

# Agile methods in European embedded software development organisations: a survey on the actual use and usefulness of Extreme Programming and Scrum

O. Salo and P. Abrahamsson

**Abstract:** Press releases, scientific publications and anecdotal evidence demonstrate that organisations worldwide are adopting agile software development methods at increasing speed. Little is still known about the current usefulness of agile methods in the complex environment of the embedded software development industry. Embedded devices are already commonplace in regular households. The goal of this survey is to provide first-hand knowledge of the adoption and experience of two of the most known agile methods, namely Extreme Programming and Scrum, in a number of European organisations of embedded software known to be interested and active in experimenting with agile software development methods. The survey involved 13 industrial organisations in eight European countries and 35 individual software development projects. The focus of the questionnaire was to enquire into the level of use as well as the experienced usefulness – or expected usefulness if there was no experience available – of the two agile methods and their individual practices. The results show that the embedded industry has been able to apply agile methods in its development processes. The results also indicate that the appreciation of the agile methods and their individual practices appears to increase once adopted and applied in practice.

## 1 Introduction

For over a decade now, there has been an ever increasing variety of agile methods available to organisations to challenge their traditional plan-driven way of producing software. For some time, the claims in favour of and against the agile methods were largely based on anecdotal evidence rather than scientific evaluation [1, 2]. Nowadays, however, an increasing amount of evidence is available on agile methods in the form of experiments, case studies and experience reports (e.g. [3, 4]) on various aspects of agile methods applied in different software development contexts. In addition, some survey studies have been conducted (e.g. [5–7]) to provide a broader overview on the status of agile methods in software development organisations. For example, the results of the Forrester survey [6] indicate that whereas as much as 14% of North American and European enterprises are currently using agile processes, 19% are either interested or planning to do so in the near future.

It has been suggested that the early adopters of agile methods have been small high-tech product companies [6, 7] working for ‘in-house teams developing software for quick-to-market applications’ [7, p. 18]. Originally, for example, Extreme Programming (XP) [8] was in fact targeted especially at small co-located teams developing non-critical products. Currently, however, there is evidence

available that the adoption of agile processes increases with company size [6]. Furthermore, the adoption of agile methods in large and distributed environments is being addressed more frequently [e.g. 9–11]. Meanwhile, the second wave of agile adoption has extended from small high-tech product companies to beyond the high-tech industry, for example, to the insurance and telecom businesses [6].

There is some evidence available on agile development in the context of embedded software development. It is rather rare, however, that an existing agile method has been adopted as such. Often, the reported experiences refer to the adoption of individual agile practices or certain fundamentals of agile software development to complement the existing processes of an organisation [e.g. 12, 13]. In some cases (e.g. [14]), an existing agile method is adopted in development projects of embedded software, and yet – often radically – modified to fit the complex context of embedded software development where the concurrent development of hardware appears to dominate the evolution of software. Embedded development, however, seems to share many characteristics of agile development such as the notion of iterativeness by necessity as well as a test-driven mindset [15].

The variety of agile methods, nowadays, includes a number of specific techniques (e.g. test-driven development (TDD) [16]) and practices (e.g. pair-programming) of software development. Among the first and perhaps best known agile methods are Scrum [17] and XP. Scrum is aimed at providing an agile approach for managing software projects while increasing the probability of successful development of software [18], whereas XP focuses more on the project level activities of implementing software. Both approaches,

however, embody the central principles of agile software development [19].

Despite the proliferation of agile methods and the number of organisations reporting going agile, it is not surprising that a concern addressed in the Forrester report was that ‘as awareness of agile processes grows, so does confusion about what it really means to go agile’ [6, p. 1]. Similarly, the Reifer survey revealed the definition of agile methods as a ‘devised list of variants and invariants on user perceptions’ [7, p. 18]. Consequently, this survey focuses especially on the adoption of two specific and well-documented agile methods and their individual practices, that is, XP and Scrum. For one, this specific focus was considered to assist in avoiding the misconceptions and different perceptions behind the broad umbrella term ‘agile’ [6, 7].

On the other hand, the goal of this survey study was: (1) to provide industrial insight into two of the most known agile methods, especially in the context of embedded software development in Europe, (2) to fathom the level of their adoption and (3) to understand how useful the adopted agile methods and their individual practices are experienced to be. To provide first-hand industrial insight, the survey study focuses on a selected group of European embedded software organisations known to be interested and active in exploring and adopting agile methods. The survey covered 13 organisations of various sizes from a total of eight European countries.

