Effects of Personality on Pair Programming

Jo E. Hannay, Erik Arisholm, *Member, IEEE*, Harald Engvik, and Dag I.K. Sjøberg, *Member, IEEE*

Abstract— Personality tests in various guises are commonly used in recruitment and career counseling industries. Such tests have also been considered as instruments for predicting the job performance of software professionals both individually and in teams. However, research suggests that other human-related factors, such as motivation, general mental ability, expertise and task complexity also affect performance in general. This paper reports on a study of the impact of the Big Five personality traits on the performance of pair programmers together with the impact of expertise and task complexity. The study involved 196 software professionals in three countries forming 98 pairs. The analysis consisted of a confirmatory part and an exploratory part. The results show that (1) our data does not confirm a meta-analysis-based model of the impact of certain personality traits on performance; and (2) personality traits in general have modest predictive value on pair programming performance compared with expertise, task complexity, and country. We conclude that more effort should be spent on investigating other performance-related predictors such as expertise, and task complexity, as well as other promising predictors, such as programming skill and learning. We also conclude that effort should be spent on elaborating on the effects of personality on various measures of collaboration, which in turn may be used to predict and influence performance. Insights into such malleable, rather than static, factors may then be used to improve pair-programming performance.

Index Terms— Pair programming, Personality, Big Five, Expertise, Task Complexity, Performance

1 Introduction

Pair programming is the practice where two programmers work together on the same programming task using one computer and one keyboard [9], [29], [30], [38], [111], [115]. Several flavors of this collaboration are possible. One might define distinct roles, where one programmer, the "driver", is in charge of the keyboard and focuses on the actual coding, while the other, the "navigator", observes and comments on the coding, searches for alternative solutions, or contributes in other ways to solving the task. These roles may be held throughout a work session, they may be switched several times during a work session, or they may be ignored altogether letting the keyboard pass freely between the two programmers at will.

The close and direct way of collaborating in pair programming might intensify both the benefits and problems of small-group collaboration in general [40]. This raises issues concerning the interaction between the individuals in a pair that influences pair performance. Several researchers have henceforth sought to identify and investigate various human factors that are postulated to affect this interaction [1], [43], [23], [26], [35], [55], [65], [71], [94], [105], [110], [117]. These factors include diverse issues such as personality, gender, expertise, attitudes, preferences, ethnicity and generation.

Personality has been a subject of interest in the context of programming and software engineering for some time. For example, Weinberg predicted, in his famous 1971 book, that "attention to the subject of personality should make substantial contributions to increased programmer performance" [111], a position he reaffirms in the 1998 edition of the book [112]. Shneiderman in his equally famous book "Software Psychology" states: "Personality variables play a critical role in determining interaction among programmers and in the work style of individual programmers" [97]. However, both authors admit to a lack of empirical evidence on the impact of personality on performance: "Personality tests have not been used successfully for selecting programmers who will become good programmers" [111], [112], "Unfortunately too little is known about the impact of personality factors" [97]. More recently, however, empirical studies lead Devito Da Cunha and Greathead to conclude that "if a company organizes its employees according to personality types and their potential abilities, productivity and quality may be improved" [33], and Dick and Zarnett conclude that "Building a development team with the necessary personality traits that are beneficial to pair programming will result in greater success with extreme programming than a team built based on technical skills alone" [35].

Somewhat contrary to this optimism, studies on the impact on general job performance show that the effects of personality are relatively small [6]. In the context of pair programming, personality traits have been applied to the issue of pair composition (Section 3). However, this literature is not explicit on what, or how large, the effects of personality are when it comes to pair programmers.

There should be a debate as to whether personality does matter in software engineering. Personality as such is well researched, and several existing personality tests exhibit both reliability and validity. It is therefore meaningful to speak of a person's personality as measured by these tests. However, it is a separate issue whether a person's personality may be used to predict behavior or performance in a reliable and valid manner. In addition, other factors than personality might be stronger predictors of performance.

This paper reports on an empirical study on the effects of personality on pair programming performance. We also investigated other factors in conjunction to personality, namely expertise, task complexity and country of employment.

A total of 196 professional IT consultants from ten companies in three different countries participated as subjects in the study. The analysis of the data consisted of two parts. The first part was confirmatory, relative to a model based

• Jo E. Hannay and Erik Arisholm, are with the Department of Software Engineering, Simula Research Laboratory, Pb. 134, NO-1325 Lysaker, Norway and the Department of Informatics, University of Oslo, Pb. 1080 Blindern, 0316 Oslo, Norway. Email: {johannay,erika}@simula.no.
• Harald Engvik is with the Department of Psychology, University of Oslo, Pb. 1094, 0317 Oslo, Norway. Email: harald.engvik@psykologi.uio.no.
• Dag I.K. Sjøberg is with the Department of Informatics, University of Oslo, Pb. 1080 Blindern, 0316 Oslo, Norway. Email: dagsj@ifi.no.

Manuscript received xx xxxx xxxx; revised xx xxxx xxxx; accepted xx xxxx xxxx; published online xx xxxx xxxx.
Recommended for acceptance by NN
For
information on obtaining reprints of this article, please send e-mail to: tsecomputer.org, and reference IEEECS Log Number TSE-XXXX-XX-XXXX.
Digital Object Identifier

on past empirical results. The findings from this part of the analysis were not strongly in favor of the proposed model. In the second part, we therefore conducted an exploratory analysis.

In the next section (Section 2) we present the Big Five model as well as other models of personality. We then summarize related work (Section 3) and give an overview of our study (Section 4). We proceed to describe our models (Section 5) and our analysis (Section 6). Section 7 discusses implications of our findings, and Section 8 concludes.

2 Personality

There exist a multitude of personality models. Any given model may have several alternative operationalizations, which give rise to the actual tests that are administered to measure a person's personality according to that model. Personality tests are in extensive commercial and governmental use by, among others, recruitment and career counseling agencies and the military. Although several of these tests may originally have had theoretical or empirical underpinnings in psychological research, many of them are simplified or altered over time for specific purposes with little or no scientific control. (See [87] for critical anecdotes and the history of personality testing.) At the same time, personality research in academia has developed well-researched models and tests. Two models that in recent years have dominated personality research [6] consist of five factors and go under the names of the *Five Factor Model* (FFM) [31] and the *Big Five* [44], [46]. The FFM posits that traits are situated in a comprehensive model of genetic and environmental causes and contexts. The Big Five posits that the most important personality differences in people's lives will become encoded as terms in their natural language, the so-called *Lexical Hypothesis* [44]. These two models are often seen as one, and their respective factors correlate quite well, e.g., [48]. However, the two models are conceptually different and their theoretical bases imply different approaches to designing indicators for the factors. In our study, we used the Big Five, although we consider findings found for both models in our confirmatory analysis.

2.1 The Big Five

The Big Five model consists of five personality factors (traits) [44], [46]. The five traits are (with descriptions from [89]):

Extraversion (Factor 1) Assesses quantity and intensity of interpersonal interaction; activity level; need for stimulation; and capacity for joy.

Agreeableness (Factor 2) Assesses the quality of one's interpersonal orientation along a continuum from compassion to antagonism in thoughts, feelings, and actions.

Conscientiousness (Factor 3) Assesses the individual's degree of organization, persistence, and motivation in goal-directed behavior. Contrasts dependable, fastidious people with those who are lackadaisical and sloppy.

Emotional stability/Neuroticism (Factor 4) Assesses adjustment versus emotional stability. Identifies individuals prone to psychological distress, unrealistic ideas, excessive cravings or urges, and maladaptive coping responses.

Openness to experience (Factor 5) Assesses proactive seeking and appreciation of experience for its own sake; toleration for and exploration of the unfamiliar.

A number of well-established operationalizations (scales, markers) exist for the Big Five. One well-known scale is the Big Five Factor Markers (BFFM) [64], [48], [47], [45], which comes in a long version with 100 indicators—20 per trait (BFFM-100) and a short version with 50 indicators—10 per trait (BFFM-50). The indicators are unipolar, but come in positive and negative keys; that is, some items indicate a positive contribution to a trait, while others indicate a negative contribution. The 100 indicators are self-assessment questionnaire items on a seven-point Likert scale. The scales appear in several validated language translations.

The five constructs of the Big Five are given as universal traits in the human population, and many researchers hold that the constructs provide a genuine insight into the concept of personality. The model allows researchers to determine a person's personality in terms of traits that are scientifically validated. According to Barrick et al. [6], the emergence of the FFM and the Big Five in the mid 1980s established a marked shift in personality research. The overall conclusion from research up to that point was that personality and job performance were not related in any meaningful manner. However, the FFM and the Big Five finally allowed researchers to establish personality in a meaningful way and that at least some aspects of personality are related to performance.

2.2 Other Personality Models

Perhaps the most widely known and commercially used personality model is the Myers-Briggs Type Indicator (MBTI) [82], [83], which was inspired loosely by Jung's psychological types. The MBTI is built around two sets of *functions*: the perceiving functions (sensing–intuition scale), which describe how a person acquires information, and the judging functions (thinking–feeling scale), which describe how a person processes information. Then, the *attitudes* (extraverted–introverted) describe whether these functions show in the external world of action, people and behavior, or in the internal world of ideas and reflection. The *lifestyles* (judging–perceiving) describe a person's preference to show the allotted judging function or the perceiving function to the outside world. Thus, the MBTI ascribes a complex personality type to a person, rather than orthogonal traits as does the Big Five model.

Other related models also build complex types; for example, the Keirsey Temperament Sorter [67], [66], which is inspired by the MBTI and by Hippocrates' four humors, builds a type according to four onion-like *rings*. Other models in frequent commercial use include variants of the Felder-Silverman Learning Styles (FSLS) [37], the 16PF [20], [21], the DiSC Personality Profile Assessment based on [77], the Minnesota Multiphasic Personality Inventory (MMPI) [60], [18], and the Thematic Apperception Test (TAT) [32], [113].

Several commercially used models and tests have been criticized in the academic community for having poor conceptual foundations and for having low reliability and validity [42], [91], [92]. In particular, many personality tests have been associated with the "Forer Effect"[1], which applies to general and vague descriptions that are likely to evoke feelings of recognition in anyone, regardless of actual personality.

[1] B. T. Forer [39] administered a personality test to his students. He then simply discarded their responses and gave all students the exact same personality analysis copied from an astrology book. The students were subsequently asked to rate the evaluation on a five-point Likert scale according to how accurately they felt that the evaluation described them. The mean was 4.26. Forer's study has been replicated numerous times, and averages remain around 4 [36], [58]. The Forer effect is also referred to as the "Barnum Effect" [78].

2.3 Personalty as a Predictor of Performance

The Big Five model may provide insight to the concept of personality. However, the question of whether it is possible to predict task performance on the basis of personality (no matter how well-defined personality might be) is a different matter.

A multitude of studies have been conducted on the effects of personality on (team) performance, and several researchers have undertaken meta-analyses of these studies. Table 1 summarizes the results of three meta-analyses. The second-order meta-analysis (a meta-analysis of meta-analyses) by Barrick et al. [6] reports results on individual performance as well as on team performance.

For team performance, some notion of "team personality" must be devised. The most common ways of aggregating personality scores into team scores is by taking the mean, the minimum, the maximum, or the variance of the individual scores, as in Table 1. Which aggregation one chooses should reflect the type of collaboration involved, which in turn might depend on both the task characteristics and the predictor variable. For example, according to Steiner's task typology [103], [40], the mean of the predictor variable should be of interest in an *additive* task since performance is thought to be a sum of the team members' individual contributions. On the other hand, the minimum is appropriate in *conjunctive* tasks, because team performance depends on the weakest link of the team. However, the person with the highest level of, say, *Extraversion* might dominate the team which may warrant that the maximum of this predictor variable is the appropriate team aggregate.

According to Barrick et al., the general effects of personality on job performance are "somewhat disappointing" [6] and "modest...even in the best of cases" [6]. Thus, personality may have little direct effect on job performance in terms of efficiency. However, personality might have more substantial indirect effects on job performance via social factors that influence teamwork. In fact, the effects on teamwork are higher than on overall job performance for all five traits [6]. This suggests that it may be more relevant to study effects of personality in the context of collaborative performance rather than on individual performance.

TABLE 1
Summary of Meta-Analyses of the Effect of Personality (FFM and Big Five) on Performance

	Team			*Indvidual*
	Barrick et al. [6]	Peeters et al. [88]	Bell [10]	Barrick et al. [6]
Extraversion				
mean	positive		positive	
minimum				
maximum				
variance				
Agreeableness				
mean	positive	positive[a]	positive	
minimum			positive	
maximum				
variance		negative		
Conscientiousness				positive
mean	positive	positive[a]	positive	
minimum			positive	
maximum				
variance		negative	negative	
Emotional stability				positive
mean	positive	negative[b]	positive	
minimum			positive	
maximum				
variance		negative[b]		
Openness				
mean			positive	
minimum				
maximum			positive	
variance		negative[c]	negative	

[a] *main effect and for professionals but not for students*
[b] *for students only*
[c] *for professionals only*

3 Personality and Pair programming

In order to gain a comprehensive overview of related work on pair programming and personality, we conducted a systematic literature review, partly following guidelines suggested in [69]. We searched the ACM Digital Library, Compendex, IEEE Xplore, and ISI Web of Science with the following basic search string: "pair programming OR collaborative programming." In addition, we hand-searched all volumes of the following thematic conference proceedings for research papers: XP, XP/Agile Universe, and Agile Development Conference. The search string was applied to the titles, abstracts, and keywords of the articles in the above electronic databases and conference proceedings. This search strategy resulted in a total of 214 unique citations.

The first author subsequently read the titles and abstracts of these 214 studies for relevance to personality. If it was unclear from the title, abstract, and keywords whether a study conformed to our inclusion criteria, it was included for a detailed review. At this stage, all studies that indicated any reference to personality or related topics such as pair compatibility, pair matching, etc. were included. This screening process resulted in 12 citations that were subsequently retrieved and reviewed by the first author. Of these 12 articles, 10 did in fact describe studies that bear relevance to our discussion. Table 2 summarizes relevant issues from these articles as well as from our present study reported in this paper.

Three of the studies in Table 2 used the Myers-Briggs Personality Type Indicator (MBTI), two used Felder-Silverman Learning Styles (FSLS), and one used the Keirsey Temperament Sorter (KTS). Several studies used their own trait indicators and/or additional trait indicators. None, apart from ours, used the FFM or Big Five model.

Other independent variables investigated were skill (which is conceptually different from expertise), competence, self-esteem and work ethics. None investigated interaction effects on expertise, task complexity or country. Dependent variables included attitudes toward pair programming, compatibility measures, and performance measures. Pairs were formed according to personality variables in three of the studies.

Rationales for the studies ranged from adages ("opposites attract") [117], through anecdotal claims cited from the software engineering/computer science literature that pairs with mixed personalities complement each other [27], to empirical evidence. There were no explicit references to theory for explaining effects of personality on pair programming. All rationales for studies that employed the MBTI, or the related KTS, argued that opposite or mixed pairs would do better than homogeneous pairs. None argued that homogeneous pairs would be better performers.

