solutions to solve problems that arise at work	4.75	0.25

As it can be seen, the rating is extremely high in most of the values. This reflects the fact that the teams have had very high Voice results for many years in a row, indicating that team members have been very satisfied in working with their team and conditions.

When considering that, it is interesting to notice that even though the ratings of year 2011 have already been high, there has also been clear growth in some parts of the survey between the years 2011 and 2012.

As job satisfaction is one of the main focus areas of the Voice survey, many questions and increments related to job satisfaction are noticeable. A set of questions that clearly indicates the increment in job satisfaction includes, for example:

- “How satisfied you are with your overall situation in Tieto” with increment of 0.68
- “How satisfied you are with your current job” with increment of 0.66
- “Tieto is making a voyage I really would like to follow” with increment of 0.53
- “I would gladly recommend a good friend to apply for a job at Tieto” with increment of 0.36

In addition, there are some strong indicators that the team members perceive that the team now works better as a whole. This can be considered to indicate improved performance for a team as a whole and also continues to reflect a higher level of happiness as the team finds out their teamwork is more fulfilling. Such questions include for example:

- “I constantly benefit from the knowledge and experience of the others” with increment of 0.97
- “We often try new ways of working” with increment of 0.63
- “In my team everyone takes responsibility for problems that arise in their work” with increment of 0.58
- “In my team we make sure that new ideas are evaluated irrespective of who suggest them” with increment of 0.39

Considering the increasing trend, a suggestion can be made that Voice survey indicates the increase in categories of happiness and performance.

6.1.5.3 Summary

Voices survey is a yearly internal employee survey of Tieto, which mostly focuses on job satisfaction.

Changes between year 2011 and 2012 were compared and there was a clear increasing trend to be noticed. The highest changes were gone through and they give a clear indication that job satisfaction has increased during the period and the team seems to have improved its teamwork, which would also indicate performance gain.

Therefore, Voice survey seems to support the conclusion that the level of happiness and performance had a growing trend during the period of research.

6.1.6 Reliability and generality

When evaluating the reliability of the results and findings, we can separate findings into two categories. The first category is reliability of results about general performance, quality and happiness changes. Second category is the value of individual changes.

In both of the categories, we can consider the reliability to be very good.

In the first category, there is multiple kind of triangulation in a place. First of all there is a research method triangulation when qualitative data provided as author's perception through action research is triangulated with quantitative data from three different surveys. The surveys themselves provide triangulation working as different data sources around the same subject. Especially there is a high overlap on purpose of TPD™ survey and Voice survey, which of both of the surveys show very similar trend. This trend is also in line with author's perception. In addition, Team survey also shows that majority of the changes were considered positive by teams' members and therefore further support the results from other sources.

The second category considers individual changes, such as evaluated in Team survey.

Team survey is strongly in line with author's perception, especially in changes that teams

considered having highest impact. This triangulation leads also to reasonably high reliability for those results as well. In addition, it is further supported by TPD™ and Voice surveys, that point out positive general trend which is in line with the fact that majority of the individual changes were also considered to be positive by the teams.

When considering possibility to generalize the results, it has to be taken into account with all the findings that the number of teams involved was only two. In addition both of them worked closely in the same organization and author was participant in both teams during the research which also might have increased similarity between the teams. This can be seen to lower the generality of the research. It is still important to notice that there is also no evidence that would suggest that the findings would apply to other organizations as well, as the reliability in given environment was very high. Author personally thinks that the findings could be very applicable to other similar kind of environments, but also acknowledges further research in different organizations would be required before the level of generalization could be considered high.

6.1.7 Summary and conclusions

All of the quantitative data, as well as the author's perception, do show clear increment in performance, quality and happiness during the research period.

It can also be seen in different surveys, such as TPD™ and Voice, that the major improvements have seemed to occur in team development, growing trust and improvement how the team behaves internally. Having such a clear correlation between results of separate surveys is also clearly increasing validity of the increasing trend.

This is also supported by the author's perception and team practices survey, which shows that the majority of practice and tool changes were considered as positive by teams. It also further indicates that the majority of the practices that provided the greatest positive effect for all three are considered in general of being part of the family

of Agile practices. Therefore there can be seen a connection between improved quality, performance and happiness with the adoption of Agile practices.

The perceived growth of quality, performance and happiness also show that the company goal for research was fulfilled.

In addition, from the experiences and gathered data, support can be found for our purpose related findings and also some emergent key findings can be isolated.

The purpose related findings are elaborated on more in chapter 6.3, and emergent key findings are elaborated on more in chapter 6.2.

6.2 Emergent key findings

6.2.1 Introduction

This chapter describes such key findings of the study that were not expected by author based on earlier studies or theory.

The key findings are described in their own sub-chapters, in which each finding is described more closely.

6.2.2 Continuous flow is better than sprints for multi-product owning product development team

Overview

Continuous flow, such as provided by Kanban method and its WIP limits worked better for both teams than Scrum sprints. This seemed to stem from the fact that both teams also had other commitments than just the ones that were picked for any given sprint. These commitments could be for example unexpected maintenance work or 3rd line customer support.

As the teams were highly committed to do their best, the additional commitment to sprint scope that could so easily be violated by external factors did greatly degrade the happiness of the teams. In addition, there was no real benefit for the team on sticking to the sprints in this kind of environment where product owner was internal and actively participating in daily activities, due to which the sprint reviews lost most of their value.

This is a very relevant matter to consider due to the fact that Scrum with its sprint based approach has become widely popular and is taken into use in all kind of environments. Findings of this study indicate that it does not work in all kind of environments and can lead to decreased happiness and performance.

Story

Both of the teams that participated in the research were introduced with the Scrum practices before of the Kanban method.

Team A had earlier been working mostly with ad-hoc practices that had formed on the top of the waterfall model projects. Their daily work was very lightweight in perspective of any processes or formal practices. The Scrum framework was introduced to them by the author in the start of the new major project, where author was acting as a project manager. The practices put into place were initially greeted with positive excitement and common agreement that they could provide something that the team had been missing so far.

