***Title page with author details**

Satisfaction and its Correlates in Agile Software Development

Martin Kropp[a], Andreas Meier[b], Craig Anslow[c], Robert Biddle[d]

[a] *University of Applied Sciences Northwestern Switzerland, Windisch, Switzerland*
[b] *Zurich University of Applied Sciences, Winterthur, Switzerland*
[c] *Victoria University of Wellington, Wellington, New Zealand*
[d] *Carleton University, Ottawa, Canada*

Abstract

In this paper we address the topic software development team members satisfaction with their development process. We present an in-depth analysis of the results of a nationwide survey about software development in Switzerland. We wanted to find out if satisfaction relates to the applied development method, and to the use of various practices, and influences on business, team and software issues. We found that higher satisfaction is reported more by those using Agile development than with plan-driven processes. We explored the different perspectives of developers and those with a management role and found a high consistency of satisfaction between Agile developers and Agile management, and differences with those using working plan-driven methods. We found that certain practices and influences have high correlations to satisfaction, and that collaborative processes are closely related to satisfaction. We then explored the relationship between satisfaction and various other perspectives. Our results in this analysis are principally descriptive, but we think they can be a relevant contribution to understand the challenges for everyone involved in Agile development.

Email addresses: martin.kropp@fhnw.ch (Martin Kropp),
meea@fhnw.ch (Andreas Meier), craig@ecs.vuw.ac.nz (Craig Anslow),
robert.biddle@carleton.ca (Robert Biddle)

Satisfaction and its Correlates in Agile Software Development

Martin Kropp[a], Andreas Meier[b], Craig Anslow[c], Robert Biddle[d]

[a]*University of Applied Sciences Northwestern Switzerland, Windisch, Switzerland*
[b]*Zurich University of Applied Sciences, Winterthur, Switzerland*
[c]*Victoria University of Wellington, Wellington, New Zealand*
[d]*Carleton University, Ottawa, Canada*

Abstract

In this paper we address the topic of software development team members satisfaction with their development process. We present an in-depth analysis of the results of a nationwide survey about software development in Switzerland. We wanted to find out if satisfaction relates to the applied development method, and to the use of various practices, and impacts on business, team and software issues. We found that higher satisfaction is reported more by those using Agile development than with plan-driven processes. We explored the different perspectives of developers and those with a management role and found a high consistency of satisfaction between Agile developers and Agile management, and differences with those using working plan-driven methods. We found that certain practices and impacts have high correlations to satisfaction, and that collaborative processes are closely related to satisfaction. We then explored the relationship between satisfaction and various other perspectives. Our results in this analysis are principally descriptive, but we think they can be a relevant contribution to understand the challenges for everyone involved in Agile development.

1. Introduction

In the last decade Agile software development methods have been widely used in industry and become mainstream, as recent studies show [1, 2]. The studies typically report “management of changing priorities”, “faster time to market”, “team morale”, “team productivity” and “people development” as top benefits from performing Agile practices. While the very first principle of the Agile Manifesto begins with “Our highest priority is to satisfy the customer. . . ” [3], studies also show that Agile team members themselves report stronger satisfaction compared with their experience with plan-driven approaches (e.g. [4]). However, not much is known about the most powerful reasons for the satisfaction. We explore potential reasons in this paper.

We examine the following research questions:

RQ1 : How does the applied software development method relate to satisfaction of team members?

We wanted to find out if Agile development leads to higher satisfaction than traditional plan-driven approaches. This question has also driven earlier research, as we discuss later, though such interest was more common when Agile methods were new. We also wanted to find out if the view on satisfaction of management is similar to that of individual professionals. We define the terms Agile and plan-driven according to Boehm and Turner [5].

RQ2 : How does satisfaction correlate to the applied practices? Most importantly, we wanted to find out which practices relate most strongly to satisfaction.

RQ3 : Does satisfaction depend on the impacts achieved with the development method?

We also wanted to find out if and how satisfaction relates to the results achieved with Agility. For this we were asking how Agility influences certain business aspects (e.g. time-to-market), team aspects (e.g. team morale), and software aspects (e.g. software architecture). In this paper we use the term “impacts” for these outcomes of a process.

The goal of our analysis was a deeper understanding about the effects of Agile development and to get indicators about the human aspects of Agile software development.

To address our research questions we analyze the results of a nationwide study of Agile software development in Switzerland, conducted in 2016. In the study we conducted two independent surveys, one for company representatives (i.e. typically upper management), and another for individual professionals.

In the next section, we review earlier work on satisfaction in software development, especially that with a focus on Agile processes. We then outline the nature of our survey, the source of our study data, and the main results concerning satisfaction. The results are then explored in more detail, investigating relationships in the data in order to better understand the potential reasons for satisfaction or dissatisfaction. In particular, we explore how development practices and various impacts relate to satisfaction. We then explore several other issues, including personal experience, stress, and potential hindrances to success with Agile methods. We then discuss our results and present our conclusions.

Email addresses: martin.kropp@fhnw.ch (Martin Kropp), meea@zhaw.ch (Andreas Meier), craig@ecs.vuw.ac.nz (Craig Anslow), robert.biddle@carleton.ca (Robert Biddle)

This paper is an extended version of one presented at the ACM International Conference on Evaluation and Assessment in Software Engineering (EASE), in Christchurch, New Zealand, in 2018 [6]. The current paper has more detail, especially in section 5, where as well as technical practices, both collaborative and planning practices are now discussed, with Figures 8b and 8c. In section 6.3, we now present our survey results about hindrances to the development process, with Tables 9 and 10, and Figures 12 and 13.

2. Related Work

The first empirical study on satisfaction in Agile development was conducted by Mannaro et al. in 2004 [7]. Their focus was on Extreme Programming (XP), where they surveyed 55 XP and 67 non-XP professionals using the Goal-Question-Metrics (*GQM*) approach [8]. They found that satisfaction was greater among XP professionals than others on a number of measures, not only in general, but also on a variety of specific issues, such as reduced stress, increased productivity, and better attitude.

In 2006, Melnik and Maurer presented results of a large (n=756) online survey [9], also based on the GQM approach; they also discussed a large survey that had recently been conducted by *Computerworld* magazine. They applied statistical inference and found evidence that Agile practitioners were more satisfied than others, and also that more experience with Agile methods increased that effect. They also reported that the effect was found both for programmers and managers.

In 2007, Tessem and Maurer presented results of a case study of satisfaction in a large Agile team at a company producing software for the petroleum industry [10]. The team used Scrum, but with some practices (such as pair-programming) from XP. The study was based on interviews with team members and consideration of the general Job Characteristics Model (*JCM*) of Hackman and Oldham [11]. This study also found strong support for satisfaction with Agile methods, and pointed to alignment with five elements of the JCM, including the positive effects of autonomy, of variety in work, of good communication with others, of significance of the work, and of addressing "complete" units of work (e.g. user stories).

Tripp and Riemenschneider have addressed the issue of satisfaction in Agile development looking for theoretical underpinnings [12, 13]. They explored satisfaction in Agile development with Hackman and Oldham's JCM, taking a quantitative approach to see how well results from an Agile development survey match the model. They first used regression and factor analysis [13]. They focused on Coding standards, Daily stand-up, Refactoring, Pair programming, Unit testing, Iterative planning, and Automated builds. They did find evidence that the Agile practices relate to most elements of the JCM, though interestingly did not find evidence for the "autonomy" element. Their later analysis applied the more sophisticated approach of Structural Equation Modeling[12]. The approach distinguishes Agile project management practices and Agile software-development practices, and suggests how each relates to the JCM. The project management practices included were Daily stand-up meeting, Iterative delivery, Retrospectives, and Burndown (charts). The software development practices included were Automated (unit) testing, Automated builds, Continuous integration, Coding standards, Refactoring and Pair programming. The findings of the study suggested that project management practices directly influence satisfaction, software development practices do support some elements of the JCM, but do not directly support satisfaction. The authors highlight the interdependence of the practices, and also consider that the "autonomy" element of the JCM may not align well with the team emphasis in Agile development.

This interplay of "technical" and "collaborative" practices also features in studies of other aspects of Agile development. For example, following their field studies of collaboration in 6 Agile teams, Robinson and Sharp make the point that collaboration works as well as it does because the practices have a structure to address important technical issues [14]. Following the analysis of their quantitative study of performance in Agile teams, Wood et al. [15] make a similar point: it is not merely that teamwork leads to better performance, but rather that the teamwork works with the technical practices.

Dybå and Dingsøyr [16] provide a literature review about empirical studies of Agile software development. They mention studies that report improved customer satisfaction when using Agile methodologies. They also report about satisfaction from the developer perspective, mentioning a higher satisfaction with the product and customer collaboration.

Lindsjørn et al. [17] analyze the relationship between of Hoegl and Gemuenden's Teamwork Quality measure (*TWQ*) [18] on various aspects of software development, and report a strong positive impact of teamwork quality on work satisfaction.

In our study we use a broader range of practices (more technical practices, collaboration practices and planning practices) and also set satisfaction in relation to the impacts in business, software, and team aspects. We take a descriptive approach, and explore various concrete issues.

