

Towards Individualized Software Engineering: Empirical Studies Should Collect Psychometrics

Robert Feldt
Richard Torkar
Blekinge Institute of
Technology
SE-372 25 Ronneby, Sweden
rfd@bth.se

Lefteris Angelis
Dept. of Informatics
Aristotle University of
Thessaloniki
54124 Thessaloniki, Greece
lef@csd.auth.gr

Maria Samuelsson
Dept. of Informatics
University West
SE-461 86 Trollhättan,
Sweden

ABSTRACT

Even though software is developed by humans, research in software engineering primarily focuses on the technologies, methods and processes they use while disregarding the importance of the humans themselves. In this paper we argue that most studies in software engineering should give much more weight to human factors. In particular empirical software engineering studies involving human developers should always consider collecting psychometric data on the humans involved. We focus on personality as one important psychometric factor and present initial results from an empirical study investigating correlations between personality and attitudes to software engineering processes and tools. We discuss what are currently hindering a more wide-spread use of psychometrics and how overcoming these hurdles could lead to a more individualized software engineering.

Categories and Subject Descriptors

D.2.8 [Software Engineering]: Metrics—*complexity measures, performance measures*
; J.4 [Social and Behavioral Sciences]: Psychology

General Terms

Human Factors

Keywords

Software Engineering, Empirical Research, Personality, Psychometrics

1. INTRODUCTION

Software Engineering (SE) aims to develop techniques, methods and processes to enable human software engineers to develop large and complex software systems in a more optimal way. SE is a broad field of study. In addition to the software itself and the many technical aspects of

its construction, development and maintenance, the field covers most of the areas of modern business, e.g. quality, customer expectations, design, legal issues and intellectual property, project and product management, and strategies. Even though much has been learned since the area was born in the 60's we think there is a large gap in how it has developed and what is currently known. This is a reflection of the fact that there has been too much of a focus on the techniques, methods, processes and organizations involved and too little focus on the humans.

Empirical SE studies typically investigate how a certain development team or organization currently develop software. It might also introduce some new technique, method or process¹ and evaluate if it improves the effectiveness or efficiency of development and/or maintenance of the software. In a controlled laboratory setting where researchers have the maximum amount of control they can randomize the assignment of individuals to groups and methods and isolate the effects of the new method being evaluated. This way they can avoid that differences between the individuals in the experiment affect the results.

When a new method is empirically evaluated in a real-world, industrial setting we do not have this level of control. Many factors affect which humans will actually be in the project; often they are the ones available, i.e. that are already in the project or team. The knowledge, abilities, attitudes and personalities of these people will affect how they react to changes in general and the introduced method in particular, how they feel about their work and the other participants and how they view each other and thus collaborate. The results of this evaluation can critically depend on these persons being studied. In many cases, the effects arising from psychological factors might dwarf any effect of the method being evaluated.

Despite early interest in human factors, and in particular personal characteristics of the humans involved in software engineering processes, these issues have been largely overlooked [4, 10, 20]. For example, even though a few studies have considered the personality of developers, they have used dated models and metrics, that classify people into a few groups or types, to do so [2, 3, 9, 18]. This has prevented detailed knowledge about how psychological aspects correlates with software engineering performance and prevented the use of this knowledge in improvement efforts.

¹In the following we use 'method' to refer to any SE 'idea' being evaluated; this can be any technique, method, process, or change etc. depending on the study in question.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHASE'08, May 13, 2008, Leipzig, Germany.

Copyright 2008 ACM 978-1-60558-039-5/08/05 ...\$5.00.

Psychometrics is concerned with theory and techniques for quantitative measurement in psychology and social sciences [12]. In practice, this often means the measurement of knowledge, abilities, attitudes, emotions, personality and motivation. It is a broad area with a long and tumultuous history [12]. In the following we will focus on the sub-area of personality and present initial results from a study that investigated correlations between personality factors and attitudes towards software engineering. In the discussion we will then connect back to the broader area of psychometrics.

2. PERSONALITY TESTING

One of the main views within personality psychology is that personality can be described by a set of traits, i.e. fixed set of patterns in how a person behaves, feels and thinks [1, 15]. These traits can be used to summarize, explain and predict how a person will act in different situations.

Even though different trait-based theories had been proposed since the 1930's it was not until the 1970's that they gained more widespread acceptance and interest. During this period many practical tests for personality traits were created and larger, empirical studies provided extensive data sets. There was much interest in applying these tests to match people to jobs and to put together successful and creative teams.

A major step was the Myers-Briggs Type Indicator (MBTI), based on Jungian theory, which to this day is the most used personality test [8, 13]. It has four dimensions: Extraversion (E) vs. Introversion (I), Sensing (S) vs. Intuition (N), Thinking (T) vs. Feeling (F) and Judging (J) vs. Perceiving (P). Based on 93 forced-choice (only two options of which one has to be chosen) items a licensed MBTI assessor can find the type of a person based on the largest score for each bipolar dimension. In theory, each of the sixteen different personality types measured by MBTI can be viewed as collections of packaged traits.

There have been a few studies on personality in software development projects; some have focused on programmers and some, on the somewhat wider notion of software engineers or, software development teams in general. All have used the MBTI.

