Agile Software Development Practice Adoption Survey

Narendra Kurapati[1], Venkata Sarath Chandra Manyam[1], and Kai Petersen[1,2]

[1] Blekinge Institute of Technology, 37140 Karlskrona, Sweden
[2] Ericsson AB, Box 518, SE-371 23, Karlskrona, Sweden
{naku10,vemc10}@student.bth.se, kai.petersen@bth.se,
kai.petersen@ericsson.com

Abstract. Agile methodologies are often not used "out of the box" by practitioners, instead they select the practices that fit their needs best. However, little is known which agile practices the practitioners choose. This study investigates agile practice adoption by asking practitioners which practices they are using on project and organizational level. We investigated how commonly used individual agile practices are, combinations of practices and their frequency of usage, as well as the degree of compliance to agile methodologies (Scrum and XP), and as how successful practitioners perceive the adoption. The research method used is survey. The survey has been sent to over 600 respondents, and has been posted on LinkedIn, Yahoo, and Google groups. In total 109 answers have been received. Practitioners can use the knowledge of the commonality of individual practices and combinations of practices as support in focusing future research efforts, and as decision support in selecting agile practices.

Keywords: Software Development, Agile Practices, Adoption, Survey.

1 Introduction

In response to the need of reacting to changes in customer needs quickly agile methodologies have gained considerable importance in the software development industry. A variety of software development methodologies (e.g. Scrum, Extreme Programming (XP, etc.) and their related practices have gained attentions from research, with the majority of empirical research focusing on XP [2]. Overall, many practices are considered as being part of the agile toolbox, different articles identifying 21 [9] up till 32 [12] agile practices.

Given that companies work in different contexts the "out of the box" agile methodologies are often not followed as they are described in the books. Instead, companies select the practices that fit their needs [5,12,11]. As the main focus of past research was on individual case studies [2] there is a research gap with regard to which practices the software industry at large is using. We only identified two surveys of relevance focusing on agile practices [1,10]. The survey by [1] focuses on which methodologies were used, but does not ask for actual usage on

C. Wohlin (Ed.): XP 2012, LNBIP 111, pp. 16–30, 2012.

practice level. Sochova [10] only included a very limited number of agile practices in her survey. Overall, this raises the need for a survey study on agile practice adoption to better understand, which agile practices are actually used in industry.

In response to the above mentioned research gap we conducted a survey to find out which agile practices are used in industry. In particular, the survey aims at answering the following research questions:

- *RQ1:* How commonly used are individual agile practices?
- *RQ2:* Which agile practices are used together by practitioners, and how common are the combinations?
- *RQ3:* To what degree does the software industry comply to Extreme Programming (XP) and Scrum?
- *RQ4:* As how successful do the practitioners perceive the adoption of agile practices in terms of customer satisfaction and employee satisfaction?

The remainder of the paper is structured as follows: Section 2 presents related work. Section 3 describes the research design. Section 4 contains the results of the survey. Section 5 concludes the paper.

2 Related Work

The related work focuses on the identification of agile practices and related methodologies, and population oriented studies that have a specific focus on agile practice adoption.

2.1 Literature on Agile Practices

Dybå et al. [2] conducted a systematic review on empirical research investigating agile software development. They found that the main focus of past research was on XP, while very few studies targeted other methodologies, such as Scrum. With regard to classification of research methods four surveys were identified, none focusing on agile practices used within the methodologies.

Williams [12] conducted a literature review and identified 32 agile and lean software development practices that are related to well known agile methodologies, namely XP, Scrum, feature driven development (FDD), and Lean. Some practices are cross-referenced given that the practices are similar, but have different names in different agile methodologies. She also points out that organizations tend to select practices creating hybrid methodologies.

Shashank and Darse [9] conducted a systematic literature review focusing on the identification of agile practices and how they are adapted, e.g. different ways of how pair programming is done. They identified 21 different practices and their adaptations in form of sub-practices.

Petersen [8] identified 22 agile practices and 4 lean practices. The practices have been mapped to agile and lean principles to highlight the differences and similarities between lean and agile software development.

Jalali and Wohlin [3] identified 25 agile practices used in global software development through a systematic review of literature.

Koch [6] provides a summary of agile methodologies and practices. After presenting the agile principles and practices he links the practices to different software development methodologies.

The studies [12,9,8,3] have been used as input for the construction of the agile survey used in this study. The practice list used in the survey presents a consolidated view of agile practices, where some practices were merged and others split. The linkage of agile practices used in our survey and agile methodologies was based on descriptions in [6,12].

2.2 Agile Surveys

We identified two surveys focusing on the adoption and use of agile practices in industry.