3. Study Setup

3.1. Study Basis

Our study was based on a nationwide online survey conducted by us in Switzerland in 2016. The survey is about the usage of development methods and practices in the IT industry, and about the impacts of applying Agile methods on projects. More detail is available about the survey instrument and the general results in the survey report [19].

The survey addressed both Agile and plan-driven companies, as well as both Agile and plan-driven IT professionals, or any hybrids. There were in fact two independent surveys: one for companies, and one for individual IT professionals.

In the company survey we address representatives of the company or the development department of a company, i.e. typically upper management level. To ensure a company was represented only once in the company survey, we sent personalized links to one management representative of each company.

Table 1: Distribution of the roles and sizes of the companies in the survey of company representatives (top), and survey of individual professionals (bottom).

Survey of Company Representatives

Role	%
CEO	34%
CTO	17%
Development Manager	11%
Team Leader	10%
CIO	7%
Project Manager	6%
Designer / Architect	2%
Software Developer	2%
Product Manager	1%
Researcher	1%
Other	9%

Size	%
Micro enterprise (≤ 9)	25%
Small enterprise (10-49)	37%
Med. enterprise (50-249)	19%
Large enterprise ≥ 250)	19%

Survey of Individual Professionals

Role	%
Senior Software Developer	17%
Software Developer	12%
Project Manager	14%
Team Leader	10%
Designer/Architect	10%
Others (< 10% each)	37%

Size	%
Micro enterprise (≤ 9)	12%
Small enterprise (10-49)	26%
Med. enterprise (50-249)	14%
Large enterprise ≥ 250)	48%

The IT professional survey was anonymous, and we invited wider participation. We sent invitations with a link to the survey via email and through professional social media like LinkedIn and XING (a career-oriented social networking site popular in German-speaking markets). Participants were typically directly involved in software development, and we describe the demographics in the section below.

The questions were the same for both surveys, with one addtion for the professionals. For that survey we added a set of questions (called "MyAgile") about their personal perspective on various issues.

3.2. Participant Demographics

We emailed 1399 companies directly with personal access code for the company representative, and about 5000[1] IT professionals in Switzerland with an anonymous link to the survey. 142 companies and 185 IT professionals filled out the complete survey. The addresses of the companies and the professionals were collated from the participating IT associations SwissICT[2] and SWEN[3], as well as from our own institutional databases. Table 2 shows the details about the survey responses. The impression value of the IT professional survey indicates the number of people visiting the survey website.

Table 2: Survey Responses

	Company Survey	Individual Survey
Impressions (gross)	1399	529
Response rate	18.16%	62%
Completion rate	10.15%	31%

Table 1 (top) shows the demographics of respondents in the survey of company representatives. It shows that 34% of the participants were Chief Executive Officers and 17% were Chief Technology Officers. "Other" includes roles like Business Analysts, Agile coach, founder, owner, and CFOs. The table also shows the distribution of the sizes of the participating companies following the official categories of the Swiss Statistical Office[4]. More than 60% are micro and small enterprises. Among the large enterprises there were four with more than 10,000 employees.

Table 1 (bottom) shows the demographics of the respondents in the survey of individual professionals. These people typically have roles much more directly involved with software development, with the largest categories of roles being Senior Software Developers (17%), Software Developers (12%), Project Managers (13%), Team Leader (10%), and Designer/ Architects (10%). We had a high number of "Others" (each < 10%)), which include roles like QA Testers, UX Designers, Scrum Masters, Agile Coaches, Product Owners, and also some managers of small companies. The IT professionals were also working mostly in a company, with the company sizes shown in the table, but were participating and speaking for themselves.

From the 182 participating professionals, 102 participants provided the company name. The professional participants came from 59 different companies. Table 3 shows the distribution of participants per company. The first row shows that there were 44 companies with one participant; 29 participants came from only 4 companies (two of those were in the financial domain). For 80 participants we don't know from which company they are. We must therefore be cautious about the potential lack of representativeness in our results.

Table 3: Distribution of participants per company in survey of individual professionals.

Participants per Company	Number of Companies
1	44
2	6
3	3
4	2
5	1
7	1
8	1
9	1
80	N/A

The main categories of the companies are IT Services/IT

[1] We do not know the exact number, since these mailings were partially done by partner associations

[2] www.swissict.ch

[3] http://www.swen-network.ch

[4] http://www.bfs.admin.ch/bfs/portal/en/index/themen/06/02/blank/key/01/groesse.html

Consulting (30%), Software Industry/Development (28%). Public Service and Finance/Insurance companies make 8% each. Next comes Telecommunication with 7%. The rest are 4% and below. The participation is a reasonable reflection of the character of software development in Switzerland according to the official governmental statistical office.

3.3. *Study Questions and Analysis*

The survey instrument questions and general results are available in full as part of the general report [19]. The questions and our analysis were principally based on Likert scales, and was therefore a quantitative approach based on self-reported experience and perception. Qualitative analysis was minimal, and limited to write-in answers to some questions where our categories could not be exhaustive.

Throughout the questionnaire, we asked participants about the nature of the software development process in their workplace. Some questions were broad, such as whether the process was more plan-driven, or more agile, and how satisfied they were with their process. We also included questions about experience, self-ratings, role, and company background.

The survey question about satisfaction came very early in the survey, and asked a simple direct question: "How satisfied are you with your current methodology?" As the survey progressed, professionals were asked about a range of their experiences in their software development environment. There were several sets of questions about *practices*: first technical practices, then collaboration practices, and lastly planning practices (Table 4 left). Each of these sets comprised several questions. Later, there were question sets about *impacts*, meaning ways in which the process inluenced outcomes: first business impacts, then team impacts, and then software impacts (Table 4 right). We acknowledge that in some cases these categorizations are more distinct than ideal: some measures could well feature in several categories. All this gave us information about how widespread the practices and experiences were. Beyond this, however, by considering these aspects together, we hoped to gain some insight about how they might be connected. For example, we might expect that the practices of Unit Testing and Test Driven Development might be related to such impacts as Software Quality and Defect Rate.

The main bases for our questions were our earlier Swiss Agile Studies [20, 1]. We have chosen those practices that are typically seen as "agile" practices (from other surveys, own experiences, discussions with companies). We were also influenced by the study by Version One [2]. It is possible we had inadvertently missed some topics, but experience with the earlier Swiss Agile Studies allowed the addition of topics suggested in feedback.

4. Basic Findings

In this section we present results about the distribution of applied methodologies and satisfaction.

Table 4: *Agile Practices*: technical, collaborative, and planning. *Agile Impacts*: business, team, and software. We asked about practices first, then impacts, in the order shown in each column. Participants responding with the level they experienced, a scale of 1–5.

Practices	**Impacts**
Technical Practices	**Business Impacts**
Unit testing	Time to market
Coding standards	Manage changing priorities
Automated builds	Alignment between IT & business objectives
Refactoring	
Continuous integration	Project visibility
Software Craftsmanship	Handling of project risk
DevOps	Development process
Clean Code	Mgmt of distributed teams
Behavior Driven Development	Requirements management
Acceptance Test Driven Dev.	Delivery predictability
Test Driven Development	
Automated acceptance testing	
Continuous delivery	
Collaborative Practices	**Team Impacts**
Dedicated product owner	Team productivity
On-site customer	People development
Daily stand-up	Effectiveness of meetings
Retrospective	Impediment management
Open work area	Engagement of product owner
Team-based estimation	Team morale / motivation
Collective code ownership	Stress at work
Pair programming	Working overtime
Single team	
Self-organizing team	
Planning Practices	**Software Impacts**
Release planning	Product / software innovation
Iteration planning	Software quality
User stories	Software maintainability
Taskboard	Engineering discipline
Burndown charts	Software architecture
Story mapping	Defect rate
Prioritized backlogs	
Short Iterations	

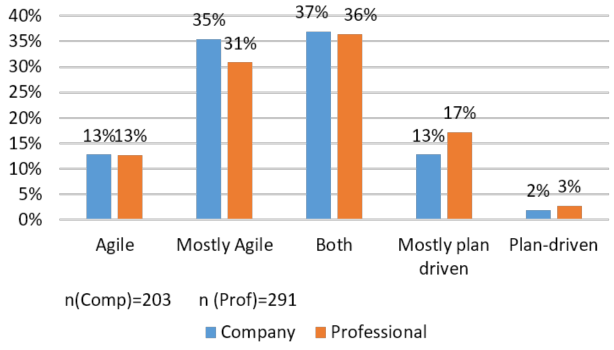


Figure 1: Percentage of companies and individual professionals doing agile on a scale from pure agile to pure plan-driven.

Figure 1 shows the results of the company representatives and individual professionals to the question: *1.1 Is your com-*

pany currently practicing plan-driven or agile software development? The participants could choose on a scale from (pure) Agile, mostly Agile, both, mostly plan-driven, and (pure) plan-driven. Aggregated, 85% of the companies and 80% of the professionals answered that they apply Agile development, at least to some extent; however, only 13% for both, companies and professional, responded that they apply only Agile development. The survey question concerning satisfaction asked *1.3 How satisfied are you with your current methodology?* Possible answers were on a scale from 1 (unsatisfied) to 4 (very satisfied). We have chosen a 4-point Likert scale to force a choice and avoid equivocation. Figure 2 shows the satisfaction results of all participating companies and all individual professionals. In the survey of companies, most representatives responding indicated satisfaction. In the survey of professionals, however, the results were balanced between unsatisfied and satisfied. We speculate that the difference between company representatives and individual professionals may stem from the representatives wanting to present a more positive view of their organization, or may indicate some detachment from the actual experience of software development.