TABLE 2
Summary of Studies on Personality and Pair Programming

1	Chao et al. [23] *survey* *experiment*	*Subjects:* 60 professionals + 21 students (survey), 58 students (experiment) *Personality Variables:* Open-minded, Logical, Responsible, Attentive. *Dependent Variables:* Quality, Compatibility *Results:* The experiment indicated that high/high and high/low combinations on either one of Open-minded and Responsible could result in higher quality code than the low/low combination (non-significant).
2	Choi [26] *survey* *experiment*	*Subjects:* 44 professionals (survey), 128 students (experiment) *Personality Variables:* MBTI. *Dependent Variables:* Code Productivity, Code Design, Communication, Satisfaction, Confidence, Compatibility *Results:* Pairs who are alike in perception or judgment, but not both, are found to perform better on Code Productivity and Code Design than pairs who are totally different or alike on these traits (significant). The survey asked professional programmers which predefined factors they thought most influences pair programming and revealed that personality, communication and gender as the perceived most influential factors. (The other predefined factors were programming skill, cognitive programming style, familiarity, and pair protocol.)
3	Dick et al. [35] *action research*	*Subjects:* 8 professionals (including authors) *Personality Variables:* Communication, Comfortable, Confidence, Compromise *Dependent Variables:* General Effectiveness *Results:* The four personality variables are posited as beneficial for pair programming.
4	Hanks [55] *survey*	*Subjects:* 115 students *Personality Variables:* Confidence *Dependent Variables:* Attitudes toward pair programming *Results:* Students with the greatest confidence had the most positive responses to pair programming attitude questions (non-significant). Results are reported to contradict those of Thomas et al. [105]. (It should be noted that the construct of "confidence" seems to be quite different from that of Thomas et al.).
5	Katira et al. [65] *regression*	*Subjects:* 564 students *Personality Variables:* MBTI *Other independent Variables:* Skill (actual), Technical competence (perceived), Self-esteem *Dependent Variables:* Compatibility (self-assessed) *Results:* Differences in personality types led to higher compatibility for one session of the experiment (significant), but not the other. However, 90% of pairs (randomly allocated and irrespective of personality) report compatibility.
6	Layman [71] *regression*	*Subjects:* 78 students *Personality Variables:* MBTI, FSLS *Other independent Variables:* Skill, Work ethic, Time management preference *Dependent Variables:* Changes in attitudes *Results:* Personality type (MBTI) and learning style (FSLS) had little effect on attitude change. Students who disliked collaborative experiences were predominantly reflective learners, introverts, and strong coders.
7	Sftetsos et al. [94] *experiment*	*Subjects:* 70 students *Personality Variables:* KTS *Dependent Variables:* Performance (measured by communication, velocity, productivity and customer satisfaction), Collaboration-viability (measured by developer satisfaction, knowledge acquisition and participation, i.e., collaboration satisfaction ratio, nuisance ratio, voluntary or mandatory preference, and driver or navigator preference) *Results:* Better performance and collaboration-viability for pairs with mixed temperaments (significant).
8	Thomas et al. [105] *survey*	*Subjects:* 64 students *Personality Variables:* Self-confidence (9 point scale from Code-Warrior to Code-a-phobe) *Dependent Variables:* Attitudes, Performance *Results:* Evidence that students who have considerable self-confidence do not enjoy the experience of pair programming as much as other students and that students produce their best work when placed in pairs with students of similar self-confidence levels.
9	Visram [110] *analysis*	*Subjects:* N/A *Personality Variables:* EI *Dependent Variables:* N/A *Results:* Advice for successful pair programming [52], [116] relates to Goleman's traits of Emotional Intelligence [49].
10	Williams et al. [117] *regression*	*Subjects:* 1350 Students *Personality Variables:* MBTI, FSLS *Other independent Variables:* Skill (perceived, actual), Self-esteem, Work ethic, Time management preference *Dependent Variables:* Compatibility (self-assessed) *Results:* Different sensing-intuition (MBTI) and sensing-intuitive scores (FSLS) correlated with highly compatible pairs (significant). However, 93% of pairs (randomly allocated and irrespective of personality) report compatibility.
11	Our study *regression* *path analysis* *regression trees*	*Subjects:* 196 professionals *Personality Variables:* Big Five *Other independent Variables:* Expertise, Task Complexity, Country *Dependent Variables:* Pair Performance *Results:* Low support of hypotheses. Other factors than personality have greater impact.

EI=Emotional Intelligence [49], *FSLS=Felder-Silverman Learning Styles* [37], *KTS=Keirsey Temperament Sorter* [67], *MBTI=Myers-Briggs Personality Type Indicator* [82]

4 Overview of Our Study

Our study was an integrated part of an experiment reported in [2], which compared the performance of professional pair programmers with that of solo programmers. Our study focused on the 196 programmers forming the 98 pairs of that experiment. These programmers were recruited from software consultancy companies in Norway, Sweden and the UK in the second half of 2004 and in the first half of 2005.

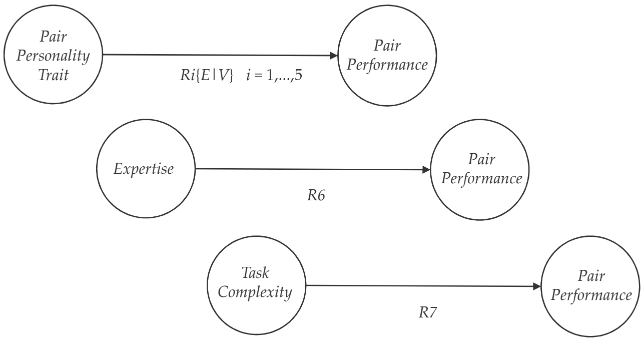
The pairs were formed so that both individuals in a pair had the same level of expertise. Each individual's level of expertise was rated by his or her workplace manager. The subjects did not know in advance who their partner would be during the study, and pairs were formed across companies (but within the same country). Within each level of expertise, pairs were assigned randomly to one of two treatments pertaining to task complexity.

Each pair participated for one day and their session was divided into four stages. First, the subjects were given an introductory presentation that included practical matters as well as an introduction to the concept of pair programming, which focused on the active collaboration in pair programming and which involved a short description of the two roles (driver and navigator). The subjects were told that they could decide for themselves how often and when to switch roles, but that they had to try both roles (even if only for five minutes). After the presentation, the subjects answered a prequestionnaire about their education and experience before they started performing the training task (T_0) and the pretest task (T_1) individually. Then, the subjects started to perform the main tasks (T_2-T_4) as well as a time sink task (T_5) in pairs. The tasks T_2-T_5 were done on two different versions of the program according to the task complexity treatment. To support the logistics of the study, the subjects used a web-based experiment support tool [4] to answer questionnaires, download code and documents, and to upload task solutions. For each task a test case was provided that each subject or pair used to test the solution. Eight hours were allotted to the completion of tasks T_0–T_4, of which six hours were thought to be sufficient for T_2–T_4. Further details are provided in [3]. At the end, the option was given to complete the Big Five personality test.[2] All but 11 subjects completed the personality test.

5 Research Questions and Hypotheses

There is little theory that links personality trait models to task performance or team work. Instead, predictions in the literature are based on commonsense reasoning[3]; for example, "In jobs involving considerable interpersonal interaction, being more dependable, thorough, persistent and hard working (high in conscientiousness), as well as being calm, secure and not depressed or hostile (high in emotional stability), should result in more effective interactions with co-workers or customers" [6]. Similarly, we are not aware of any theory linking personality models to (pair) programming performance. Our confirmatory analyses were therefore based on prior empirical research as described in the following.

We conducted two confirmatory analyses, one univariate analysis and one multivariate analysis. We first tested the univariate models implied by the meta-analyses in Table 1. Common in personality research, univariate models give postulates for the isolated effect of one personality trait (and one team aggregate) at a time, on some performance measure. The conceptual models, giving concepts and postulated relationships, are indicated in Fig. 1. The different team aggregates (mean, variance, minimum, maximum) of Table 1 are conceptualized into *Elevation* (mean, minimum, maximum) and *Variability* (variance) (see Section 5.1). Each relationship $Ri\{E|V\}$ (*E* for *Elevation* and *V* for *Variability*) is then derived from the observations in Table 1. Each relationship gives rise to one model.



R1E: *Elevation* in *Extraversion* increases *Pair Performance*
R2E: *Elevation* in *Agreeableness* increases *Pair Performance*
R2V: *Variability* in *Agreeableness* decreases *Pair Performance*
R3E: *Elevation* in *Conscientiousness* increases *Pair Performance*
R3V: *Variability* in *Conscientiousness* decreases *Pair Performance*
R4E: *Elevation* in *Emotional Stability* increases *Pair Performance*
R4V: *Variability* in *Emotional Stability* decreases *Pair Performance*
R5E: *Elevation* in *Openness* increases *Pair Performance*
R5V: *Variability* in *Openness* decreases *Pair Performance*
R6: *Expertise* increases *Pair Performance*
R7: *Task Complexity* decreases *Pair Performance*

Fig. 1. Univariate Conceptual Models.

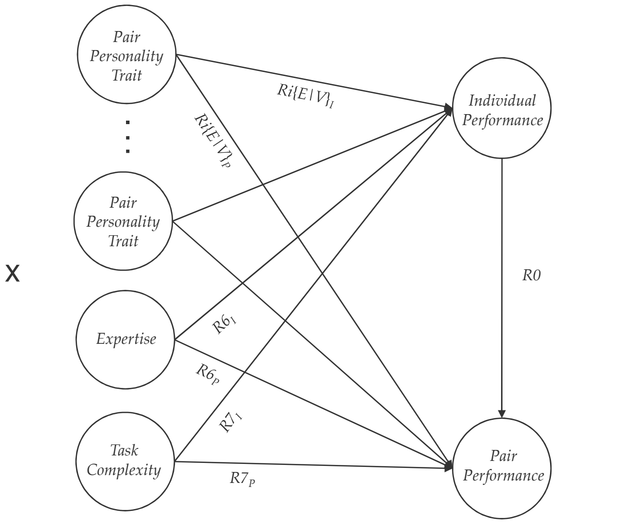
We were also interested in the effects of expertise and task complexity. Based on the prior studies on expertise and task complexity [3], [54], we extended the univariate case with relationships *R6* and *R7* (Fig. 1).

Our second confirmatory analysis related to a multivariate model that expresses the simultaneous effects of all independent variables (with relevant interactions). Fig. 2 shows the multivariate conceptual model. (The X indicates interactions between the independent (i.e., predictor) variables.) This model also included *Individual Performance* as a predictor of *Pair Performance*. Moreover, *Personality* was included as a predictor of *Individual Performance*. Thus, the total effect $Ri\{E|V\}$ of a personality trait on *Pair Performance* is split into the direct effect $Ri\{E|V\}_P$ on *Pair Performance* and the indirect effect $Ri\{E|V\}_I$ that is mediated by *Individual Performance*. A similar constellation is given for *Expertise* and *Task Complexity*. Apart from expressing effects on both *Individual* and *Pair Performance*, this model allows one to analyze the *Gain* in the effect of a predictor variable on *Performance* when moving from individuals to pairs. (This is equivalent to the effect of a predictor variable on the gain in performance when moving from individuals to pairs.) In this context, *Gain* may be understood as the synergy effect of pairing, relative to the performance potential of the individuals in a pair. Note that a predictor's *Gain* in effect may well be the reverse of its effect on *Pair Performance*.

There is no hypothesized *Gain* in effect due to *Personality*. However, there is a hypothesized *Gain* in effects due to *Expertise* and *Task Complexity* ($R6_G$ and $R6_G$+). These relationships are postulated based on the findings in [2] that suggest that pairing up is most beneficial for lower levels of expertise on

[2] The personality test was optional due to ethical reasons.

[3] Commonsense reasoning is arguably distinct from theoretical reasoning [76], [73].



Rj is the total effect on *Pair Performance* consisting of the direct effect *Rj_P* and the indirect effect *Rj_I* on *Individual Performance* via the effect *R0* of *Indivdual Performance on Pair Performance.*

R1E: *Elevation* in *Extraversion* increases *Pair Performance*
R2E: *Elevation* in *Agreeableness* increases *Pair Performance*,
R2E+: ... and more so for high levels of *Expertise*
R2V: *Variability* in *Agreeableness* decreases *Pair Performance*
$R3E_I$: *Elevation* in *Conscientiousness* increases *Individual Performance*
R3E: *Elevation* in *Conscientiousness* increases *Pair Performance*
R3E+: ... and more so for high levels of *Expertise*
R3V: *Variability* in *Conscientiousness* decreases *Pair Performance*
$R4E_I$: *Elevation* in *Emotional Stability* increases *Individual Performance*
R4E: *Elevation* in *Emotional Stability* increases *Pair Performance*,
R4E-: ... but is negatively related for low levels of *Expertise*
R4V-: *Variability* in *Emotional Stability* decreases *Pair Performance*
... for low levels of *Expertise*
R5E: *Elevation* in *Openness* increases *Pair Performance*
R5V: *Variability* in *Openness* decreases *Pair Performance*,
R5V+: ... and more so for high levels of *Expertise*
$R6_I$: *Expertise* increases *Individual Performance*,
$R6_I$+: ... and more so for higher, than lower, *Task Complexity*
R6: *Expertise* increases *Pair Performance*,
R6+: ... and more so for higher, than lower, *Task Complexity*
$R6_G$: *Expertise* decreases *Pair Gain*,
$R6_G$+: ... and more so for lower, than higher, *Task Complexity*
$R7_I$: *Task Complexity* decreases *Individual Performance*
R7: *Task Complexity* decreases *Pair Performance*

Fig. 2. Multivariate Conceptual Model.

more complex tasks.[4]

The postulated interaction effects (apart from *$R6_G$+*) were taken from the meta-analyses (Table 1) where we chose to translate interaction effects on students and professionals to interaction effects on *Expertise* in our model.

Our third analysis was exploratory. For that analysis, we included *Country* as a predictor variable.

The conceptual models in Figs. 1 and 2 have concepts and relationships, which are the basic building blocks of scientific theories [5], [51], [57], [114]. What is missing for this model to become an explanatory theory in the sense of [51], [107] are propositions that explain why each relationship *Ri* holds. To our knowledge, such explanatory propositions are not available beyond the commonsense reasoning alluded to above. However this reasoning diverges in content and is not part of a wider explanatory framework, and we therefore omit propositions from the models.

5.1 Constructs and Indicators

The conceptual models of Figs. 1 and 2 depict certain relationships between the concepts of personality, expertise, task complexity and performance. These concepts and relationships need to be operationalized into observable variables (*indicators*) before they can be studied in an empirical study. Once a concept is associated with indicators, we refer to the concept by the more technical term *construct* [95]. The constellation of a construct and its indicators is often called a *measurement model*. The measurement models for our constructs are depicted in Fig. 3. Once the indicators are established, one may present an *analysis model*, which is the conceptual model as expressed by its indicators.[5] We will present analysis models in Section 6.

Disciplines such as social science and psychology have established many constructs whose corresponding indicators have been extensively tested for construct validity and reliability; an example being the Big Five personality traits and their indicators. In contrast, the constructs of empirical software engineering have not reached the same level of maturity: they are rarely validated with respect to reliability and construct validity, the constructs' definitions are in many cases not uniformly understood or agreed upon, and for the most part, constructs are given only a single indicator. Nevertheless, on the way to reaching a higher level of maturity, researchers must make use of current understanding and current measures, even if they are incompletely understood. In this fragile process it is essential that one is explicit about one's concepts and their operationalizations.

In Sections 5.2 and 5.3, we describe the indicators for the concepts of the conceptual models shown in Figs. 1 and 2.

5.2 Independent Variables

We first describe the independent variables.

Construct: *Pair Personality Trait (Elevation, Variability)*

Indicators: *Pair BFFM-100 Factor Score (mean and difference)* The indicators for individual personality are the BFFM-100 questionnaire items for the Big Five model (Section 2). Each personality trait (construct) is operationalized through 20 indicators. Fig. 3 *a* illustrates the case for *Extraversion*. In Fig. 3 *a*, the links between construct and indicators are represented by arrows from the construct to the indicators. This directionality corresponds to the idea that a personality trait is something that is inherent in an individual and that gives rise to observable behavior in the corresponding indicators, i.e., the specific answers given on the questionnaire items. Thus, variation in the indicators reflect variation in the construct. Measurement models of this kind are therefore referred to as *reflective*, and are the standard constellations of factor analysis [8], [50].