Dogs and Klimmer [1] conducted a survey in 2004 where they received 84 responses. The goal of the survey was to capture which agile methodologies are most frequently used, success with respect to defects and user experience, and the perceived usefulness of different agile practices. The most commonly used methodology was XP (38.6%) followed by FDD (14.55%), RUP (11.9%) and Scrum (7.2%). Furthermore, a number of methodologies with less than 5% of all answers were identified. A ranking of how many responses state that a practice was used successfully showed that more than half of the identified practices received more than half of the total number of responses. However, the study does not look into which practices are actually selected independently of a methodology.

Sochova [10] conducted a survey on agile adoption receiving 181 responses in a three month period in 2009. The survey focused on reasons to start agile, difficulty of using agile practices, and actual usage of agile practices. The focus was on 9 agile practices (stand-up meeting, backlog, burn-down, pair-programming, TDD, estimations in points, planning poker, customer demo, and retrospective). With respect to ease of use Scrum, backlog, and burn-down, retrospective, and customer demo were perceived as the easiest. Pair programming, TDD, and estimation were perceived as hard to learn. With regard to usage the least used practices are planning poker, TDD, pair programming with more than 30 people not using them. Scrum standup (not used by 2) and customer demo (not used by 3) are used by almost all of the respondents. The survey is limited in the sense that it only focuses on very few practices.

Korhonen [7] surveyed three agile teams with regard to their agile adoption in different points in time. The practices considered were daily practices (user stories, product backlog, and short iteration), team practices (refactoring, Scrum, self-organized teams), and programming practices (collective code ownership, pair programming, refactoring, tests written at the same time as code, TDD, and continuous integration). The specific focus was on determining, which practices were used by teams without programming responsibility. Their general findings were that teams with no direct responsibility for programming adopted agile

practices related to daily practices and team practices. Only the programming team would in addition to that also adopt programming practices, but at the same time rely on the other two categories of practices as well.

Overall, the related work shows that there are investigations on agile practices, where the practices are mostly investigated through case studies. We found very few surveys, where one (Sochova [10]) was of limited rigor, not discussing e.g. validity threats. This motivates the work presented in this paper, focusing on agile practice adoption in software industry through a survey to get a broader picture of what practices are actually used in the industry. Finding this out is of interest as several researchers recognized that companies tailor their practice selection to their needs [5,12,11].

3 Survey Design

Sampling and Population: The survey was sent out to 600 practitioners that were sampled by diversity (different countries, companies, and domains), and was posted on LinkedIn, Yahoo, and Google groups. The population comprises software industry practitioners who are experienced in agile software development. In total we received 109 valid answers from practitioners.

Survey Structure: The survey consisted of six different parts, namely introduction, demographics, agile practice adoption, agile practice adaptation, employee and customer satisfaction, and contact details. In total the survey comprised of 217 questions.

Part 1: Introduction: The introduction shortly explained the purpose, benefits for the respondents, definition of population (who should fill in the survey), estimated duration of the survey (30 Minutes), and information about the researchers conducting the study.

Part 2: Demographics: This part captured information about the respondent and his/her organization. The practitioners also selected whether they want to answer the survey for a single project or their organization. The reason for doing so was that many roles in software organizations are not necessarily involved in the project work. For example, in market-driven development market analysts package requirements and based on the availability of requirements one or several projects are initiated, i.e. there is a pre-activity before the actual development projects start.

Part 3: Agile Practice Adoption: The respondents got a list of agile practices with a short description of the practice, and selected those that they use. An overview of the practices is given in Table 1. As mentioned earlier, several reviews on agile practices have been conducted [12,9,8,3], these have been used to create a consolidated list of practices for the survey. The mapping of the practices to the two methodologies checked for level of adoption is based on the book by Koch [6] on agile software development (see Appendix E for XP and Appendix H for Scrum) and the analysis of compliance is based on this book.