This paper is structured as follows: in Section 2, the research design is defined; in Section 3, the results of the study are presented; and in Section 4, the conclusions are drawn and the limitations of the study are discussed.

## 2 Design of the study

In this section the background to the research and the design of the questionnaire-based survey study are introduced.

### 2.1 Background

A European wide research initiative on agile methods, that is, Agile-ITEA project (<http://www.agile-itea.org>), was conducted between 2004 and 2006. Its focus was on the research and deployment of agile methods in embedded software development in Europe. The Agile-ITEA consortium consisted of 22 partner organisations, both industry and research oriented, from eight European countries who all had the aim of exploring the potential of agile methods in their various development settings.

The questionnaire-based survey provides an understanding on the following aspects: (1) what is the level of adoption of XP and Scrum practices among individual software projects in different embedded software organisations interested in agile software development and (2) how useful are the adopted methods perceived to be.

### 2.2 Survey study design

The research was conducted as a questionnaire-based survey research. The survey can be considered as a descriptive survey and thus provides a descriptive analysis only (i.e. frequencies and cross tabulation). According to Oppenheim [20] descriptive surveys are not intended to explain or to show causal relationships between variables, but they focus on describing what proportion of a sample has a certain opinion or how often certain events occur, that is, are associated with each other.

In selecting the respondent organisations for the survey a purposive sampling [21] was used. Nardi [21] defines this

method of sampling as involving a selection of a unique sample on purpose because of its specific trait that is important for the study. In this study, the industrial participant organisations of Agile-ITEA provided a sample of organisations that were known to be interested in agile methods, and known to have access and involvement in the ongoing research on agile software development methods in Europe. Their current experiences and knowledge on adopting agile software methods was considered interesting for the numerous organisations still currently considering the adoption of agile methods [6].

Thus, a total of 18 industrial embedded software organisations from the Agile-ITEA project formed the survey’s sample group. From the research viewpoint, this also means that the respondents are from organisations that can be assumed to already be somewhat familiar with agile software development or interested in exploring it. In addition, the respondent’s organisations are more likely to have investigated or even deployed agile practices compared with any randomly sampled software development organisation, while probably also having higher expectations about agile methods than any randomly selected organisation. The questionnaire and covering letter were placed on the website of the research organisation. An additional covering note was also e-mailed to the contact persons of the Agile-ITEA project of the sample organisations. They were instructed to forward the website link to the questionnaire to various project managers and software developers in their organisations. This was also designed to avoid bias in favour of agile methods and to provide a broader view of each organisation. It should also be noted that the questionnaire was targeted at individual project managers and software developers in the respondent organisation – of both traditional and agile software projects. For this reason, the understanding, current knowledge and attitudes of each individual respondent are still a very subjective matter.

Only one response was requested per project, multiple projects were targeted in each organisation to provide a more detailed view on the current situation. A deadline for returning the questionnaire was set at four weeks from the initial contact, and a reminder was sent to the contact persons after a few weeks, as suggested by Rea and Parker [22].

Alreck and Settle [23] identify three principal methods of data collection in a questionnaire-based survey study: personal interviewing, telephone interviewing and mail data collection. However, currently an increasingly popular way is to conduct a computer assisted web-based survey [21]. There are multiple reasons for selecting web-based data collection in this research. For one, this enabled data storage in electronic format as well as immediate access to the data from the database. Researchers were therefore able to keep daily track of the received responses. On the other hand, this was also considered to eliminate the human errors resulting of manually entering the data from questionnaires in a database, as suggested by Nardi [21]. The web-based data collection also overcomes some limitations of ordinary mail surveys and other data collection mechanisms in terms of speed and cost. In addition, a low-response rate – very rarely over 30% – is often considered a problem especially for direct mail data collection [23]. On the other hand, web-based surveys indicate an increase in the response rates [21].

## 3 Results

In the following sub-sections, the results of the survey are analysed from the viewpoint of the goals set for the study.

Section 3.1 shows the demographic data from the companies and projects participating in the study. This is followed by an evaluation of what agile software methods had been used in the respondent projects of the target organisations (3.2). Finally, in section 3.3, the perceived usefulness of the adopted agile methods is reported.

### 3.1 Background of the respondent organisations and projects

In total, the questionnaire study provided data from 35 projects from 13 out of the 18 target software organisations from the eight European countries involved in the project. The study involved: three organisations from Belgium, one from Bulgaria, three from Finland, one from France, one from Italy, one from the Netherlands, one from Slovenia and two from Spain.