Even though the changes were welcomed, it soon became clear that adjusting to sprint based working was not easy in the given environment.

The fact that the team had multiple products to support, caused it to constantly fail its sprints. This was caused mostly due to the fact that in addition to the development project, the team had maintenance responsibilities for several other products. In addition, some of the team members had organization wide responsibilities which could

take some of their time in very short notice. All this caused the team to have very little predictability of the real effort they could be able to commit to the sprints.

The team also felt their commitment to sprints be very artificial sort and conflict with their commitment towards their whole product portfolio while also added needless overhead in form of regular planning meetings. In addition, the team did feel that too often the commitment to scope in such a short iteration would conflict with commitment to quality. This all caused negative stress for the team, clearly decreasing the happiness. The team did try sprint based working for three months, before it was very clear that in the given environment and for given team the sprint based work would have more disadvantages than advantages.

After three months of failures in sprinting, the flow based model was taken into use. The model was a slightly simplified version of what is defined in Kanban method. The model used takes use of visualizing the workflow through a virtual task board and having work in progress limits, however, for example following lead times or different service classes were not introduced for the team. The team was quick to adapt the new way of working, and was very satisfied with it. One of the best indicators of the team's increased happiness was that they also started to tell other teams in the organization of this new way of working that really worked well for them. This triggered some other teams in the organization to also start taking flow based approach into use.

The team has now been working with flow based approach for over year and a half and is clearly happy with it.

Team B had a longer history with Scrum and they had got used to working with it comfortably. Team B had only one product to maintain back then and therefore it was quite straight forward to commit only to the sprints of that product. Certain amount of problem was however caused by the need to provide 3rd line customer support for that product, which often was so urgent that it had to break the sprint. These problems

were, however, considered as mandatory nuisance by the team, and the team was determined to work with Scrum even with these little flaws in their sprint commitment.

In the case of this team, the switch for flow-based approach was initiated by the author. The reason to change in this case was due to the great success that Team A was having with such a method. In addition, there were some new projects starting that would increase the number of products that the team had to support, which seemed to cause the risk of the team spreading its commitment in many ways and having similar issues that Team A had with sprints.

When the author suggested the change, there was some resistance towards it. At first, the resistance was so hard that it was decided that the team would still go on with the sprints and postpone trying out the flow for some time. After the author consistently brought the flow based approach up in discussions, the team decided to experiment with it for a short period.

Most of the team adjusted easily to the new way of working, even though there still was some resistance from one team member. This resistance was mostly due to the fact that he was worried if the team as a whole could perform as well as there were no such a tight deadlines as the sprints had provided.

After the trial time for trying the flow base approached had passed, the team did not want to return working with sprints anymore. And now that the team has had more situations where it has multiple products to support and has to concentrate on those instead of a single project, the advantages gained by continuous flow have become very clear.

Conclusion

Using continuous flow of work, instead of sprints, increased happiness for both teams, with also a dramatic positive effect on teams' performance and the produced quality.

The most important environmental factor seemed to be the fact that the team could not focus solely on one project, but had to provide product support and maintenance as well.

Considering both teams it is also important to notice that even though sprint based commitment and reviews were removed with the sprints, many other Scrum based practices were still left in place: the practices that were kept included, for example were daily meetings and retrospective meetings. Reviews and sprint planning also just seemed to take a different form by becoming part of continuous flow instead of being formally fixed into time as they were with sprints.

This is the author's perception and in addition, it is clearly visible in the team survey, where the sprint based approach did get really low scoring from the team members while moving to the flow-based approach was seen to have a highly positive impact.

It is of course worth noting that Scrum defines that a team should have the possibility to commit only for the sprint during the time of the sprint, which these teams clearly could not reach. Having a possibility to only commit for the sprint would have most likely improved teams' success with sprints and eliminated some of the negative impact that the teams were seeing; however, it was very clear that the current organization and business environment could not have supported such a way of working, which is a common case when team is having multiple products to support.

Therefore findings of this study suggest that at least in the environment where a team has to commit to more tasks than a single project, a continuous flow of work approach is likely to lead to a higher level of happiness, higher performance and improved quality compared to sprint based approach.

6.2.3 The most valuable practices are those that improve visibility to the workflow and enable its continuous improvement

Overview

One of the most essential characteristics that all Agile methodologies and frameworks seem to have is their natural tendency to bring things more visible for everyone involved.

This study clearly highlights the value of visibility to succeed in all three: performance, quality and happiness. It also points out that specific practices that are described as part of some Agile frameworks or methodologies, deal extremely well in improving both visibility and communication. This is especially important when considering visibility and maintainability of team's workflow.

This finding is relevant for providing insight of which kinds of practices and tools the teams and managers should focus on early to gain the benefits of the improved visibility.

Story

Both of the teams got first involved with practices for improving visibility and communication when adopting Scrum framework. One of the first things taken into use were daily standup meetings as well as a virtual task board for sharing the teams' work. In addition, other meetings such as sprint planning, sprint review and retrospective meetings were taken into use, of which all can be considered as practices that aim to improve communication and visibility.

When the team practices kept on evolving and the teams moved more towards pull flow based approach for sharing work, many of the Scrum practices were abandoned. There were still some Scrum practices that teams wanted to keep on using and found highly useful, including for example daily standup meetings and task board. They did go

through some evolution during the time as well, which in both cases evolved towards following more the flow of the work and optimizing it instead of following the individual progress of each team member.

The evolution also led the teams to take into use many other Kanban method based practices such as work-in-progress limits for different phases of the work process.

Eventually the task boards of both teams did evolve into a status where they really clearly visualized the true workflow of the individual team. This was emergent in a way that even though the author expected that something such could happen, the practices were not taken onto that direction by his direct guidance, but instead the fine tuning suggestions were brought up by the members of the teams. This was seen by author as an indication that teams found those very valuable and therefore were more willing to put additional effort in further improving them.