ID	25 Agile Practices	XP	SCRUM
Of	Office (office structure that supports agile development)	√	
PP	Pair Programming (two people communicate and program on the same computer)	√	
PG	Planning Game/On-site customer (Interaction between customer and developer for effort estimation, scope, and timing of release)	√	
TP	Tracking Progress (tracking of progress of project)		√
40H	40 Hour Week (number of working hours for project per week)	√	
Ref	Refactoring (Restructuring code for better understandability and reduced complexity)	√	
Ret	Retrospective (Discussing good and weaker aspects at the end of the iteration)		
SR	Short/small Releases (Delivering less more often to the customer)	√	
SD	Simple Design (Goal to design the simplest solution)	√	
Sp	Sprint/Iteration (Iterative development considering a sprint/time box)		√
SPM	Sprint Planning Meeting (Meeting to select features for following sprint/iteration)		√
SRM	Sprint Review Meeting (Review meeting to discuss about the work after each sprint/iteration)		√
S-ups	Stand-ups (Short every-day meeting to discuss what was done the last day)		√
St	Stories/Features (Feature/Stories are short statements of the functionality desired by the system user/customer)		√
Team	Team (Work in teams, e.g. co-located, distributed, project manager, SCRUM Master)	√	√
TDD	Test Driven Development (Unit tests developed before coding the software)	√	
Testing	Testing (Continuous testing, unit, integration, acceptance)		
CS	Coding Standards (Coding rules followed by programmer)	√	
CO	Collective Ownership (Anyone can change code, not owned by individuals)	√	
Comm	Communication (Different channels of communication among team members, e.g. face to face, Skype, etc.)		
C&CM	Configuration and Change Management (Enables identification of historical tracking/versioning and change management process)		
CI	Continuous Integration (Regular integration of code)	√	
Doc	Documentation (Importance/emphasis on documentation, e.g. architecture)		
IW	Informative Workshops (Feedback mechanism among agile teams which supports in the daily work)		
Me	Metaphors (Terms used to define the system between customer and developer for reduction of ambiguities)	√	

Fig. 1. Identified Practices and their Mapping to XP and Scrum

Part 4: Agile Practice Adaptation: For each practice selected in Part 3 the practitioners provided answers of how they adapted the practice to their organization by choosing sub-practices.

Part 5: Employee satisfaction and customer satisfaction: This part focused on the outcome achieved when using the agile practices with respect to perceived employee and customer satisfaction. Only a sub-set of the respondents answering Part 1, Part 2, and Part 3 completely answered with respect to customer satisfaction. In our analysis, hence only a sub-set of the answers is reprsented.

Part 6: Contact details: The respondents were free to provide their contact details so that the results of the survey can be made available to them. Furthermore, getting the contact details allowed us to ask further questions with respect to the answers received.

The focus of the study presented here was on agile practice adoption, which makes use of the answers received in Part 2, 3, and 5 of this survey.

Prior to running the survey from October 11, 2011 till November 11, 2012 the survey was reviewed from two researchers (one full professor and one PhD student) and two practitioners, who in informal interviews provided feedback and suggested changes, that were incorporated.

Analysis: The analysis for RQ1 was done using descriptive statistics. As RQ2 is focusing on combinations of agile practices used in industry, we used hierarchical cluster analysis and agglomerative clustering to find similar groups of practices. As a distance measure Euclidean distance was used. RQ3 was also analyzed through descriptive statistics.

Validity Threats: In surveys there is always a risk that questions are misunderstood. In order to reduce the risk we conducted interviews with two practitioners and two researchers who work/do research in agile software development.

Furthermore, the outcome might be biased with respect to similarities of the respondents. However, respondents from different domains, experience levels, etc. answered the survey, even though a limited number of responses was obtained. Hence, this threat is partially reduced. One threat remaining is that respondents represent different roles and project types. Given that a previous study (cf. [7]) showed that depending on the type of project there are different usages of agile practices, there might be a risk that the results are biased towards programming oriented projects, as the majority of respondents were programmers. With regard to the project managers, we also do not know whether they used agile in a programming project.

Evaluation apprehension was avoided by guaranteeing anonymity to the respondents, and not forcing them to provide their contact details if they do not want to.

Hypotheses guessing is a threat, which means that the practitioners might provide answers the researcher wants to hear. However, we only revealed the information that we are seeking to find which practices are used in industry; not, for example, that we intend to check conformance to development methods, which would likely have biased the practitioner to select certain practices.

Given that a web survey was posted in on-line communities, and requests for filling in the survey by e-mail were sent, there was no control for the researchers with respect to external validity (i.e. the general applicability of the results). What can be observed is that few practitioners from military domain have answered the survey, however, for other domains such as information systems, outsourced, commercial, end-user, and embedded several answers have been received.

The survey is long and hence maturation is a threat to validity. Given that we captured not only practice adoption, but also adaptation (how each individual practice is used by them), there is a risk that the practitioners might get bored. Though, in order to get a complete picture of agile practice adaptation there is a need to ask detailed questions of how agile practices are used. This is more a threat for the overall survey, as the questions relevant to this study were asked in Part 3, which was very early in the survey. Hence, for the results presented in this paper the threat of maturation is low.

Only a sub-set of the respondents answered questions with respect to their satisfaction with respect to agile, so they do not represent the full set of respondents based in which we captured agile practice adoption. However, we decided to still present the results as they give some indication of whether the agile practice adoption as presented in this survey was a success.