A construct's magnitude of influence on an indicator is given by so-called loadings ($\lambda_1, \lambda_2, \ldots, \lambda_{20}$ in Fig. 3 *a*). The loadings may be based on theory and/or empirical data, and setting them belongs to the measurement model-building stage. Subsequently, the loadings may be used to calculate a score on the construct based on measurement scores on its indicators; i.e., one may calculate a person's *Extraversion* score based on his/her score on the corresponding questionnaire items. There are various ways to do this [86], but in BFFM-100, a principal components approach is taken.

[4] Note that our present analysis is using parts of the same data as that study. However, the present study investigates these effects in the presence of *Personality* factors.

[5] In structural equation modeling it is possible to conduct statistical analysis directly on the construct level [70], [75]. We do not do this here since several of our constructs are not validated yet.

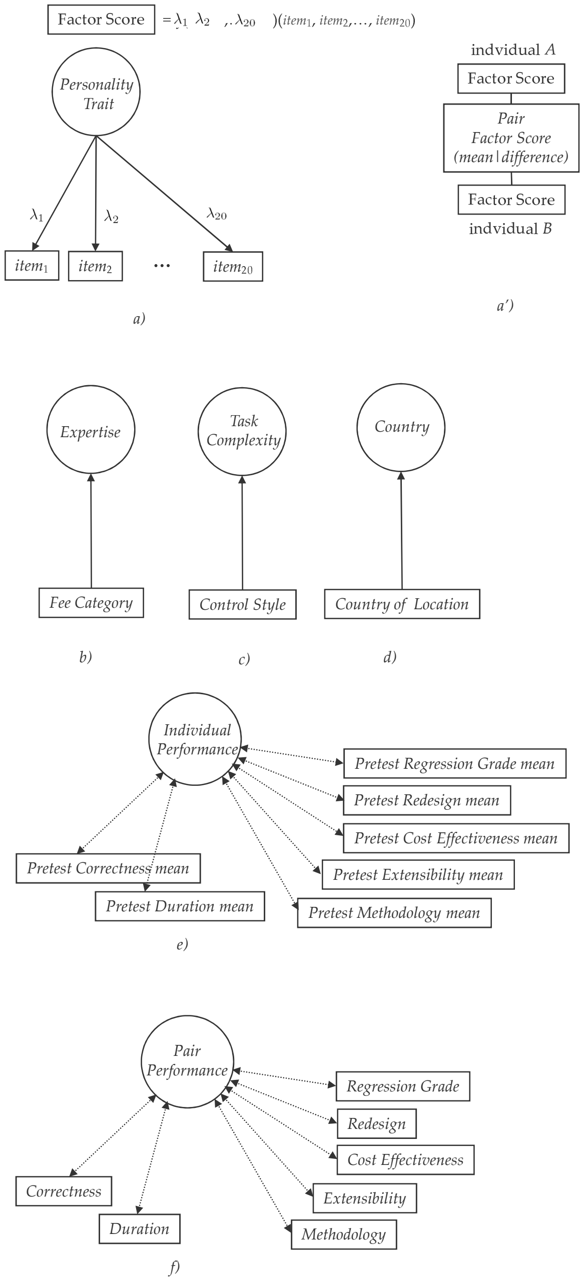


Fig. 3. Measurement Models.

We computed principal components in SPSS asking for five components and then we computed factor scores by the "regression method" (indicated by the $f(\lambda_1, \lambda_2, ..., \lambda_{20})(...)$ expression Fig. 3 *a*). Following tradition, the scores were standardized to a mean of 50 and a standard deviation of 10. The English version of the scale can be found at [64]. The Norwegian version of the BFFM-100 is developed by Dr. Harald Engvik (co-author of this paper). The Swedish version is developed by Dr. Martin Bäckström; see [64] for information on these translations.

As mentioned in Section 2.3, common team personality aggregates are the mean, variance, minimum, or maximum of the team's individuals' scores. However, the mean, minimum, and maximum may conceptually all be seen as indicators for the same sub-construct of *Team Personality*, namely *Elevation*, and the variance may be seen as an indicator for a sub-construct *Variability*. Empirically, this is supported by the agreement of results for mean, minimum, and maximum across the meta-analyses in Table 1, and, in our data, by the fact that the mean, minimum, and maximum correlate bivariately in the range 0.402–0.951, all at $p < 0.01$. Also, Peeters et al. [88] refer to [7], [68], [80], [84], [108] and promote *Elevation* and *Variability* as the appropriate sub-constructs. Additional indicators for *Elevation* are summed individual scores for the trait or the proportion of high-scoring individuals for a trait. *Variability* is also expressed as the standard deviation.

Based on this, we chose trait elevation in terms of the mean of the two scores (which is analytically equivalent to sums and more sensitive than the minimum, maximum, or proportions for groups consisting only of two persons) and trait variance in terms of the variance of the two scores, which in the dyad case is equivalent to the difference between the two scores (Fig. 3 *a′*).

Construct: *Expertise*

Indicator: *Fee Category*

The conceptual model also postulates that pair performance is affected by *Expertise*. In our study, we operationalized *Expertise* by a single indicator *Fee Category* (Fig. 3 *b*). This three-category ordinal variable (*junior*, *intermediate*, *senior*) is the consulting fee level charged for a subject by his/her company. This indicator is relevant to pair programming because much of the debate around pair programming revolves around whether it is worth spending the extra manpower associated with pair programming. A manager in each company selected the subjects from the company's pool of consultants and rated them on to their Java programming experience according to how they would rate them for similar kinds of real projects. Consequently, a few consultants with ample general work or programming experience (but very little OO or Java experience) could still be rated as *junior* by the companies. This operationalization of *Expertise* in terms of *Fee Category* is related to the standard operationalization in terms of amount of domain-specific experience found in other disciplines.

Unlike personality traits, which give rise to observations in the indicators and whose measurement models are therefore reflective, *Expertise* is a construct that is more naturally expressed by way of a *formative* measurement model. In formative measurement models, indicators are viewed as expressing various aspects of a construct. In contrast to a reflective measurement model where convergent validity is an issue, formative indicators should not correlate closely, since each indicator expresses a different aspect of the construct [11], [16], [34], [86]. Formative measurement models are depicted with arrows leading from the indicators to the construct (Fig. 3 *b*), since the indicators are seen to contribute to the definition of the construct.[6]

Each pair in this study consisted of two individuals with a similar level of programmer expertise (*junior/junior*, *intermediate/intermediate*, and *senior/senior*). This choice was motivated by previous studies on pair programming that reported that pairs consisting of individuals with similar competence levels collaborated more successfully than those with different competence levels [19], [43], [115].

[6] In this framework, it is important to be aware that the operationalizations do not constitute *nominal* definitions of their concepts; that is, *Expertise* is *not* fully defined in terms of its indicators. Neither is *Extraversion* fully defined in terms of the scores on the questionnaire items. Instead, both *Expertise* and *Extraversion* are concepts that we as researchers hold in our minds *a priori*, and which we operationalize in a given research setting by giving indicators that are as appropriate as possible under logistic and intellectual constraints. Such an operationalization may be seen as a *denotative* definition of a concept; that is, the concept is given meaning by listing examples. However, the full definition of the concept may lie outside what is practically possible or it may not be fully understood theoretically [28, p. 138].

The *Expertise* concept is one of the classic concepts of social and behavioral science and has undergone several stages of elaboration. Sternberg [104] summarizes the evolution of the concept through three stages: *superiority of information processing* [85], through *quantity of knowledge*, to *superiority in organization of knowledge*. According to Sternberg, *Expertise* can, at present, be viewed along eight dimensions: *different cognitive processes*, *higher quantity of knowledge*, *superior knowledge organization*, *superior analytic ability*, *superior creative ability*, *superior automatization* and *superior practical ability*; see Hærem [53]. One of the important trends currently is that *Expertise* is treated as domain-specific and mutually dependent on task complexity [54].

We are interested in expertise pertaining to programming tasks. The development and validation of instruments to assess programming expertise according to, say, the above eight dimensions has not come very far, and empirical research on expertise in the software engineering literature is not abundant. For example, among the 103 articles reporting software engineering experiments surveyed in [99] only three [17], [90], [61] explicitly investigated an expertise construct related to programming in some way (operationalized respectively, by type of mental model, by degree of semantic knowledge, and by students versus professionals). In related disciplines, expertise is often operationalized by amount of domain-specific experience in various ways [102], [96], since it is postulated that the mental representations that lead to increased levels along any of the eight dimensions develop over time; see [53] for a review of operationalizations of IT-expertise in the management literature.

The programming expertise construct for software engineering should be viewed as "under construction". In the process of determining which aspects this construct should embody, one is somewhat justified in conducting studies that consider one aspect at a time.

Construct: *Task Complexity*

Indicator: *Control Style*

Like *Expertise*, *Task Complexity* is also a concept that has been researched extensively [53]. Again, we chose a formative measurement model and a single indicator, namely *Control Style*, which is the control style (*centralized*, *delegated*) in the application on which the subjects performed the tasks (Fig. 3 *c*). In a centralized control style, a few large "control classes" coordinate a set of simple classes [119]. In contrast, in a delegated control design, a well-defined set of responsibilities are distributed among a number of classes [119], and the classes play specific roles and occupy well-known positions in the application architecture [119], [120]. Applications written in a delegated control style, might be more elegant and in accordance with existing responsibility-driven design principles [119], [120], but might also be harder to comprehend and change due to a larger amount of delocalized plans [101]. An experiment that investigated the impact of the two control styles on the exact same application and the same tasks as those used in our study confirmed that, on average, the delegated control style was more difficult to change than was the centralized control-style implementation [3]. That experiment was conducted with individual programmers.

Construct: *Country*

Indicator: *Country of Location*

Scientific results are only useful to the extent to which they apply in a range of situations. The study was conducted across three Western European countries in an attempt to challenge the robustness of the results across countries that, to some extent, can be said to share the same work culture. Under the simplifying assumption that corporate culture overrides individual culture, we therefore chose to operationalize the *Country* construct by the country (*Norway*, *Sweden*, *UK*) in which the subject's company was located (Fig. 3 *d*).

5.3 Mediator and Dependent Variables

The mediator (multivariate model only) and dependent variables pertain to the concept *Performance*; that is, the mediator variable pertains to *Individual Performance*, while the dependent variable pertains to *Pair Performance*. The indicators for these two constructs are the same, but while those for *Pair Performance* measure the overall performance on the pair programming tasks T_2–T_4, the indicators for *Individual Performance* measure the performance on the individual pretest task T_1, but averaged over the two individuals in a pair (Fig. 3 *e*, *f*). The indicators for both constructs are aspects of performance. However, we have not validated the construct with respect to these indicators, and hence we are not sure to what extent a formative or reflective measurement model is appropriate. We therefore tentatively present a measurement model for further elaboration in Fig. 3 *e*,*f*. The double-headed arrows signify that the model may be deemed reflective, formative, or a combination of the two, in the future.

Construct: *Individual/Pair Performance*

Indicators: *Correctness*, a binary functional correctness variable 0/1, where 1 was awarded if the change task(s) was/were implemented correctly, and 0 was given if there were serious logical errors in the solution(s).

Duration, a continuous variable recording the time spent by a pair on the task/tasks.

Methodology, a five-point ordinal variable registering the extent to which the solution made use of good object-oriented coding principles.

Extensibility, a five-point ordinal variable registering the degree to which the solution may be extended easily with further functionality.

Cost effectiveness, a five-point ordinal variable registering the extent to which the solution was simple and reused existing code.

Redesign, a five-point ordinal variable registering the extent to which the solution changed the design of the system.

Regression grade, a four-point ordinal variable (*no solution*, *major deviations*, *minor deviations*, *correct*) registering the difference between actual and expected output of a simple test case covering the main functions of the system.

Note that *Duration* is of interest primarily for correct solutions (*Correctness* = 1). In the present scheme for the *Pair Performance* construct, we therefore disregarded incorrect solutions when analyzing *Duration*. Future operationalizations should incorporate *Correctness* and *Duration* in a better manner.

The *Correctness* variable was assessed independently by two senior consultants who were not among the subjects and who were not informed about the research questions of the study. *Duration* was recorded by the subjects in the web-based experiment support tool. The last five variables were assessed by a single expert programmer, who scored all solutions using a web-based system that had facilities for viewing details related to the task descriptions and task solutions for each subject, including the source code and source code differences, as well as the results from the automated test case execution (actual, expected and difference between actual and expected output).

5.4 Situational Variables

The study's *situational variables* are variables other than the independent and dependent variables; such as subjects, tasks, systems and settings.

Subjects. The intended target population of this study was professional software developers (software professionals). Whereas generalization from the variables of our study must generally follow principles of analytical generalization [56], [95], the fact that numerous subjects are necessary for conclusion validity (power and significance) also allows one to consider statistical generalization on subjects. To obtain a broad sample of software professionals, 196 Java consultants were hired from a total of 10 software consultancy companies in Norway, Sweden, and the UK.

In order to recruit these subjects, several Java consultancy companies were contacted through their sales channels. The companies were paid normal consultancy fees for the time spent on the experiment. Fees were paid according to fee category and market value, that is, *seniors* were more expensive than *intermediates*, who, in turn, were more expensive than *juniors*. For a few companies, a fixed honorarium (the same payment) for all three developer categories was agreed upon. The participating developers did not receive any information regarding the categorization or actual payment.

An overview of the subjects' education and experience can be found in [2]. Most of the subjects had no experience with pair programming. Furthermore, they performed maintenance change tasks on a program of which they had no prior knowledge.

With regards to personality scores, our sample of programmers were more homogeneous than a reference group of 1100 Norwegian army conscripted recruits.[7] This finding is accentuated by the fact that this reference group is itself homogeneous, consisting as it does, entirely of young men. Moreover, our sample of programmers score lower (p <0.0001) on *Extraversion*, lower on *Emotional Stability* (p=0.0065), and higher on *Openness to Experience* (p <0.0001) than does the reference group. This trend is strongest for the subjects from the UK. This is expected, since our subjects have a higher level of education than average, and education level is known to correlate with the *Openness* trait. The stereotype of programmers being neurotic, introvert and intellectual [22], [81], [100], [121] is hence confirmed, especially in the UK part of our sample.

Tasks and Applications. The empirical study included the programming of six change tasks: a training task (T_0), a pretest task (T_1), three (incremental) main tasks (T_2, T_3, and T_4), and a final time sink task (T_5).

Individual Training Task (T_0): All subjects were asked to change a small program so that it could read numbers from the keyboard and print them out in reverse order. The purpose of this task was to familiarize the subjects with the study's environment.

Individual Pretest Task (T_1) ATM: All subjects implemented the same change on the same system. The initial system (before changes) consisted of seven classes and 354 lines of code. The change consisted of adding transaction log functionality and printing an account statement for a bank teller machine and was not related to the main tasks. This pretest task provided the means to assess individual performance on the mediator variable.

