

15

THE VIEWS OF EXPERTS ON THE CURRENT STATE OF AGILE METHOD TAILORING

Kieran Conboy

Cairnes Graduate School
National University of Ireland
Galway, Ireland

Brian Fitzgerald

Lero Software Engineering Research Centre
University of Limerick
Limerick, Ireland

Abstract

As stated in the conference theme, the failure of information systems and information technology projects remains stubbornly high. Agile methods have recently emerged as a new and seemingly popular alternative approach to systems development. Purveyors of these methods claim they solve many of the problems that have plagued the field for over 40 years, and there is now anecdotal evidence to suggest that these benefits are being realized and that diffusion of agile methods is rapidly increasing. However, a key factor in the diffusion of any technology or method is its ability to be customized. Innovative customization, tailoring, and fragmentation of systems development methods are viewed by many as a necessary step to avoid project failure like so many projects in the past. The ability to tailor any method is considered critically important given the complex and unique nature of each and every ISD environment, and in particular, one would logically expect that a method labeled as agile should be malleable. However, it is still unclear whether agile methods are amenable to tailoring. On one hand, purveyors of these methods advocate and often recommend tailoring. On the other hand, however, tailoring of agile methods has been described as a potential minefield due to the fact that their practices are interconnected, synergistic, and socially embedded in the development effort. This study develops a better understanding of agile method tailoring in practice through semi-structured delphi

Please use the following format when citing this chapter:

Conboy, K., and Fitzgerald, B., 2007, in IFIP International Federation for Information Processing, Volume 235, Organizational Dynamics of Technology-Based Innovation: Diversifying the Research Agenda, eds. McMaster, T., Wastell, D., Ferneley, E., and DeGross, J. (Boston: Springer), pp. 217-234.

interviews with 40 ISD expert practitioners and academics. The study sought to ascertain their opinion on the tailoring of agile methods in general, and then honed in on specific critical success factors (CSFs) of tailoring, namely built-in contingency, clear rationale behind method practices, independence of method practices, and disciplined and educated tailoring of practices. The study found that these factors are largely ignored by the agile method movement except in rare instances, and concludes with a set of recommendations for agile method creators and users to ensure agile methods experience higher diffusion rates than at present.

Keywords Agile method, tailoring, systems development, software development, delphi, expert opinion

1 INTRODUCTION

For the last 40 years, information system development (ISD) projects have been troubled by time and budget overruns, and inferior, ineffective systems, resulting in many dissatisfied customers, users, and developers. Many methods, method hybrids, and method variants have been developed and implemented in the hope of overcoming these problems, and yet at the start of this century the ISD community is still seeking what Brooks (1987) calls “the silver bullet.” The late 1990s and early 2000s have seen the emergence of what are commonly referred to as agile methods. These seek to “restore credibility to the word *method*,” and eradicate the problems that have plagued ISD for so long (Fowler and Highsmith 2001, p. 29). A number of methods are included in this family, the most popular being *eXtreme Programming* (XP) (Beck 1999), the *dynamic systems development method* (DSDM) (Stapleton 1997), *Scrum* (Schwaber and Beedle 2002), *Crystal* (Cockburn 2001), *agile modeling* (Ambler 2002), *feature driven design* (Coad et al. 1999), and *lean software development* (LSD) (Poppendieck 2001). These methods represent a popular initiative that complements the critique of formalized ISD methods over the past decade or so (Baskerville et al. 1992; Fitzgerald 1994, 1996), and have been well received by practitioners and academics alike. In fact, after 40 years of research showing that developers rarely adhere to formalized methods (Fitzgerald 1997, 1998; Hardy et al. 1995; Jenkins et al. 1984; Necco et al. 1987), there is now anecdotal evidence to suggest use of agile methods has been growing rapidly since their inception.

While there is general agreement across the ISD community regarding the increasing popularity of agile methods, there is a lot of uncertainty as to whether they should be fragmented or tailored. The ability to tailor a method is considered to be very important given that it is now accepted that no single method can provide an exact fit for the needs of every ISD project given the diversity and uniqueness of ISD environments (Brinkkemper 1996; Iivari 1989). One of the main reasons underpinning the lack of adoption of traditional methods is their lack of flexibility and malleability (Brinkkemper 1996; Iivari 1989). Apart from the fact that any method should be tailorable, the very name *agile* suggests that the method should be easily adjusted to suit its environment.

Some agile method texts and papers do highlight the ease with which agile methods can be tailored. Beck and Fowler’s (2001, p. xi) text on planning XP projects authoritatively states that “no two XP projects will ever act exactly alike,” and “once you get comfortable with the basic process, you will grow it to fit your situation more precisely.”

Some studies have advocated an *à la carte* approach such as XP Lite, where an existing agile method is “defanged” and a subset method used (Stephens and Rosenberg 2003). Conversely, however, some state that the whole is better than the sum of its parts and that agile methods are only beneficial when used in their entirety and not tailored and fragmented (Beck 1999). Furthermore, Stephens and Rosenberg (2003) view the tailoring of agile methods as a potential minefield.

The objective of this paper is to review the current state of agile method tailoring. To achieve this, a set of method tailoring critical factors were drawn from the literature (section 2), and used as a lens to analyze what agile method literature exists in relation to each (section 3). The primary research approach is outlined in section 4, and describes how the researchers selected 40 ISD experts, ascertaining their opinions on the current state of agile method use regarding tailoring in general, as well as in relation to each exemplar. These findings are discussed in section 5.

2 CRITICAL SUCCESS FACTORS FOR METHOD TAILORING

The first factor of method tailoring cited in the literature is Iivari’s (1989) notion of *built-in contingency*. Numerous studies advocate contingent use of methods, based on the belief that there is no single silver bullet method that is applicable in all circumstances (e.g., Avison and Wood-Harper 1991; Benyon and Skidmore 1987; Davis 1982; Gremillion and Pyburn 1983; Iivari 1989; McMaster et al. 1998; Naumann et al. 1980; Sullivan 1985; Vidgen and Madsen 2003). These studies have proposed many features or *situation dependencies* (Kumar and Welke 1992) of an ISD project environment which should be considered when selecting a method or part of a method. Such features include developer proficiency and experience, type of IS and customer, the development culture, and programming language. However, the concept of built-in contingency requires that the method itself contains an encompassing framework, allowing it to be adjusted to fit any context as well as providing guidance on the tailoring process, as opposed to being just one method to be selected from a pool of many.