4 Results

4.1 Survey Demographics

Table 1 shows the results of system type for projects and organizations. When defining the system types we followed the recommendation by Jones [4]. Observe that the total number of responses is higher than 109 as an organization or team can work on different types of systems at the same time (e.g. a commercial end user system). What can be observed is that the majority of the responses come from the information system domain (38%), followed by outsourced (20%) and commercial (19%). All types are accounted for with regard to the total responses. On project level, no answers have been received from the military domain.

Table 1. Number of Responses per System Type

System type	Project	Organization	Responses Total	Percentage Total
Information Systems	16	50	66	38%
Outsourced (developed under contract)	7	28	35	20%
Commercial (marketed to external client, e.g. sold on CD)	8	25	33	19%
End user (private, for personnel use, e.g. banking software)	4	11	15	9%
Embedded	2	12	14	8%
Other	3	4	7	4%
Military	0	4	4	2%
Total	40	134	174	100%

Table 2 shows the distribution of responses by role, showing that all roles are covered in the survey, in particular programmers, project managers, agile coaches, and business analysts are well represented. Furthermore, the respondents are experienced in software development, which is indicated by the average experience.

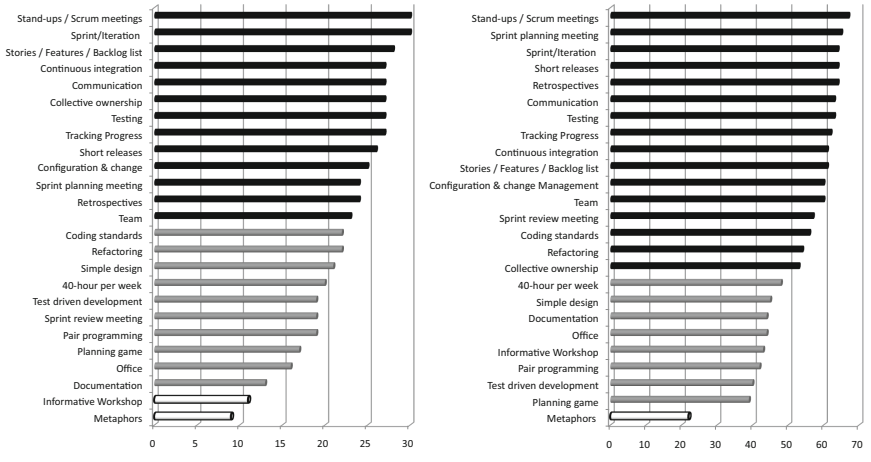
Table 2. Respondents and Experience

Role	Responses	Percentage	Avg. exp (years)
Programmer	55	24.34	11
Project Manager	52	23.01	10
Agile Coach	29	12.83	4.7
Business Analyst	27	11.95	9.7
System Designer	20	8.85	11.8
System Analyst	18	7.96	8
Quality Assurance	14	6.19	5
Researcher	11	4.87	9.75
Total	226	100.00	

4.2 Commonality of Agile Practice Usage

Here we investigated the commonality of each individual agile practice, as is shown in Figure 2. As a means for structuring the data we define three categories, namely:

- Common: Used by $> 2/3$ of the respondents (represented by black bars).
- Less common: Used by $[1/3;2/3]$ of the respondents (represented by gray bars).
- Seldom: Used by $< 1/3$ of the respondents (represented by white bars).



(a) Use of Agile Practices Project Level (33 respondents)

(b) Use of Agile Practices Organizational Level (76 respondents)

Fig. 2. Frequeny of Agile Practice Usage

Project: Common practices are stand-ups, sprint and iteration, stories and features, continuous integration, communication, collective ownership, testing, tracking progress, short releases, configuration and change management, sprint planning meeting, retrospectives, and team.

Less common practices are coding standards, refactoring, simple design, 40-hour week, test-driven development, sprint review meeting, pair programming, planning game, office, and documentation.

Seldom practices are informative workshop and metaphors.

Organization: Common practices are stand-ups, sprint planning meeting, sprint and iteration, short releases, retrospectives, communication, testing, tracking progress, continuous integration, stories and features, configuration and change management, team, sprint review meeting, coding standards, refactoring, and collective ownership.

Less common practices are 40-hour week, simple design, documentation, office, informative workshop, pair programming, test-driven development, and planning game.

Metaphors is a practice applied rarely.

Comparison: Comparing the responses on project and organizational level we can see that the answers show a high level of agreement with respect to how commonly the practices are used. With regard to common practices on organizational level we find sprint review meeting, coding standards, and refactoring, which are less common practices on project level. Otherwise, the common practices are the same. With regard to the less common practices there is also a high agreement, project level having coding standards, refactoring, and sprint review meeting in that category, while organizational level has informative workshop, which is rated less common on project level. Metaphors fall in the seldom category for projects and organizations, while informative workshops are seldom for project level, but not organizational level.