Main Tasks (T_2–T_4) Coffee Machine: These tasks were based on two alternative Java systems that were designed and implemented with a *centralized* and *delegated* control design strategy, respectively (see *Control Style* in Section 5.2). Further details are provided in [2]. The two design alternatives were coded using similar coding styles, naming conventions, and amounts of comments. Names of identifiers (e.g., variables and methods) were long and reasonably descriptive. UML sequence diagrams of the main scenario for the two designs were given to help clarify the designs. The tasks consisted of three incremental changes to the coffee machine as follows:

T_2: *Implement a coin return button.*
T_3: *Introduce bouillon as a new drink choice.*
T_4: *Implement a check for ingredient availability.*

Time Sink Task (T_5) Coffee Machine: The final task in an empirical study needs special attention as a result of potential "ceiling effects". Consequently, the final change task (T_5) in this study was not included in the analysis. Thus, the analysis of duration and effort is not threatened by whether the pairs actually managed to complete the last task, while at the same time, the presence of the last task helped to put time pressure on the pairs during the study. The time sink task was a further incremental change:

T_5: *Make ones own drink by selecting from the available ingredients.*

Pilot studies were conducted to ensure that it would be very likely that all pairs would complete tasks T_0–T_4 within a maximum time span of eight hours. All pairs did indeed complete T_0–T_4 within the allotted eight hours.

Except for the Java source code, which contained class, method, and variable names and comments in English, all subjects received the material in the appropriate language.

Settings The study was conducted in the subjects' own offices, or in offices at Simula Research Laboratory. The work environment at Simula was similar to that of a client's site with respect to resources, development tools, etc. However, special measures were taken to ensure that pairs did not disturb each other or listen to each others' conversations during the study. To ensure accurate duration and effort data, the subjects were told to take breaks only between the tasks, not to answer telephone calls or talk to colleagues (other than the pair programming partner) during the study.

We wanted the subjects to perform the tasks with satisfactory quality in as short a time as possible because most software engineering jobs impose relatively severe time constraints on the tasks to be performed. What constitutes the best way to impose a realistic time pressure depends, to some extent, on the size, duration, and location of a study [98]. In this study, we used the following strategy: The subjects were told that they would be paid a fixed rate for five hours, regardless of how much time they would actually need. This was meant to encourage the subjects to finish as quickly as possible and to discourage them from working slowly in order to receive higher payment. However, to maintain motivation, once the five hours had passed, we told those subjects who had not yet finished that they would be paid for additional hours if they attempted to complete their tasks. The subjects were allowed to leave when they were finished. Those who did not finish had to leave after eight hours.

Finally, strict confidentiality was guaranteed regarding

[7] All Norwegian men aged 18-19 are conscripted to the army.

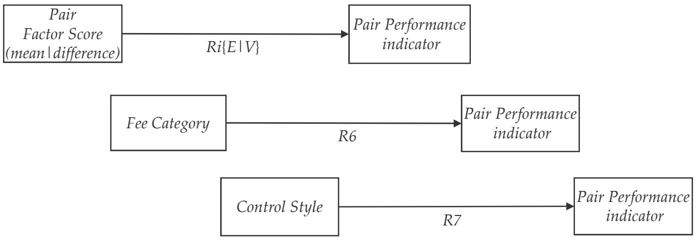

Pair
Factor Score
(mean|difference)
Ri{E|V}
Pair Performance indicator
Fee Category
R6
Pair Performance indicator
Control Style
R7
Pair Performance indicator

Fig. 4. Univariate Analysis Models.

TABLE 3
Summary of Univariate Analyses

R1E:	*Elevation* in *Extraversion* increases *Pair Performance*	
R2E:	*Elevation* in *Agreeableness* increases *Pair Performance*	✓
R2V:	*Variability* in *Agreeableness* decreases *Pair Performance*	
R3E:	*Elevation* in *Conscientiousness* increases *Pair Performance*	
R3V:	*Variability* in *Conscientiousness* decreases *Pair Performance*	✓
R4E:	*Elevation* in *Emotional Stability* increases *Pair Performance*	✓
R4V:	*Variability* in *Emotional Stability* decreases *Pair Performance*	
R5E:	*Elevation* in *Openness* increases *Pair Performance*	✓
R5V:	*Variability* in *Openness* decreases *Pair Performance*	✓
R6:	*Expertise* increases *Pair Performance*	✓
R7:	*Task Complexity* decreases *Pair Performance*	✓

information about the subjects' performance. Furthermore, the subjects signed a confidentiality agreement where they agreed not to reveal any information about the study to their peers.

6 Analysis

We used SPSS 16.0, AMOS 16.0, SAS 9.1, Enterprise Guide 2.1, and *jmp* 7 for statistical analysis.[8] Our data consisted of variables for 98 pairs made up from 196 individuals. However, *Duration* was only analyzed for pairs with correct solutions (*Correctness* = 1), i.e., 80 pairs. Descriptive data is given in Table 8 in the Appendix. We first dealt with missing values. We then proceeded with a confirmatory analysis of the conceptual models in Figs. 1 and 2. The confirmatory analysis spurred an exploratory analysis.

6.1 Missing Values

Personality scores were missing for 11 subjects, and pretest measures were missing in one way or another for eight subjects. The mechanism leading to missing values (and not the actual pattern of missingness) determines the most appropriate method for dealing with missing values [74]. The nature of our constructs implies that our missing data is neither *missing completely at random* (MCAR) (e.g., because it is conceivable that people with personality types that deviate substantially from the mean may refrain from submitting themselves to personality tests), nor *missing at random* (MAR) (e.g., because there is no other variable, say *depression*, such that the probability that personality scores are missing vary according to depression, but do not vary according to personality or anything else within each level of depression). It is not recommended to subject datasets whose missingness is neither MCAR nor MAR to deletion strategies (i.e., *listwise deletion* and *pairwise deletion*) [74].

We therefore employed the expectation-maximization (EM) imputation algorithm [74] in SPSS to estimate missing data. The EM imputation method performs satisfactorily when population distributions are not multivariate normal (although normality is a theoretical prerequisite). The data used to estimate the missing values should preferably include variables that are not used in the subsequent analysis [62], [106]. In our case, the missing personality scores for the 11 cases were imputed using data on the individual level that was either not used further in the analysis or was used subsequently in pair-aggregated form (except for *Country of Location* and *Fee Category*, which were used unaltered subsequently in the analysis of pairs). We imputed personality scores on the factor score level rather than on the indicator level. The reason we did this, was that distributions at indicator level were highly erratic, while distributions at factor level were approximately normal.

In addition, values were missing at the pair level in one to two cases for the *Pair Performance* indicators. These values were imputed using pair-aggregated forms of the same data that was used to impute missing values at the individual level, together with aggregated forms of the Big Five factor scores. Thus, the subsequent analysis was performed using data for all 98 pairs.

6.2 Confirmatory Analysis

The univariate conceptual models in Fig. 1 were tested by variants of univariate generalized linear regression models. Fig. 4 shows the analysis models that correspond to the conceptual models of Fig. 1. Table 3 shows the conclusions of the analyses as interpreted on the conceptual level. Details of the analyses are given in Table 9 in the Appendix. Effect sizes for personality are small compared with those for *Fee Category* and *Control Style*. For example, a one-point increase in *Openness mean* yields a predicted mean improvement of a meager 0.013 minutes in *Duration*, while stepping up from *Junior* to *Intermediate* yields a mean improvement of about 20 minutes. The largest effect size in favor of any hypothesized relationship on personality is 0.052 in *Regression Grade* on *Openness*.

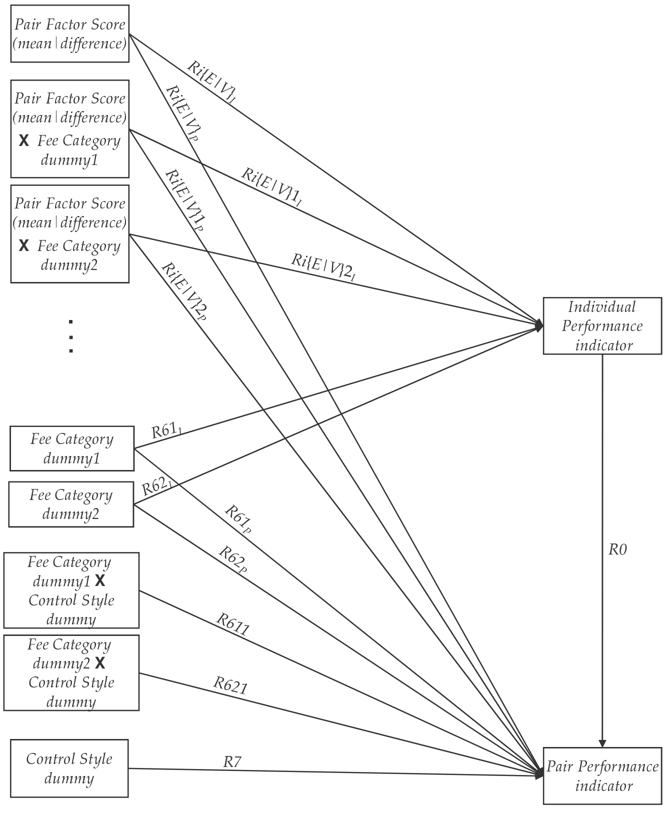
The multivariate model in Fig. 2 was tested by confirmatory path analyses in AMOS. In our context, a path analysis is a generalization of multiple regression analysis. In path analyses, mediator variables can be modeled, and covariances/correlations between independent variables may be explicitly omitted from the model, see [95], [75], [70]. A separate analysis was conducted for each of the dependent variables.[9] The multivariate analysis model is indicated in Fig. 5.

Categorical indicators for independent variables are translated into (0/1) dummy variables in the statistical analysis. Thus, the three-category indicator *Fee Category* (Fig. 3 *b*) for *Expertise* is translated into a tuple (*Fee Category dummy2*,*Fee Category dummy1*), where *junior* is represented by (0,0), *intermediate* by (1,0), and *senior* by (0,1). For example, for *Fee Category*, the relationship $R6_I$ in the conceptual model in Fig. 2 is split into $R61_I$ and $R62_I$ from, respectively, *Fee Category dummy1* and *Fee Category dummy2*. The *Control Style* variable is translated into *Control Style dummy1* where *centralized* is 0 and *delegated* is 1.

The path analysis produces four estimates per independent variable: Rk_I, Rk_P, Rk, Rk_G denote, respectively, relations on

[8] AMOS and SPSS are trademarks of SPSS Inc. SAS, Enterprise Guide, and *jmp* are trademarks of SAS Institute Inc.

[9] Had the measurement model for *Individual/Pair Performance* (Fig. 3 *e*, *f*) been validated, we could have used latent variable analysis or structural equation modeling [75], [70].


Pair Factor Score (mean | difference)
Pair Factor Score (mean | difference) X Fee Category dummy1
Pair Factor Score (mean | difference) X Fee Category dummy2
Fee Category dummy1
Fee Category dummy2
Fee Category dummy1 X Control Style dummy
Fee Category dummy2 X Control Style dummy
Control Style dummy
Individual Performance indicator
Pair Performance indicator
Ri[E|V]_I
Ri[E|V]_P
Ri[E|V]1_I
Ri[E|V]1_P
Ri[E|V]2_I
Ri[E|V]2_P
R61_I
R62_I
R61_P
R62_P
R611
R621
R7
R0

Total effect: $Rk = Rk_P + (Rk_I * R0)$. Gain: $Rk_G = Rk - Rk_I$.

Fig. 5. Multivariate Analysis Model.

individual performance, pair performance, total effect, and gain. For example, the total effect of *Extraversion average* on *Pair Performance* is $R1E = R1E_P + R1E_I$*$R0$. The gain, or synergy effect of pairing up, relative to working alone, is $R1E_G = R1E - R1E_I$.

The pretest did not distinguish between control styles, so we were unable to test the the relationships $R7_I$ and the interactions of $R6$ with $R7_I$. Table 4 shows the conclusions of the analyses as interpreted on the conceptual level. Details of the analyses are given in Table 10 in the Appendix.

Although analysis model fit is acceptable (Table 11 in the Appendix), significant effects are few and there is no substantial support for the conceptual model as a whole. Also, the R^2 values are not particularly high for several of the analysis models (Table 11). This means that there is considerable unexplained variance. This, too, may indicate that additional or other variables might be more appropriate for describing our data, or that there are variables that add disturbances to the model.

From here, one might start altering the analysis model and then check if model fit increases. One could include or exclude variables and relations following any number of elimination or introduction strategies. However, in a confirmatory mode, we feel strongly that such model searching should only be done in the presence of rival theories to guide these steps, and such theories are lacking. Even in an exploratory mode, such "correlation hunting" is not particularly reliable. It might also be the case that non-linear models would be more appropriate. After all, personality traits may be like culinary ingredients: good or bad up to a certain level with beneficial/adverse effects diminishing after that. In the next section, we switch to a purely exploratory mode.

TABLE 4
Summary of Multivariate Analyses

$R1E$:	*Elevation* in *Extraversion* increases *Pair Performance*	
$R2E$:	*Elevation* in *Agreeableness* increases *Pair Performance*,	
$R2E$+:	... and more so for high levels of *Expertise*	
$R2V$:	*Variability* in *Agreeableness* decreases *Pair Performance*	
$R3E_I$:	*Elevation* in *Conscientiousness* increases *Individual Performance*	
$R3E$:	*Elevation* in *Conscientiousness* increases *Pair Performance*	
$R3E$+:	... and more so for high levels of *Expertise*	
$R3V$:	*Variability* in *Conscientiousness* decreases *Pair Performance*	✓
$R4E_I$:	*Elevation* in *Emotional Stability* increases *Individual Performance*	
$R4E$:	*Elevation* in *Emotional Stability* increases *Pair Performance*,	✓
$R4E$-:	... but is negatively related for low levels of *Expertise*	
$R4V$-:	*Variability* in *Emotional Stability* decreases *Pair Performance* ... for low levels of *Expertise*	
$R5E$:	*Elevation* in *Openness* increases *Pair Performance*	
$R5V$:	*Variability* in *Openness* decreases *Pair Performance*,	✓
$R5V$+:	... and more so for high levels of *Expertise*	✓
$R6_I$:	*Expertise* increases *Individual Performance*,	
$R6_I$+:	... and more so for higher, than lower, *Task Complexity*	n/a
$R6$:	*Expertise* increases *Pair Performance*,	✓
$R6$+:	... and more so for higher, than lower, *Task Complexity*	✓
$R6_G$:	*Expertise* decreases *Pair Gain*,	
$R6_G$+:	... and more so for lower, than higher, *Task Complexity*	
$R7_I$:	*Task Complexity* decreases *Individual Performance*	n/a
$R7$:	*Task Complexity* decreases *Pair Performance*	✓

6.3 Exploratory Analysis

For the exploratory analysis, we used *jmp*, which implements decision tree analysis. As in regression analysis, the starting point of a decision tree analysis is a dependent variable and a set of independent variables. The independent variables of our analysis were the same as those of the confirmatory analysis. However, mediator variables cannot be modeled straightforwardly in decision tree analysis. Gain was therefore expressed as a separate dependent variable, *Pair Gain*, whose indicators are the differences between each *Pair Performance* indicator and the corresponding *Individual Performance* indicator (Fig. 6).[10] As for *Pair Performance* and *Individual Performance* (Fig. 3), the measurement model for *Pair Gain* is not validated and issues regarding formative versus reflective definitions are left for future work. We included *Country* in the exploratory analysis to check for differences across countries.