Researchers have developed a variety of method engineering frameworks (e.g., Brinkkemper 1996; Cronholm and Goldkuhl 1994; Grundy and Venable 1996; Harmesen 1997; Harmesen et al. 1994; Smolander et al. 1990; Tolvanen and Lyytinen 1993;), all of which are designed to construct or “engineer” a development method according to stakeholders’ needs, where methods are built from the ground up using “existing proven method fragments” (Fitzgerald et al. 2002, p. 150). This is an alternative approach to contingency-based use that suggests ISD projects may not always be fulfilled by a set of available methods due to the uniqueness of ISD situations (Brinkkemper 1996; Kumar and Welke 1992). Of the various method engineering frameworks in existence, almost all require that the *rationale behind these fragments be clear*, thus aiding the method creator. A further key attribute of a method amenable to engineering is that its *practices are highly independent*, allowing them to be separated and combined without fear of unknown knock-on effects (Kumar and Welke 1992; Stephens and Rosenberg 2003).

Finally, there is also evidence to suggest that while methods are tailored in practice, this is often done without adequate consideration being given to the practices being

Table 1. CSFs of Method Tailoring

Critical Factors	Literature
Contingency Built Into Method	Iivari 1989
Clearly Stated Rationale Behind Method and Practices	Brinkkemper 1996; Cronholm and Goldkuhl 1994; Grundy and Venable 1996; Harmesen 1997; Harmesen et al. 1994; Smolander et al. 1990; Tolvanen and Lyytinen 1993
Independent Practices	Brinkkemper 1996; Kumar and Welke 1992; Stephens and Rosenberg 2003
Disciplined and Educated Tailoring of Practices	Kumar and Welke 1992

dropped (Kumar and Welke 1992). Empirical research shows that method use in practice is rather limited (Fitzgerald 1996; Hidding 1996), and in fact, an empirical study by Fitzgerald (1998) found that only 6 percent of developers rigorously adhere to methods at all. This does not necessarily mean that all of these developers did not duly consider each practice or method before dropping them, but such a scenario is highly unlikely given that adherence is so low. The final exemplar, therefore, requires *disciplined and educated tailoring of practices*, whereby the ISD team test each practice or at least give due consideration to the merits, demerits, and suitability of that practice before dropping it.

3 A CRITICAL CONSIDERATION OF METHOD TAILORING IN AGILE METHODS

3.1 Built-In Contingency

There is evidence to support the existence of contingency based approaches to agile method selection and tailoring across the literature, and some researchers have proposed situational characteristics upon which such selection efforts should be based (e.g., Boehm and Turner 2004; Koch 2005). These include team size, relationship with the customer, criticality of the system, dynamism of the environment, developer competency, team culture, and existing tools and processes. In some cases, the characteristics proposed are intended to be used to choose between agile methods (e.g., Koch 2005), while others aid the selection decision between agile methods and traditional plan-driven approaches (e.g., Boehm and Turner 2003).

However, a key exemplar proposed in Table 1 was that a method should guide the tailoring process through what Iivari (1989) calls “built-in contingency.” From an analysis of the literature, it seems that agile methods have still not adequately dealt with this issue. The creators of each method still offer detailed, intricate, step-by-step instructions on how to follow their method, but although they acknowledge tailoring should be conducted, only Crystal offers recommendations as to how this should be accomplished (Cockburn 2001). This shortcoming of agile methods is similar to that of their traditional

counterparts where the need for flexibility is acknowledged but not addressed. Therefore, an analysis of the literature would suggest that problems cited by Fitzgerald et al. (2002) in relation to traditional methods still hold true for the agile method movement: tailoring is still left to the intuition of developers, it is still carried out in an *ad hoc* fashion, and little is learned about tailoring across projects.

3.2 Clarity and Rationale Behind Practices

The method tailoring literature suggests that a method's practices should be clear and rational in order to aid tailoring decisions and processes. The existing agile method literature on this issue is scarce and what does exist is inconclusive. On one hand, there are a couple of dissenting texts which describe agile methods, and XP in particular, as irrational and vague (McBreen 2003; Stephens and Rosenberg 2003). In addition, quite a few also highlight the fact that some agile method practices are nonprescriptive, and represent a high level of abstraction, lending themselves to inconsistent interpretation and implementation (Abrahamsson et al. 2002; Boehm and Turner 2004; Koch 2005). On the other hand, however, there is at least one proprietary text to accompany every agile method, with an entire series of texts dedicated to XP. In many of these, a whole chapter is dedicated to each practice. Therefore, it is hard to argue that the creators of these methods are guilty of releasing them without adequate explanation of their constitution and rationale.

3.3 Independence of Practices

One of the most distinctive features of some agile methods is that their practices are not independent, but are instead very tightly coupled (Auer and Miller 2002), interdependent (Beck 1999), and synergistic (Martin 2003). As Beck (1999, p. 121) states, "any one practice doesn't stand well on its own...and they require the other practices to keep them in balance." Boehm and Turner (2004, p. 16) cite an unnamed "agilist" who dismisses partial use of agile practices and claims that "the pieces fit together like a fine Swiss watch." Stephens and Rosenberg (2003) liken this to a "self-referential safety net" (p. 81) where even if some practices add no value, it is impossible to remove them if they are necessary in order to hold the other ones in place. Stephens and Rosenberg state that "although XP is supposedly adaptable to a wide variety of projects, its authors have got it exactly the wrong way around" (p. 82). They consider this to be analogous to a house of cards or a circle of snakes:

The tightly meshed nature of XP's practices and activities makes them like a ring of poisonous snakes, daisy-chained together. All it takes is for one of the snakes to wriggle loose, and you've got a very angry poisonous serpent heading your way (p. 82).