It was also noticeable that when considering practices that formed around daily standup meetings or the task board, the team was much more active in suggesting improvements than with any other practices.

Visualization through task board and daily standup meetings also seemed to become the central points for continuous improvement of team practices and processes. The visibility they provided seemed to be essential in bringing up the issues that the team seemed to consider most important to deal with to improve the way they work.

Conclusion

While studying the team through both discussions and surveys, the practices that the team rated highest of all three; performance, quality and happiness, were ones that have strong impact on the visualization of the workflow.

Through the results of this research and personal experience outside of the research as well, the author suggests that visualizing workflow and continuously maintaining it

through daily meetings are essential tools for enabling continuous improvement process in any team. This is even more strongly highlighted in software development where one's work product is quite abstract for most of the time and its readiness cannot be closely evaluated. By making the workflow visible, it is easier to notice the issues and impediments to the productivity for the whole flow and therefore this enables to deal with issues as soon as they appear and by that way to enable continuous improvement.

When taking a look at the top five list of the team survey, it can clearly be noticed that three out of five items are related into visualizing and communicating about work and workflow of the team.

Table 4: Top five of team survey

Change	Perf.	Qual.	Job sat.	Total
1. Moving from sprint based approach to Kanban-flow	1.80	1.33	1.83	1.65
2. Introduction of more extensive build automation	1.50	1.38	1.25	1.38
3. Introduction of daily meetings	1.57	1.29	1.14	1.33
4. Introduction of virtual taskboard for projects	1.42	1.14	1.43	1.33
5. Changes to team's members	1.33	1.33	1.00	1.22

Clearly the virtual task board is essential in bringing the workflow visible for all the team members and enabling the team to observe the functionality of the workflow.

The daily meetings are in practice a way to update the status of work in flow between the whole team as well as to react and adjust the workflow to improve the team's productivity.

Movement from sprint-based approach to Kanban-flow also highlights the perception of the importance on focusing more on flow of the work than on fixed scope or commitment of specific deliverables in a short period of time.

The meaning of all these practices to the continuous improvement of the whole team cannot be highlighted too much.

There was, however, one conflict between perception of the author and the team. The team valued retrospective meetings very low, the author on the other hand considers them to be an essential part that is also needed to enable continuous improvement of workflow and team productivity. It was very clear that whenever there was need to solve an issue which would take more than a day to solve or would need more in-depth discussion of how to approach it, the issue did not really start moving forward through daily meetings only. For that the retrospective meetings were essential.

It seems though that the retrospective meetings in the way that they were implemented did still leave a feeling for not being that important for the team and at least they were something that the team did not feel like much of fun. This might be that the retrospective practices that were tried were not suitable for these teams, however, it also raises a valid question to seek for a different kind of approach to bring bigger improvements as part of flow without a need for additional retrospective meetings.

All in all, the finding clearly indicates that if team wants to be able to enable continuous improvement and find quick performance gain, it would be well advised to focus on visualizing and concentrating on continuously improving its workflow.

6.2.4 The focus of practices should be on team and not on project

Overview

It seems that many of the methodologies, processes and literature related to software development are strongly focused on projects or individual products.

However, the findings of this study indicate that in software product development environment, better results could be reached if the focus would be on the team instead of project or projects that the team is working on.

The relevancy of these findings stems from the fact that it raises a concern that currently software development seems to be approached from a very project-centric angle which might lead to worse results when applied to the software product development.

Story

Team A mainly focused on the development project of product M, however, it still had responsibility for maintenance and technical support of many other products. In addition, some of the team members also had responsibilities outside of the team's own project and maintenance work. They had, for example, specific skills or knowledge that other teams of the organization might need every now and then.

Because the main focus of the team was a development project, the team initially set up the practices around the development project. This included, for example, visualizing the workflow around the project workflow and having daily meetings focused on the project.

In addition, the content of the sprints was picked from the project's backlog.

However, the visibility for the whole work of the team never really got into a good level as there seemed to regularly pop up needs to work on maintenance or help other teams. These distractions could take from a few hours to even weeks and often did come up in such a tight schedule that the team could not take them into account in sprint planning.

The team tried to tackle these issues in many ways such as aiming at shorter sprints and adding buffer to the content of the sprint to take this variation into account, however, in the end none of these seemed to have a good enough impact.

When considering options, the author came to the conclusion that the team's commitment to the project was only partial as it was shared between the project and maintenance work. This caused a situation that the commitment level on the shared

part would differ quite a lot between team members and there was a strong variance in these levels depending on other work.

Due to that, the author decided that it would be better if the team commitment would be shared in higher degree by sharing the maintenance commitment also for the whole team. To reach that, the author suggested to the team that they would move from project based task board and daily meetings to team based, which would include all the work on a set of products that the team was either developing or maintaining.

The team was willing to give that a try and therefore practices were adapted to focus on all of the work of the team and on products that the team had on its responsibility.

After an initial trial period, the team decided to keep the team based approach which they felt to be clearly better than the earlier project based approach. Some concrete examples of improvement included better visibility and joint commitment for the work that the team was doing, however, the team also started to share some responsibilities with products that had earlier been only maintained by a single member of the team.

Team B had been working more on maintenance based approach of a single product for some time already and was in a better situation in the sense that they had only one product to focus on. For them, the project based approach was working well as their one product could be considered as one project.

However, later on team B also got involved in development of new products. There were some indications that similar issues started to appear as for team A, mainly with visibility for the whole work of the team getting worse. In that phase, the author suggested to them also to move to the team based approach for both maintenance and new products due to good experiences with Team A.

Team B also found that approach good for them and they decided to keep the team based approach.

Conclusion

The majority of the methods and literature seems to focus on projects or products instead of teams; however, during the research the switch for more team based approach seemed to increase teams' happiness and performance, already on short term. In addition, it enabled the team to improve sharing the responsibilities and knowledge better.

Both teams moved from project or product oriented work management to team based work management and seemed to benefit from it.