Decision tree analysis is an iterative process that successively splits the original n observations in a dependent variable in halves, thus creating a binary tree structure. Fig. 7 shows the resulting decision tree for *Correctness Gain*. Any split is associated to an independent variable such that all observations in one partition are less in the independent variable than all observations in the other partition (for ordinal or continuous independent variable), or according to categories (for categorical independent variables). Each split

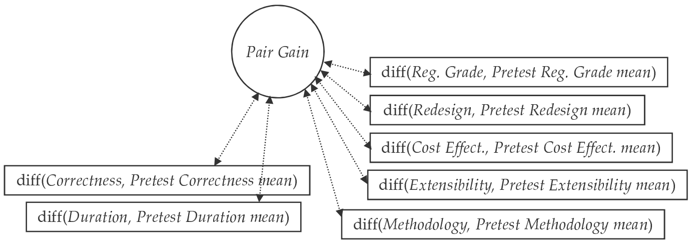

Pair Gain
diff(Reg. Grade, Pretest Reg. Grade mean)
diff(Redesign, Pretest Redesign mean)
diff(Cost Effect., Pretest Cost Effect. mean)
diff(Extensibility, Pretest Extensibility mean)
diff(Methodology, Pretest Methodology mean)
diff(Correctness, Pretest Correctness mean)
diff(Duration, Pretest Duration mean)

Fig. 6. Measurement Model—*Pair Gain*.

[10] This moves a part of the model's structure to a single variable. This is generally regarded as an inferior modeling solution [25], [24], partly because two sources of variance are collapsed into one.

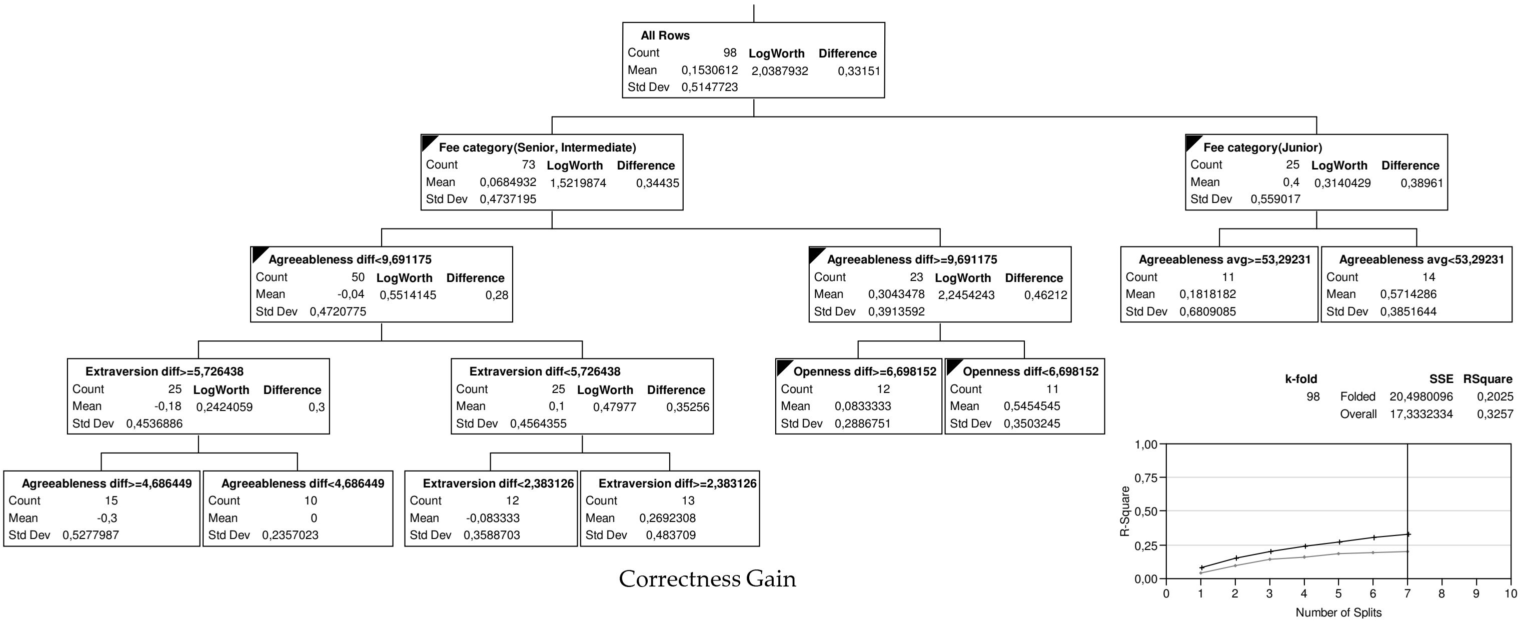

All Rows
Count 98 LogWorth Difference
Mean 0,1530612 2,0387932 0,33151
Std Dev 0,5147723
Fee category(Senior, Intermediate)
Count 73 LogWorth Difference
Mean 0,0684932 1,5219874 0,34435
Std Dev 0,4737195
Fee category(Junior)
Count 25 LogWorth Difference
Mean 0,4 0,3140429 0,38961
Std Dev 0,559017
Agreeableness diff<9,691175
Count 50 LogWorth Difference
Mean -0,04 0,5514145 0,28
Std Dev 0,4720775
Agreeableness diff>=9,691175
Count 23 LogWorth Difference
Mean 0,3043478 2,2454243 0,46212
Std Dev 0,3913592
Agreeableness avg>=53,29231
Count 11
Mean 0,1818182
Std Dev 0,6809085
Agreeableness avg<53,29231
Count 14
Mean 0,5714286
Std Dev 0,3851644
Extraversion diff>=5,726438
Count 25 LogWorth Difference
Mean -0,18 0,2424059 0,3
Std Dev 0,4536886
Extraversion diff<5,726438
Count 25 LogWorth Difference
Mean 0,1 0,47977 0,35256
Std Dev 0,4564355
Openness diff>=6,698152
Count 12
Mean 0,0833333
Std Dev 0,2886751
Openness diff<6,698152
Count 11
Mean 0,5454545
Std Dev 0,3503245
Agreeableness diff>=4,686449
Count 15
Mean -0,3
Std Dev 0,5277987
Agreeableness diff<4,686449
Count 10
Mean 0
Std Dev 0,2357023
Extraversion diff<2,383126
Count 12
Mean -0,083333
Std Dev 0,3588703
Extraversion diff>=2,383126
Count 13
Mean 0,2692308
Std Dev 0,483709
Correctness Gain
k-fold SSE RSquare
98 Folded 20,4980096 0,2025
Overall 17,3332334 0,3257
1,00
0,75
0,50
0,25
0,00
R-Square
0 1 2 3 4 5 6 7 8 9 10
Number of Splits

Fig. 7. Decision Tree for *Correctness Gain*.

is chosen as the one that maximizes some split criterion. In our case, that criterion is to maximize the statistical significance of the resulting difference in the dependent variable. In order to obtain a reasonable robustness toward outliers, and following [13], the process was set to terminate when partition sizes went below 10.

Decision tree analysis is independent of any assumptions on normality or types of data. Splits nearer the root split more of the observations and signify more general effects than splits further away from the root. Nonlinear effects are reflected by successive and asymmetrical splits (in the sense of producing an unbalanced tree) with respect to the same independent variable. Interaction effects are reflected by asymmetrical splits of different variables.

In Fig. 7, significant splits are marked with black triangles in the children nodes.[11] The tree should be read as follows (when focusing on significant splits): The main effect for *Correctness gain* was in the *Fee Category* variable, where *juniors* achieved the largest gain. Thus, *intermediates* and *seniors* had a significantly lower gain when paired, especially those with lower *Agreeableness difference*. Of the latter, less *Openness difference* was associated with greater gain.

We developed similar trees for the other *Pair Gain* indicators. (The analysis for *Duration Gain* excluded two outliers as defined by Mahalanobis distance, jacknife distance, and T^2 statistics.) These trees are summarized in Table 5 under the "All" columns. The numbers indicate the split sequence in the tree (a '*' indicates a significant split). As an example, consider the tree for *Correctness Gain* in Fig. 7. This tree is summarized in Table 5 under the column for *Correctness Gain* "All": The first split was on *Fee Category* and was significant. The second split was on *Agreeableness difference* (significant), the third on *Openness difference* (significant), the fourth and fifth on *Extraversion difference* (non-significant), the sixth on *Extraversion average* (non-significant), and the seventh on *Agreeableness difference* (non-significant).

A study of the "All" columns reveals that personality factors do not dominate among the significant effects on the *Pair Gain* indicators, except for on *Regression Grade Gain*.

We also analyzed exploratory models where the personality traits were the only independent variables, with and without *Country* (columns "*P+C*" and "*P*", respectively, in Table 5), as well as the direct effects of all independent variables on the *Pair Performance* indicators (column "*P.P.*") in Table 5. At the bottom of each column, the R^2 of an n-fold crossvalidation is given, that is, the average R^2 over n predictions where $n-1$ observations are used to predict the nth observation. And at the very bottom, the overall R^2, which indicates the ratio of explained variance, is given. Effect sizes for the personality traits were more comparable to those of the other predictors than they were in the linear models of the confirmatory analyses. This may signal the inappropriateness of describing personality effects by linear models.

From Table 5 one can note that in the absence of *Fee Category* and *Control Style* (and *Country*), main effects of personality manifest themselves significantly in all models (except for *Duration*). This can be seen clearly in columns "*P**" and "*P+C**" versus column "All*" of Table 6 which aggregates the findings in Table 5 per model type. Table 6 ranks each independent variable according to how early associated splits occurred, by the formula *max splits + 1 - split number*. In our case, *max splits* was seven, that is, no trees had more than seven splits. Thus, if, say, *Fee Category* assumed a significant second split in a model in which only personality factors were included, then that split would contribute 7+1-2=6 to the ranking for *Fee Category* in the "*P*" columns. The '*'-columns only count significant splits.

The two most prevalent personality influences in the absence of other predictors, are *Extraversion difference* and *Conscientiousness mean*. In fact, *Extraversion difference* is the most general significant predictor as ranked over all exploratory models. Note that *Extraversion difference* was not included in the confirmatory models of Section 6.2.

One of the important questions for pair programming,

[11] The "LogWorth" index in each parent node is a significance measure of the difference in mean values for the observations in each child node with regards to the dependent variable. Specifically, LogWorth = $-\log_{10}(p)$, where p is the adjusted probability of the observed data under the hypothesis of the means being equal. Thus, p-values less than 0.05 are reflected by LogWorth-values greater than 1.30. The adjusted p-value takes into account the number of different ways splits can occur. It is fair compared to the unadjusted p-value and to the Bonferroni p-value, see [93].

TABLE 5
Exploratory Analysis

	Correctness				*Duration*				*Methodology*				*Extensibility*				*Cost Effectiveness*				*Redesign*				*Regression Grade*			
	Gain			*P.P.*	*Gain*			*P.P.*	*Gain*			*P.P.*	*Gain*			*P.P.*	*Gain*			*P.P.*	*Gain*			*P.P.*	*Gain*			*P.P.*
	All	*P*	*P+C*		All	*P*	*P+C*		All	*P*	*P+C*		All	*P*	*P+C*		All	*P*	*P+C*		All	*P*	*P+C*		All	*P*	*P+C*	
Fee Category	1*				2*			1*2*				4	2							5				5	3			
Control Style				4					1*			1*	1*			1*	1*			1*	1*			1*	5 7			
Country					1 5		1 4		2*		2	2*5			6	3	2*5		1*	2*	2*4*		5	2*3*7	6		5	
Extraversion avg	6				3	1	2										6		2*	6						6 7	6	
Extraversion diff	4 5				4		3		3*7	1*4*	1*6	3*6		2*4*	2*4*				4 5	3	3	1*3*	1*3*	4*6	1*	1*	1*	1*
Agreeableness avg													3 4 7					7										7
Agreeableness diff	2*7	3	3	1																		6	6					
Conscientiousness avg		4 6 7	4 6 7					4 5	5 6*	3 6	3 5 7			1*5 6	1*5	2*7	3	3	3		6	4*5	4*		4	3	3	
Conscientiousness diff				3		4 6				2 5	4		6	3*	3*	5		1*				2*	2*		2*	2*	2*	2*
Emotional Stability avg						2 5													7								7	6
Emotional Stability diff						3										6		6								5		4 5
Openness avg		2*	2*	5					4	7								4 5	6		5 7*					4	4	3
Openness diff	3*	1*5	1*5	2			5	3*					5	7	7	4	4	2		4 7								
n-Folded R^2	.20	.32	.32	.22	.21	.19	.13	.24	.31	.27	.19	.21	.35	.29	.29	.41	.23	.27	.19	.22	.60	.32	.31	.63	.20	.24	.21	.19
Overall R^2	.33	.43	.43	.31	.32	.33	.26	.34	.42	.39	.32	.32	.45	.41	.40	.51	.34	.38	.33	.35	.66	.41	.40	.68	.32	.36	.34	.33

however, is whether it pays off to pair up. Thus, we focus on *Pair Gain*. Moreover, a main objective was to compare the effects of personality with other readily available predictors. In concluding our exploratory analysis, we therefore focus on the full model for *Pair Gain* (the "All*" column in Table 6). Table 7 summarizes the exploratory analysis from this perspective, on the conceptual level, where the predictors are ranked in order of decreasing impact.

7 Discussion

Our confirmatory analyses did not support the proposed models of Section 6.2. Our exploratory analysis showed that the three main effects in our data were due to indicators of *Task Complexity*, *Country*, and *Expertise*, while *Extraversion difference* was the strongest personality trait factor. Personality effects were sometimes found as interaction effects to the main effects. However, the effects of personality were not consistent, and it is hard to propose a model that incorporates personality as predictors of *Pair Performance* or *Pair Gain*. In fact, if one omits personality from the multivariate model (Fig 5) of the confirmatory analysis in Section 6.2, one obtains better fit indices and comparable R^2 values than for the original model.

7.1 Implications

The strongest predictor among the personality factors was *Extraversion Variability*. However, our findings suggest that there are stronger predictors than personality for *Pair Performance* and *Pair Gain*, even when nonlinearity is catered for in the analysis. This entails that if there are limited resources available for screening employees one should concentrate on e.g., matching expertise and task complexity rather than allocating human resources based on personality tests.

Our study only investigated short-term effects over a single day. Some group dynamics evolve and stabilize over time. Moreover, factors that influence group dynamics might manifest themselves only after days, or months of collaboration (see e.g., [40]). It is therefore possible that longitudinal studies might uncover more persistent findings regarding the effects of personality.

Studies suggest that pair programming sessions should be kept short (e.g., 1.5–4 hours) [15], [109], and proponents of pair programming advocate that pairs should rotate partners routinely [115]. In this sense, pair programming is an extreme form of collaboration, and one that perhaps does not give time for the effects of more subtle influences to emerge.

We therefore see our findings as relevant to pair programming, but acknowledge that personality may well manifest itself stronger in longer studies. However, our findings confirm the body of evidence from other disciplines that suggests that personality is only a moderate predictor for performance.

7.2 Threats to Validity

Every empirical study suffers from methodological shortcomings. This section discusses some of the most pressing issues in this respect.