3.4 Disciplined and Educated Tailoring of Practices

Many efforts have been made to tailor agile methods to suit a variety of contexts such as large teams (Bowers et al. 2002; Cao et al. 2004; Crispin and House 2003; Kahkonen

2004; Lindvall et al. 2004), start-ups (Auer and Miller 2002), distributed development environments (Kircher et al. 2001; Stotts et al. 2003), greenfield sites (Rasmusson 2003), educational environments (Johnson and Caristi 2003; McDowell et al. 2003; Melnik and Mauer 2003; Wainer 2003), open source development (Kircher and Levine 2001), outsourcing arrangements (Kussmaul et al. 2004), and systems maintenance (Poole and Huisman 2001). However, there is little empirical evidence focusing specifically on the extent to which such tailoring is done in a disciplined and educated manner, and it is not known if teams evaluate all practices before deciding whether to adopt each or not.

4 RESEARCH APPROACH

This study involved a set of delphi interviews with experts in the field of ISD. The delphi method was originally devised “in order to obtain the most reliable opinion consensus of a group of experts” (Dalkey and Helmer 1963), and there are many reasons why the delphi method is desirable when conducting applied social research (see Dalkey and Helmer 1963; Linstone and Turoff 1975; Moore 1987). First, combining the judgment of a large number of people offers a better chance of getting closer to the truth. Second, it is easier to understand social phenomena by obtaining the views of the actors. Given the ambiguous interpretation and use of agile methods and the fact that they are socially oriented methods (Beck 1999; Koch 2005; Schwaber and Beedle 2002), this advantage is highly relevant in the context of this study. Finally, complex and ill-defined problems can often be addressed only by pooled intelligence, and such difficulties are prevalent not just in the field of agile methods, but also in the study of ISD and the study of agility across all disciplines.

There is no single prescribed format for conducting a Delphi study. It is “flexible in its design” (Okoli and Pawlowski 2004, p. 29) and, indeed, many hybrids and variants exist (Moore 1987). In this instance, the decision was made to invite 40 experts to take part in the study, all of whom accepted (see Table 2). Group size theory varies in its suggestions regarding the ideal number of delphi study participants. Some general rules-of-thumb indicate 5 to 10 people for a homogenous population, but 15 to 40 people for a heterogeneous population (i.e., people coming from different social and professional stratifications such as academics and practitioners, as is the case in this study) (Delbecq et al. 1975; Uhl 1983).

Verifying expertise is somewhat difficult as it can be judged by status, experience, or “a myriad of other things” (Brown 1968, p. 211). Amethodical selection of participants or allowing every willing person to take part is considered highly unscientific (Clayton 1997; Sackman 1975), and so systematic classification and selection was conducted. The skills and background of experts required for this study are listed in Table 3, along with the basis for identification and selection. As well as selecting a mix of practitioners and academics, the selection process also ensured that at least half of the participants had experience using or researching traditional, pre-agile methods, so as to enable comparison and critical reflection. It is also worth noting that the minimum criteria were lower in relation to the selection of agile-oriented practitioners and academics, as more stringent criteria requiring 7 years industry experience or a large number of agile method publications is somewhat unrealistic given that these methods have such recent origins.

Table 2. List of Delphi Participants

Practitioner	Organization
Pekka Abrahamson	VTT Finland
Ben Aveling	Alcatel Australia
Sean Baker	Iona Technologies
Paul Bohan	American Power Conversion
Dave DesAutel	Accenture
Alistair Cockburn	Cockburn & Associates
Bill Curtis	TeraQuest
Niall Donnelly	Iona Technologies
Michael Freeley	Accenture
Tom Gilb	Gilb Consulting
Sean Griffin	Qumas
Brian Hanly	eXoftware
James Harte	PA Consulting
Ger Hartnett	Intel
Michael Hennessy	EDS
Liam Kidd	Dept. of Communications
Larry Lumsden	Curam Software
John McAvoy	Motorola
John O'Flaherty	MAC
Mary Poppendieck	Cutter Consortium
Academic	University
Ivan Aaen	Aalborg University
Richard Baskerville	Georgia State University
Michael Cusumano	Sloan MIT
Rob Fichman	Boston College
Guy Fitzgerald	Brunel University
Jim Herbsleb	Carnegie Mellon University
Mike Holcombe	University of Sheffield
Linda Levine	Software Engineering Institute
Kalle Lyytinen	Case Western Reserve
Angela Martin	University of Wellington
Lars Mathiassen	Georgia State University
Peter Middleton	Queens University Belfast
Mark Paulk	Carnegie Mellon University
Jan Pries-Heje	IT University of Copenhagen
Nancy Russo	Northern Illinois University
Helen Sharp	Open University
Ian Sommerville	Lancaster University
Duane Truex	Georgia State University
Laurie Williams	North Carolina State
Robert Zmud	University of Oklahoma

Table 3. Classification of Experts and Listing of Participants

Desired Background or Skill Set	Method of Expert Identification	Minimum Selection Criteria
Practitioners who have used agile methods	Membership of relevant societies (Agile Alliance, DSDM Consortium etc) Personal contacts	> 4 years agile method experience > 2 years agile project management experience
Practitioners who have worked in ISD, and are aware of agile methods	Membership of relevant societies (ITAA, Cutter Consortium etc) Personal contacts	> 7 years ISD experience > 3 years ISD management experience
Academics who have researched agile methods	Literature review of relevant academic and practitioner journals and conferences	≥ 3 agile method publications in refereed journal/conferences
Academics who have researched ISD and are aware of agile methods	Literature review of relevant academic and practitioner journals and conferences	≥ 5 ISD publications in refereed journal/conferences

Data was collected through personal face-to-face interviews, which is considered the superior data gathering technique FOR interpretivist studies such as this (Yin 2003). Personal interviews are also well suited for exploratory research because they allow expansive discussions which illuminate additional factors of importance (Oppenheim 1992; Yin 2003). Also, the information gathered is likely to be more accurate than information collected by other methods since the interviewer can avoid inaccurate or incomplete answers by explaining the questions to the interviewee (Oppenheim 1992).

A guiding script was prepared for use throughout the interviews to establish a structure for the direction and scope of the research, to ensure the researcher covers all aspects of the study with each respondent, to manufacture some element of distance between the interviewer and interviewee, and to permit the researcher to compare and contrast responses (McCracken 1988). The researcher circulated the guiding questions in advance to allow participants to consider their responses prior to the interview. The questions were largely open-ended, allowing respondents freedom to convey their experiences and views, and expression of the socially complex contexts that underpin ISD and agile method use (Oppenheim 1992; Yin 2003).