4.3 Combination of Agile Practices

Table 3 shows the results of the hierarchical cluster analysis on project and organizational level for practices. The data is sorted in ascending order for distance. If many respondents choose a similar set of practices they are likely to end up in one cluster. Overall, the table shows which practices are used together on project and organizational level. The following information shown in the table should be highlighted:

- When comparing project and organization, the distance between items on organizational level is larger than on project level. One possible explanation might be that practices on organizational level are more spread as an organization might run projects with varying practices in each project.
- Similarities between project and organization: Test driven development (TDD) and pair programming (PP) are in the same cluster for project and organization, even though the distance is much lower on project level (2.236 on project level in comparison to 4.472 on organizational level). Cluster 7 (Stand-ups, Sprint/iteration, Sprint planning meeting, retrospective) for project is the same as Cluster 3 for organization with similar distance values (3,073 on project level and 3,231 on organizational level) that are both relatively low given that the largest distance is 4.144 and 6.476 for project and organization, respectively. The identity of clusters 3 and 7, while having low distance values at the same time, would indicate that the clusters are distinctive when combining practices.

After identifying similar groups we investigated the frequency of responses that fell into the previously identified clusters, as shown in Table 3. The goal is to identify the most frequently used combination of agile practices.

For structuring the data (Figure 3) we divide the usage of combinations of practices in three categories, namely:

- Common: Used by $> 2/3$ of the respondents.
- Less common: Used by $[1/3;2/3]$ of the respondents.
- Seldom: Used by $<1/3$ of the respondents.

Table 3. Cluster Analysis

Project				Organization			
Cluster	1st item	2nd item	Distance	Cluster	1st item	2nd item	Distance
1	S-ups	Sp	1.414	1	SPM	Sp	2.236
2	SD	Ref	1.732	2	S-ups	Ret	3.000
3	Testing	St	2.000	3	Cluster 2	Cluster 1	3.231
4	TDD	PP	2.236	4	St	TP	3.317
5	Cluster 1	SPM	2.236	5	CI	Ref	3.317
6	CI	SR	2.449	6	Testing	Team	3.317
7	Cluster 5	Ret	2.641	7	C&CM	Comm	3.317
8	Cluster 3	C&CM	2.646	8	Cluster 6	Cluster 4	3.532
9	Cluster 6	TP	2.828	9	Cluster 8	Cluster 3	3.766
10	CS	40H	2.828	10	Cluster 5	CO	3.803
11	Team	SRM	2.828	11	Cluster 9	SR	4.023
12	Cluster 8	Cluster 2	2.911	12	Cluster 11	Cluster 7	4.040
13	Cluster 4	Of	2.914	13	Cluster 10	CS	4.320
14	Cluster 9	CO	2.940	14	Cluster 12	SRM	4.356
15	IW	PG	3.000	15	TDD	PP	4.472
16	Me	Doc	3.000	16	Cluster 14	Cluster 13	4.516
17	Cluster 14	Cluster 12	3.067	17	Cluster 15	SD	4.996
18	Cluster 11	Cluster 7	3.073	18	IW	Of	5.000
19	Cluster 17	Comm	3.328	19	Cluster 16	40H	5.137
20	Cluster 19	Cluster 10	3.381	20	Cluster 18	Cluster 17	5.334
21	Cluster 20	Cluster 18	3.481	21	Doc	PG	5.385
22	Cluster 21	Cluster 13	3.705	22	Cluster 20	Cluster 19	5.493
23	Cluster 22	Cluster 15	3.837	23	Cluster 22	Cluster 21	5.560
24	Cluster 23	Cluster 16	4.144	24	Cluster 23	Me	6.476

Project: On project level common combinations of practices are represented by clusters 1 (stand-ups, Sprint/iteration), 3 (testing, stories/features), 6 (continuous integration, short releases), and 5 (stand-ups, sprint/iteration, and sprint planning meeting).

Less common combinations are represented by clusters 9 (continuous integration, short releases, tracking progress), 8 (testing, stories/features, change and configuration management), 7 (retrospective, sprint planning meeting, stand-ups, sprint/iteration), 2 (simple design, refactoring), 14 (continuous integration, short releases, tracking progress, collective ownership), 11 (team, sprint review meeting), 10 (coding standards, 40 hour week), 4 (test-driven development, pair programming), 18 (team, sprint review meeting, stand-ups, sprint/iteration, and sprint planning meeting, sprint planning meeting), 12 (testing, stories/features, change and configuration management, simple design, refactoring), and 13 (test-driven development, pair programming, and office).