We were especially interested to explore whether Agile development is associated with more satisfaction. Figure 3 shows the analysis of the above question divided into three participation categories. We aggregated the "pure Agile" and "mostly Agile" companies into one "Agile" group, the "pure plan-driven" and "mostly plan-driven" into a "plan-driven (PD)" group and kept the "both" group standalone.

Figure 3 shows a very high satisfaction rate, both for companies and the individual professionals, with very similar values. In the "Both" category the companies still report high satisfaction, while the professionals are not quite so satisfied. However, in the "plan-driven" category companies, i.e. company representatives, still report a high level of satisfaction with the methodology (71%), while only 16% of the professionals report to be satisfied or very satisfied. But 40% of the plan-driven individual professionals report they are unsatisfied with the methodology.

To investigate further, we can compare the level of satisfaction (1–4) reported with the level of agility (from 1: plan-driven to 5: Agile). This is shown in Figure 4, on the left, where each level of Agility is shown on the horizontal axis, and the distribution of satisfaction responses for each is shown by a boxplot.[5] The self-reported level of Agility may not be accurate, so we also show (on the right of the figure) how the level of Agility compares to the mean level reported for a number of Agile technical practices. As we can see, this demonstrates a strong relationship, suggesting a link from the practices, to perception of Agility, to satisfaction.

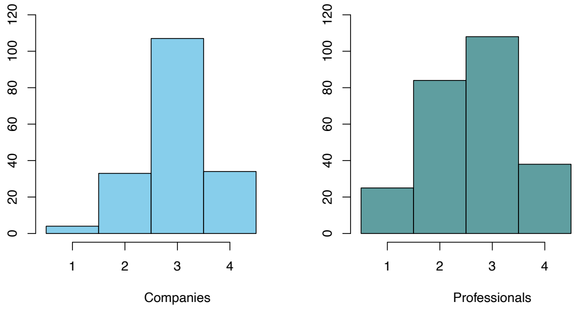


Figure 2: Distribution of reported satisfaction, on a scale from 1 (unsatisfied) to 4 (very satisfied): Company representatives (left) and individual professionals (right).

The company survey data was provided by representatives who were mostly managers, typically senior managers. However, the survey of professionals also included a number of people responding who gave job titles indicating a management role. We therefore explored the levels of satisfaction by such managers compared with developers. We counted as managers anyone with "manager" (e.g. product manager, project manager) or "coach" in their title, 62 in all; we counted as developers anyone with "developer" or similar in their title, 64 in all. The results are shown in Figure 5 illustrating that the level of satisfaction rises with the level of claimed Agile adoption, both for professionals with a management role, and those who are developers. We also explored the manager/developer distinction in many other aspects of the data for individual professionals, and found few differences.

5. Potential Reasons for Satisfaction

In this section, we explore the potential *reasons* for satisfaction, using the data from answers to other questions in the survey. In particular, we use the answers from the survey of professionals, because they were more directly involved with software development, and they were answering for themselves alone. We omit the answers from the companies, i.e. management, here, because they might tend to claim more positive satisfaction than individuals, as Figure 3 indicates.

In a survey of this nature, we actually cannot detect reasons, or causes, for satisfaction, but merely answers that exhibit a close relationship. The survey questions follow a Likert scale approach, and so allow detection of similar patterns using ordinal statistics. We identify the similarities we find, and discuss *how* these relationships might arise.

To examine the relationship between satisfaction and other issues, we compared the answers for satisfaction and for other issues on a person-by-person basis, where each person responded to the same questions. We computed correlation statistics, comparing satisfaction answers with the matching answers for other questions. A correlation shows that when one figure is low, so is the other, and similarly for high. To compute the correlation, we use Spearman's non-parametric "rho" (ρ) method, rather than Pearson's r, because our Likert scale data is ordinal, and this approach supports more conservative results. A rho approaching 1 is an extremely close match, a rho approaching -1 is extremely close but opposite, and a rho approaching 0 is a

[5] Although the Likert data is ordinal, we use boxplots to show distribution in a compact manner. The thick line indicates the median, the coloured box indicates the inner quartiles, the whiskers indicates the outer quartiles, and circles show outliers. Diamond markers show the mean.

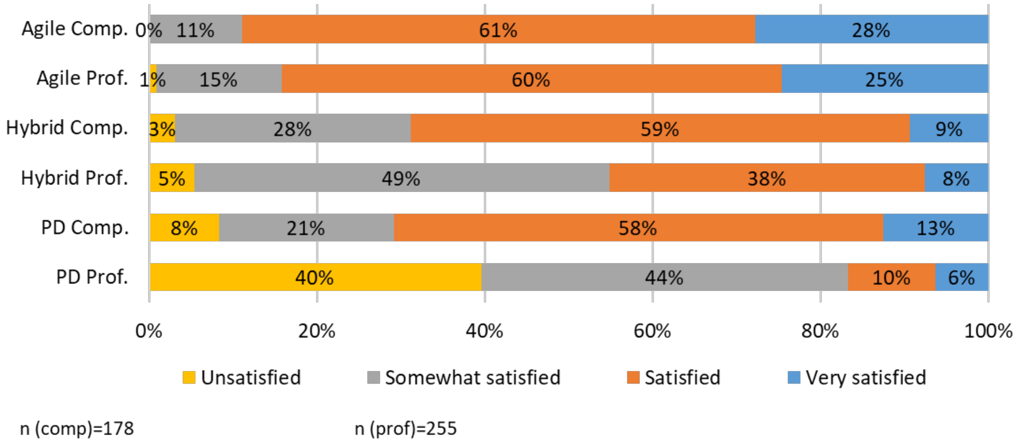


Figure 3: Satisfaction with the methodology aggregated to agile (Agile: pure agile and mostly agile), hybrid, plan-driven (PD: mostly plan-driven, pure plan-driven) for company representatives ("comp") and individual professionals ("prof"), hence "Agile Comp", "Agile Prof", "Hybrid Comp", "Hybrid Prof", "PD Comp", "PD Prof".

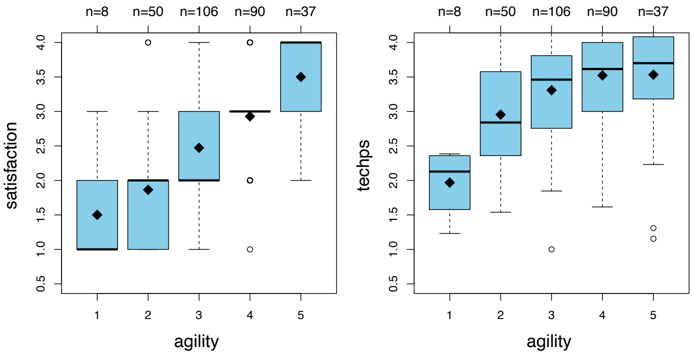


Figure 4: Satisfaction levels by level of agility claimed (left) 1–4, and mean level of technical practices by level of agility (right) 1–5 claimed. Together these show that satisfaction is related to level of agility, and that the claimed level is indeed based on the level of actual technical practices used. (The boxplots show the medians as heavy black lines, inner quartiles as coloured boxes, outer quartiles as whiskers, and the means as diamonds. Numbers at the top show number of particpants in that level.)

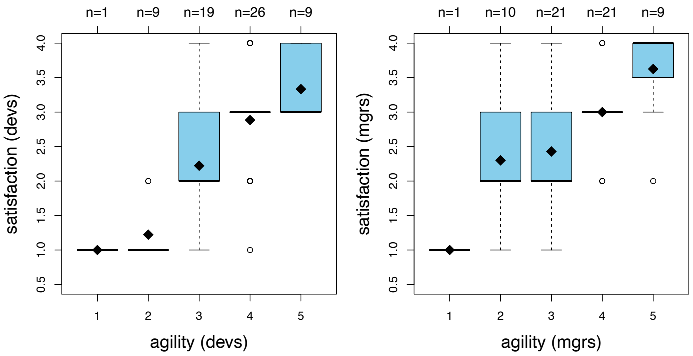


Figure 5: Satisfaction levels by level of agility, for developers (left), and managers (right), both taken from the survey of professionals, showing the relationship between satisfaction and agility is true for both.

very poor match. Note that in our data, our primary referent is the satisfaction question which was rated on a 1–4 scale, while our questions about practices and impacts were rated on a 1–5 scale. This means that the maximum correlation coefficient is .8 rather than 1.