The interviews lasted between 50 and 120 minutes, with the average length being approximately 85 minutes. The interviews were conducted in a responsive (Rubin and Rubin 2005; Wengraf 2001), or reflexive (Trauth and O'Connor 1991) manner, allowing the researcher to follow up on insights uncovered mid-interview, and adjust the content and schedule of the interview accordingly. Furthermore, the researcher kept a diary of questions asked during each interview, and analyzed their effectiveness, making refinements and additions to the set of questions prior to the next meeting. In order to aid analysis of the data after the interviews, all were recorded with each interviewee's consent, and were subsequently transcribed, proof-read, and annotated by the researcher. In any cases of ambiguity, clarification was sought from the corresponding interviewee, either via telephone or e-mail.

Coding is often used in qualitative research to provide structured and coherent analysis of qualitative data (Miles and Huberman 1999; Stake 1995; Wengraf 2001;

Rubin and Rubin 2005; Salkind 2006). This involves systematically labeling concepts, themes, and artefacts so as to be able to retrieve and examine all data units that refer to each subject across the interviews. The coding structure adopted in this research consisted of three distinct mechanisms. First, an ID code was attached to each piece of text extracted from a transcript (A1...A20 for academics and P1...P20 for practitioners) to ensure participant anonymity. Second, a classification schema was built to analyze the data from the delphi interviews as recommended by Rubin and Rubin (2005), Miles and Huberman (1999), Stake (1995), and Yin (2003). This mechanism acted as set of “intellectual bins” (Miles and Huberman 1999), which were then used to segment the data and remove any irrelevant data collected throughout the interviews. This process also considered the fact that some factors are repeated across various components of the framework, and these links were identified when coding and analyzing the data. Finally, pattern coding was used in order to “identify any emergent themes, configurations or explanations” (Miles and Huberman 1999, p. 41). This approach aims to aggregate and summarize the previous codes, identifying themes, and inferences across them all.

5 FINDINGS

5.1 General Perspectives on Agile Method Tailoring

Most respondents were of the opinion that agile methods are not designed with tailoring in mind. Some pointed to the religious undertones which often emanate from agile method creators and early adopters. A number recalled hearing methods such as XP and Scrum described as *bibles* (A1, A8, A9, P4, P7, P11, P12) and *doctrines* (A3, A10) and other religious connotations, indicating that these methods should never be tailored or even questioned. One practitioner recalled a conversation he had with what he called an *agile apostle*, where the practices of XP were continually referred to as *the 12 commandments* (P4). Three participants did note that such extremity was now rarer, and that even those driving the agile movement don't feel as *emotional* (P11, P17) about it anymore. However, some interviewees still believed that such compromising attitudes are often superficial, and that those behind the agile movement are as fanatical as ever:

We had one agile consultant on our project for a while, who said he wasn't an agile evangelist and believed in nothing more than a “try first” approach where XP practices are dropped if they are not working. After 6 months of working with him, I'd say his philosophy was “try first, and if it doesn't work then try harder.” (P7)

I keep hearing that that the agile method extremists are easing off a little these days. But I met one of them at a conference a month ago, and ever since he made reference to me and my team burning in the depths of hell if we didn't convert to agile, I'm not so sure. (P12)

The dogmatic sentiments emanating from the creators of agile methods is typified by what some respondents identified as a fervent belief that the ISD environment should

be tailored to the method rather than the method to the environment. While some respondents felt that *lubricating* (P3) the organization to the method is sometimes appropriate, others suggested that agile method *radicals* (A7) often take this to a ludicrous and unrealistic extreme:

I can see how organizational structures can be tinkered to get the best out of a method. But for God's sake, we worked on a consulting engagement for public sector bodies across the E.U. We were one project of about 30 that these bodies were dealing with, yet this guy came on board, and told us we should get them to change everything just for us. When we told him we were distributed across a continent, his only solution was "Get everybody in the same room." When we said a 40 hour week was not possible because the on-site customers we were given worked split shifts in their own organization, he said "Change it so they work 8 hours." When we said we couldn't do 2 week sprints because the government bodies worked on an 8 week testing schedule, he said "Get them to change it to 2 weeks!" (P19)

Some research suggests that method purveyors should show the limitations of their method, and where it fits in with those already in existence (Benyon and Skidmore 1987; Iivari 1989; Naumann et al. 1980; Sullivan 1985). Most respondents indicated that purveyors of contemporary agile methods do not usually envisage their methods to be part of a larger ensemble, and pay little heed to alternatives. In fact, two respondents felt that on the rare occasions when a method's proprietary literature does mention alternative approaches, it is usually done to highlight their inadequacies and to accentuate the strengths of the new method being promoted.

Given that tailoring to context is essential to maximizing value and therefore agility, it is clear from the responses that the purveyors of agile methods need to encourage tailoring of their methods. It can be argued that some proprietary texts do concede that the method has limitations and is not suitable in all circumstances (e.g., Beck 1999; Cockburn 2001). If so, then the findings in this study suggest that this message needs to be conveyed to the ISD community in a clearer fashion than at present.

5.2 Built-In Contingency

Most contended that while some proprietary texts accompanying agile methods may state their limitations and where they are and are not appropriate, contingencies are rarely built in to guide the developers on how the method should be tailored to suit a particular project context. To illustrate this, one practitioner described his longing for a simple document accompanying an agile method, telling him "*if a circumstance exists do this step, if a different circumstance exists do this step, and so on*" (P5). All 20 practitioners described some form of tailoring they had conducted to adapt agile methods to a large organization or team, to a distributed environment, or to a critical system or public sector project. In 19 of those cases, the tailoring effort was done in an *ad hoc* manner or based on *intuition* (P6, P20), and was not in any way directed by any guideline inherent in the method being used. One practitioner described his experiences of tailoring without guidance:

We changed a lot of things about XP. It took a long time to perfect given we were flying in the dark, on a trial and error basis, but we got there. And I think we are more agile. I just wish the option to use these alternatives could have been part of the method. It would have saved a lot of time, effort, and uncertainty. (P1)

The absence of built-in contingency is not specific to agile methods, as this has been a problem associated with traditional methods as far back as the late 1980s (Iivari 1989). However, it is of more concern in an agile method context, given that agile theory demands that a method be as amenable to tailoring as possible.