In the questionnaire, background data was requested on both the respondent organisation and the project. The respondents included projects (11) from small organisations (less than 50 employees), 11 projects from medium sized organisations (between 50 and 250 employees) as well as 13 projects from large (over 250 employees) organisations. The number of respondent projects per organisation varied from one to seven among the participant organisations.

Fig. 1 shows the projects participating in the survey in terms of the ‘agility’ framework proposed by Boehm and Turner [24]. The alignment of the respondents (i.e. each project) within each dimension is defined by bullet points. Each bullet represents a number of projects having a particular characteristic (shown by size of the bullet as well as the number within each bullet). Because of some missing data, two projects are left out of the figure. The criticality, size and dynamism of the project follow the scaling of Boehm and Turner [24]. For practical reasons, the scales of culture and personnel experience have been altered. Originally, Boehm and Turner suggest the scaling of culture varies from 10% to 90% from order to chaos (from ‘culture where people feel comfortable and empowered by having their roles defined by clear policies and procedures’

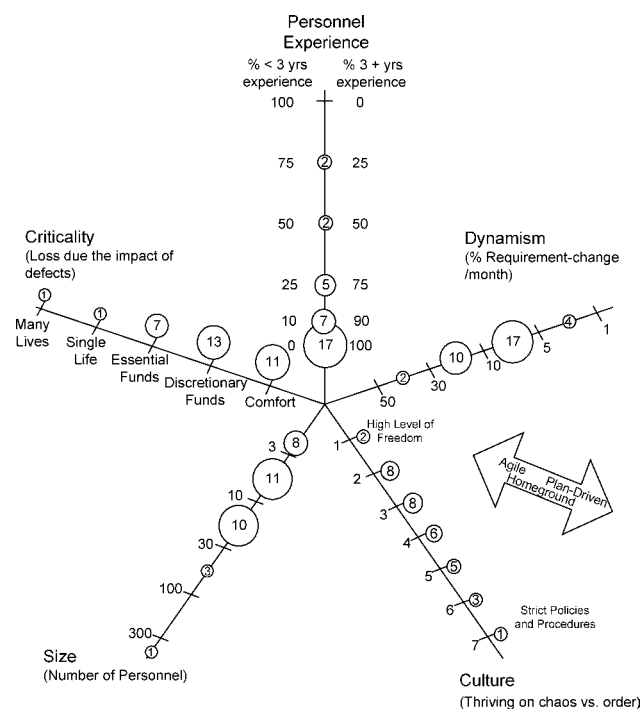


Fig. 1 Agile homeground of the respondent projects

to ‘culture where people feel comfortable and empowered by having many degrees of freedom’ [24, p. 55]). In the questionnaire, however, the degrees of freedom were defined in a seven-point semantic differential scale [21] where the opposing values shifted from a high level of freedom to strict policies and procedures in a project. In addition, it was considered difficult for software developers and project managers to estimate the level of experience of the personnel in project teams, using the levels of Cockburn [25] that Boehm and Turner [24] have modified. Thus, the experience level was defined in the questionnaire in years of experience (<3 years, between 3–6 years and over 6 years). This was, then, inserted into the personnel experience scale to illustrate the division of project personnel into less experienced (<3 years) and more experienced (3+ years).

It should also be noted that both ordinal and nominal measures were used in collecting the background information. In Fig. 1, the responses of ordinal measures, where the exact value was not requested, are aligned with the degrees in the scale (i.e. size and dynamism). In addition, the responses acquired with nominal scale are aligned (Fig. 1) with the exact scale degree (i.e. culture, criticality and experience).

Overall, it can be said that the agility homeground level of the individual respondent projects varied largely depending on the dimension. None of the projects seemed to align in either of the extremes for all the dimensions. However, especially from the viewpoint of the experience of the personnel, criticality and project size, the majority of the respondent projects seem to operate in a homeground suitable for agile software development.

### 3.2 Actual use of agile methods

In the questionnaire, a number of agile software development methods and practices were raised to find out what agile software methods have actually been adopted in the respondent projects. The verbal frequency scale [23] was used to define the level of adoption of each of the listed items. The verbal frequency scale includes ‘five words that indicate how often an action has been taken’ [23, p. 118]. In the questionnaire, the scale included the following five adoption levels: (1) systematically used throughout the project, (2) mostly used throughout the project, (3) sometimes used in the project, (4) rarely used during the project and (5) never used during the project. In addition, the respondents were provided with the following answer options: (6) not applicable and (7) I do not know. This was considered to provide more accuracy and insight into the responses. In this paper, the focus of evaluating the current use of agile methods centres on two of the most familiar agile methods, namely XP and Scrum. The practices of XP, as illustrated in Fig. 2, comply largely with the first version of XP [26] which, at the time of conducting the