The teams' perception of change to use team based task board was seen mostly as a minor improvement when looking at the team survey in all three aspects, and in addition, it alone was seen in a similar way by the author; however, on the long run it did have a significant impact on how the team improved its cooperation and got more involved in continuous improvement. The author sees one major reason for this to be that the commitment for the teams' work was now shared on more in-depth level and the visibility of work as a whole had improved.

This finding is important in the aspect that when considering current Agile methodologies and frameworks, it is important to think on a team level instead of focusing the highest priority on project or product. It seems that focusing on team level will also lead to better results when considering those high priority projects and products.

6.3 Purpose related findings

6.3.1 Introduction

This chapter describes findings that answer the original research questions.

The key findings are described in their own sub-chapters, in which each finding is described more closely.

6.3.2 There is no fit-for-all methodology, but team based tailoring is required

Overview

Both of the teams went through plenty of evolution with their practices during the period of the research. It proved to be true that neither of the teams did find any single predefined Agile methodology that would have suited perfectly for their needs, but tailoring was required.

The teams did eventually end up with very similar core sets of practices which were mostly related to visibility and managing workflow. That emergent finding is described more in-depth in chapter 6.2.3.

Even with such a core set that was found, there were some team based differences of how they were applied to practice.

This is underlined by the results of the teams' practices survey that is gone through more in-depth in chapter 6.1.3.

Conclusion

Even though both teams ended up with a similar set of core practices, some level of differences also still existed. In addition, there seemed to be a continuous need for change in the practices which for some reason was due to changing external conditions such as the phase of the projects. Therefore, the finding suggests that there is no silver bullet of methodologies on detailed level.

This finding was as expected, leading into a conclusion that there is always need for fine tuning practices for the teams instead of applying something straight from the box.

It is, however, important to take a note that there was a set of core practices that both teams found to have a very positive impact, which could indicate that there is a good set to start with and to adapt those to match the needs of the team.

6.3.3 Some Agile practices are exceptionally good for improving all three: performance, quality and happiness

Overview

It was expected by the author that some Agile or Lean practices would prove to improve all three: performance, quality and happiness.

The findings of this research indicate that such practices were found that were applicable for both of the teams.

This can be seen especially in Team Practices Survey which is discussed more in-detail in chapter 6.1.3. The finding is also underlined by the author's experience during the research when looking at the list that combines the results of the different aspects as part of Team Practices Survey, and limiting the list to include only those which have average of ≥ 1.0 , such a list is shown in Table 5.

Table 5: Team practices survey, average value ≥ 1.0 in all categories

Change	Perf.	Qual.	Job sat.	Total
1. Moving from sprint based approach to Kanban-flow	1.80	1.33	1.83	1.65
2. Introduction of more extensive build automation	1.50	1.38	1.25	1.38
3. Introduction of daily meetings	1.57	1.29	1.14	1.33
4. Introduction of virtual taskboard for projects	1.42	1.14	1.43	1.33
5. Changes to team's members	1.33	1.33	1.00	1.22

As it can be seen, there are five issues that have had a remarkable positive effect on each of the aspects that were studied. Of those five, three are clearly Agile or Lean practices and the other two could be considered to have been triggered by Agile practices.

Conclusion

A set of concrete practices was found which seemed to have remarkable improvement on all three: performance, quality and job satisfaction.

The ones in the set that can most clearly be considered as Agile or Lean practices are introduction of Kanban-flow, introduction of daily meetings and introduction of virtual task board.

It is also worth pointing out that the specified sets are items that are strongly focused on improving visibility and communication related to work and workflow, which is an important emergent finding and is examined more closely in chapter 6.2.3.

Therefore, it can be concluded that such a set of Agile practices was found that improved all three in the scope of this research.

7 Summary and conclusions

7.1 Objective and findings of the thesis

The main organizational object of the thesis was to improve the produced quality and performance of two software development teams, while maintaining a high level of happiness. Author considers that goal was fulfilled and all of the data is supporting this.

In addition, survey was taken that goes through different tools and practices that were taken into use and teams' impression on their impact for performance, quality and happiness.

Two expected findings were defined in the start of the research, including

- There is no fit for all methodology, but team based tailoring is required
- Some Agile practices are especially good in improving all three: performance, quality and happiness.

In both cases the result was as expected. These findings are gone through more in detail in chapter 6.3.

There were also many emergent findings during the research, which of three most valuable are

- Continuous flow is better than sprints for multi-product owning product development team
- The most valuable practices are those that improve visibility to the workflow and enable its continuous improvement
- The focus of practices should be on team and not on project

Emergent findings are gone through in chapter 6.2.

7.2 Future use of the results

The results provide valuable usage for both organization and software development community.

From organizational perspective, especially the finding 6.2.3 focusing on value of visualization of workflow can prove to be valuable if the teams in the organization will apply those practices more widely as that can lead to meaningful increment in performance, quality and happiness in whole organization. Similar advantages can be considered also to be found in wider software development community.

The finding that the focus should be on teams instead of projects, described in chapter 6.2.4, is something that could lead to remarkable improvements in organization. As the organization is currently very project oriented, such a change could have a high positive impact. This is also important insight to be shared outside the organization as currently the mainstream software development seems to be very project focused.

It has to be taken into account with all the findings that the number of teams involved was only two, and both of them worked closely in the same organization. This lowers the generalization value of the findings. However, the data that was collected was triangulated in many levels, including for example research methods and different survey sources, and therefore can be considered as highly valid in the given environment. This validity would also suggest that the findings should be applicable at least in similar environments as the research environment was.

Considering all this, the results of this research could also be used as baseline for further research with a larger number of teams and organizations which could lead to higher level of generalization.

7.3 In closing

The research was interesting to do in many ways. First of all, being strongly focused on the author's daily work it provided him with more discipline to focus on the improvement progress also in more scientific ways.