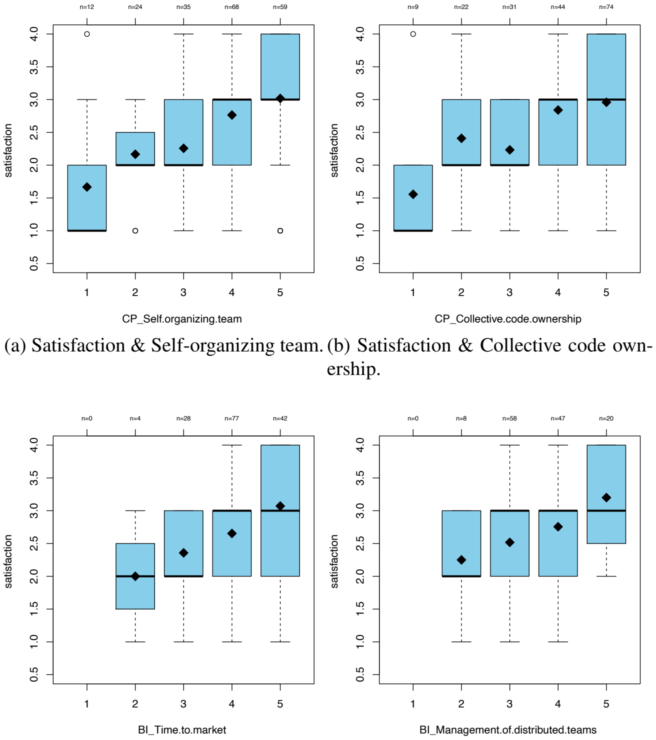
We also calculated significance, the probability that such a result might occur by chance, and dismissed results above an alpha level of 0.05.

Table 5, in the upper section, shows the highest correlations of satisfaction with various answers about software development practices. We sorted the results in decreasing order of rho, so more highly correlated answers are shown first. (More precisely, in order to detect any reverse correlations, we sort by absolute value of rho, but report the true value). In the table, we can see that the highest correlation for satisfaction with practices comes from the collaborative practice of a self-organizing team, followed by that of collective code ownership and Story mapping, and these are the only practices with $\rho > 0.3$. Moreover, the top 5 are all either collaborative practices or planning practices. Although 3 technical practices are in the top 10, the pattern seems clear: it is collaboration and planning practices that most closely match satisfaction.

Moving from practices to impacts, we use the same technique, with the results shown in the lower section of Table 5. Here the most high correlated answer is about time to market. This could be an indication that fast time to market might generate higher satisfaction. Interestingly, the second most highly correlated answer is about management of distributed teams. This might seem odd, because Agile methods are often regarded as poor on this aspect, but the finding simply means that when management of distributed teams is done well, satisfaction is high. Note also row 5 in lower section of Table 5, Software architecture, the highest and only "Software Impact" measure in the top 10. Row 6 is Stress at work: we reverse-coded this aspect, so a high result means lower stress: it makes sense that this is related with high satisfaction. Overall, it is interesting

Table 5: Satisfaction correlations for Agile practices and impacts. Technical practices are prefixed TP, collaborative practices with CP, and planning practices with PP; business impacts with BI, software impacts with SI, team impacts with TI.

#	**Practices Questions**	rho	p.value
1	CP Self organizing team	0.446	<.001
2	CP Collective code ownership	0.375	<.001
3	PP Story mapping	0.306	<.001
4	PP Short Iterations	0.299	<.001
5	CP Single team integrated development and testing	0.293	<.001
6	TP Software Craftsmanship	0.275	0.001
7	PP Prioritized backlogs	0.258	<.001
8	CP Team based estimation	0.247	<.001
9	TP Refactoring	0.245	<.001
10	TP Acceptance Test Driven Development ATDD	0.235	0.001
#	**Impacts Questions**	rho	p.value
1	BI Time to market	0.333	<.001
2	BI Management of distributed teams	0.289	0.001
3	BI Handling of project risk	0.261	0.001
4	BI Development process	0.249	0.002
5	SI Software architecture	0.239	0.003
6	TI Stress at work	0.224	0.007
7	BI Ability to manage changing priorities	0.218	0.006
8	BI Delivery predictability	0.216	0.008
9	TI People development	0.213	0.009
10	BI Project visibility	0.193	0.019



(a) Satisfaction & Self-organizing team.

(b) Satisfaction & Collective code ownership.

(c) Satisfaction & Time to Market.

(d) Satisfaction & Management of Distributed Teams.

Figure 6: Satisfaction levels corresponding levels for most correlated practices (top) and impacts (bottom).

that 7 of the top 10 are business impacts. This suggests that success with business aspects might have a strong impact on, or is necessary for, software professionals' satisfaction.

Considering the practices and the impacts together, it is tempting to see a general picture: satisfaction is highly correlated with collaborative and planning practices, together with success in business aspects. However, this is not the whole story. Referring again to Table 5, we can see that even the highest correlations are only in the range of .3 or .4, and so nowhere near a perfect correlation. This is not surprising, because software development is complex, and we should not expect any one practice or impact to lead to perfect satisfaction. Rather, it makes more sense that several aspects would be necessary for high satisfaction. Moreover, consideration of only correlation is quite limited, and will miss some important patterns, such as close matches for part of a distribution, but divergence elsewhere.

The relationships over the range of responses between these most correlated responses and satisfaction are shown in Figure 6. For each level of response about the practice or impact, the range of responses for satisfaction is shown. Figure 6a, for example, shows that each level of response about Self-organizing teams is associated with a range of satisfaction responses, but the central part of range (the inner quartiles represented by the coloured box) steadily increases with the level of response about Self-organizing teams. A similar pattern is shown in each of the sub-figures, although is less strong in Figure 6d, perhaps because management of distributed teams is not important in all environments.

5.1. *Dominant Issues*

To explore this, we considered several approaches. For example, in studies of complex processes, the approach indicated might be multiple regression, where satisfaction is the dependent variable (DV), and the practices and impacts are the independent variables (IVs), and a formula relating them is sought. We feel, however, that this is more suitable for underlying continuous physical processes. Accordingly, we took an approach that looks for critical points in the data that affect satisfaction. To do this, we used create a Regression Tree [21] using Recursive Partitioning [22][6] In this approach, the analysis begins with the whole data set, and determines which IV, and at what point, best distinctly divides the DV. Thus we obtain two sets, one with lower satisfaction, and one with higher. The process is then applied recursively.

We applied this approach first to the practices, and obtained the trees shown in Figure 7 on the left tree. As we might expect from the earlier correlation analysis, the primary factor is the collaborative practice of a self-organizing team. The tree is

[6] An updated version of this document is available at: `https://cran.r-project.org/web/packages/rpart/vignettes/longintro.pdf`.

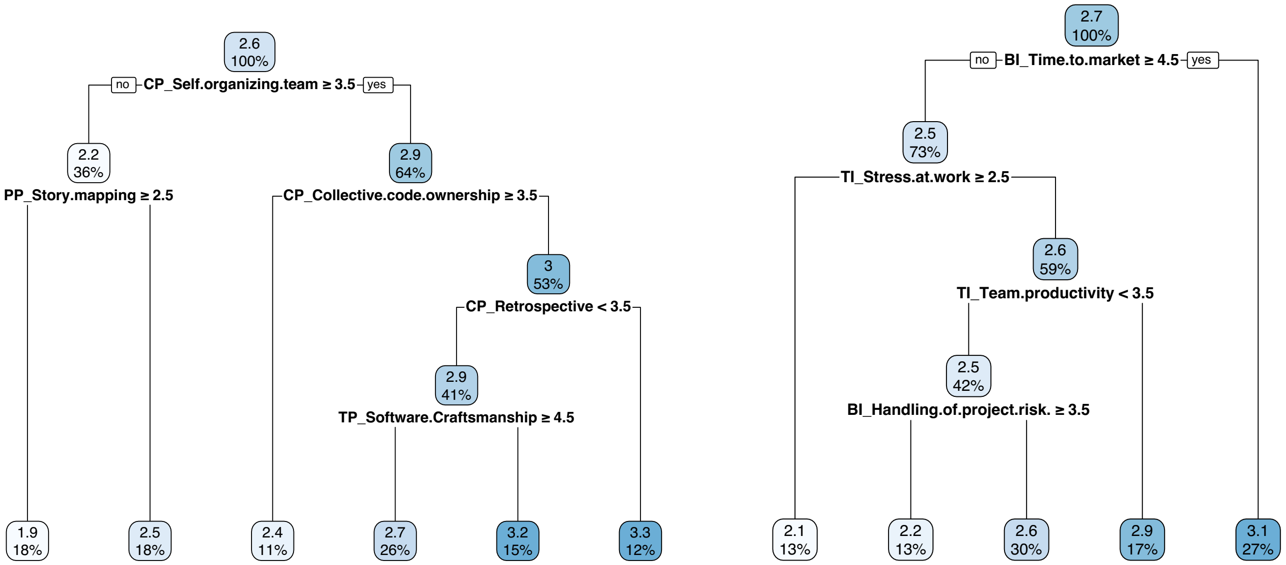


Figure 7: Recursive partition trees for satisfaction factors in practices (left) and impacts (right). Nodes show implied satisfaction level and % of data, leaves limited to 10%. Tone density of nodes indicates levels of satisfaction: darker means higher.

split between results for that question on a rating of 3.5 (on the Likert scale of 1–5), with the lower to the left, and the higher to the right. On the right, we next see, again as we might expect from the correlations, the factor of collective code ownership. Where it is at or above 3.5, the next factor is the collaborative practice of retrospectives, and that gives the highest result for satisfaction: a mean result of 3.1. We can explore the other branches for the tree, to see the effects of other factors. On the left side of the tree we can see the factors related to low satisfaction: the lack of user stories and story mapping appears strongly related to low satisfaction. Overall, the impression is similar to what we expected from the correlations, collaborative practices are paramount, though technical practices also play a role, and we now have more detail to identify which combinations lead to the best results. There is one important caveat. In the tree, note that the right-hand branch indicating higher satisfaction comes from *lower* emphasis on retrospectives. The tree on the right shows the pattern for impacts. Here we see that the primary factor is time to market, and for lesser levels the important issues are stress, productivity, and risk management.