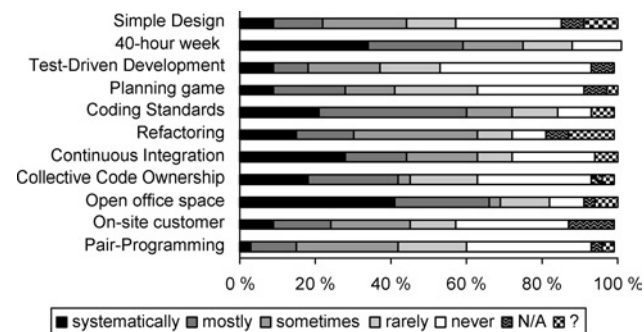


Fig. 2 Use of XP practices in the respondent projects

**Table 1: XP practices and of their usage**

XP practice	Range	Responses
	Responses	Responses
	'systematically' or 'mostly', %	'rarely' or 'never', %
1) open office space	66	22
2) coding standards	60	21
3) 40 h week (sustainable pace)	59	26
4) continuous integration	44	31
5) collective code ownership	42	48
6) refactoring	30	18
7) planning game	28	50
8) on-site customer	24	42
9) simple design	22	41
10) test-driven development	18	56
11) pair-programming	15	51

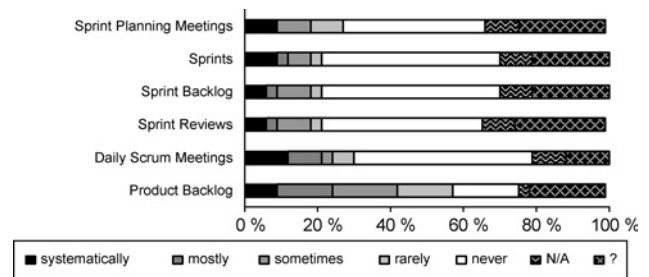
survey, was considered more familiar for the respondents than its updated version [27]. In Fig. 2, 10 of the 12 XP practices, as defined in [26], are addressed excluding the practices of small releases and metaphor. In addition, the questionnaire focused on the test-driven aspect of testing in XP.

Fig. 2 illustrates the reported usage of XP practices in the respondent projects; the darker the colour the more systematically the practice is reported to have been applied. The responses of non-usage (i.e. 'never') are illustrated in white. In addition, the responses of non-applicable (N/A) and 'don't know' (?) are illustrated with different patterns of black and white.

As can be seen in Fig. 2, a large proportion of the respondents are using XP practices at least to a certain extent. In the Table 1 the XP practices are organised in the order of their usage – from the most to the least used. The first 'Range' of each practice illustrates the total percentage of the responses in the categories of 'systematically' and 'mostly' used whereas the second range illustrates the other extreme, that is, 'rarely' and 'never' used.

Thus, according to the data in this survey (Table 1), the five most used XP practices among the respondents, reported either as systematically or mostly used during the project are: (1) open office space, (2) coding standards, (3) 40 h week, (4) continuous integration and (5) collective code ownership. However, on the basis of the responses it cannot be estimated whether the respondents have applied each practice as part of XP. In fact, it could be argued that especially the three most used practices can be easily applied independently of the XP method or agile development. On the other hand, the least used XP practices, such as simple design, TDD, planning game (iterative planning) and on-site customer, are likely to require more fundamental and extensive changes in the development approach, technologies and mindset.

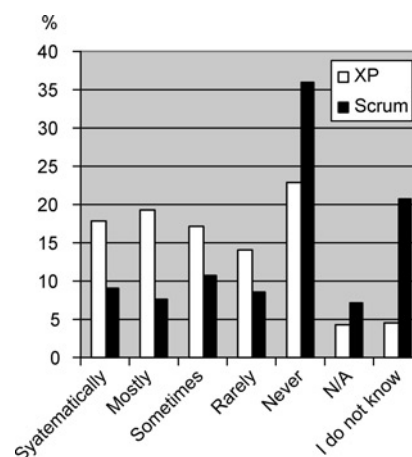
In addition, whereas TDD and pair programming were reported as the least 'systematically' or 'mostly' used practices of XP, they were also the top findings in the 'rarely' or 'never applied' categories (TDD 56% and pair-programming 51%). Collective code ownership, however, seems to be the most controversial of the practices,

**Fig. 3** Use of Scrum practices in the respondent projects

gaining a very high percentage of 'sometimes' responses (33%). In addition, the most non-applicable XP practice among the respondents was the on-site customer (12%). The three XP practices that did not gain any 'non-applicable' responses were continuous integration, coding standards and the 40 h week. The two XP practices that gained 'I do not know' responses were refactoring and simple design. However, on the basis of the survey, it cannot be evaluated whether this was related to the unfamiliarity of the practices, lack of experience with them, or both.