Managing and initiating continuous changes in a team can be very challenging, however, iterative framework such as provided by action research and many Agile methodologies can really make it easier. It is also often hard to evaluate which of the changes in team behavior are really caused by the change of practices and their effectiveness and which of are related to the behavior of individuals in the team or external events.

The author also did find it sometimes difficult to decide how much he should be pushing some specific methods and practices forward, and how much should he leave for his team to figure out. Often the author had plenty of knowledge about practices that the team members did not have, however, offering those for the team straight away could have led into a situation where the team would not have felt ownership of those changes and been as interested in adopting them. That seems to be quite a common challenge in all change management.

Also, the constant change of conditions in software product development and changing bias between new development and maintenance gave a challenge of their own as continuous adjustment was needed on the well functioning practices as well.

It was also a very good learning process for author. While doing the research work, the author also went through much more literature related to Agile practices as well as management than he would have done otherwise. Even though several improvement suggestions and management practices came quite naturally for the author, the additional insights gained from theory enabled the author to succeed in his team and change management duties even more effectively.

Bibliography

Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Washington: Blue Hole Press.

Jeffries, R. E. (2014). *What is Extreme Programming?* Retrieved 3 8, 2014, from xprogramming.com: <http://xprogramming.com/what-is-extreme-programming/>

Lahti, K. (2008). *Adapting to and assessing Scrum in game development*. HANKEN - Swedish School of Economics and Business Administration.

Manifesto for Agile software development. (2001). Retrieved 2 27, 2014, from <http://agilemanifesto.org/>

Maslow, A. (1943). *A Theory of Human Motivation*. Retrieved 4 27, 2014, from Classics in history of psychology: <http://psychclassics.yorku.ca/Maslow/motivation.htm>

McNiff, J. (2013). Action Research: Principles and practices. In J. McNiff, *Action Research: Principles and practices* (Third edition ed.). Routledge Ltd.

Poppendieck, M., & Poppendieck, T. (2007). *Implementing Lean Software Development : From Concept to Cash*. Boston: Addison-Wesley.

Poppendieck, M., & Poppendieck, T. (2010). *Leading Lean Software Development*. Boston: Addison-Wesley.

Schwaber, K., & Beedle, M. (2002). *Agile Software Development with Scrum*. New Jersey: Prentice Hall.

Schwaber, K., & Sutherland, J. (2013). *Scrum Guide*. Retrieved 3 8, 2014, from Scrum.org: www.scrum.org/Scrum-Guide

Seligman, M. (2004, 2). *Martin Seligman: The new era of positive psychology*. Retrieved 4 27, 2014, from TED:

https://www.ted.com/talks/martin_seligman_on_the_state_of_psychology

VersionOne. (2014, 27 2). *Annual State of Agile Survey*. Retrieved 27 2, 2014, from VersionOne.com: <http://stateofagile.versionone.com/>

Appendices

Appendix 1: Team survey results

Table 6: Tools, practices and actions team survey

Category	Change	Major negative effect	Minor negative effect	No effect	Minor positive effect	Major positive effect
Quality	24. Defining the state of the team and setting common goals	0	0	5	1	0
Performance	27. Introduction of unit testing	2	3	1	0	0
Job satisfaction	29. Introduction of Scrum sprints	1	1	1	0	0
Quality	23. Mapping team values	0	0	4	1	0

Performance	25. Defining negative testing as part of developer testing	0	2	4	0	0
Job satisfaction	28. Introduction of retrospective meetings	0	3	1	1	0
Quality	29. Introduction of Scrum sprints	0	0	2	1	0
Performance	22. Introduction of definition of done	0	0	6	0	0
Job satisfaction	26. Introduction of sprint demos	1	0	3	1	0
Quality	28. Introduction	0	0	3	2	0

	of retrospective meetings					
Performance	26. Introduction of sprint demos	0	1	3	1	0
Job satisfaction	27. Introduction of unit testing	1	1	3	0	1
Quality	21. Changing retrospective meetings more structural	0	0	3	2	0
Performance	29. Introduction of Scrum sprints	0	1	1	1	0
Job satisfaction	25. Defining negative testing as part of developer testing	0	0	6	0	0

Quality	26. Introduction of sprint demos	0	1	2	1	1
Performance	24. Defining the state of the team and setting common goals	0	0	5	1	0
Job satisfaction	22. Introduction of definition of done	0	0	5	1	0
Quality	18. Introduction of user stories	0	0	4	0	1
Performance	20. Introduction of workflow phase based definition of dones	0	1	2	2	0
Job	9.	0	1	3	2	0

satisfaction	Introduction of developer cross-testing					
Quality	19. Introduction of WIP limits on the task board	0	0	3	2	0
Performance	23. Mapping team values	0	0	3	1	0
Job satisfaction	15. Defect root cause analysis meetings	0	0	4	1	0
Quality	17. Introduction of labday	0	0	3	3	0
Performance	15. Defect root cause analysis meetings	0	0	3	2	0
Job satisfaction	20. Introduction of workflow	0	0	4	1	0

	phase based definition of dones					
Quality	11. Organizational change: Customer services as part of imaging team	0	0	3	0	1
Performance	21. Changing retrospective meetings more structural	0	0	3	2	0
Job satisfaction	23. Mapping team values	0	0	4	1	0
Quality	16. Transition from using project task board to team task board	0	0	3	1	1
Performance	14. Introduction	0	1	2	2	1

	of monthly meetings					
Job satisfaction	10. Organizing team into two core teams and supporting functions	0	0	2	1	0
Quality	7. Changing daily meeting structure from Scrum-style to Kanban-style	0	0	3	1	1
Performance	17. Introduction of labday	0	0	3	3	0
Job satisfaction	24. Defining the state of the team and setting common goals	0	1	3	1	1

Quality	8. Replacing sprint demos with user story based demos	0	0	2	3	0
Performance	16. Transition from using project task board to team task board	0	0	3	1	1
Job satisfaction	21. Changing retrospective meetings more structural	0	0	3	2	0
Quality	14. Introduction of monthly meetings	0	0	3	2	1
Performance	28. Introduction of retrospective meetings	0	0	3	1	1