7.2.1 Statistical Conclusion Validity

We conducted univariate and multivariate linear analyses in order to replicate the findings of univariate linear analyses

TABLE 6
Exploratory Analysis Aggregated

	All*	All	*P**	*P*	*P+C**	*P+C*	*P.P.**	*P.P.*	Total*	Total
Fee Category	13	24	0	0	0	0	13	23	26	47
Control Style	28	32	0	0	0	0	28	32	56	64
Country	22	37	0	0	7	32	23	32	52	101
Extraversion mean	0	16	0	9	6	14	0	2	6	39
Extraversion diff	12	33	40	40	36	50	16	25	104	148
Agreeableness mean	0	10	0	1	0	0	0	1	0	12
Agreeableness diff	6	7	0	7	0	7	0	7	6	28
Conscientiousness mean	2	16	11	43	11	40	6	14	30	113
Conscientiousness diff	6	8	24	39	17	21	6	14	53	82
Emotional Stability mean	0	0	0	9	0	2	0	2	0	13
Emotional Stability diff	0	0	0	10	0	0	0	9	0	19
Openness mean	1	8	6	18	6	12	0	8	13	46
Openness diff	5	12	7	17	7	14	5	20	24	63

TABLE 7
Exploratory Analysis Conclusion

Rank	Predictor
1	*Task Complexity*
2	*Country*
3	*Expertise*
4	*Extraversion Variability*
5	*Agreeableness Variability*
6	*Conscientiousness Variability*
7	*Openness Variability*
8	*Conscientiousness Elevation*
9	*Openness Elevation*
10	*Extraversion Elevation*
11	*Agreeableness Elevation*
12	*Emotional Stability Elevation*
13	*Emotional Stability Variability*

common in personality research. Univariate analyses only show relationships between one independent variable and a dependent variable at a time disregarding the levels of other independent variables. For example, an overall positive effect of a variable may actually be negative when seen at various levels of another variable even under acceptable collinearity [41]. Hence, our confirmatory univariate results (Section 6.2) must be read accordingly. Multiple regression solves this problem, but in our case (Section 6.2), it also introduced large models, the threat of overspecification, and the threat of multicollinearity. The large model was a result of our desire to test a model that expresses known empirical results. The threat of overspecification was thus a consequence of this and was part of the ensuing discussion of whether the model is suitable. Multicollinearity was avoided by excluding correlated indicators for the same trait. Other than that, the independent variables are not known to correlate substantially. Several of the indicators are well-known to have non-normal population distributions. We used the possibilities of generalized linear models, bootstrapping and Bayesian estimation to cater for non-normality as far as possible. The exploratory analysis (Section 6.3) used regression trees, which lose power at each successive level. Our counter measure was to stop splitting when partition sizes decreased below a recommended limit. Finally, reliability is unknown for the indicators for which a single expert made subjective assessments.

7.2.2 Internal Validity

Internal validity pertains to conclusions of causation. In our setting, internal validity can be threatened if relations have the wrong direction or if an observed correlation is interpreted as a causation but where both variables are in fact influenced by a third variable not in the model. Our models are sound in many of these respects, but it may be that certain relations are missing; for example, *Personality* indicators may well influence the *Expertise* indicator. However, our choices of models were directed by our overall research goals (examining the effects of personality and other factors on performance), and we leave possibly better specified analysis models to future research.

7.2.3 External Validity

External validity concerns the degree to which conclusions drawn on the specific variables of an empirical study are valid for variations of those variables [95]. Together with construct validity, external validity determines how well a study may be generalized to other situations within the intended theoretical or practical scope. For our study, external validity pertains especially to the situational variables, and thus indirectly to the independent variables that are associated with them. We assume that the subjects were representative for developers in the three countries and that the study's results are robust over variations of subjects, and hence, variations in indicators of *Personality* and *Expertise*, in these three countries. We base this on the reasonably large sample size of subjects and companies and to the relatively high degree of randomness in both the selected companies and the developers within the companies. It is not clear to us how robust any of these variables are in countries with substantially different corporate cultures from those of our study.

This study has a specific operationalization of the aggregate construct of pair programming, which, in our opinion, represents a real-life pair-programming situation. However, the results for this operationalization may fail to transfer to variations on this operationalization, e.g., to longer pair programming sessions in which longer-term group dynamics, such as *pair jelling* [118], might be influential, or where subjects are not dissuaded to engage in distracting activities.

7.2.4 Construct Validity

Construct validity concerns the degree to which the specific variables of an empirical study represent the intended constructs in the conceptual or theoretical model [95]. Our choices of constructs and indicators were presented in Section 5.1 and in Section 6.3.

Although we have operationalized the *Expertise* construct in a way that is common in industry and that is related to common operationalizations in a wide range of research, it is debatable whether this construct is reliable and valid. For our data, variability within each *Fee Category* is larger than between categories.[12] The indicators that were used in this study are not validated. That is why we analyzed each indicator separately, even though we try to summarize our findings at the construct level. It is not clear that our *Performance* construct is the same as the concepts of performance of the meta-analyses that we referred to when presenting the conceptual models. However, at the current level of knowledge we postulate that they are related. The connotative definition of *Performance* irrespective of operationalizations and decisions regarding reflective and formative definitions in terms of indicators, are crucial unresolved issues. Assuming for the moment a formative stance where indicators represent different aspects of a construct, it is important to acknowledge that our indicators are just that: aspects of the construct, and not by themselves a full representative for the construct. Task complexity, performance and expertise are interrelated concepts, and our *Task Complexity* construct also has short-comings that are under investigation. This also relates to whether the tasks that were administered are suitable to measure *Performance*. Nevertheless, our indicators can reasonably be said to represent the intended constructs.

The Lexical Hypothesis implies that different language versions of the Big Five indicators may not be equivalent. This means that differences observed in *Country* with regards to *Personality* may partly be due to such non-equivalence. The different language versions in our study are validated extensively to minimize this threat.

8 Conclusion

Personality may be a valid predictor for long-term team performance. However, we found no strong indications that personality affects pair programming performance or pair gain in a consistent manner, especially when including predictors pertaining to expertise, task complexity, and country.

In the short term, we therefore think that it is worth for industry and research alike to focus on other predictors of performance, including expertise and task complexity. Of particular interest might be relevant factors that are under current investigation, such as team learning, team motivation, and programming skill. In contrast to personality traits, which are fixed in a person, malleable factors such as

[12] Work is ongoing to develop better indicators for *Expertise* that are more reliable and that are on an interval scale rather than ordered categorical. Work is also ongoing to develop a better *Performance* construct.

TABLE 8

Descriptive Data n=98, n=80 for *Duration* (*Correctness*=1)

	min	max	mean	stddev
Dependent variables				
Correctness	0	1	0.82	0.39
Duration	30	163	62.59	24.37
Methodology	9	15	13.00	1.33
Extensibility	9	15	13.20	1.35
Cost effectiveness	3	15	12.49	2.14
Redesign	3	11	5.83	1.48
Regression grade	0	9	7.19	1.95
Independent variables				
Extraversion				
mean	32.84	63.91	50.00	7.35
minimum	25.3	63.68	44.74	8.58
maximum	37.77	79.23	55.26	8.46
difference	0.02	30.29	7.43	6.10
Agreeableness				
mean	35.1	69.39	50.00	6.75
minimum	14.03	65.08	44.08	8.38
maximum	38.58	74.9	55.92	7.75
difference	0.01	29.96	8.37	6.25
Conscientiousnes				
mean	24.89	69.27	50.00	7.81
minimum	22.64	67.96	45.10	8.55
maximum	27.13	79.85	54.90	8.91
difference	0.21	25.94	6.93	5.52
Emotional Stability				
mean	29.32	66.58	50.00	8.13
minimum	16.31	65.33	45.15	9.80
maximum	35.98	70.25	54.85	7.58
difference	0.05	19.17	6.85	4.61
Openness				
mean	21.21	71.37	50.00	7.68
minimum	11.23	67.4	44.89	9.39
maximum	31.18	75.33	55.11	7.74
difference	0.1	25.84	7.23	5.51
Fee Category n:	*junior*=25, *intermediate*=35, *senior*=38			
Control Style n:	*centralized*=51, *delegated*=47			
Country n:	*Norway*=41 *Sweden*=28, *UK*=29			

TABLE 9

Univariate Analyses, n=98, n=80 for *Duration* (*Correctness*=1)

	Correctness	*Duration*	*Methodology*	*Extensibility*	*Cost effect.*	*Redesign*	*Reg. grade*
	e^b	$b\ \Gamma$	b	b	b	b	b
Extraversion							
mean R1E	.999	.011*	-.016	-.023	-.048	.046*	-.042
minimum	1.007	.005	-.008	-.009	-.040	.018	-.045
maximum	.992	.011*	-.017	-.026	-.031	.051**	-.017
difference (*R1V*)	.980	.006	-.012	-.023	.013	.045	.040
Agreeableness							
mean R2E	1.042	.003	-.021	.009	-.026	-.006	.023
minimum	.998	.003	-.013	.006	-.016	-.003	.008
maximum	1.069†	.000	-.017	.006	-.021	-.006	.025
difference R2V	1.100	-.003	-.002	-.002	-.003	-.003	.017
Conscientiousness							
mean R3E	1.019	.004	-.015	-.055**	-.064*	.002	.000
minimum	1.031	-.003	-.016	-.047**	-.063*	.016	.007
maximum	1.001	.008	-.008	-.041**	-.040	-.011	-.006
difference R3V	.955	.016*	.011	.004	.033	-.047†	-.023
Emotional Stability							
mean R4E	1.026	-.008†	-.021	-.020	-.024	.023	.013
minimum	1.019	-.004	-.010	-.016	-.021	.019	-.001
maximum	1.029	-.012*	-.031	-.018	-.021	.022	.031
difference R4V	.995	-.007	-.027	.017	.028	-.017	.062
Openness							
mean R5E	1.030	-.013**	.002	.008	.022	.017	.052*
minimum	1.034	-.010*	.002	.016	.025	.009	.034
maximum	1.009	-.012**	.001	-.006	.006	.019	.051*
difference R5V	.950	.002	-.003	-.041†	-.043	.009	.002
Fee Category R6 χ^2:	.162	26.827**	1.417	1.713	3.231	1.429	2.367
junior–intermediate	.040	20.191**	-.131	-.035	-.266	-.345	-.644
junior–senior	.020	30.140**	-.384	-.381	-.907	.036	-.715
intermediate–senior	-.020	9.949†	-.253	-.346	-.641	.381	-.071
Control Style R7 χ^2:	.110	.187	8.965**	53.821**	15.785**	113.688**	.978
centralized–delegated	.100	4.105	-.768**	-1.603**	-1.588**	2.155**	.387
Country χ^2:	1.109	6.298*	1.880	.535	5.341	.621	1.175
Norway–Sweden	.100	4.105	-.025	-.123	-.799	-.178	-.217
Norway–UK	.030	-11.550	-.408	-.123	-1.102	.126	-.508
Sweden–UK	-.080	-15.655*	-.383	.260	-.303	.304	-.291

**p* < 0.05 *(two-tailed),* ***p* < 0.01 *(two-tailed),* †*p* < 0.05 *(one-tailed). Effects for the categorical variables are differences in means (Bonferroni-adjusted p-values).*

learning, motivation and skill lend themselves to programs of improvement.

In the long term, we think it is worthwhile to analyze the relationships between personality and pair collaboration in more detail. In particular, using collaboration as a mediator variable might reveal larger effects of personality. Analyzing the nature of collaboration would focus on the idea-generating process prior to the selection of a solution. This idea-generating stage may take the form of any of Steiner's task types. The form this stage takes may depend on the nature of collaboration within a pair, and this in turn might depend on personality. For example, a certain personality combination in a pair might enforce the idea-generating process as additive. Thus, the exact way to characterize a pair's joint personality may in fact depend on the individual personalities. The investigation of such reciprocal effects of personality (and also nonlinear effects) demands more advanced analysis methods (e.g., path analysis) than those that are commonly used at present.

Acknowledgements

The authors are grateful to Tore Dybå for compiling the initial set of articles for the review of related work, and to Guri Bollingmo, Gordon Cheung, David Howell, David Matheson, Gerty Lensvelt-Mulders, Ed Rigdon, and Stas Kolenikov for helpful remarks and pointers. The authors wish to thank the anonymous referees for detailed and insightful comments.

Appendix

This appendix contains descriptive data and the details of the statistical analyses underlying the results of Section 6.2.

Table 8 gives the descriptive data. Note that *Duration* is considered only for *Correctness* = 1 (correct solutions).

Table 9 shows the univariate analyses. In reaching the conclusion in Table 3, we considered all three indicators for *Elevation*. For example, the hypothesized relationship *R2E* is supported significantly at *Correctness*. One-tailed significance is applicable for the hypothesized relations. Otherwise, two-tailed significance is applicable. Logistic regression was applied for *Correctness* and the results are presented in terms of odds ratios (e^b). A generalized linear model (gamma log link) was applied for *Duration* (b Γ). Linear regression was applied for the other indicators. Appropriate variants were applied for the categorical independent variables.

Table 10 shows the multivariate analysis. Peeters et al. [88] report that trait elevation and trait variation are negatively correlated in general, but only significantly for agreeableness and conscientiousness. Note also that there are inherent correlations between dummy variables representing the same measure. There are also correlations between interaction terms and their components. All these correlations were modeled, even though they are omitted from Fig. 5 and Table 10 in order to avoid clutter.

The parameters of path models are estimated by a maximum likelihood algorithm that maximizes a fit index. Path analysis engines also output powerful model fit indices, and they cater for categorical dependent variables (by the way of Bayesian estimation) and data that violates multivariate normality (by the way of bootstrapped estimates).

The path weights in Table 10 are interpreted as multiple regression coefficients, that is, a weight shows the influence of an independent variable on the dependent variable, given that all other independent variables are held constant at any level. The path weights are computed using the Bayesian Metropolis-Hastings MCMC algorithm [79], [59] in AMOS (convergence criterion < 1.007, tuning parameter 0.35), which

TABLE 10

Multivariate Analyses, n=98, n=80 for *Duration* (*Correctness*=1)