Fig. 3 shows the reported usage of Scrum practices, as defined in [28]. It should be noted that in Scrum, the Sprint Backlog is defined as a specific part of Sprint Planning activity rather than an individual practice. The Product Backlog seems to be a most favoured Scrum practice. Actually, 24% of the respondents claim to use it systematically or mostly throughout the project. However, all the Scrum practices seem to receive a high percentage of 'never' and 'I do not know' responses. Only a small proportion of the respondents (ranging from 9% to 21%) claimed to apply the other Scrum practices either systematically or mostly in their project. This may indicate unfamiliarity with Scrum among a large proportion of the respondent projects at the time of the survey. Furthermore, in the questionnaire the concept of Product Backlog was not strictly associated with Scrum and, thus, it could have been considered as a more universal concept for managing requirements in projects.

In Fig. 4, the data on the adoption of XP and Scrum in the respondent projects is illustrated. The percentage of the responses in each point of the scale is presented as an average on all the responses of all the practices of the method. In this way, the data of the two methods are somewhat comparable. Fig. 4 demonstrates a clear trend of higher adoption of XP practices in the respondent projects.

**Fig. 4** Adoption of XP and Scrum methods (%)

Whereas the responses concerning the use (systematically, mostly, sometimes and rarely) are consistently higher in XP, the responses indicating the ‘non-use’ or ‘non-applicability’ (never and N/A) as well as the responses referring to the unfamiliarity of the practices (I do not know) are consistently higher in Scrum. On average, 54% of the responses referred to systematically, mostly or sometimes applying XP. The corresponding figure in Scrum is 27%.

### 3.3 Experienced and expected usefulness of agile methods and practices

In this section, the results of two kinds of cross-tabulation are presented and discussed. First, the questionnaire responses where at least some level of use of a practice of XP or Scrum (systematic, mostly, sometimes or rarely) is reported are mapped with the reported level of usefulness. The related research question is: ‘how useful have the adopted methods been regarded?’ Secondly, the responses where there has been no usage of certain agile practice (i.e. none, not applicable or I do not know) are mapped with the estimated level of usefulness (in the future projects). The related research question of this mapping is: ‘what is the potential usefulness of the agile practices?’

In the questionnaire the level of current use of an agile method was requested using the scale of: (1) systematically used throughout the project, (2) mostly used throughout the project, (3) sometimes used in the project, (4) rarely used during the project and (5) never used during the project. In addition, the respondents were provided an opportunity to select the following answers: (6) not applicable and (7) I do not know. Instead, the intensity of the usefulness (estimated or experience based) was requested using the scale of: (1) extremely useful, (2) very useful, (3) useful, (4) not useful, (5) harmful, (6) not applicable and (7) I do not know.

In Fig. 5, the perceived usefulness of XP and Scrum methods are illustrated.

To illustrate the experienced usefulness of the two agile methods, Fig. 5 includes only the responses which have reported the adoption of the agile method at least to some extent (systematically, mostly, sometimes or rarely). Thus, the responses of ‘never’ ‘not applicable’ or ‘I do not know’ are excluded from the analysis. In addition, the responses illustrated in Fig. 5 are counted from the responses of all the practices of an individual method (XP or Scrum).

While examining the XP data, first it can be seen that the proportion of positive experiences is remarkably high. Nearly 90% of the responses where practices of XP had been applied (at least to some extent) could be considered positive (extremely useful, very useful or useful) whereas

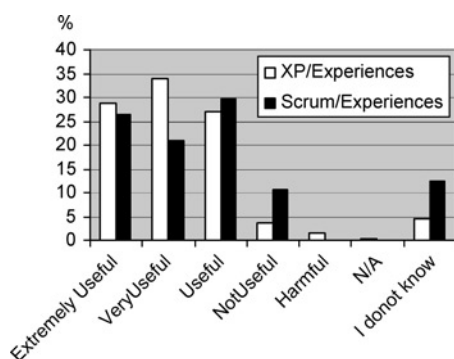


Fig. 5 Experienced usefulness of XP and Scrum

only 5.8% could be characterised as strictly negative ones (not useful or harmful). In the data of Scrum, the corresponding figures were 77% of strictly positive responses, whereas nearly 11% of the responses referred to negative experiences of Scrum (not useful). Thus, the comparison of the responses reveals the somewhat more positive experiences with XP than Scrum (90% against 77%). However, it should be noted that the experienced usefulness of Scrum gained a very high percentage of ‘I do not know’ responses compared with XP (4.5% against 12.3%).