5.3 Clarity and Rationale Behind Practices

Respondents noted two problems with current agile methods in relation to this exemplar which hinder the ability to tailor. First, two academics and one practitioner maintained that the level of abstraction across agile method practices is quite varied even within a single method, making it difficult to make a structured tailoring decision. One used XP as an example, describing the *pair programming* practice as “*prescriptive, operational, and detailed*,” while the *simple design* and *metaphor* practices are more abstract and open to wider interpretation (A3).

Second, quite a few felt that the rationale behind some agile method fragments is not always that clear, and as one practitioner stated, “*unless you understand the rationale, you can’t make an informed decision about extending that step, tailoring it, [or] dropping it*” (P10). Of these respondents, six explicitly mentioned the *system metaphor* practice in XP as an example. However, it must be said that numerous practitioners and academics strongly disagreed, and felt that the rationale behind agile method fragments is usually explained clearly.

As stated earlier, the emergence of most contemporary agile methods has been accompanied by proprietary texts which clearly describe the purpose and rationale of the method’s practices (e.g., Beck 1999; Cockburn 2001; Schwaber and Beedle 2002). Therefore, it is possible that the perceived lack of clarity regarding these practices is a function of something else, perhaps because on occasion these methods are being communicated second- and third-hand without the aid of proprietary documentation, a possibility raised by one respondent.

5.4 Independence of Practices

All of the respondents agreed that tailoring and fragmentation of today’s agile methods is made increasingly difficult by the fact that many of their practices are highly interdependent and tightly coupled, confirming what some literature has already suggested (Auer and Miller 2002; Beck 1999; Martin 2003; Stephens and Rosenberg 2003). Various participants described these practices as being *interrelated* (A5, P7), *interconnected* (P9), *fused* (P11), *meshed* (P8), *knitted* (A6), *tightly coupled* (P4), *tethered* (A13), *tied-in* (A19), and *synergistic* (A18), which together form a set of *checks and balances* (P6) or *cogs and pulleys* (P19). Even though one academic considered this to

be an issue not specific to agile methods, but one that has “*plagued development since time immemorial*” (A7), he still conceded that it is certainly an issue exacerbated in contemporary agile methods.

It is also clear that such tight coupling is causing significant problems in practice. Many of the practitioners wanted to remove certain non-value-adding practices on past projects, but were reticent due to the embedded nature of these practices and uncertainty about knock-on effects. This reiterates Stephens and Rosenberg’s (2003, p. 8) condemnation of what they called a “self-referential safety net,” whereby no fragment can be removed, regardless of its limitations, due of the other fragments which are dependent on it. The fact that this hinders tailoring should be of some concern, and paves the way for the development of more modular and segmented agile methods. It also highlights an urgent need to learn more about what practices are interconnected and interdependent, as all 20 academics felt that, like so many other aspects of agile methods, very little is known as yet about such relationships and the knock-on effects the removal of practice will have.

5.5 Disciplined and Educated Tailoring of Practices

Despite the fact that agile methods are not conducive to tailoring, quite a few respondents ironically noted that current tailoring efforts in practice go far beyond *tweaking* (P5) and *fine tuning* (P1), and often see developers “*hacking off practices like body parts*” (P17). According to the respondents, this has resulted in a *sporadic* (P3) and *patchy* (P15) use of agile methods, where in many cases only a very small minority of practices are actually implemented. Academics spoke of their observations, with one concluding that “*its very rare to even see more than four or five [of the 12 XP practices] used*” (A9). Another spoke of 5 years of research looking at over 20 agile method projects, of which none employed “anything close” to the full range of any method’s practices (A19).

This observation was supported by many of the practitioners’ own experiences on their current or most recent project. Of 16 practitioners currently or recently working on an XP project, Figure 1 shows that in reality only 25 percent of those projects were adopting more than half of the practices.

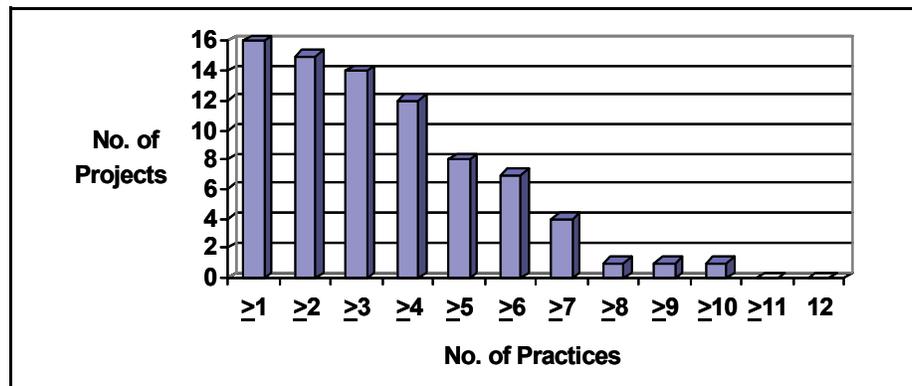


Figure 1. Sporadic Adoption of XP Practices

The fact that few practices are adopted is not that revelatory, as research has shown this to be true of method use in general (Fitzgerald 1997; Hardy et al. 1995; Jenkins et al. 1984; Necco et al. 1987; Powell et al. 1998). What is of greater concern is that many of the respondents believed that developers often do not have knowledge or experience of the practices they remove, and that there are many fragments “they know absolutely nothing about, but still love to drop” (A12). Of the 16 practitioners using XP, 15 conceded that many of the practices were dropped without due consideration, with only one stating that on his project, the pros and cons of each practice were debated before deciding whether to adopt or not (P20).

Although all participants agreed that method tailoring contributes to agility as a general rule, most did not believe this rule applied to the “*blind hacking of practices*” (A10), or when tailoring occurs just because the developers “*didn’t feel like doing all of the practices*” (A17). As one of the interviewees contended,

Picasso has got to be the most agile guy there ever was- he mastered the traditional form and then he played with it. (A5)

The need for diligent tailoring was considered to be even more imperative by one academic, given the uncharted nature of today’s agile methods (A11). He argued that some of the flawed parts of older methods have been exposed through years of application, but that it was inexcusable for an ISD team to discard parts of agile methods when so little is still known about their use.