5.2. *Low vs. High Satisfaction Experiences*

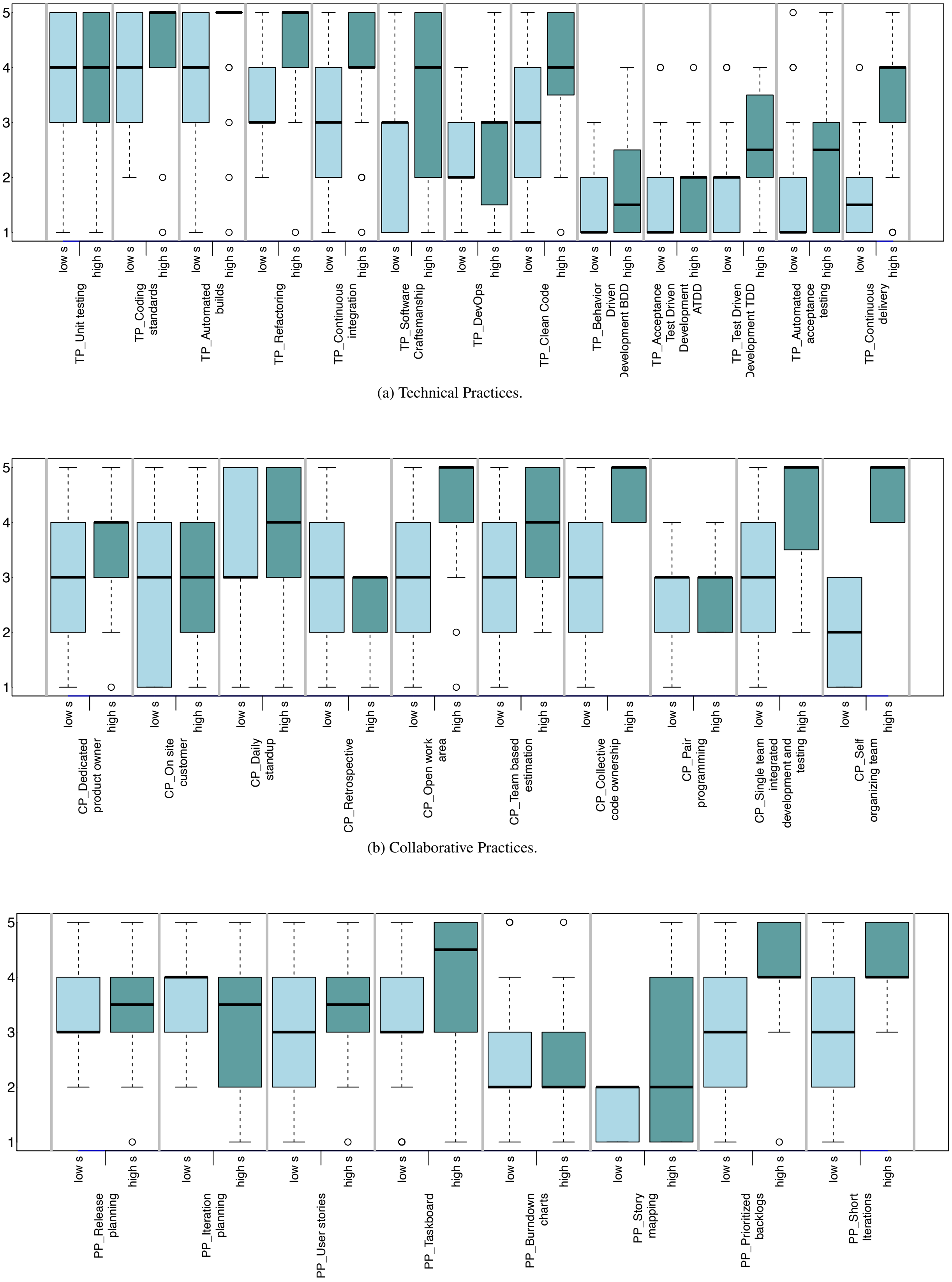
To explore the reality of low and high satisfaction, we divided participants into two groups, those with satisfaction lower than the median, and those with satisfaction higher. We then looked at the range of responses to other questions to see how they differed between the two groups. In this way, we hope to gain understanding of how various issues differ together, rather than simply looking at each issue individually. We focussed on responses about practices, distinguishing the responses from participants with lower satisfaction from those with higher satisfaction. To compare these, we created boxplots showing the distribution of participant responses about use of each practice. For each practice, we show one set of boxplots depicting the responses from participants with lower satisfaction (lighter colour) and another for those with higher satisfaction (darker colour), as shown in Figure 8. This approach highlights the different patterns of responses between the two groups.

Figure 8a shows these distinctions for technical practices. As can be seen, the median response for every question is the same or higher for the high satisfaction group. We see that for some practices the distributions are very similar: e.g. unit testing is high in both. For some practices, however, there was a stark difference: e.g. refactoring, continuous integration, software craftsmanship, clean code, test-driven development, and continuous delivery. These would seem to relate to code quality.

Figure 8b shows the differences for collaborative practices. The distinctions are most clear for an open area, collective code ownership, team integrated testing, and of course a self-organizing team. All those appear to relate to *cohesion* within the team. Perhaps surprisingly, practices such as an on-site customer and pair-programmer show little difference between groups. The striking reason is that neither practice is widely adopted.

Figure 8c shows the differences for planning practices. Release planning, iteration planning, user stories, and burndown charts show little difference. The largest distinctions are for taskboards, prioritized backlogs, and short iterations. Perhaps the theme is one of a sense of progress. One might expect burndown charts to also support that, but use was low for both groups. It might be that the popularity of taskboards means people feel little need for burndown charts.

Considering all these patterns together suggests satisfaction relates to concern for quality work, team cohesion, and support for tracking progress. None of these are surprising, and indeed



(a) Technical Practices.

(b) Collaborative Practices.

(c) Planning Practices.

Figure 8: Results for questions about Practices, showing those in low (lighter colour) vs. high (darker colour) satisfaction groups.

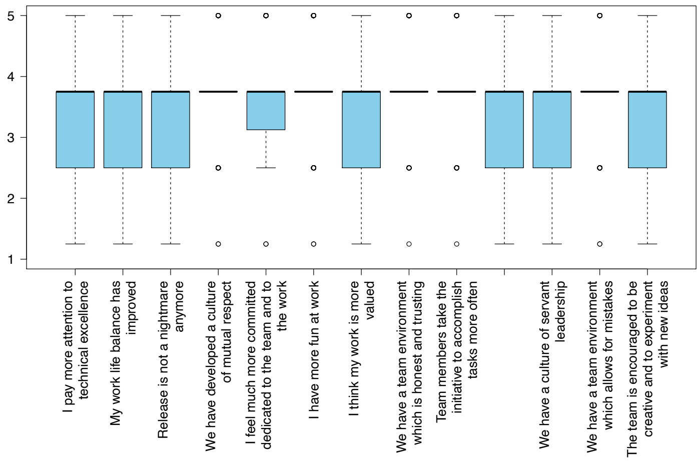


Figure 9: Results for “My Agile” questions 1–13 from Table 6.

#	My Agile
1	I pay more attention to technical excellence
2	My work life balance has improved
3	Release is not a nightmare anymore
4	We have developed a culture of mutual respect
5	I feel much more committed/dedicated to the team and to the work
6	I have more fun at work
7	I think my work is more valued
8	We have a team environment which is honest and trusting
9	Team members take the initiative to accomplish tasks more often
10	The team has been empowered to make decisions about how to do their work and execute on those decisions without outside interference
11	We have a culture of servant leadership
12	We have a team environment which allows for mistakes
13	The team is encouraged to be creative and to experiment with new ideas

Table 6: “My Agile” questions, where each question was ranked on a Likert scale of 1–5.

they reflect findings from qualitative studies, e.g. [4]. What might be seen as more surprising is that, despite widespread emphasis on such characteristics from Agile software development advocacy, many of our participants seem to work in environments where they are lacking.

6. Other perspectives

6.1. *“My Agile”*

In the survey, professionals were also asked questions about their personal perspective on Agile processes, “My Agile”: see Table 6. We have presented these results in more detail elsewhere [23], and here outline the relationship with satisfaction. We acknowledge that the “My Agile” questions themselves are problematic, in that they presume experience of a change to Agile from something else, and may suggest it would be a positive change.