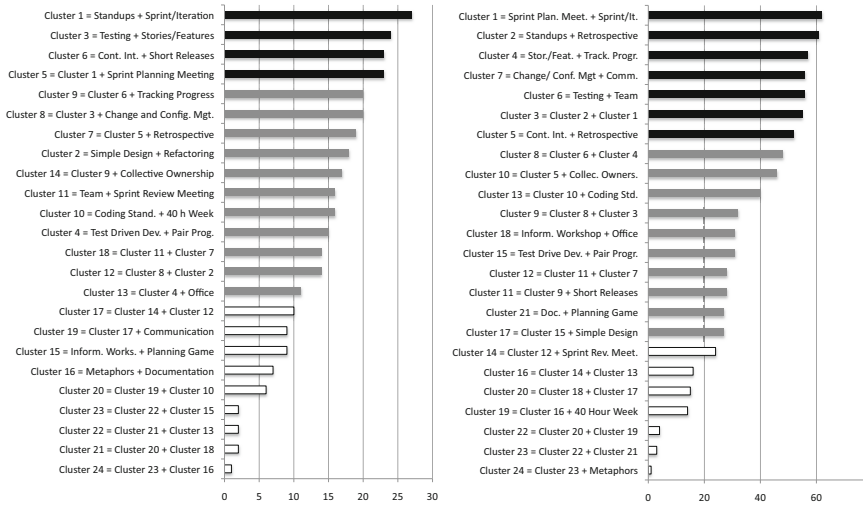
Seldom combinations of practices, which contain a larger set of practices, are represented by clusters 17, 19, 15, 16, 20, 23, 22, 21, and 24.

Organization: Common practice combinations on organizational level are clusters 1 (sprint planning meeting), 2 (stand-ups, retrospective), 4 (stories and features, tracking progress), 7 (change/configuration management, communication), 6 (testing, team), 3 (stand-ups, retrospective, sprint planning meeting), and 5 (continuous integration, retrospective).

Clusters 8 (testing, team, stories/features, tracking progress), 10 (continuous integration, retrospective, collective ownership), 13 (continuous integration, retrospective, collective ownership, coding standards), 9 (testing, team, stories/features, tracking progress, stand-ups, retrospective, sprint planning meeting), 18 (Informative workshop,office), 15 (test-driven development, pair programming), 12 (testing, team, stories/features, tracking progress, stand-ups, retrospective,

sprint planning meeting, short releases, change/configuration management, communication), 11 (testing, team, stories/features, tracking progress, stand-ups, retrospective, sprint planning meeting, short releases), 21 (documentation, planning game), 17 (test-driven development, pair programming, simple design) represent less common practice combinations.

Seldom combinations contain many practices and are represented by clusters 14, 16, 20, 19, 22, 23, 24.



(a) Project Level (33 respondents) (b) Organizational Level (76 respondents)

Fig. 3. Cluster Analysis for Agile Practice Combination - Frequencies

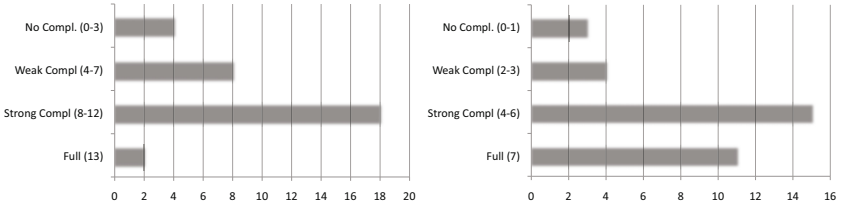
4.4 Compliance to Agile Development Processes (XP and Scrum)

The compliance is measured as the number of practices adopted belonging to the XP and Scrum methodologies, both containing a different number of practices (see Table 1). The compliance is structured as follows for XP and Scrum:

- Full compliance: All practices are fulfilled, which means 13 practices for XP and 7 practices for Scrum.
- Strong compliance: Most of the practices are fulfilled, which means 8-12 practices for XP and 4-6 practices for Scrum.
- Weak compliance: Few practices in relation to the total number of practices are fulfilled, which means for XP 4-7 practices and for Scrum 2-3 practices.
- No compliance: None or very few practices are fulfilled, meaning 0-3 for XP and 0-1 for Scrum.

Project: Figure 4 shows the compliance to XP and Scrum on project level. It is visible that Scrum has a higher compliance level than XP. For Scrum 33.33% of all respondents are fully compliant, and 45.45% are strongly compliant. Only few projects have weak (12.12%) or no (9.09%) compliance. For XP only 6.06% are fully compliant, while 54.55% are strongly compliant. There are, however, more projects that have weak (24.24%) or no (12.12%) compliance to XP.

Table 4 shows a cross-analysis of the compliance to Scrum and XP. It, for example, shows that one project using Scrum with full compliance is also fully compliant to XP, 10 projects using Scrum with full compliance are strongly compliant with XP, and so forth. The interesting observation here is that around 50 % of all projects are at least strongly compliant to both methodologies. There are few projects that use one of the methodologies and then have a weak compliance to the other (see e.g. strong compliance to XP and weak compliance to Scrum), showing clearly that neither of the methodologies is used in isolation from others.