Job satisfaction	13. Changes in management positions	0	0	3	2	0
Quality	10. Organizing team into two core teams and supporting functions	0	0	1	3	0
Performance	18. Introduction of user stories	0	0	3	1	1
Job satisfaction	12. Introduction on product backlog	0	0	3	1	1
Quality	25. Defining negative testing as part of developer testing	0	0	2	3	1
Performance	9. Introduction of developer	0	1	1	3	1

	cross-testing					
Job satisfaction	18. Introduction of user stories	0	0	3	1	1
Quality	15. Defect root cause analysis meetings	0	0	1	3	1
Performance	13. Changes in management positions	0	0	2	2	1
Job satisfaction	19. Introduction of WIP limits on the task board	0	0	2	3	0
Quality	22. Introduction of definition of done	0	0	2	2	2
Performance	19. Introduction of WIP limits	0	0	2	2	1

	on the task board					
Job satisfaction	14. Introduction of monthly meetings	0	0	3	2	1
Quality	12. Introduction on product backlog	0	0	1	3	1
Performance	11. Organizational change: Customer services as part of imaging team	0	0	2	0	2
Job satisfaction	16. Transition from using project task board to team task board	0	0	2	2	1
Quality	20. Introduction	0	0	1	3	1

	of workflow phase based definition of dones					
Performance	12. Introduction on product backlog	0	0	1	3	1
Job satisfaction	17. Introduction of labday	0	0	1	4	1
Quality	13. Changes in management positions	0	0	2	1	2
Performance	6. Introduction of ReSharper as a tool	0	0	2	2	2
Job satisfaction	11. Organizational change: Customer services as part of	0	0	2	0	2

	imaging team					
Quality	4. Introduction of virtual taskboard for projects	0	0	1	3	1
Performance	7. Changing daily meeting structure from Scrum-style to Kanban-style	0	0	1	3	1
Job satisfaction	6. Introduction of ReSharper as a tool	0	0	1	4	1
Quality	1. Moving from sprint based approach to Kanban-flow	0	0	0	3	1

Performance	4. Introduction of virtual taskboard for projects	0	0	1	1	3
Job satisfaction	8. Replacing sprint demos with user story based demos	0	0	1	3	1
Quality	2. Introduction of more extensive build automation	0	0	0	4	2
Performance	2. Introduction of more extensive build automation	0	0	1	1	4
Job satisfaction	3. Introduction of daily	0	0	1	2	2

	meetings					
Quality	3. Introduction of daily meetings	0	0	0	3	2
Performance	10. Organizing team into two core teams and supporting functions	0	0	0	2	2
Job satisfaction	7. Changing daily meeting structure from Scrum-style to Kanban-style	0	0	1	2	2
Quality	9. Introduction of developer cross-testing	0	0	0	3	3
Performance	8. Replacing sprint demos with user	0	0	0	2	3

	story based demos					
Job satisfaction	5. Changes to team's members	0	0	1	1	2
Quality	27. Introduction of unit testing	0	0	0	3	3
Performance	5. Changes to team's members	0	0	0	1	3
Job satisfaction	2. Introduction of more extensive build automation	0	0	1	2	3
Quality	6. Introduction of ReSharper as a tool	0	0	1	1	4
Performance	3. Introduction of daily	0	0	0	1	4

	meetings					
Job satisfaction	4. Introduction of virtual taskboard for projects	0	0	1	1	3
Quality	5. Changes to team's members	0	0	0	1	3
Performance	1. Moving from sprint based approach to Kanban-flow	0	0	0	0	3
Job satisfaction	1. Moving from sprint based approach to Kanban-flow	0	0	0	1	3

Appendix 2: TPD™ results

Table 7: TPD™ survey results 1 and 2 including differences

Question	% of “No” -answers (H1/2012)	% of “No” - answers (2H/2012)	Max. Difference 2012
1. <i>There should be more flexibility in my job content.</i>	83 %	86 %	3 %
2. We are all very busy but we don't seem to be pulling in the same direction.	83 %	100 %	17 %
3. People tend not to say what they really think or how they really feel.	50 %	71 %	21 %
4. There is little loyalty between members of the team.	100 %	100 %	0 %
5. Often, the wrong kinds of skills are developed within our team.	83 %	86 %	3 %
6. We are often in	83 %	100 %	17 %

conflict with teams in other departments.			
7. The team do not get enough feedback.	83 %	86 %	3 %
8. Information is not shared well enough within the team.	83 %	100 %	17 %
9. There doesn't seem to be enough focus on hitting our team objectives.	100 %	86 %	-14 %
10. Team spirit is quite low right now.	100 %	100 %	0 %
11. Team members are not trusted to make decisions on their own.	83 %	100 %	17 %
12. In this team the way we do things is rarely challenged.	83 %	57 %	-26 %
13. Decisions are taken at the wrong level, often by the wrong people.	83 %	100 %	17 %

14. <i>It's sometimes difficult to help out team members as our jobs seem to be so different.</i>	67 %	86 %	19 %
15. Longer term planning meetings don't happen enough.	83 %	86 %	3 %
16. Conflict is often destructive in this team.	83 %	100 %	17 %
17. I do not receive enough feedback from other team members.	83 %	86 %	3 %
18. Personal development is not taken seriously enough.	67 %	100 %	33 %
19. We don't talk about our position or raise our profile well enough within the broader organisation.	67 %	86 %	19 %
20. Team members are not supported when dealing with others.	100 %	100 %	0 %