#	My Agile	Practice	rho	p.value
1	The team has been empowered to make decisions about how to do their work…	CP Self organizing team	0.378	<.001
2	I feel much more committed dedicated to the team and to the work	CP Pair programming	0.371	<.001
3	The team is encouraged to be creative and to experiment with new ideas	CP Self organizing team	0.362	<.001
4	Team members take the initiative to accomplish tasks more often	CP Self organizing team	0.355	<.001
5	We have a culture of servant leadership	CP Self organizing team	0.321	<.001
6	We have a team environment which allows for mistakes	CP Self organizing team	0.317	<.001
7	I think my work is more valued	TP Software Craftsmanship	0.309	0.001
8	I think my work is more valued	PP Story mapping	0.300	<.001
9	We have a team environment which allows for mistakes	CP Pair programming	0.299	<.001
10	We have developed a culture of mutual respect	CP Self organizing team	0.298	<.001

Table 7: Correlations between “My Agile” questions and practices (top 10 significant).

The question we asked was: *To what extent do you agree with the following statements?* The participants could choose on a scale from “completely disagree”, to “completely agree” with a 1–5 scale. The general results for each question are shown in the boxplots in Figure 9. One thing we can immediately see is that the results are fairly consistent, with every scale showing the same median, although some distributions are very tight (e.g. “more fun at work”.)

We explored the relationship with satisfaction using the recursive partition approach, obtaining the tree shown in Figure 10. As we can see here, two factors stand out. The dominant finding is a relationship between satisfaction and the factor “I pay more attention to technical excellence”: showing the importance of quality to professionals.

We were interested in the relationship between the results for these questions and those for the practices, so we calculated pairwise correlations for each of the “My Agile” questions with each of the practices questions. We used the same Spearman’s correlation technique as described in section 5, and report the top 10 significant correlations in Table 7. As can be seen, we see several of the same factors we have highlighted before. In particular, having a self-organizing team is the practice most strongly linked to high scores in the “My Agile” questions,

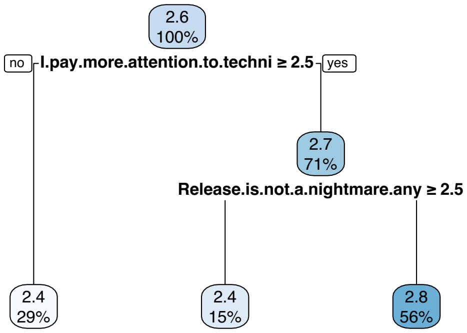

2.6
100%
no
I.pay.more.attention.to.techni ≥ 2.5
yes
2.7
71%
Release.is.not.a.nightmare.any ≥ 2.5
2.4
29%
2.4
15%
2.8
56%

Figure 10: Satisfaction factors in answers to “My Agile” questions.

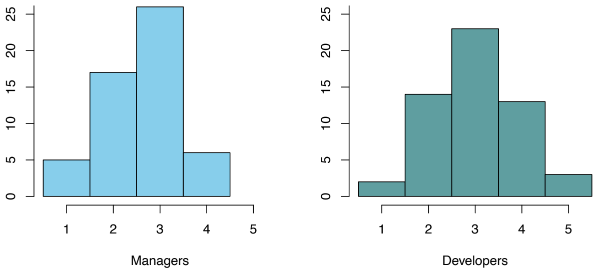


Figure 11: Reported stress by managers (left) and developers (right), on a scale from 1 (unstressed) to 5 (very stressed).

Table 8: Stress correlations for impacts.

	Question	rho	p.value
1	SI Defect rate	-0.439	<.001
2	TI Team morale motivation	-0.413	<.001
3	SI Software architecture	-0.374	<.001
4	SI Software quality	-0.362	<.001
5	BI Requirements management	-0.353	0.001
6	SI Engineering discipline	-0.337	0.001
7	SI Software maintainability	-0.335	0.001
8	TI Engagement of customer product owner	-0.333	0.001
9	BI Ability to manage changing priorities	-0.323	0.002
10	TI Effectiveness of meetings	-0.321	0.002

though some technical practices also appear in the top 10.

Considering the relationship with satisfaction, we looked for differences in the “My Agile” topics with for participants with low and high satisfaction. We did find some relationship, but very little. We again calculated the Spearman’s correlation coefficient between each scale and the satisfaction results, but only found a few correlations, all under $\rho = .25$. While personal experience and overall satisfaction concern similar issues, we found little insight emerging from our analysis.

6.2. Stress

In this paper our focus is on satisfaction, but in other analysis of the same survey, we addressed the subject of stress [24]. In the survey we directly asked IT professionals about their stress at work. They answered on a scale from 1 (significantly less stressed) to 5 (significantly more stressed). Figure 11 displays histograms of the results, showing separately responses from partipants with titles indicating a manageral role from those with titles indicating development work. As we can see there is a range of answers, with most developers reporting a neutral level, and most professionals with management responsibities reporting somewhat less. Although these results are not extreme, they do suggest some reason for concern, with sizeable numbers reporting they are more stressed or significantly more stressed (levels 4 and 5).

To explore how the practices and impacts related to the stress, we looked for correlations. To compute the correlation, we use again Spearman’s correlation measure. Our speculation was a relationship between collaborative processes overall, and stress. We therefore calculated a composite score based on all collaborative practices, and compared it with the stress data. We did not find a strong connection: $\rho = -0.16, p = .05$.

We then explored each of the practices, and each of the impacts, calculating the correlation of each individually with stress. As described in our earlier papers [25, 24] we had determined a hypothesis, so in this analysis we modified p-levels with the Bonferroni correction for multiple tests, and used an alpha level of 0.05. For practices, we found the *only* practice with a significant effect was the “Self-Organizing Team” collaborative practice showing $\rho = -0.27, p = 0.02$ (Bonferroni corrected). On further inspection, we found this relationship was strongest for those with management responsibilities, with $\rho = -0.54$.

Exploring impacts, we found a more diverse picture. Table 8 shows the top 10 correlations, ranked by $|\rho|$. The p-levels again reflect Bonferroni correction for multiple tests, and we omit any results above an alpha level of 0.05.

As can be seen, the impacts that play a role are varied, with software, business, and team impacts all involved. Perhaps most notably, several software impacts (SI) rate highly: lower defect levels, good software architecture, and overall software quality are all associated with lower stress. The business Impacts (BI) also relate to good process outcomes, such as requirements management and ability to manage changing priorities. Team Impacts (TI) reflect a positive environment, such as good morale, an engaged customer, and effective meetings. Looking at differences between managers and developers, we found most of the impact relationships concerned managers, but it was developers who most highly rated low defect rates, ability to manage changing priorities, and morale as most related to reduced stress.

Considering these findings, it seems reasonable to directly consider the relationship between stress and satisfaction. We might expect, for example, that more stress is inversely related to satisfaction. We calculated Spearman’s correlation coefficient for these two ratings, however, and found $\rho = -0.22, p = 0.007$. So there is a significant negative correlation, but at 0.22 it is not very strong. We speculate that stress alone is not the determining factor. As suggested in some earlier work (e.g. [4]), work might be stressful but also satisfying.

6.3. Hindrances

Another section in our survey concerned potential hindrances to Agile software processes. These issue were introduced with the question *How much do the following aspects hinder you to further adopt agile software development in your company?* The issues are shown in Table 9. Participants were asked to respond on five point Likert scale, with "Not at all" being 1, and "Very Strong" being 5. The responses are summarized in Figure 12, where the responses for each topic are shown as boxplots. As can be seen, eight of the ten topics received a median score of 2 ("A little"), and the two others received a median score of 3. However, the ranges shown on the boxplots indicate wide differences.

To highlight the relationship between satisfaction and the possible hindrances, as in section 5.2, we separated the participants into two groups with lower and higher satisfaction ratings, and then looked at the ten hindrance ratings. This is shown in Figure 13, where the two boxplots are shown for each issue: lighter for those that are associated with lower satisfaction, and darker for those associated with higher satisfaction. This allows us to see the pattern of differences between the two groups, and allow the overall distinctions become more clear. As we might expect, almost all hindrance issues show a relationship with satisfaction: hindrances will necessarily hinder something. One hindrance does stand out dramatically, with the leftmost boxplot pair on the graph: "Ability to change organizational culture". For this hindrance, lower satisfaction partipants rated this as really problematic (median 4: "Strong"), whereas higher satisfaction participants rated this not as much of an issue (median 1: "Not at all").

We then looked for correlations between satisfaction and the potential hindrances. These are shown in Table 10. As before the table shows the ordinal correlation coefficient, Spearman's rho ρ, in decreasing order by absolute value, and only showing those with $p < .05$. Again, because satisfaction was rated 1–4 and the hindrances 1–5, the maximum correlation is 0.8. All correlations are negative, not surprisingly showing an increase in hindrance is related to a decrease in satisfaction. In particular, the four strongest (negative) correlations all show a common pattern: *General organizational resistance to change*, *Lack of management support*, *Ability to change organizational culture*, and *Concerns about loss of management control*. This is exploratory post-hoc analysis, so we do not correct for multiple tests, though most values would be well below the alpha level.

7. Discussion

Our research questions were about how satisfaction relates to the development approach, to specific practices, and to specific impacts perceived.

In our findings described in the previous sections, we first noticed confirmation that, for individual professionals, Agile development is associated with greater satisfaction than plan-driven development. We then explored why this might be. We wanted to fine the practices and the impacts most closely related satisfaction.

Table 9: *Agile Hindrances* topics, which participants were asked to rate on a Likert scale of 1–5.