	Correctness	*Duration*	*Methodology*	*Extensibility*	*Cost effect.*	*Redesign*	*Reg. grade*
R0	-.042	.784†‡	.096	.146	.358	-.206	.683†‡
Extraversion mean							
$R1E_I$	-.003	.380	-.002	-.006	-.012	-.002	-.022
$R1E_P$	-.003	.181	-.001	-.008	-.030	-.023	-.023
R1E	-.003	.479	-.001	-.009	-.035	-.022	-.039
$R1E_G$	.000	.099	.001	-.003	-.022	-.020	-.016
Agreeableness mean							
$R2E_I$	-.001	.212	-.011	-.021	-.002	.001	-.035
$R2E_P$	.013	-.509	-.041	.002	-.033	.036	.023
R2E	.013	-.343	-.042	-.001	-.034	.036	.000
$R2E_G$	.014	-.556	-.031	.020	-.032	.035	.035
Agreeableness mean **X** *Fee Category*							
$R2E1_I$	.000	-.295	.002	-.002	-.021	.006	.043
$R2E1_P$	-.020	.547	-.002	-.035	-.051	-.012	-.045
R2E1	-.020	.319	-.002	-.035	-.059	-.014	-.017
$R2E1_G$	-.021	.613	-.004	-.033	-.038	-.020	-.060
$R2E2_I$	.027	.021	.030	.059	.035	-.004	.078
$R2E2_P$	.013	-.177	.036	.038	.055	-.049	.030
R2E2	.012	-.158	.039	.047	.067	-.049	.083
$R2E2_G$	-.015	-.179	.010	-.012	.033	-.045	.005
Agreeableness difference							
$R2V_I$	-.001	-.042	.000	-.004	-.004	-.005	-.001
$R2V_P$	.011	-.559	-.014	-.001	-.016	.001	.024
R2V	.011	-.593	-.014	-.001	-.018	.002	.024
$R2V_G$	.012	-.551	-.015	.003	-.014	.007	.025
Conscientiousness mean							
$R3E_I$	-.023	.330	-.045	-.061	-.071	.031	-.047
$R3E_P$	.012	-1.168	-.002	.011	.062	-.001	-.078
R3E	.013	-.907	-.007	.002	.036	-.007	-.110
$R3E_G$	.036	-1.238	.038	.064	.107	-.038	-.063
Conscientiousness mean **X** *Fee Category*							
$R3E1_I$	.019	-.055	.074§	.071§	.078	-.048	.040
$R3E1_P$	-.011	1.281	-.014	-.048	-.117	.011	.099
R3E1	-.012	1.238	-.006	-.038	-.089	.021	.126
$R3E1_G$	-.031	1.293	-.080	-.109	-.167	.069	.086
$R3E2_I$	.016	-.013	.024	.053	.051	-.013	.050
$R3E2_P$	-.015	1.142	-.003	-.091	-.155	-.044	.049
R3E2	-.016	1.133	-.001	-.083	-.137	-.041	.083
$R3E2_G$	-.032	1.146	-.025	-.136	-.188	-.028	.033
Conscientiousness difference							
$R3V_I$	-.011	.394	-.014	-.026	-.031	.004	.000
$R3V_P$	-.010	.956†	.013	-.021	.022	-.009	-.022
R3V	-.010	1.263†‡	.011	-.025	.011	-.009	-.021
$R3V_G$	.002	.868	.025	.001	.042	-.013	-.022
Emotional Stability mean							
$R4E_I$	-.001	-.512	.012	.022	.014	-.016	-.013
$R4E_P$	-.013	-1.205	-.015	-.010	-.005	-.012	.087
R4E	-.013	-1.607	-.014	-.006	.000	-.008	.079
$R4E_G$	-.012	-1.095	-.026	-.028	-.014	.008	.091
Emotional Stability mean **X** *Fee Category*							
$R4E1_I$	.009	.537	-.036	-.052	-.027	.042	.020
$R4E1_P$	.034†	.695	-.031	.010	-.006	.014	-.080
R4E1	.034†‡	1.120	-.034	.002	-.016	.005	-.067
$R4E1_G$	.025	.583	.002	.054	.012	-.037	-.086
$R4E2_I$	.008	-.072	-.007	-.023	.003	.026	.020
$R4E2_P$	.018	.496	.007	-.019	-.020	.018	-.075
R4E2	.018	.441	.006	-.023	-.019	.013	-.061
$R4E2_G$	.010	.512	.013	.000	-.022	-.013	-.081
Emotional Stability difference							
$R4V_I$	.010	-.388	.036	.051	.019	-.016	-.001
$R4V_P$	-.010	-2.448	-.002	-.032	.026	-.019	.126
R4V	-.011	-2.752	.001	-.024	.033	-.016	.126
$R4V_G$	-.021	-2.364	-.035	-.075	.013	.000	.127
Emotional Stability difference **X** *Fee Category*							
$R4V1_I$	-.011	.546	-.023	-.036	.006	.013	.010
$R4V1_P$	.004	1.876	-.103	.014	-.123	.002	-.087
R4V1	.005	2.307	-.105	.009	-.120	-.001	-.081
$R4V1_G$	.016	1.762	-.082	.045	-.126	-.013	-.090
$R4V2_I$	-.030	.142	-.021	-.058	-.017	.001	-.018
$R4V2_P$	.006	2.183	-.023	.018	-.005	-.001	-.094
R4V2	.008	2.293	-.025	.009	-.011	-.001	-.106
$R4V2_G$	.038	2.152	-.004	.067	.006	-.002	-.088
Openness mean							
$R5E_I$	.007	-.027	.025	.025	.022	-.022	.020
$R5E_P$	.001	-.134	.023	.025	.040	-.015	.038
R5E	.001	-.156	.025	.029	.048	-.010	.052
$R5E_G$	-.006	-.129	.000	.004	.026	.012	.032
Openness difference							
$R5V_I$	.002	-.979	.003	-.014	.012	-.025	.037
$R5V_P$	.010	1.376†	.045	.029	.077	.007	.189
R5V	.010	.607	.045	.027	.081	.012	.215
$R5V_G$	.008	1.586	.042	.041	.069	.037	.178
Openness difference **X** *Fee Category*							
$R5V1_I$	-.002	.613	.040	.031	.004	-.013	-.024
$R5V1_P$	-.036†	.205	-.044	-.053	-.089	-.016	-.200†
R5V1	-.035†	.685	-.040	-.048	-.088	-.013	-.217†‡
$R5V1_G$	-.034	.072	-.080	-.079	-.092	.000	-.193
$R5V2_I$	.000	1.156	.001	.009	-.002	.008	-.032
$R5V2_P$	-.025	-2.179	.007	-.116†	-.150	-.088	-.263†‡
R5V2	-.025	-1.272	.008	-.115†	-.151	-.090	-.285†‡
$R5V2_G$	-.025	-2.428	.006	-.124	-.149	-.098	-.252
Fee Category							
$R61_I$	-1.094	-35.024	-1.375	-.110	-.582	-.374	-4.850
$R61_P$	-.003	-160.654†	3.182	4.087	10.489	-.519	2.615
R61	.038	-188.410†	3.062	4.075	10.290	-.422	-.640
$R61_G$	1.133	-153.386	4.438	4.186	10.872	-.048	4.210
$R62_I$	-2.107	-20.543	-1.814	-3.783	-3.752	-.577	-6.741
$R62_P$	-.637	-89.137	-2.697	4.928	7.685	5.434	1.615
R62	-.562	-105.483	-2.883	4.402	6.345	5.572	-2.971
$R62_G$	1.546	-84.940	-1.069	8.185	10.096	6.149	3.770
Fee Category **X** *Control Style*							
R611	.100	14.303	.147	-.429	-.706	-.222	1.998†
R621	-.104	9.228	1.626†‡	-.923	-.657	-1.506†‡	2.274†
Control Style							
R7	.025	-20.657	.224	2.038	1.976	-1.762	-2.385†‡

$^*p < 0.05$ (two-tailed), $^{\dagger}p < 0.05$ (one-tailed), $^{\S}p < 0.05$ (two-tailed bootstrap ML), $^{\ddagger}p < 0.05$ (one-tailed bootstrap ML).

handles categorical data. All prior distributions were set to uniform (diffuse) which conceptually, reduces Bayesian estimation to classical estimation and gives path weights that are close to classical maximum likelihood (ML) algorithms [72]. To handle non-normality, we also computed path weights using ML estimation with bootstrapping. These weights and 95% confidence intervals were very similar to the Bayesian weights and 95% credibility intervals. We include p-values for both estimation techniques in Table 10.[13]

Table 11 gives goodness of fit indices. The optimal Bayesian Posterior Predictive p-value is 0.5, with values on either side approaching 0 or 1 indicating inferior fit. Our obtained values of about 0.05 indicate a moderate fit. The Deviance Information criterion (DIC) and Effective parameters indices are not absolute fit measures but are provided for future model comparison [72]. For the ML Bootstrap estimation, the R^2s for the mediator (*Individual Performance*) and dependent variable (*Pair Performance*) show the proportion of variance accounted for by the analysis models. The χ^2 statistic measures the discrepancy between the observed data and the corresponding values predicted by the model. The associated p-value signifies the probability of the observed data under the null hypothesis of this discrepancy being zero. Thus, higher p-values signify better fit. The Bollen-Stine bootstrap estimation [12] adjusts these statistics in the event of non-normal data. Thus, the expected χ^2 value (the average of χ^2 values over all bootstrap iterations) is seen to be higher than the final χ^2 value. This results in an upward-adjusted (Bollen-Stine) p-value relative to the standard p-value that assumes multivariate normality. Also provided, is the Root Mean Square Error of Approximation (RMSEA) which is a parsimony-adjusted index in the sense that it favors models with fewer parameters [14]. Lower RMSEA values signify better fit, where RMSEA values less than or equal to 0.05 indicates close approximate fit, values between 0.05 and 0.08 indicates reasonable error of approximation, and values above 0.1 suggest poor fit [14], [70]. The Akaike Information Criterion (AIC) is also a parsimony-adjusted index and is useful when comparing contesting models, and lower values are favorable. The Comparative Fit Index (CFI) is an index that compares the proposed model with the model where no variables are related (the independence model). Higher CFI values are favorable, and values above 0.9 indicate reasonable fit [63], [70].

TABLE 11

Multivariate Analyses Fit Measures

	Correctness	*Duration*	*Methodology*	*Extensibility*	*Cost effect.*	*Redesign*	*Reg. grade*
Fit Measures (Bayesian)							
Post Predictive p	.040	.070	.040	.030	.040	.040	.030
DIC	640.630	629.910	644.330	649.770	642.790	644.490	643.970
Effective parameters	154.650	138.810	154.390	153.440	153.890	155.290	154.630
Fit Measures (ML Bootstrap)							
R^2 Mediator	.339	.407	.297	.329	.331	.278	.268
R^2 Dependent	.268	.717	.312	.555	.335	.696	.399
χ^2 df=174	218.279	208.486	218.273	219.550	218.371	218.131	221.027
p	.013	.038	.013	.011	.013	.012	.009
expected χ^2	217.044	235.846	215.688	216.119	216.286	215.822	216.570
Bollen-Stine p	.451	.640	.444	.438	.445	.449	.421
RMSEA	.051	.050	.051	.052	.051	.051	.053
AIC	624.279	614.486	624.273	625.550	624.371	624.131	627.027
CFI	.986	.987	.986	.986	.986	.987	.986

REFERENCES

[1] M. Ally, F. Darroch, and M. Toleman, “A framework for understanding the factors influencing pair programming success,” in *Proc. XP 2005*. Springer-Verlag, 2005, pp. 82–91.

[13] In the Bayesian approach, unknown population parameters are viewed as random, rather than fixed as in the classical approach. Thus, p-values and credibility intervals for Bayesian estimated path weights are statements about the actual unknown population distribution, rather than statements about the likelihood of present data under a null hypothesis. However, in our case we use the Bayesian estimates classically.

[2] E. Arisholm, H. Gallis, T. Dybå, and D.I.K. Sjøberg, "Evaluating pair programming with respect to system complexity and programmer expertise," *IEEE Trans. Software Eng.*, vol. 33, pp. 65–86, Feb. 2007.

[3] E. Arisholm and D.I.K. Sjøberg, "Evaluating the effect of a delegated versus centralized control style on the maintainability of object-oriented software," *IEEE Trans. Software Eng.*, vol. 30, pp. 521–534, Aug. 2004.

[4] E. Arisholm, D.I.K. Sjøberg, G.J. Carelius, and Y. Lindsjørn, "A web-based support environment for software engineering experiments," *Nordic J. Computing*, vol. 9, no. 4, pp. 231–247, 2002.

[5] S.B. Bacharach, "Organizational theories: Some criteria for evaluation," *Academy of Management Review*, vol. 14, no. 4, pp. 496–515, 1989.

[6] M.B. Barrick, M.K. Mount, and T.A. Judge, "Personality and performance at the beginning of the new millennium: What do we know and where do we go next?" *Int'l J. Selection and Assessment*, vol. 9, no. 1/2, pp. 9–30, 2001.

[7] M.R. Barrick, G.L. Stewart, M.J. Neubert, and M.K. Mount, "Relating member ability and personality to work-team processes and team effectiveness," *J. Applied Psychology*, vol. 83, no. 3, pp. 377–391, 1998.

[8] A. Basilevsky, *Statistical Factor Analysis and Related Methods: Theory and Applications*. John Wiley and Sons, Inc., 1994.

[9] K. Beck and C. Andres, *Extreme Programming Explained: Embrace Change*, 2nd ed. Addison-Wesley, 2003.

[10] S.T. Bell, "Deep-level composition variables as predictors of team performance: A meta-analysis," *J. Applied Psychology*, vol. 92, no. 3, pp. 595–615, 2007.

[11] K. Bollen and R. Lennox, "Conventional wisdom on measurement: A structural equation perspective," *Psychological Bull.*, vol. 110, no. 2, pp. 305–314, 1991.

[12] K.A. Bollen and R.A. Stine, "Bootstrapping goodness-of-fit measures in structural equation models," *Sociological Methods and Research*, vol. 21, pp. 205–229, 1992.

[13] L.C. Briand and J. Wust, "Modeling development effort in object-oriented systems using design properties," *IEEE Trans. Software Eng.*, vol. 27, no. 11, pp. 963–986, 2001.

[14] M.W. Browne and R. Cudeck, "Alternative ways of assessing model fit," in *Testing Structural Equation Models*, K.A. Bollen and J.S. Long, Eds. Sage Publications, 1993, pp. 136–162.

[15] S. Bryant, B. du Boulay, and P. Romero, "XP and pair programming practices," *J. Computer Society of India*, vol. 30, no. 5, pp. 17–20, 2007, extended version at `www.ppig.org/newsletters/2006-09/3-overview-xp.pdf` newsletter of the Psychology of Programing Interest Group (PPIG).

[16] C. Burke Jarvis, S.B. Mackenzie, and P.M. Podsakoff, "A critical review of construct indicators and measurement model misspecification in marketing and consumer research," *J. Consumer Research*, vol. 30, pp. 199–218, Sept. 2003.

[17] J.M. Burkhardt, F. Détienne, and S. Wiedenbeck, "Object-oriented program comprehension: Effect of expertise, task and phase," *Empirical Software Engineering*, vol. 7, no. 2, pp. 115–156, June 2002.

[18] J.N. Butcher, W. Dahlstrom, J.R. Graham, A. Tellegen, and B. Kaemmer, *Minnesota Multiphasic Personality Inventory-2 (MMPI-2): Manual for Administration and Scoring*. University of Minnesota Press, 1989.

[19] L. Cao and P. Xu, "Activity patterns of pair programming," in *Proc. 38th Annual Hawaii Int'l Conf. System Sciences*. IEEE Computer Society, 2005, pp. 1–10.

[20] R.B. Cattell, *The Description and Measurement of Personality*. Harcourt, Brace, & World, 1946.

[21] R.B. Cattell, *Personality and Motivation Structure and Measurement*. World Book, 1957.

[22] C.G. Cegielski and D.J. Hall, "What makes a good programmer?" *Comm. ACM*, vol. 49, no. 10, pp. 73–75, Oct. 2006.

[23] J. Chao and G. Atli, "Critical personality traits in successful pair programming," in *Proc. AGILE 2006*. IEEE Computer Society, 2006.

[24] G.W. Cheung, "Introducing the latent congruence model for improving the assessment of similarity, agreement, and fit in organizational research," *TOrganizational Research Methods*, vol. 12, pp. 6–33, 2009.

[25] G.W. Cheung, "A multiple-perspective approach to data analysis in congruence research," *Organizational Research Methods*, vol. 12, pp. 63–68, 2009.

[26] K.S. Choi, "A discovery and analysis of influencing factors of pair programming," Ph.D. dissertation, Faculty of New Jersey Institute of Technology, Department of Information Systems, 2004.

[27] A. Cockburn, *Agile Software Development*. Addison-Wesley, 2002.

[28] B.P. Cohen, *Developing Sociological Knowledge: Theory and Method*, 2nd ed. Nelson-Hall Publishers, 1989.

[29] L.L. Constantine, *Constantine on Peopleware*. Prentice Hall, 1995.

[30] J.O. Coplien, "A generative development-process pattern language," in *Pattern Languages of Program Design*, J.O. Coplien and D.C. Schmidt, Eds. Addison-Wesley, 1995, pp. 183–237.

[31] P.T. Costa and R.R. McCrae, "The NEO Personality Inventory Manual," 1985.

[32] P. Cramer, *The Development of Defense Mechanisms: Theory, Research, and Assessment*. Springer-Verlag, 1991.

[33] A. Devito Da Cunha and D. Greathead, "Does personality matter? An analysis of code-review ability," *Comm. ACM*, vol. 50, no. 5, pp. 109–112, 2007.

[34] A. Diamantopoulos and J.A. Siguaw, "Formative versus reflective indicators in organizational measure development: A comparison and empirical illustration," *British J. Management*, vol. 17, pp. 263–282, 2006.

[35] A.J. Dick and B. Zarnett, "Paired programming & personality traits," in *Proc. Third Int'l Conf. Extreme Programming and Agile Processes in Software Engineering (XP 2002)*, 2002, pp. 82–85.