(a) Compliance XP (Project) (b) Compliance Scrum (Project)

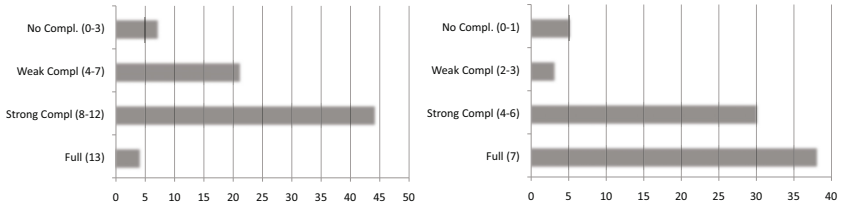
Fig. 4. Compliance to XP and Scrum (Project Level)

Table 4. Compliance on Project Level: Cross-Analysis

		XP			
		Full	Strong	Weak	No
SCRUM	Full	1 (3.03)	10 (30.30)	1 (3.03)	0
	Strong	0	7 (21.21)	7 (21.21)	1 (3.03)
	Weak	0	3 (9.09)	0	1 (3.03)
	No	0	0	0	2 (6.06)

Organization: Figure 5 shows the compliance to XP and Scrum on organizational level. A similar pattern to the one on project level can be observed. That is, organizations are more compliant to Scrum than to XP. For Scrum 50.00% are fully compliant, 39.47% are strongly compliant, 3.95% are weakly compliant, and 6.58% are not compliant. For XP, 5.25% of the organizations are fully compliant, 57.89% are strongly compliant, 27.63 are weakly compliant, and 9.21 are not compliant, showing the lower degree of compliance.

Table 5 shows that over 50% of the organizations are at least strongly compliant to Scrum and XP, which is a similar observation as for the project level. This also applies to the overall pattern that companies do not seem to concentrate solely on one of the development methods.



(a) Compliance XP (Organization) (b) Compliance Scrum (Organization)

Fig. 5. Compliance to XP and Scrum (Organizational Level)

Table 5. Compliance on Organizational Level: Cross-Analysis

		XP			
		Full	Strong	Weak	No
SCRUM	Full	3 (3.95)	27 (35.53)	7 (9.21)	1 (1.32)
	Strong	1 (1.32)	17 (22.37)	10 (13.18)	2 (2.64)
	Weak	0	0	3 (3.95)	0
	No	0	0	1 (1.32)	4 (5.26)

4.5 Success of Adoption

Table 6 shows the responses for six factors with respect to employee satisfaction. Overall, the majority of the respondents perceives the agile adoption as positive, the factor with the highest agreement was related to reduction of stress and workload due to agile practice use.

Table 6. Employee Satisfaction

Employee satisfaction factors	Positive (agree/strongly agree)	Neither agree nor disagree	Disagree Strongly disagree	Do not know	Total
Adaptation of practices helps employee in reaching their goals and eventually the teams goals	61 (92%)	2 (3%)	1 (2%)	2 (3%)	66
The adapted practices increase individual and team morale.	62 (91%)	5 (8%)	0 (0%)	1 (1%)	68
The adapted combination of practices has increased/increases productivity.	58 (89%)	4 (6%)	1 (2%)	2 (3%)	65
Stress and workload on the employee decreases with this adapted set of practices.	45 (68%)	9 (14%)	9 (14%)	3 (4%)	66
The adapted combination of practices increased/increases internal work motivation.	55 (82%)	9 (13%)	2 (3%)	1 (2%)	67
High attendance and increase in productivity has observed with this adapted set of practices.	47 (70%)	15 (22%)	3 (5%)	2 (3%)	67

Table 7. Customer Satisfaction

Customer satisfaction factors	Positive (agree/strongly agree)	Neither agree nor disagree	Disagree Strongly disagree	Do not know	Total
Customers had/has an opportunity to select methodology or practices.	23 (34%)	8 (12%)	32 (48%)	4 (6%)	67
Customer had/has an opportunity to give rapid feedback	60 (90%)	4 (6%)	2 (3%)	1 (2%)	67
Customer has satisfied with the output through frequent deliveries	55 (83%)	8 (12%)	3 (5%)	0 (0%)	66
Customer has constant insight and control over the development process	45 (68%)	9 (14%)	12 (18%)	0 (0%)	66
This adapted combination of practices created positive response form customers	55 (83%)	6 (8%)	4 (7%)	1 (2%)	66

Table 7 shows the results with respect to customer satisfaction. The results show that the aspects that were most positively perceived were related to that the customer could provide rapid feedback, and is satisfied with the output of frequent deliveries, which also resulted in a high percentage of people answering positive with respect to positive responses from customers customers.