	Hindrances	scale
1	Ability to change organizational culture	1-5
2	General organizational resistance to change	1-5
3	Availability of personnel with necessary agile experience	1-5
4	Lack of management support	1-5
5	Project complexity or size	1-5
6	Business / user / customer availability	1-5
7	Concerns about the ability to scale agile	1-5
8	Perceived time and cost to make transition	1-5
9	Concerns about loss of management control	1-5
10	Regulatory compliance	1-5

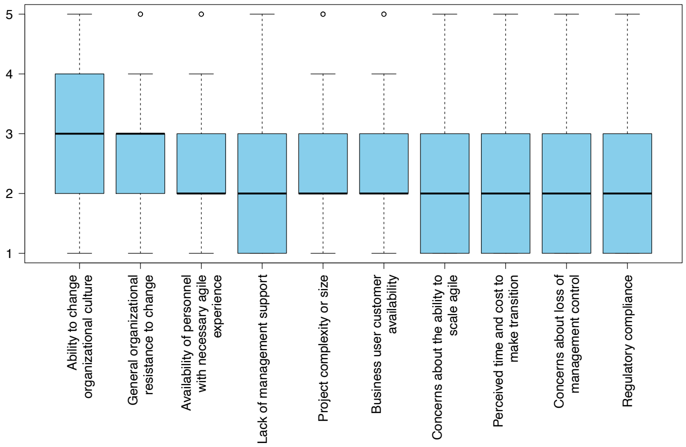


Figure 12: Results for "Hindrances" questions 1–10 from Table 9

When we looked at practices, we considered three kinds: technical, collaborative, and planning practices. What we found was that the strongest relationship with satisfaction came from collaborative practices: self-organizing teams, and collective code ownership. The technical practices, such as software craftsmanship and story mapping, do have an effect, but at lesser levels. Overall, this suggests that self-organizing teams and collective code ownership need to be taken very seriously, otherwise satisfaction might suffer.

For impacts, we enquired about business impacts, team impacts, and software impacts. The dominant factor we found was a business factor: time to market. It seems that teams find satisfaction in delivering quickly. At lesser levels, team impacts such as avoiding stress and maintaining productivity were seen to be important. Although our survey of professionals had mostly developers and low-level managers, it is interesting to see that business impacts are seen as so important: this appears to show the kind of positive relationship between software development and business goals that Agile methods emphasize.

After our main analysis, we considered several other perspectives, based on other parts of our survey. The "My Agile" section sought to find out the personal feelings about the pro-

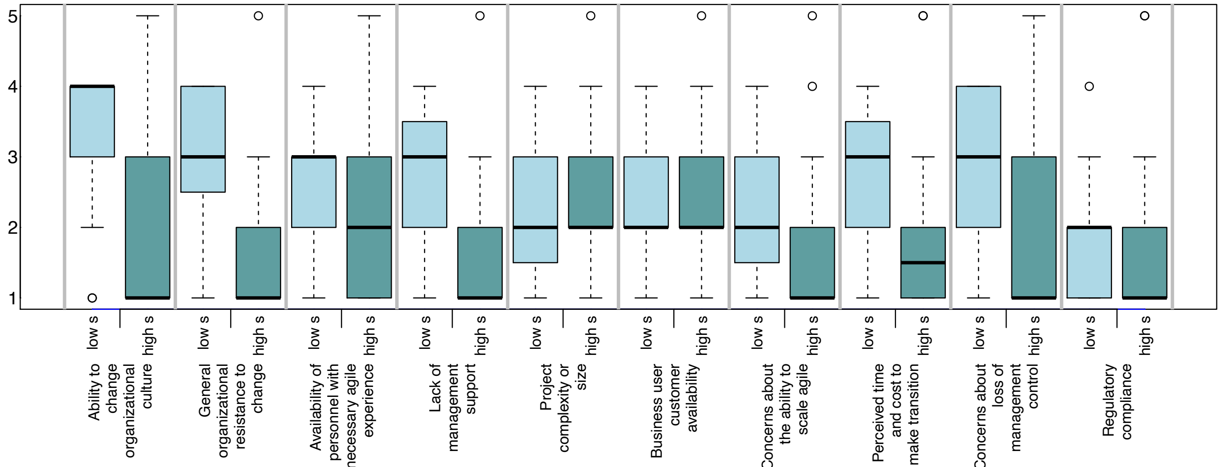


Figure 13: Results for questions about Hindrances, showing those from low (lighter colour) vs. high (darker colour) satisfaction cases.

Table 10: Satisfaction correlations for Agile hindrances.

	Question	rho	p.value
1	HI General organizational resistance to change	-0.440	<.001
2	HI Lack of management support	-0.399	<.001
3	HI Ability to change organizational culture	-0.374	<.001
4	HI Concerns about loss of management control	-0.342	<.001
5	HI Availability of personnel with necessary agile experience	-0.282	<.001
6	HI Concerns about the ability to scale agile	-0.268	<.001
7	HI Perceived time and cost to make transition	-0.237	0.002
8	HI Regulatory compliance	-0.182	0.016

cess. When we looked at the factors linked to satisfaction, the dominant one that emerged was a concern for technical quality. We found this interesting, because technical topics did not appear so important in our analysis of practices or impacts. We then looked at stress, a topic we had examined in detail earlier. That analysis showed that many issues related to satisfaction were also related to stress, so we wondered whether satisfaction was principally related to stress. That turned out not to be clear: perhaps some stress is compatible with satisfaction. Finally, we examined hindrances. In the survey we had asked about a range of potential hindrances, and our analysis showed some are strongly associated with low satisfaction. In particular, the strongest associations all concerned difficulties with management process.

This picture suggests some clear considerations for practitioners and educators. Perhaps the most important lesson relates to collaborative practices: if we expect Agile methods to lead to satisfaction, they cannot be ignored, and must be supported. As educators ourselves, we have already been influenced to emphasize the importance of these practices, even beginning to offer specific courses [26, 27, 28, 29, 30, 31]. This can be challenging because of limited opportunties to engage with real business requirements, and limited time for iterations and change. For practitioners, as our results about hindrances show, the challenges may relate to organizational support. In particular, the role of self-organization seems critical, and so studies of this are important: such as the work of Hoda et. al [32].

For researchers, there are a variety of challenges raised by our study. One arises from the anomalous finding about retrospectives discussed in the previous section: at some point too much emphasis is related to *reduced* satisfaction. So we cannot regard collaborative practices as always beneficial — or perhaps that in some cases practices like retrospectives need to be conducted with more care. More broadly, there is a research challenge identified by the dichotomy of practices and impacts with little emphasis on technical issues, but personal feeling is linked to ability to focus more on technical issues. One possibility is simply that professionals feel they know how to address technical quality, but identify collaborative practices are the key way to ensure time for such concerns. Lastly, we should again consider the association of satisfaction, self managing teams, and the hindrances related to organizational management. The Agile Manifesto was articulated almost two decades ago, and the persistance of difficulties with management practices should be cause for serious reflection.

Our study has several limitations that represent threats to validity. Considering internal validity first, and of particular importance to the topic of this paper, is that we cannot assume correlation reflects a cause and effect relationship. However, our results mean we are now able to identify *potential* cause and effect relationships to explore more specifically in later studies. Another issue is that our focus on Agile practices may bias participants to respond more positively to them. Worse, in some cases, our questions might suggest an expectation that Agile adoption would lead to improvement, thus again biasing responses. This is especially evident in our "My Agile" questions, that suggest a change to Agile would be an improvement. Moreover, younger professionals may not have experienced any other approach, so asking about "change" may not be appropriate.

Considering external validity, the important issue is the generalizability of our results. Within the Swiss context, we felt that the company types and job roles reflected the software industry well, but more careful coverage would be beneficial, especially to attempt representative balance across organizations and domains. Perhaps of greater concern is that the data is self-reported, and indeed self-selected. For example, it is possible that professionals might be more likely to self-select if they were interested in, or even advocates of, Agile methods. In future, we might be better to include questions to detect such bias in order to improve the validity of our results. Our emphasis on Agile methods might also dissuade proponents of more planned approaches from participating, so we should be especially hesitant about any negative findings about planned approaches.

8. Conclusions

In this paper we presented a study of overall satisfaction with software development method in relation to various relevant aspects, including specific practices, impacts, and hindrances. The data came from a survey of IT professionals in Switzerland in 2016.

Overall, we can describe the picture that emerges as follows. Agile development seems related to greater satisfaction primarily because of collaborative practices and business impacts. Technical practices and team impacts are important, but at lesser levels. On a personal basis, however, an ability to focus more on technical quality is seen as critical. Hindrances related to management issues are still a problematic issue.

The study has several limitations as we have discussed. In particular, the survey participants were self-selected, and the emphasis on Agile may have led to bias in favour of professionals who advocate that approach. Moreover, the survey was restricted to professionals in Switzerland, and may not reflect attitudes common elsewhere. Finally, the survey format did not allow participants to elaborate their reasoning in depth. We hope all these issues may be addressed in future work.

Acknowledgments

We especially thank the study participants for their most valuable answers, the anonymous referees of this paper, and the Swiss IT organizations swissICT and SWEN for supporting and funding the study.