When viewing the data of the individual practices of XP, the most appreciated XP practices among the respondents that had actually applied the practices (at least to some extent) were: (1) collective code ownership, (2) 40 h week, (3) coding standards and (4) simple design. No negative experiences were reported on any of these four XP practices, and collective code ownership and 40 h week gained only positive responses (i.e. extremely useful, very useful or useful). On the other hand, the least appreciated of the applied XP practices were: (1) pair-programming (20% negative responses), (2) TDD (12% negative responses) and (3) on-site customer (11% negative responses). It, however, should be noted that XP does not claim to provide new software development practices but, rather, provides a novel mixture of the existing ones [26]. Thus, based on the responses, it cannot be estimated whether the respondents have applied each practice as part of XP. For example, the most used practices of XP are the 40 h week and coding standards, both of which can be applied in any process model of software development whether agile or traditional.

Among the respondents who had not applied the individual practices of XP, a total of 28% of the responses revealed negative expectations towards the overall variety of XP practices. Still, the positive expectations were considerably high (57%) among these respondents.

Compared with XP, the practices of Scrum might be considered more method specific than independent considering, for example, starting from the Scrum-specific naming of the practices (besides the product backlog). Thus, unlike in the case of XP practices, the respondents’ claims of using the Scrum practices could be assumed to be the use of them as a part of the Scrum method. The data in Fig. 3 indicates that roughly two-thirds of the respondents had not been applying the individual Scrum practices in their projects. An exception to this is the Product Backlog, which had been applied by 74% of the respondents.

The most appreciated Scrum practice seemed to be the Product Backlog. This practice did not give negative results and only a few ‘I do not know’ responses among the respondents who had applied it, at least to some extent. In addition, the Product Backlog practice seemed to also receive most positive expectations among the respondents who had not applied it in practice. Overall, the responses concerning all the Scrum practices seem to be very favourable among the respondents who had actually applied Scrum at least to some extent. The least positive of the Scrum practices is the Sprint Planning Meeting which, nevertheless, gained a total of 71% of positive responses of its experienced usefulness.

The expected usefulness of Scrum practices (i.e. the responses of ‘none’, ‘not applicable’ or ‘I do not know’), despite the Product Backlog, seem to be roughly divided in half positive and negative expectations. However, it should be pointed out that none of the respondents considered any of the Scrum practices as harmful but rather as ‘not useful’ or ‘not applicable’ in their own operational environment.

Generally speaking, however, all the Scrum practices seemed to gain a much larger proportion of 'I do not know' responses in the survey than the practices of XP, regarding both the experienced as well as expected usefulness of the methods. For one, this could imply the high unfamiliarity with the individual practices of Scrum at the time of the survey. On the other hand, the XP practices may have more visibility in the daily work of the respondents. However, more data would be needed to make any broad generalisation or conclusion.

In addition, in both methods of study, there seems to be an improvement in attitude when comparing the expectations with the experiences. While examining the XP data, it can be seen that nearly 90% of the responses where the practices of XP had been applied (at least to some extent) could be considered positive whereas only 5.8% could be characterised as negative ones. Instead, among the respondents not having applied the individual practices of XP in practice, a total of 28% of the responses revealed negative expectations towards XP. Still, the number of positive expectations was considerably high (57%). The data of Scrum revealed the same trend; the ones having actually applied the practices reported more positive experiences (77%), than were the positive expectations of the group that had no experience of the practices (28%).

#### 4 Conclusions and limitations of the study

The goal of this study is to provide insights, experiences and knowledge on XP and Scrum methods of agile software development from European embedded software organisations known to be interested and active in adopting and exploring agile methods. This questionnaire-based survey research provides results from 35 projects from 13 software organisations of different sizes from a total of eight European countries. The focus of the questionnaire was to examine the level of adoption as well as the experienced usefulness – or expected usefulness if there was no experience available – of individual practices of XP and Scrum.