Linking this back to the discussion of agility theory, while customization and streamlining were considered important to maximize value and therefore agility, the general conclusion was that such efforts need to be done diligently, with due consideration given to each part of an entity before it is removed. It is clear from these responses that, in agile method practice at least, due diligence is often absent and ISD teams often take a minority of practices from a method without testing the other method fragments, or at least learning enough about them to justify their removal.

6 SUMMARY AND RECOMMENDATIONS

Rather than viewing method agility as being flexible and amenable to tailoring, there was a perception among respondents that agile methods are often incorrectly viewed as infallible and as possessing properties usually associated with a religion. The purveyors of these methods along with pro-agile consultants were seen by many as the primary drivers of these perceptions, although a few did concede such emotions were becoming less prevalent in the last couple of years.

The study also found that agile methods are not amenable to tailoring despite the fact that the ability to do so underpins the very meaning of what it is to be agile. The literature purported that contingencies should be built in to a method to guide the tailoring process, and these interviews found such contingencies to be distinctly absent. The findings also verified that the tightly coupled nature of agile method practices hinders the tailoring process, with numerous practitioners steering away from tailoring altogether for fear of knock-on effects. Many of the respondents also complained that despite the exist-

Table 4. Summary of Findings

CSF	Agile Method Deficiency
Built-in contingency	No built-in contingency apart from the Crystal family of methods.
Clearly stated rationale	Lack of clarity and rationale in some instances. Possibly caused by agile methods being passed on second and third hand.
Independence of practices	Low independence among agile method practice.
Disciplined and educated tailoring of practices	Indiscriminate culling of agile method practices in many cases.

tence of at least one proprietary text for each method, agile method practices sometimes lack clarity and the underpinning rationale is often unclear. It was felt that the vast majority of agile method tailoring efforts are haphazard and indiscriminate and, while it is now almost the norm to “hack off” huge parts of these methods, teams often fail to approach the tailoring process in a structured and rational manner, giving due consideration to practices before culling them.

The implication of this study is that those in practice need to develop agile methods which have built-in contingencies, independent practices, and a set of clearly stated rationale. While the research community can also play a vital role in the development, testing, and understanding of better agile methods, given that the agile method movement to date has been primarily industry-driven, it is logical to communicate these deficiencies primarily to this audience.

References

- Abrahamsson, P., Salo, O., Ronkainen, J., and Warsta, J. *Agile Software Development Methods: Review and Analysis*, Espoo, Finland: Technical Research Centre of Finland, Vtt Publications, 2002.
- Ambler, S. W. *Agile Modeling: Best Practices for the Unified Process and Extreme Programming*, New York: John Wiley & Sons, 2002.
- Auer, K., and Miller, R. *Extreme Programming Applied: Playing to Win*, Reading, MA: Addison-Wesley, 2002.
- Avison, D., and Wood-Harper, A. T. *Multiview: An Exploration in Information Systems Development*, Oxford, UK: Blackwell Scientific Publications, 1991.
- Baskerville, R., Travis, J., and Truex, D. “Systems Without Method: The Impact of New Technologies on Information Systems Development Projects,” in K. Kendall, K. Lyytinen, and J. I. DeGross (eds.), *The Impact of Computer Supported Technologies on Information Systems Development*, Amsterdam: Elsevier Science, 1992, pp. 241-269.
- Beck, K. *Extreme Programming Explained*, Reading, MA: Addison-Wesley, 1999.
- Beck, K., and Fowler, M. *Planning Extreme Programming*, Boston, MA: Addison-Wesley, 2001.
- Benyon, D., and Skidmore, S. “Towards a Toolkit for the Systems Analyst,” *The Computer Journal* (30:1), 1987, pp. 2-7.
- Boehm, B., and Turner, R. *Balancing Agility and Discipline: A Guide for the Perplexed*, Boston: Addison-Wesley, 2004.