5 Conclusion

We conducted a survey of agile practice adoption, which was sent to over 600 practitioners and posted on LinkedIn, Yahoo groups and Google groups. The survey contained questions regarding demographics, agile practice adoption and adaptation, and outcomes of agile practice usage. This study focused on the agile adoption part of the survey. In the following answers to the research questions are presented.

RQ1: How commonly used are individual agile practices? With regard to usage of individual practices we identified three groups of practices based on their commonality for projects and organization (see Section 4.2. for the frequencies).

Knowledge of the commonality of practices has important implications for practice and research. From a practitioner point of view this knowledge provides pointers of which agile practices to consider for their own development organization, given that other practitioners learn and adapt their practice selection accordingly based on their experience.

RQ2: Which agile practices are used together by practitioners, and how common are the combinations? In order to answer RQ2 we conducted a cluster analysis to determine which agile practices are used together, and investigated the frequency of practice usage in each cluster (see Section 4.3.).

It was apparent that the combinations of practices belonging to clusters with low distance (i.e. they are very similar) and that are frequently used are very rational. This adds further to the validity of the survey. For example, on project level clearly stated stories and features support testing, continuous integration facilitates short releases, and sprint/iterations are strongly connected to a sprint planning meeting.

The analysis of the commonality further supports practice selection, as it supports further investigations in research and practice not just which practices to choose based on overall frequency, but how the selection of one practice might depend on one or more other practices.

RQ3: To what degree does the software industry comply to Extreme Programming and Scrum? Overall, we found that Scrum has a higher compliance than XP on project and organizational level, both levels showing very similar patterns (see Section 4.4). The result was that practices from Scrum and XP seem to be used together, i.e. both methodologies are used complementary. From a research perspective this means that it would be interesting to investigate how to integrate agile methodologies in the best possible way. It also means

that future research needs to not only focus on single methodologies, given that the majority of past research has an XP focus.

RQ4: As how successful do the practitioners perceive the adoption of agile practices in terms of customer satisfaction and employee satisfaction? Overall, we found that the impact of agile practices was perceived as positive on customer satisfaction and employee satisfaction (see Section 4.5). It is important to highlight that these results have limitations as only a sub-set of the respondents answered this question.

Future Work: In future work the reasons of why certain agile practices and the combination thereof are chosen more frequently has to be investigated in further detail. Furthermore, as the scope of the survey also contained parts on practice adoption and outcome of agile practice usage, we will investigate how each individual practice is adopted by the companies, and with what success.

References

1. Dogs, C., Klimmer, T.: An evaluation of the usage of agile core practices. Master's thesis, Blekinge Institute of Technology, Sweden (2011)
2. Dybå, T., Dingsøyr, T.: Empirical studies of agile software development: A systematic review. Information & Software Technology 50(9-10), 833–859 (2008)
3. Jalali, S., Wohlin, C.: Agile practices in global software engineering - a systematic map. In: Proceedings of the 5th IEEE International Conference on Global Software Engineering (ICGSE 2010), pp. 45–54 (2010)
4. Jones, C.: Software assessments, benchmarks, and best practices. Addison-Wesley, Boston (2000)
5. Kniberg, H., Farhang, R.: Bootstrapping scrum and xp under crisis a story from the trenches. In: Proceedings of Agile Development Conference (AGILE 2008), pp. 436–444 (2008)
6. Koch, A.S.: Agile software development: evaluating the methods for your organization. Artech House, Boston (2005)
7. Korhonen, K.: Adopting Agile Practices in Teams with No Direct Programming Responsibility – A Case Study. In: Caivano, D., Oivo, M., Baldassarre, M.T., Visaggio, G. (eds.) PROFES 2011. LNCS, vol. 6759, pp. 30–43. Springer, Heidelberg (2011)
8. Petersen, K.: Is lean agile and agile lean? a comparison between two software development paradigms. In: Dogru, A.H., Bicer, V. (eds.) Modern Software Engineering Concepts and Practices: Advanced Approaches, pp. 19–46 (2010)
9. Shashank, S.P., Darse, D.H.P.: Finding common denominators for agile practices: A systematic literature review. Master's thesis, Blekinge Institute of Technology, Sweden (2011)
10. Sochova, Z.: Agile adoption survey (2009)
11. Vriens, C.: Certifying for cmm level 2 and iso9001 with xpatsignscrum. In: Proceedings of the Agile Development Conference (ADC 2003), pp. 120–124 (2003)
12. Williams, L.: Agile software development methodologies and practices. Advances in Computers 80, 1–44 (2010)