[36] D.H. Dickson and I.W. Kelly, "The 'Barnum Effect' in personality assessment: A review of the literature," *Psychological Reports*, vol. 57, pp. 367–382, 1985.

[37] R.M. Felder and L.K. Silverman, "Learning and teaching styles in engineering education," *Engineering Education*, vol. 78, no. 7, pp. 674–681, 1988.

[38] N.V. Flor and E.L. Hutchins, "Analyzing distributed cognition in software teams: A case study of team programming during perfective software maintenance," in *Proc. Fourth Workshop Empirical Studies of Programmers*, 1991, pp. 36–64.

[39] B.R. Forer, "The fallacy of personal validation: A classroom demonstration of gullibility," *J. Abnormal and Social Psychology*, vol. 44, pp. 118–123, 1949.

[40] D.R. Forsyth, *Group Dynamics*, 4th ed. Thomson Wadsworth, 2006.

[41] R.J. Freund, W.J. Wilson, and P. Sa, *Regression Analysis: Statistical Modeling of a Response Variable*, 2nd ed. Academic Press, 2006.

[42] A. Furnham, "The big five versus the big four: The relationship between the Myers-Briggs Type Indicator (MBTI) and NEO-PI five factor model of personality," *Personality and Individual Differences*, vol. 21, no. 2, pp. 303–307, 1996.

[43] H. Gallis, E. Arisholm, and T. Dybå, "An initial framework for research on pair programming," in *Proc. 2003 Int'l Symp. Empirical Software Engineering (ISESE'03)*, 2003, pp. 132–142.

[44] L.R. Goldberg, "An alternative description of personality: The big-five factor structure," *J. Personality and Social Psychology*, vol. 59, pp. 1216–1229, 1990.

[45] L.R. Goldberg, "The development of markers for the big-five factor structure," *Psychological Assessment*, vol. 4, no. 1, pp. 26–42, 1992.

[46] L.R. Goldberg, "The structure of phenotypic personality traits," *American Psychologist*, vol. 48, pp. 26–34, 1993.

[47] L.R. Goldberg, "A broad-bandwidth, public domain, personality inventory measuring the lower-level facets of several five-factor models," in *Personality Psychology in Europe*, I. Mervielde, I. Deary, F.D. Fruyt, and F. Ostendorf, Eds. Tilburg University Press, 1999, vol. 7, pp. 7–28.

[48] L.R. Goldberg, J.A. Johnson, H.W. Eber, R. Hogan, M.C. Ashton, C.R. Cloninger, and H.C. Gough, "The international personality item pool and the future of public-domain personality measures," *J. Research in Personality*, vol. 40, pp. 84–96, 2006.

[49] D. Goleman, "What makes a leader?" *Harvard Business Review*, vol. 76, no. 6, pp. 92–105, 1998.

[50] R.L. Gorsuch, *Factor Analysis*, 2nd ed. Lawrence Erlbaum Associates, 1983.

[51] S. Gregor, "The nature of theory in information systems," *MIS Quarterly*, vol. 30, no. 3, pp. 611–642, Sept. 2006.

[52] J. Grenning, "Launching extreme programming at a process-intensive company," *IEEE Software*, vol. 18, no. 6, pp. 27–33, 2001.

[53] T. Hærem, "Task complexity and expertise as determinants of task perceptions and performance: Why technology-structure research has been unreliable and inconclusive," Ph.D. dissertation, Norwegian School of Management BI, 2002.

[54] T. Hærem and D. Rau, "The influence of degree of expertise and objective task complexity on perceived task complexity and performance," *J. Applied Psychology*, vol. 92, no. 5, pp. 1320–1331, 2007.

[55] B. Hanks, "Student attitudes toward pair programming," in *Proc. 11th Annual Conf. Innovation and Technology in Computer Science Education (ITiCSE06)*. ACM, 2006, pp. 113–117.

[56] J.E. Hannay and M. Jørgensen, "The role of deliberate artificial design elements in software engineering experiments," *IEEE Trans. Software Eng.*, vol. 34, pp. 242–259, Mar/Apr 2008.

[57] J.E. Hannay, D.I.K. Sjøberg, and T. Dybå, "A systematic review of theory use in software engineering experiments," *IEEE Trans. Software Eng.*, vol. 33, pp. 87–107, Feb. 2007.

[58] W.E. Hanson and C.D. Claiborn, "Effects of test interpretation style and favorability in the counseling process," *J. Counseling and Development*, vol. 84, no. 3, pp. 349–358, 2006.

[59] W.K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[60] S.R. Hathaway and J.C. McKinley, "A multiphasic personality schedule (Minnesota): I. construction of the schedule," *J. Psychology*, vol. 10, pp. 249–254, 1940.

[61] M. Höst, B. Regnell, and C. Wohlin, "Using students as subjects—a comparative study of students and professionals in lead-time impact assessment," *Empirical Software Engineering*, vol. 5, no. 3, pp. 201–214, Nov. 2000.

[62] D.C. Howell, "The treatment of missing data," in *The SAGE Handbook of Social Science Methodology*, W. Outhwaite and S.P. Turner, Eds. Sage Publications Ltd., 2007.

[63] L. Hu and P.M. Bentler, "Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives," *Structural Equation Modeling*, vol. 6, pp. 1–55, 1999.

[64] "International Personality Item Pool. A scientific collaboratory for the development of advanced measures of personality traits and other individual differences," 2007. [Online]. Available: http://ipip.ori.org

[65] N. Katira, L. Williams, E. Wiebe, C. Miller, S. Balik, and E. Gehringer, "On understanding compatibility of student pair programmers," in *Proc. 35th Technical Symp. Computer Science Education (SIGCSE'04)*. ACM, 2004, pp. 7–11.

[66] D. Keirsey, *Please Understand Me II*. Prometheus Nemesis Book Company, 1988.

[67] D. Keirsey and M. Bates, *Please Understand Me*. Prometheus Book Company, 1984.

[68] S.L. Kichuk and W.H. Wiesner, "Work teams: Selecting members for optimal performance," *Canadian Psychology*, vol. 39, pp. 23–32, 1998.

[69] B.A. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," Keele University, EBSE Technical Report, EBSE-2007-01, Tech. Rep., 2007.

[70] R.B. Kline, *Principles and Practice of Structural Equation Modeling*, 2nd ed. The Guilford Press, 2005.

[71] L. Layman, "Changing students' perceptions: An analysis of the supplementary benefits of collaborative software development," in *Proc. 19th Conf. Software Engineering Education and Training (CSEET'06)*. IEEE Computer Society, 2006.

[72] S.Y. Lee, *Structural Equation Modelling. A Bayesian Approach*. Wiley, 2007.

[73] C.E. Lindblom, "Alternatives to validity. Some thoughts suggested by Campbell's guidelines," *Knowledge Creation, Diffusion, Utilization*, vol. 8, pp. 509–520, 1987.

[74] R.J.A. Little and D.B. Rubin, *Statistical Analysis with Missing Data*, 2nd ed. John Wiley & Sons, Inc., 2002.

[75] J.C. Loehlin, *Latent Variable Models: An Introduction to Factor, Path, and Structural Equation Analysis*, 4th ed. Lawrence Erlbaum Associates, 2003.

[76] B. Markovsky, "The structure of theories," in *Group Processes*, M. Foschi and E.J. Lawler, Eds. Nelson–Hall Publishers, 1994, pp. 3–24.

[77] W.M. Marston, *Emotions of Normal People*. Harcourt, Brace, & Co., 1928.

[78] P.E. Meehl, "Wanted—a good cookbook," *American Psychologist*, vol. 11, no. 3, pp. 263–272, 1956.

[79] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, "Equations of state calculations by fast computing machines," *J. Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[80] S. Mohammed and L.C. Angell, "Personality heterogeneity in teams: Which differences make a difference for team performance?" *Small Group Research*, vol. 34, pp. 651–677, 2003.

[81] J.E. Moore, "Personality characteristics of information systems professionals," in *Proc. 1991 Special Interest Group on Computer Personnel Research (SIGCPR) Annual Conference*. ACM, 1991, pp. 140–155.

[82] I.B. Myers and M.H. McCaulley, *A Guide to the Development and Use of the Myers-Briggs Type Indicator*. Consulting Psychologists Press, 1985.

[83] I.B. Myers and P. Myers, *Gifts Differing: Understanding Personality Type*. Davies-Black Publishing, 1995.

[84] G.A. Neuman, S.H. Wagner, and N.D. Christiansen, "The relationship between work-team personality composition and the job performance of teams," *Group & Organization Management*, vol. 24, pp. 28–45, 1999.

[85] A. Newell, J.C. Shaw, and H.A. Simon, "Elements of a theory of human problem solving," *Psychological Review*, vol. 65, pp. 151–166, 1958.

[86] J.C. Nunnally and I. Bernstein, *Psychometric Theory*, 3rd ed. McGraw-Hill, Inc., 1994.

[87] A.M. Paul, *The Cult of Personality Testing: How Personality Tests Are Leading Us to Miseducate Our Children, Mismanage Our Companies, and Misunderstand Ourselves*. Free Press, 2005.

[88] M.A.G. Peeters, H.F.J.M. van Tuijl, C.G. Rutte, and I.M.M.J. Reymen, "Personality and team performance: A meta-analysis," *European J. of Personality*, vol. 20, pp. 377–396, 2006.

[89] L.A. Pervin and O.P. John, *Personality: Theory and Research*, 7th ed. John Wiley & Sons, Inc., 1997.

[90] S. Ramanujan, R.W. Scamell, and J.R. Shah, "An experimental investigation of the impact of individual, program, and organizational characteristics on software maintenance effort," *J. Systems and Software*, vol. 54, no. 2, pp. 137–157, Oct. 2000.

[91] A. Saggio, C. Cooper, and P. Kline, "A confirmatory factor analysis of the Myers-Briggs Type Indicator," *Personality and Individual Differences*, vol. 30, pp. 3–9, 2001.

[92] A. Saggio and P. Kline, "The location of the Myers-Briggs Type Indicator in personality factor space," *Personality and Individual Differences*, vol. 21, no. 4, pp. 591–597, 1996.

[93] J. Sall, "Monte carlo calibration of distributions of partition statistics," SAS Institute, Tech. Rep., 2002. [Online]. Available: jmp.com/software/whitepapers/pdfs/montecarlocal.pdf

[94] P. Sfetsos, I. Stamelos, L. Angelis, and I. Deligiannis, "Investigating the impact of personality types on communication and collaboration-viability in pair programming—an empirical study," in *Proc. Seventh Int'l Conf. Extreme Programming and Agile Processes in Software Engineering (XP 2006)*, ser. Lecture Notes in Computer Science, vol. 4044. Springer-Verlag, 2006, pp. 43–52.

[95] W.R. Shadish, T.D. Cook, and D.T. Campbell, *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. Houghton Mifflin, 2002.

[96] T.M. Shaft and I. Vessey, "The relevance of application domain knowledge," *J. Management Information Systems*, vol. 15, no. 1, pp. 51—78, 1998.

[97] B. Shneiderman, *Software Psychology: Human Factors in Computer and Information Systems*. Winthrop Publishers, 1980.

[98] D.I.K. Sjøberg, B. Anda, E. Arisholm, T. Dybå, M. Jørgensen, A. Karahasanović, and M. Vokáč, "Challenges and recommendations when increasing the realism of controlled software engineering experiments," in *Empirical Methods and Studies in Software Engineering: Experiences from ESERNET*, R. Conradi and A.I. Wang, Eds. Springer-Verlag, 2003, pp. 24–38.

[99] D.I.K. Sjøberg, J.E. Hannay, O. Hansen, V.B. Kampenes, A. Karahasanović, N.K. Liborg, and A.C. Rekdal, "A survey of controlled experiments in software engineering," *IEEE Trans. Software Eng.*, vol. 31, no. 9, pp. 733–753, Sept. 2005.

[100] D.C. Smith, "The personality of the systems analyst: An investigation," *SIGCPR Computer Personnel*, vol. 12, no. 2, pp. 12–14, 1989.

[101] E. Soloway, J. Pinto, S. Letovsky, D. Littman, and R. Lampert, "Designing documentation to compensate for delocalized plans," *Comm. ACM*, vol. 31, no. 11, pp. 1259–1267, Nov. 1988.

[102] S. Sonnentag, "Expertise in professional software design," *J. Applied Psychology*, vol. 83, no. 5, pp. 703–715, 1998.

[103] I.D. Steiner, *Group Process and Productivity*. New York and London: Academic Press, 1972.

[104] R.J. Sternberg, "Cognitive conceptions of expertise," *Int'l J. Expert Systems*, vol. 7, no. 1, pp. 1–12, 1994.

[105] L. Thomas, M. Ratcliffe, and A. Robertson, "Code warriors and code-a-phobes: A study in attitude and pair programming," in *Proc. 34th Technical Symp. Computer Science Education (SIGCSE'03)*. ACM, 2003.

[106] A.S. Tippetts and P.R. Marques, "Compensating for deficiencies in perinatal data sets: Parametric perspectives," in *Treatment for Drug-Exposed Women and Children: Advances in Research Methodology. NIH Publication No. 96-3632*, E.R. Rahdert, Ed. National Institute on Drug Abuse, 1996, pp. 272–291.

[107] B. Van Fraassen, *The Scientific Image*. Oxford University Press, 1980.

[108] A.E.M. Van Vianen and C.K.W. De Dreu, "Personality in teams: Its relations to social cohesion, task cohesion, and team performance," *European J. Work and Organizational Psychology*, vol. 10, pp. 97–120, 2001.

[109] J. Vanhanen and C. Lassenius, "Effects of pair programming at the development team level: An experiment," in *Proc. 33rd EUROMICRO Conference on Software Engineering and Advanced Applications*, 2007, pp. 211–218.

[110] K. Visram, "Extreme programming: Pair-programmers, team players or future leaders?" in *Proc. Eighth IASTED Int'l Conf. Software Engineering and Applications*. Acta Press, 2004, pp. 659–664.

[111] G.M. Weinberg, *The Psychology of Computer Programming*. Van Nostrand Reinhold, 1971.

[112] G.M. Weinberg, *The Psychology of Computer Programming*, silver anniversary ed. Dorset House Publishing, 1998.

[113] D. Westen, "Clinical assessment of object relations using the TAT," *J. Personality Assessment*, vol. 56, no. 1, pp. 56–74, 1991.

[114] D.A. Whetten, "What constitutes a theoretical contribution," *Academy of Management Review*, vol. 14, no. 4, pp. 490–495, 1989.

[115] L. Williams and R.R. Kessler, *Pair Programming Illuminated*. Addison-Wesley, 2002.

[116] L. Williams, R.R. Kessler, W. Cunningham, and R. Jeffries, "Strengthening the case for pair programming," *IEEE Software*, vol. 17, no. 4, pp. 19–25, 2000.

[117] L. Williams, L. Layman, J. Osborne, and N. Katira, "Examining the compatibility of student pair programmers," in *Proc. AGILE 2006*. IEEE Computer Society, 2006.

[118] L. Williams, A. Shukla, and A.I. Antón, "An initial exploration of the relationship between pair programming and Brooks' Law," in *Proc. Agile Development Conf. (ADC'04)*, 2004, pp. 11–20.

[119] R.J. Wirfs-Brock, "Characterizing your application's control style," *Report on Object Analysis and Design*, vol. 1, no. 3, 1994.

[120] R.J. Wirfs-Brock and B. Wilkerson, "Object-oriented design: A responsibility-driven approach," *SIGPLAN Notices*, vol. 24, no. 10, pp. 71–75, 1989.

[121] C.K. Woodruff, "Personality profiles of male and female data processing personnel," in *Proc. 17th Annual Southeast Regional Conference*. ACM, 1979, pp. 124–128.