References

[1] M. Kropp, A. Meier, Swiss Agile Study 2014, Tech. Rep. ISSN: 2296-2476, Swiss Agile Study, `http://www.swissagilestudy.ch/files/2015/05/SwissAgileStudy2014.pdf` (2015).

[2] VersionOne, 11th State of Agile Survey, Tech. rep., VersionOne, Inc. (2017).

[3] Agile Manifesto Signatories, Agile Manifesto, `http://agilemanifesto.org`, 2001.

[4] E. Whitworth, R. Biddle, The social nature of agile teams, in: Proceedings of the AGILE 2007 Conference, IEEE Computer Society, Washington, DC, USA, 2007, pp. 26–36.

[5] B. Boehm, R. Turner, Using risk to balance agile and plan-driven methods, Computer 36 (6) (2003) 57–66.

[6] M. Kropp, A. Meier, C. Anslow, R. Biddle, Satisfaction, practices, and influences in agile software development, in: Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering, ACM, 2018, pp. 112–121.

[7] K. Mannaro, M. Melis, M. Marchesi, Empirical analysis on the satisfaction of IT employees comparing XP practices with other software development methodologies, in: Proc. of International Conference on Extreme Programming and Agile Processes in Software Engineering (XP), Springer, 2004, pp. 166–174.

[8] V. R. Basili, Applying the goal/question/metric paradigm in the experience factory, Software Quality Assurance and Measurement: A Worldwide Perspective (1993) 21–44.

[9] G. Melnik, F. Maurer, Comparative analysis of job satisfaction in agile and non-agile software development teams, in: Proc. of International Conference on Extreme Programming and Agile Processes in Software Engineering (XP), Springer, 2006, pp. 32–42.

[10] B. Tessem, F. Maurer, Job satisfaction and motivation in a large agile team, in: Proc. of International Conference on Extreme Programming and Agile Processes in Software Engineering (XP), Springer, 2007, pp. 54–61.

[11] J. R. Hackman, G. R. Oldham, Work redesign, Addison-Wesley, 1980.

[12] J. F. Tripp, C. Riemenschneider, J. B. Thatcher, Job satisfaction in agile development teams: Agile development as work redesign, Journal of the Association for Information Systems 17 (4) (2016) 267.

[13] J. F. Tripp, C. K. Riemenschneider, Toward an understanding of job satisfaction on agile teams: Agile development as work redesign, in: 2014 47th Hawaii International Conference on System Sciences, 2014, pp. 3993–4002. doi:10.1109/HICSS.2014.494.

[14] H. Robinson, H. Sharp, Collaboration, communication and co-ordination in agile software development practice, in: I. Mistrík, A. van der Hoek, J. Grundy, J. Whitehead (Eds.), Collaborative Software Engineering, Springer, 2010, pp. 93–108.

[15] S. Wood, G. Michaelides, C. Thomson, Successful extreme programming: Fidelity to the methodology or good teamworking?, Information & Software Technology 55 (4) (2013) 660–672.

[16] T. Dybå, T. Dingsøyr, Empirical studies of agile software development: A systematic review, Information and Software Technology 50 (9) (2008) 833 – 859.

[17] Y. Lindsjørn, D. I. Sjøberg, T. Dingsøyr, G. R. Bergersen, T. Dybå, Teamwork quality and project success in software development: A survey of agile development teams, Journal of Systems and Software 122 (2016) 274 – 286.

[18] M. Hoegl, H. G. Gemuenden, Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence, Organization science 12 (4) (2001) 435–449.

[19] M. Kropp, A. Meier, Swiss Agile Study 2016, Tech. Rep. ISSN: unpublished, Swiss Agile Study, `http://www.swissagilestudy.ch/files/2017/09/3.SwissAgileStudy.pdf` (2017).

[20] M. Kropp, A. Meier, Swiss Agile Study 2012, Tech. Rep. ISSN: 2296-2476, Swiss Agile Study, `http://www.swissagilestudy.ch/files/2015/05/SwissAgileStudy2012.pdf` (2012).

[21] L. Breiman, J. Friedman, C. Stone, R. Olshen, Classification and regression trees, CRC press, 1984.

[22] T. M. Therneau, E. J. Atkinson, Technical report series no. 66: An introduction to recursive partitioning using the RPART routines, Tech. rep., Mayo Foundation for Medical Education and Research, Rochester, Minnesota, USA (1997).

[23] R. Biddle, A. Meier, M. Kropp, C. Anslow, Myagile: sociological and cultural effects of agile on teams and their members, in: Proceedings of the 11th International Workshop on Cooperative and Human Aspects of Software Engineering, ACM, 2018, pp. 73–76.

[24] A. Meier, M. Kropp, C. Anslow, R. Biddle, Stress in agile software development: Practices and outcomes, in: J. Garbajosa, X. Wang, A. Aguiar (Eds.), Agile Processes in Software Engineering and Extreme Programming, Springer International Publishing, Cham, 2018, pp. 259–266.

[25] M. Kropp, A. Meier, R. Biddle, Agile practices, collaboration and experience: An empirical study about the effect of experience in agile software development, in: Product-Focused Software Process Improvement: 17th International Conference, PROFES 2016, Trondheim, Norway, November 22-24, 2016, Proceedings 17, Springer, 2016, pp. 416–431.

[26] C. Anslow, F. Maurer, An experience report at teaching a group based agile software development project course, in: Proceedings of the 46th ACM Technical Symposium on Computer Science Education, SIGCSE '15, ACM, New York, NY, USA, 2015, pp. 500–505. doi:10.1145/2676723.2677284.
URL `http://doi.acm.org/10.1145/2676723.2677284`

[27] A. Meier, M. Kropp, G. Perellano, Experience report of teaching agile collaboration and values: Agile software development in large student teams, in: 2016 IEEE 29th International Conference on Software Engineering Education and Training (CSEET), 2016, pp. 76–80. doi:10.1109/CSEET.2016.30.

[28] M. Kropp, A. Meier, R. Biddle, Teaching agile collaboration skills in the classroom, in: 2016 IEEE 29th International Conference on Software Engineering Education and Training (CSEET), 2016, pp. 118–127. doi:10.1109/CSEET.2016.27.

[29] A. Martin, C. Anslow, D. Johnson, Teaching agile methods to software engineering professionals: 10 years, 1000 release plans, in: H. Baumeister, H. Lichter, M. Riebisch (Eds.), Agile Processes in Software Engineering and Extreme Programming, Springer International Publishing, Cham, 2017, pp. 151–166.

[30] M. Kropp, C. Anslow, M. Mateescu, R. Burkhard, D. Vischi, C. Zahn, Enhancing agile team collaboration through the use of large digital multitouch cardwalls, in: H. Baumeister, H. Lichter, M. Riebisch (Eds.), Agile Processes in Software Engineering and Extreme Programming, Springer International Publishing, Cham, 2017, pp. 119–134.

[31] K. Lundqvist, C. Anslow, M. Homer, K. Bubendorfer, D. Carnegie, An agile conversion masters degree programme in software development, in: Proceedings of the 49th ACM Technical Symposium on Computer Science Education, SIGCSE '18, ACM, New York, NY, USA, 2018, pp. 846–851. doi:10.1145/3159450.3159540.
URL `http://doi.acm.org/10.1145/3159450.3159540`

[32] R. Hoda, J. Noble, S. Marshall, Self-organizing roles on agile software development teams, IEEE Transactions on Software Engineering 39 (3) (2013) 422–444.

*Biography

Satisfaction and its Correlates in Agile Software Development

Author Biographies

MARTIN KROPP:

Martin Kropp is professor for Software Engineering at the Institute of Mobile and Distributed Systems at the University of Applied Sciences and Arts Northwestern Switzerland. In the research group Efficient Software Development, he aims to develop concepts and tools to make software developers and software development teams more productive. His interests are in build automation, testing, refactoring and methodologies. His current focus is especially on agile methodologies and tools supporting agile teams in their daily work and their collaboration. He is co-founder of the Swiss Agile Research Network, and member of the management board of the Software Engineering Network.

ANDREAS MEIER:

Andreas Meier is a lecturer and researcher of software engineering at Zurich University of Applied Sciences. He teaches classes on programming and software engineering both on the bachelor and the master level. Additionally, he is involved in a start-up company in the software industry. His special interests are agile methodologies and their positive influence on innovation, team spirit and software systems. He is co-founder of the Swiss Agile Research Network and introduced the Swiss Agile Study to deliver independent facts and figures to the interested public.

CRAIG BIDDLE:

Dr. Craig Anslow is a Lecturer (Assistant Professor) in Software Engineering within the School of Engineering and Computer Science at Victoria University of Wellington where he leads the Human Computer Interaction and Visualization Research Group. Craig also teaches on the MSc in Software Engineering Programme within the Department of Computer Science at the University of Oxford, UK. Craig's research interests are visualization, interactive surfaces and spaces, VR for healthcare, and human aspects of software development.

ROBERT BIDDLE:

Robert Biddle is a Professor at Carleton University in Ottawa, Canada, appointed both to the School of Computer Science and the Institute of Cognitive Science. His research is primarily in human factors in cyber-security and software design, especially creating and evaluating innovative designs for computer security software and collaborative software development.