21. We do not seem to learn from our mistakes, often carrying on without reviewing properly.	83 %	86 %	3 %
22. We usually manage to get the job done, but sometimes it's a bit of a last minute rush.	33 %	71 %	38 %
23. It's not much fun working in this team.	83 %	100 %	17 %
24. Command and control would be a good way of describing how things are run around here.	100 %	100 %	0 %
25. We don't seem to actively use many 'creative thinking tools or techniques' in our discussions and meetings.	100 %	86 %	-14 %
26. Problem solving is more about blame and punishment rather than a genuine desire to solve	100 %	100 %	0 %

things and learn from mistakes.			
27. <i>When a key person is away then work tends to pile up in that area or cause problems.</i>	67 %	71 %	4 %
28. The objectives of our team keep changing.	83 %	71 %	-12 %
29. It would help if people were more willing to admit their mistakes.	33 %	86 %	53 %
30. I do not feel supported by my colleagues.	83 %	100 %	17 %
31. The subject of 'training' comes up sometimes, but nothing really seems to happen.	100 %	86 %	-14 %
32. We are not very good at listening to our internal or external customers.	83 %	100 %	17 %

33. There is little interest in what team members are doing until something goes wrong.	83 %	86 %	3 %
34. We should spend more time questioning the way we work.	67 %	71 %	4 %
35. Accountability, blame and credit are often given to those who are popular and those who are not.	83 %	100 %	17 %
36. I am quite stressed at the moment.	83 %	100 %	17 %
37. Team members often feel frustrated because they are not consulted about issues.	100 %	100 %	0 %
38. We are too scared of risk as a team, we tend to choose the safer options.	83 %	100 %	17 %
39. Members of the team	83 %	86 %	3 %

are not involved enough in decision-making.			
40. <i>Too much time is spent defining territory, roles, boundaries and responsibilities.</i>	100 %	100 %	0 %
41. I do not know exactly what my objectives are.	83 %	86 %	3 %
42. There is a sense of hostility among parts of this team.	100 %	100 %	0 %
43. There is not enough listening going on within our team.	83 %	100 %	17 %
44. People are not really helped to develop.	83 %	100 %	17 %
45. We have too little influence on the rest of the organisation.	83 %	86 %	3 %
46. Our team is not organised to make the best use of our resources, either individually or as a	83 %	86 %	3 %

team.			
47. It can be difficult to get things done because of all the forms and processes that need to be adhered to.	100 %	100 %	0 %
48. There are too many complaints from 'customers' (internal/external).	83 %	86 %	3 %
49. I am sure some team members would rather phone in sick than come in when feeling slightly unwell.	83 %	86 %	3 %
50. Mistakes are rarely accepted within this team.	83 %	100 %	17 %
51. There does not seem to be any process for actively seeking and coming up with ideas.	83 %	86 %	3 %

52. We seem to make more bad decisions than good ones.	83 %	100 %	17 %
53. <i>I have no clear idea of what other team members are doing.</i>	83 %	100 %	17 %
54. I do not understand how my own objectives relate to those of the team.	83 %	86 %	3 %
55. More time should be devoted to discussing and valuing our differences within the team.	67 %	100 %	33 %
56. People seem unwilling to take the views of others into account quite a lot of the time.	83 %	100 %	17 %
57. We rarely spend time/money on team building and team development.	83 %	86 %	3 %

58. Information does not flow freely enough between our team and other teams.	50 %	100 %	50 %
59. I often find myself struggling with new tasks with little or no guidance.	100 %	100 %	0 %
60. We often do not know what other teams are doing in other departments.	50 %	86 %	36 %
61. I feel that we could achieve much more as a team.	67 %	100 %	33 %
62. We are more like a collection of individuals than a team.	83 %	100 %	17 %
63. When work is delegated we are not trusted to complete the task independently.	100 %	100 %	0 %
64. Some team members are resistant to change.	83 %	86 %	3 %

65. Problems tend to be passed around with no one really owning or attempting to resolve them.	83 %	86 %	3 %
--	------	------	-----

Appendix 3: Voice 2011 results

Table 8: Voice 2011 survey results

Question	1	2	3	4	5
... I constantly benefit from the knowledge and experience of others	10 %	0 %	30 %	30 %	30 %
How satisfied are you with your overall situation in Tieto?	0 %	20 %	20 %	30 %	30 %
How satisfied are you with your current job?	0 %	10 %	20 %	50 %	20 %
... we often try out new ways of working	0 %	0 %	10 %	60 %	30 %
In my team everyone takes responsibility for problems that arise in their work	0 %	20 %	10 %	20 %	50 %
Tieto is making a voyage I really would like to follow	0 %	10 %	10 %	70 %	10 %
In my team we make sure that new ideas are evaluated irrespective of who suggests them	0 %	10 %	10 %	30 %	50 %
... we encourage and support new ideas	0 %	0 %	20 %	40 %	40 %
I would gladly recommend a good friend to apply for a	0 %	0 %	40	30	30

job at Tieto			%	%	%
In our team we always do our best to find solutions that would add value to the customers' business.	0 %	0 %	20 %	30 %	50 %
... we learn from our mistakes and continuously improve the way we do things	0 %	0 %	30 %	40 %	30 %
I would stay on at Tieto even if I were offered a similar job at approximately the same pay and benefits in another company	0 %	10 %	10 %	40 %	40 %
... we have an atmosphere of trust where we can openly talk about mistakes and disagreements	0 %	0 %	10 %	50 %	40 %
I feel that I develop and expand my competence at work	0 %	0 %	20 %	60 %	20 %
I feel I have good possibilities to make a career at Tieto	0 %	20 %	30 %	40 %	10 %
I feel content with my overall situation in Tieto	0 %	10 %	20 %	40 %	30 %
I believe Tieto will become one of the winners within its field	0 %	0 %	30 %	50 %	20 %
How satisfied are you with the processes available?	0 %	10 %	10 %	60 %	20 %