- Boehm, B., and Turner, R. "Using Risk to Balance Agile and Plan-Driven Methods," *IEEE Software* (36:6), 2003, pp. 57-66.
- Bowers, J., May, J., Melander, E., Baarman, M., and Ayoob, A. "Tailoring XP for Large Mission Critical Software Development," in D. Wells and L. Williams (eds.), *XP/Agile Universe*, Chicago, IL, August 4-7, 2002, pp. 100-111.
- Brinkkemper, S. "Method Engineering: Engineering of Information Systems Development Methods and Tools," *Information and Software Technology* (38:4), 1996, pp. 275-280.
- Brooks, F. "No Silver Bullet: Essence and Accidents of Software Engineering," *IEEE Computer* (20:4), 1987, pp. 10-19.
- Brown, B. *Delphi Process: A Methodology Used for the Elicitation of Opinions of Experts*, Santa Monica, CA: Rand Corporation, 1968.
- Cao, L., Mohan, K., Xu, P., and Ramesh, B. "How Extreme Does Extreme Programming Have to Be? Adapting XP Practices to Large-Scale Projects," in R. Sprague (ed.), *Proceedings of the 37th Hawaii International Conference on System Sciences*, Los Alamitos, CA: IEEE Computer Society Press, 2004.
- Clayton, M. "Delphi: A Technique to Harness Expert Opinion for Critical Decision-Making Tasks in Education," *Educational Psychology* (17:4), 1997, pp. 373-387.
- Coad, P., De Luca, J., and Lefebvre, E. *Java Modeling in Color*, Englewood Cliffs, NJ: Prentice Hall, 1999.
- Cockburn, A. *Crystal Clear: A Human-Powered Software Development Methodology for Small Teams*, Reading, MA: Addison-Wesley, 2001.
- Crispin, L., and House, T. *Testing Extreme Programming*, Boston: Pearson, 2003.
- Cronholm, S., and Goldkuhl, G. "Meanings and Motivates of Method Customization in Case Environments: Observations and Categorizations from an Empirical Study," in B. Theodoulidis (ed.), *Proceedings of the 5th Workshop on the next Generation of Case Tools*, University of Twente, Enschede, The Netherlands, 1994, pp. 67-79.
- Dalkey, N., and Helmer, O. "An Experimental Application of the Delphi Method to the Use of Experts," *Journal of the Institute of Management Science* (9:3), 1963, pp. 458-467.
- Davis, G. B. "Strategies for Information Requirements Determination," *IBM Systems Journal* (21:1), 1982, pp. 4-30.
- Delbecq, A. L., Van De Ven, A. H., and Gustafson, H. *Group Techniques for Program Planning*, Glenview IL: Scott Foresman and Company, 1975.
- Fitzgerald, B. "An Empirical Investigation into the Adoption of Systems Development Methodologies," *Information and Management* (34:6), 1998, pp. 317-328.
- Fitzgerald, B. "Formalized Systems Development Methodologies: A Critical Perspective," *Information Systems Journal* (6:1), 1996, pp. 3-23.
- Fitzgerald, B. "The Systems Development Dilemma: Whether to Adopt Formalized Systems Development Methodologies or Not?," in W. Baets (ed.), *Proceedings of the Second European Conference on Information Systems*, Nijenrode University, Utrecht, The Netherlands, 1994, pp. 691-706.
- Fitzgerald, B. "The Use of Systems Development Methodologies in Practice: A Field Study," *Information Systems Journal* (7:3), 1997, pp. 201-212.
- Fitzgerald, B., Russo, N., and Stolterman, E. *Information Systems Development: Methods in Action*, London: McGraw-Hill, 2002.
- Fowler, M., and Highsmith, J. "The Agile Manifesto," *Software Development* (9:8), 2001, pp. 28-32.
- Gremillion, L., and Pyburn, P. "Breaking the Systems Development Bottleneck," *Harvard Business Review* (61:2), 1983, pp. 130-137.
- Grundy, J., and Venable, J. "Towards an Integrated Environment for Method Engineering," in S. Brinkkemper, K. Lyytinen, and R. Welke (eds.), *Proceedings of the IFIP TC8 Working Conference on Method Engineering: Principles of Method Construction and Tool Support*, London: Chapman & Hall, 1996, pp. 45-62.

- Hardy, C., Thompson, J., and Edwards, H. "The Use, Limitations and Customization of Structured Systems Development Methods in the United Kingdom," *Information and Software Technology* (37:9), 1995, pp. 467-477.
- Harmesen, F. *Situational Method Engineering*, Ph.D. Thesis, Twente University, Enschede, Netherlands, 1997.
- Harmesen, F., Brinkkemper, S., and Oei, H. "Situational Method Engineering for I.S. Project Approaches," in A. A. Verrijn-Stuart and T. Olle (eds.), *Methods and Associated Tools for the Is Life Cycle*, Amsterdam: Elsevier Science, 1994, pp. 169-194.
- Hidding, G. "Method Engineering: Experiences in Practice," in S. Brinkkemper, K. Lyytinen, and R. Welke (eds.), *Method Engineering: Principles of Method Construction and Tool Support*, London: Chapman & Hall, 1996, pp. 47-56.
- Iivari, J. "A Methodology for I.S. Development as Organizational Change," in H. Klein and K. Kumar (eds.), *Systems Development for Human Progress*, Amsterdam: North Holland, 1989, pp. 197-217.
- Jenkins, A., Naumann, J., and Wetherbe, J. "Empirical Investigation of Systems Development Practices and Results," *Information & Management* (7:1), 1984, pp. 73-82.
- Johnson, D., and Caristi, J. "Extreme Programming and the Software Design Course," in M. Marchesi, G. Succi, D. Wells, and L. Williams (eds.), *Extreme Programming Perspectives*, Reading, MA: Addison-Wesley, 2003, pp. 47-59.
- Kahkonen, T. "Agile Methods for Large Organizations- Building Communities of Practice," in T. Little (ed.), *Proceedings of the Agile Development Conference 2004*, Los Alamitos, CA: IEEE Computer Society Press, 2004, pp. 2-11.
- Kircher, M., Jain, P., Corsaro, A., and Levine, D. "Distributed Extreme Programming," in *Proceedings of XP 2001: Extreme Programming and Flexible Processes in Software Engineering*, Villasimius, Sardinia, Italy, May 2001, pp. 66-71.
- Kircher, M., and Levine, D. "The XP of Tao: Extreme Programming of Large Open-Source Frameworks," *Extreme Programming Examined*, Reading, MA: Addison-Wesley, 2001, pp. 463-485.
- Koch, A. *Agile Software Development: Evaluating the Methods for Your Organization*, Norwood, MA: Artech House, 2005.
- Kumar, K., and Welke, R. J. "Methodology Engineering: A Proposal for Situation-Specific Methodology Construction," in W. Cotterman and J. Senn (eds.), *Challenges and Strategies for Research in Systems Development*, New York: John Wiley & Sons, 1992, pp. 257-269.
- Kussmaul, C., Jack, R., and Sponsler, B. "Outsourcing and Offshoring with Agility: A Case Study," in C. Zannier, H. Erdogmus, and L. Lindstrom (eds.), *XP/Agile Universe*, Calgary, Canada, August 15-18, 2004, pp. 147-154.
- Lindvall, M., Muthig, D., Dagnino, A., Wallin, C., Stupperich, M., Kiefer, D., May, J., and Kahkonen, T. "Agile Software Development in Large Organizations," *IEEE Computer* (37:12), 2004, pp. 27-34.
- Linstone, H., and Turoff, M. "Introduction," in H. Linstone and M. Turoff (eds.), *The Delphi Method: Techniques and Applications*, Reading, MA: Addison-Wesley, 1975, pp. 3-12.
- Martin, R. *Agile Software Development: Principles, Patterns and Practices*, Upper Saddle River, NJ: Prentice Hall, 2003.
- McBreen, P. *Questioning Extreme Programming*, Boston: Addison-Wesley, 2003.
- McCracken, G. *Qualitative Research: The Long Interview*, Beverly Hills, CA: SAGE Publications, 1998.
- McDowell, C., Werener, L., and Bullock, H. "The Impact of Pair Programming on Student Performance, Perception and Persistence," in *Proceedings of the 25th International Conference on Software Engineering*, Portland, OR, May 3-10, 2003, pp. 602-607.
- McMaster, T., Vidgen, R., and Wastell, D. "Networks of Association and Due Process in IS Development," in T. J. Larsen, L. Levine, and J. I. DeGross (eds.), *Information Systems: Current Issues and Future Changes*, Laxenburg, Austria: IFIP, 1998, pp. 25-34.