The principal results concerning XP are as follows: 54% of the responses referred to systematically, mostly or sometimes applying the practices of XP. The most used XP practices among the respondents were: (1) open office space (66% of the responses), (2) coding standards (61%), (3) 40 h week (59%), (4) continuous integration (44%) and (5) collective ownership (42%). The XP practices that gained most 'never applied' responses were: (1) TDD (41%), (2) pair-programming (33%), (3) collective code ownership and on-site customer (both 30%) as well as and (4) simple design and planning game (both 28%). The most non-applicable XP practice among the respondents was the on-site customer (12%). In addition, nearly 90% of the responses where practices of XP had been applied (at least to some extent) could be considered positive (extremely useful, very useful or useful), whereas only 5.8% could be characterised as strictly negative ones (not useful or harmful).

The principal results concerning Scrum are as follows: 27% of the responses referred to systematically, mostly or sometimes applying the practices of Scrum. The Product Backlog was reported to be the most favoured Scrum practice. Altogether, 24% of the respondents claim to use it systematically or mostly throughout their project. However, all the Scrum practices seem to receive a high percentage of 'never' and 'don't know' responses. Only a small proportion of the respondents (ranging from 9% to 21% depending on the practice) claimed to apply the

other Scrum practices either systematically or mostly in their project. In addition, 77% of responses where practices of Scrum had been applied (at least to some extent) could be considered positive (extremely useful, very useful or useful), whereas nearly 11% of the responses referred to negative experiences of Scrum (not useful).

Furthermore, the results suggest an improvement in attitude when comparing the respondents of no experience with XP and Scrum practices. Nearly 90% of the responses where practices of XP had been applied (at least to some extent) could be considered positive whereas only 5.8% could be characterised as negative ones. Instead, among the respondents not having applied the individual practices of XP in practice, a total of 28% of the responses revealed negative expectations. However, the number of positive expectations was still considerably high (57%). The data of Scrum revealed the same trend; the ones having actually applied its practices reported more positive experiences (77%), than the positive expectations of the group that had no experience of Scrum (28%). Thus, the experienced usefulness of the practices was clearly higher than the expected usefulness among the respondents not having applied the practices of XP and Scrum in their projects.

Because of the purposive sampling of the respondent organisations (embedded software development organisations known to be interested in adopting agile methods) there is a limited amount of responses (35 projects from 13 organisations) available. Thus, no broad generalisations can be made based on this evidence alone. It should be noted that the purposive sampling used in this survey also means that the responses may well be more favourable (regarding, for example, the expected usefulness) for the agile methods as results gained from a randomly sampled group of software development organisations. However, it should also be noted that the respondents of each organisation included personnel from different projects; not only those actively participating in the Agile-ITEA research project. The possible bias caused by the above fact, however, was a necessary choice founded on the goals of this study, that is, to provide first-hand knowledge of the adoption and experience of two of the most known agile methods, namely, XP and Scrum, in a set of European organisations of embedded software known to be interested and active in experimenting with agile software development methods.

In addition, the survey results are naturally bound to be affected by the subjective opinions, attitudes and current knowledge of the individual respondents, and provide a very narrow viewpoint of each individual organisation as a whole. The different backgrounds, knowledge and experiences of the respondents with regard to traditional and agile software development are also likely to affect their understanding and interpretations of the method-specific questions on XP and Scrum in the questionnaire. The 'I do not know' alternative was provided in order to avoid erroneous responses because of lack of knowledge concerning the topic at hand. In fact, over 20% of the Scrum responses referred to this alternative, possibly indicating a fairly low level of knowledge of Scrum and its specific terms. It is highly likely that the adaptation of XP and Scrum has also required adaptation of the different practices to fit the complex environment of embedded software development. Thus, the application and adaptation of different practices of XP and Scrum, as well as the extent of use of the two methods, are also bound to differ from organisation to organisation and from project to project. This study does not reveal the respondents' experienced application and adaptation of XP and Scrum and, consequently, it is not

evaluated how this may have affected the results. Generally speaking, however, it could be argued on the basis of the results of this study that embedded software development organisations seem to be able to apply the two agile methods, namely, XP and Scrum, and their individual practices in their projects and report fairly positive results of their application.

## 5 Acknowledgments

This research work has been carried on in the Agile-ITEA and Flexi-ITEA projects funded by TEKES (National Technology Agency of Finland).