In my team we always try practical solutions to solve problems that arise at work	0 %	10 %	0 %	20 %	70 %
I have a clear understanding of the competences that will be required from me in the future	0 %	0 %	10 %	70 %	20 %
My manager gives constructive feedback on work performance	0 %	0 %	20 %	70 %	10 %
The way we in our team divide our work between us, makes it easier for us to achieve our goals	0 %	0 %	10 %	50 %	40 %
I feel involved in the decisions taken in my team	0 %	10 %	10 %	50 %	30 %
I can clearly see how my work contributes to achieving Tieto's overall goals	0 %	0 %	20 %	40 %	40 %
How satisfied are you with the tools available?	0 %	10 %	0 %	50 %	40 %
... actions are taken very quickly when a decision has been made	0 %	0 %	10 %	30 %	60 %
... there are good systems for finding the information I need in order to be able to carry out my work	0 %	10 %	10 %	60 %	20 %
How satisfied are you with your working conditions?	0 %	0 %	30 %	20 %	50 %

I am well aware of how satisfied our customers are	0 %	10 %	0 %	80 %	10 %
I am empowered to deal with problems arising in my work without having to seek my manager's permission first	0 %	0 %	0 %	30 %	70 %
... we make decisions urgently when necessary	0 %	0 %	10 %	20 %	70 %
When I do a good job my contribution is recognised	0 %	0 %	40 %	30 %	30 %
In my team we actively use customer feedback to improve our products and services	0 %	0 %	10 %	50 %	40 %
I am familiar with the latest development in products and services within our business (IT services)	0 %	0 %	30 %	50 %	20 %
My colleagues care about me	0 %	0 %	20 %	60 %	20 %
It is my responsibility to make sure that I develop professionally within Tieto	0 %	0 %	0 %	30 %	70 %
In Tieto we have the right work processes in order to achieve successful deliveries	0 %	0 %	10 %	60 %	30 %
I feel respected and valued in my work at Tieto	0 %	10 %	0 %	60 %	30 %

Answer options were labeled by defining the scale to be between “Strongly disagree” (1) and “Strongly agree” (5). In satisfaction related questions the scale was labeled to be between “Not satisfied at all” (1) and “Very satisfied” (5).

Appendix 4: Voice 2012 results

Table 9: Voice 2012 survey results

Question	1	2	3	4	5
... I constantly benefit from the knowledge and experience of others	0 %	0 %	0 %	33 %	67 %
How satisfied are you with your overall situation in Tieto?	0 %	8 %	0 %	33 %	58 %
How satisfied are you with your current job?	0 %	0 %	8 %	33 %	58 %
... we often try out new ways of working	0 %	0 %	0 %	17 %	83 %
In my team everyone takes responsibility for problems that arise in their work	0 %	0 %	0 %	42 %	58 %
Tieto is making a voyage I really would like to follow	0 %	0 %	17 %	33 %	50 %
In my team we make sure that new ideas are evaluated irrespective of who suggests them	0 %	0 %	8 %	25 %	67 %
... we encourage and support new ideas	0 %	0 %	0 %	42 %	58 %
I would gladly recommend a good friend to apply for a job	0	8	0	50	42

at Tieto	%	%	%	%	%
In our team we always do our best to find solutions that would add value to the customers' business.	0 %	0 %	0 %	36 %	64 %
... we learn from our mistakes and continuously improve the way we do things	0 %	0 %	8 %	50 %	42 %
I would stay on at Tieto even if I were offered a similar job at approximately the same pay and benefits in another company	0 %	0 %	8 %	42 %	50 %
... we have an atmosphere of trust where we can openly talk about mistakes and disagreements	0 %	0 %	8 %	25 %	67 %
I feel that I develop and expand my competence at work	0 %	0 %	17 %	42 %	42 %
I feel I have good possibilities to make a career at Tieto	0 %	8 %	33 %	42 %	17 %
I feel content with my overall situation in Tieto	0 %	9 %	0 %	55 %	36 %
I believe Tieto will become one of the winners within its field	0 %	0 %	17 %	50 %	33 %
How satisfied are you with the processes available?	0 %	0 %	17 %	50 %	33 %
In my team we always try practical solutions to solve	0	0	0	25	75

problems that arise at work	%	%	%	%	%
I have a clear understanding of the competences that will be required from me in the future	0 %	0 %	17 %	42 %	42 %
My manager gives constructive feedback on work performance	0 %	0 %	17 %	58 %	25 %
The way we in our team divide our work between us, makes it easier for us to achieve our goals	0 %	0 %	8 %	33 %	58 %
I feel involved in the decisions taken in my team	0 %	0 %	17 %	50 %	33 %
I can clearly see how my work contributes to achieving Tieto's overall goals	0 %	0 %	17 %	42 %	42 %
How satisfied are you with the tools available?	0 %	0 %	17 %	42 %	42 %
... actions are taken very quickly when a decision has been made	0 %	0 %	17 %	17 %	67 %
... there are good systems for finding the information I need in order to be able to carry out my work	0 %	8 %	17 %	50 %	25 %
How satisfied are you with your working conditions?	0 %	8 %	8 %	42 %	42 %
I am well aware of how satisfied our customers are	0	9	18	45	27

	%	%	%	%	%
I am empowered to deal with problems arising in my work without having to seek my manager's permission first	0 %	0 %	8 %	17 %	75 %
... we make decisions urgently when necessary	0 %	0 %	17 %	17 %	67 %
When I do a good job my contribution is recognised	0 %	8 %	25 %	42 %	25 %
In my team we actively use customer feedback to improve our products and services	0 %	9 %	0 %	45 %	45 %
I am familiar with the latest development in products and services within our business (IT services)	0 %	0 %	25 %	67 %	8 %
My colleagues care about me	8 %	0 %	8 %	58 %	25 %
It is my responsibility to make sure that I develop professionally within Tieto	0 %	0 %	0 %	42 %	58 %
In Tieto we have the right work processes in order to achieve successful deliveries	0 %	0 %	25 %	42 %	33 %
I feel respected and valued in my work at Tieto	0 %	8 %	8 %	58 %	25 %

Answer options were labeled by defining the scale to be between “Strongly disagree” (1) and “Strongly agree” (5). In satisfaction related questions the scale was labeled to be between “Not satisfied at all” (1) and “Very satisfied” (5).