- Melnik, G., and Mauer, F. "Agile Methods in Learning Environments: Lessons Learned," in F. Maurer and D. Wells (eds.), *XP/Agile Universe 2003*, New Orleans, LA, August 10-13, 2003, pp. 172-184.
- Miles, M., and Huberman, A. *Qualitative Data Analysis*, London: SAGE Publications, 1999.
- Moore, C. *Group Techniques for Idea Building*, London: SAGE Publications, 1987.
- Naumann, J., Davis, G., and McKeen, J. "Determining Information Requirements: a Contingency Method for Selection of a Requirements Assurance Strategy," *The Journal of Systems and Software* (1:4), 1980, pp. 273-281.
- Necco, C., Gordon, C., and Tsai, N. "Systems Analysis and Design: Current Practices," *MIS Quarterly* (11:3), 1987, pp. 461-476.
- Okoli, C., and Pawlowski, S. "The Delphi Method as a Research Tool: An Example, Design Considerations and Applications," *Information & Management* (42:1), 2004, pp. 15-29.
- Oppenheim, A. *Questionnaire Design, Interviewing and Attitude Measurement*, New York: Continuum, 1992.
- Poole, C., and Huisman, J. "Using Extreme Programming in a Maintenance Environment," *IEEE Software* (18:6), 2001, pp. 42-50.
- Poppendieck, M. "Lean Programming," *Software Development Magazine* (9:5), 2001, pp. 71-75.
- Powell, T. A., Jones, D. L., and Cutts, D. C. *Web Site Engineering/Beyond Web Page Design*, Upper Saddle River, NJ: Prentice Hall, 1998.
- Rasmusson, J. "Introducing XP into Greenfield Projects: Lessons Learned," *IEEE Computer* (20:3), 2003, pp. 21-28.
- Rubin, H., and Rubin, I. *Qualitative Interviewing: The Art of Hearing Data*, Thousand Oaks, CA: SAGE Publications, 2005.
- Sackman, H. *Delphi Critique*, Lexington, MA: D. C. Heath and Company, 1975.
- Salkind, N. *Exploring Research*, Upper Saddle River, NJ: Pearson, 2006.
- Schwaber, K., and Beedle, M. *Agile Software Development with Scrum*, Upper Saddle River, NJ: Prentice Hall, 2002.
- Smolander, K., Tahvanainen, V., and Lyytinen, K. "How to Combine Tools and Methods in Practice: A Field Study," in B. Steinholtz, A. Sølvsberg, and L. Bergman (eds.), *Lecture Notes in Computer Science, Second Nordic Conference Caise '90*, Stockholm, Sweden, 1990, pp. 195-214.
- Stake, R. *The Art of Case Study Research*, Thousand Oaks, CA: SAGE Publications, 1995.
- Stapleton, J. *DSDM: Dynamic Systems Development Method*, Harlow, England: Addison-Wesley, 1997.
- Stephens, M., and Rosenberg, D. *Extreme Programming Refactored: The Case Against XP*, Berkeley, CA: Apress, 2003.
- Stotts, D., Williams, L., Nagappan, N., Baheti, P., Jen, D., and Jackson, A. "Virtual Teaming: Experiments and Experiences with Distributed Pair Programming," in F. Maurer and D. Wells (eds.), *Extreme Programming/Agile Universe 2003*, New Orleans, August 11-14, 2003, pp. 129-141.
- Sullivan, C. H. "Systems Planning in the Information Age," *Sloan Business Review* (26:2), 1985, pp. 3-11.
- Tolvanen, J., and Lyytinen, K. "Flexible Method Adaptation in Case: The Metamodeling Approach," *Scandinavian Journal of Information Systems* (5:1), 1993, pp. 51-77.
- Trauth, E., and O'Connor, B. "A Study of the Interaction Between Information, Technology and Society," in H-E. Nissen, H. K. Klein, and R. Hirschheim (eds.), *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Amsterdam: North-Holland, 1991, pp. 131-144.
- Uhl, N. *Using Research for Strategic Planning*, San Francisco: Jossey-Bass, 1983.
- Vidgen, R., and Madsen, S. "Exploring the Socio-Technical Dimension of Information System Development: Use Cases and Job Satisfaction," paper presented at the 11th European Conference on Information Systems, Naples, Italy, June 19-21, 2003.

- Wainer, M. "Adaptations for Teaching Software Development with Extreme Programming: An Experience Report," in F. Maurer and D. Wells (eds.), *XP/Agile Universe*, New Orleans, August 10-13, 2003, pp. 199-207.
- Wengraf, T. *Qualitative Research Interviewing*, London: SAGE Publications, 2001.
- Yin, R. *Case Study Research: Design and Methods*, Thousand Oaks, CA: SAGE Publications, 2003.

About the Authors

Kieran Conboy is a lecturer in Information Systems at National University of Ireland, Galway, Ireland. His current research focuses on agile methods for systems development as well as agility across other disciplines. Kieran's other research interests include systems analysis and management accounting in systems development projects. Kieran was a management consultant with Accenture, where he worked on a variety of projects across Europe and the United States. Kieran can be reached by e-mail at kieran.conboy@nuigalway.ie.

Brian Fitzgerald holds the Frederick A Krehbiel II Chair in Innovation in Global Business & Technology, at the University of Limerick, Ireland, where he is also Research Fellow and Science Foundation Ireland Principal Investigator. Brian's research interests lie primarily in the area of software development, a broad area which encompasses the use of development methods, globally distributed software development, agile methods, and open source software. Having worked in industry prior to taking up an academic position, Brian has more than 20 years experience in the software field. This experience was gained in a variety of companies (Citibank, Eircom, IDS Computing, Ridge Tool Company) in a number of countries (Ireland, Belgium, Germany). Brian can be reached by e-mail at bf@ul.ie.