## 6 References

- 1 Lindvall, M., Basili, V.R., Boehm, B., *et al.*: 'Empirical findings in agile methods'. Proc. XP/Agile Universe 2002, Chicago, IL, USA, 2002, pp. 197–207
- 2 Abrahamsson, P., Salo, O., Ronkainen, J., *et al.*: 'Agile software development methods: review and analysis' (VTT Electronics, Espoo VTT Publications 478, 2002)
- 3 Mann, C., and Maurer, F.: 'A case study on the impact of Scrum on overtime and customer satisfaction'. Proc. Agile 2005 Conf., Marriot, Denver City Center, USA, 2005, pp. 70–79
- 4 Hulkko, H., and Abrahamsson, P.: 'A multiple case study on the impact of pair programming on product quality'. Proc. 27th Int. Conf. Software Engineering (ICSE 2005), St. Louis, Missouri, USA, 2005, pp. 495–504
- 5 Sillitti, A., Ceschi, M., Russo, B., *et al.*: 'Managing uncertainty in requirements: a survey in documentation-driven and agile companies'. Proc. Software Metrics 2005, 11th IEEE Int. Symp., 2005, pp. 17–27
- 6 Schwaber, C., and Fichera, R.: 'Corporate IT leads the second wave of agile adoption' (Forrester Research, Inc., 2005)
- 7 Reifer, D.J.: 'How good are agile methods?', *IEEE Softw.*, 2002, **19**, (4), pp. 16–18
- 8 Beck, K.: 'Embracing change with extreme programming', *IEEE Comput.*, 1999, **32**, pp. 70–77
- 9 Lindvall, M., Muthig, D., Dagnino, A., *et al.*: 'Agile software development in large organizations', *Comput.*, 2004, **37**, pp. 26–34
- 10 Karlsson, E.-A., Andersson, L.-G., and Leion, P.: 'Daily build and feature development in large distributed projects'. Proc. Conf. Software Engineering (ICSE 2000), Limerick, Ireland, 2000, pp. 649–658
- 11 Eckstein, J.: 'Agile software development in large-diving into the deep' (Dorset House Publishing, 2004)
- 12 Manhart, P., and Schneider, K.: 'Breaking the ice for agile development of embedded software: an industry experience report'. Proc. 26th Int. Conf. Software Engineering (ICSE 2004), EICC, Scotland, UK, 2004, pp. 378–386
- 13 Smigelschi, D.: 'Combining predictability with extreme programming in embedded multimedia project'. Proc. 3rd Int. Conf. Extreme Programming and Agile Processes in Software Engineering (XP 2002), Sardinia, Italy, 2002, pp. 191–192
- 14 Grenning, J.: 'Launching XP at a process-intensive company', *IEEE Softw.*, 2001, **18**, pp. 3–9
- 15 Ronkainen, J., and Abrahamsson, P.: 'Software development under stringent hardware constraints: do agile methods have a chance?'. Proc. 4th Int. Conf. Extreme Programming and Agile Processes in Software Engineering (XP 2003), Genova, Italy, 2003, pp. 73–79
- 16 Beck, K.: 'Test-driven development by example' (Addison-Wesley, 2003, 1st edn.)
- 17 Schwaber, K.: 'Scrum development process'. Proc. OOPSLA'95 Workshop on Business Object Design and Implementation, 1995, pp. 117–134
- 18 Schwaber, K.: 'Agile project management with Scrum' (Microsoft Press, 2004)
- 19 Available at: <http://www.agilemanifesto.org/principles.html>, accessed November 2006
- 20 Oppenheim, A.N.: 'Questionnaire design, interviewing and attitude measurement' (Pinter Publishers, 1996, New edn.)
- 21 Nardi, P.M.: 'Doing survey research: a guide to quantitative methods' (Pearson Education, Inc., 2003)
- 22 Rea, L.M., and Parker, R.A.: 'Designing and conducting survey research: a comprehensive guide' (Jossey-Bass, Inc., 1997, 2nd edn.)
- 23 Alreck, P.L., and Settle, R.B.: 'The survey research handbook: guidelines and strategies for conducting a survey' (IRWIN Professional Publishing, 1995, 2nd edn.)
- 24 Boehm, B., and Turner, R.: 'Balancing agility and discipline: a guide for the perplexed' (Addison-Wesley, 2004)
- 25 Cockburn, A.: 'Agile software development' (Addison-Wesley, 2002)
- 26 Beck, K.: 'Extreme programming explained: embrace change' (Addison Wesley Longman, Inc., 2000)
- 27 Beck, K., and Andres, C.: 'Extreme Programming explained: embrace change' (Addison-Wesley, 2004, 2nd edn.)
- 28 Schwaber, K., and Beedle, M.: 'Agile software development with Scrum' (Prentice-Hall, 2002)

Copyright of IET Software is the property of Institution of Engineering & Technology and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.