In [2] a sample of 100 software engineers were administered the MBTI. The main findings were that NT and ST types were over-represented in the sample, while SF and NF were underrepresented. Capretz also summarizes previous, related research which seems to indicate that programmers are often introverts and that the types ISTJ, INTJ and INTP are the most frequent. In contrast, Smith found [18] that among 37 system analysts 35% were ISTJ while 30% were ESTJ. Recently, Chao and Atli [3] executed a survey with 60 responses (programmers). They found no evidence that there is a difference in code quality between different ways of pairing personality types. Some studies have also used MBTI personality types to understand and improve software engineering education [9].

However, the MBTI has been called into question by research in psychology since the late 1980's [14]. Empirical results have questioned whether MBTI measures qualitatively distinct types, and especially the Judging/Perceiving dimension seems weak. The alternative is a descriptive model called the Five-Factor Model (FFM) [14]. The five factors refer to broad personality dimensions that have been found in empirical research: Openness (O), Conscientiousness (C),

Extraversion (E), Agreeableness (A), and Neuroticism (N). In recent descriptions the Neuroticism factor is more aptly called Emotional Stability [6]. In the following we will use the traditional term, but note that increased levels for the N factor actually refers to increasing emotional stability. The Openness factor is sometimes also called Intelligence but we will avoid that term since it can be confounded with IQ testing.

IPIP is a freely available set of items and scales for psychometrics based on the five-factor personality model [6, 7]. The IPIP scales have several benefits compared to the MBTI: they have been found to better describe personality than the MBTI; they can be further refined to more detailed trait descriptions if needed (each factor is a 'super' factor collecting a number of sub-factors/traits together), they provide numerical scores for each factor which makes more detailed statistical analysis possible, they do not require licensed assessors, and they are freely available.

3. PERSONALITY & SE ATTITUDES

As a first step in studying human and personality factors and their effect on software engineering we have administered a web-based questionnaire to software engineers in companies in Sweden. The questionnaire has two main parts: a personality test (Part I) and a set of questions to probe the attitudes and working style and habits of the respondent in areas related to software engineering (Part II).

Part I is the IPIP 50-item scale with 50 items. Part II contains 56 additional questions on attitudes and working habits related to software engineering. It is divided into sub-sections that ask the respondents and their self-image, about software engineering tools and processes used, the organization in which they work, and about their view on SE research. The answer alternatives are mainly categorical, most with Likert scales [11] having four alternatives to force respondents to take a stand. Most of the questions had been previously used in a degree study by one of the authors; only eight new were added [16]. Thus we were already confident that the wordings of the questions and the sets of answer alternatives were suitable. Even so, a pre-test was performed with the help of a software developer. We used this to ensure that it would take, on average, less than 30 minutes to answer the questionnaire.

By contacting a large group of companies by phone we ended up with 10 diverse companies willing to participate. However, few of them were willing to allow many developers to take the test, citing upcoming deadlines and time constraints as the main reasons. Some also showed resistance towards unproven methods that are not common within software engineering or computer science. In the end we got 47 responses from ten different companies. Seven of the companies are small- to medium-sized software development or software consultancy companies, one was a small subsidiary company within a large Swedish telecom company, one was a real-time and embedded software development department within a large industry, and one was the local branch of a large Nordic IT consultancy company.

Of the 47 respondents, 11 were female and 36 male. Most of the respondents were between 24 and 55 years old. They had a mixed level of software engineering experience ranging from less than 2 years to more than 10 years. About half of respondents described themselves foremost as being 'Programmers' about a third as either 'Project Managers',

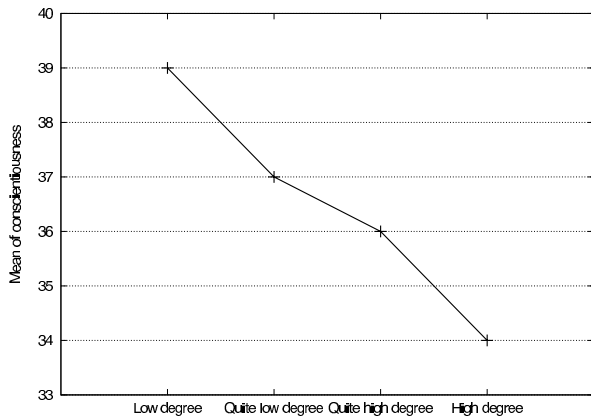


Figure 1: Mean of conscientiousness vs. question 74 (To what degree do you feel there is a need to change the current working manner?)

‘Product Managers’ or ‘System Architects’. We are doing an extensive statistical analysis of the results; here we only give some initial results.

In order to test the association between personality factors and the questions in Part II, we performed ONE-WAY ANOVA analyses. This shows if there is significant difference in the means of the personality factor across the categories of a question. The use of ANOVA is warranted since the 5 variables were tested for normality by the Kolmogorov-Smirnov test and found to not differ significantly from the normal distribution ($p > 0.1$ for all variables). In Figure 1 we have plotted the average value for the Conscientiousness personality factor against the four possible answers on question 74: ‘To what degree do you feel there is a need to change the current working manner?’ This is a statistically significant correlation. With further ANOVA analyses we see that higher levels of conscientiousness correlate with preferring to work alone, a low need to change the current working procedures, preferring non-technical tasks above technical tasks, and thinking software engineering research is not so important. We think it is clear that such correlations can affect the introduction and thus evaluation of new SE methods. Thus, it is crucial that we have information about personality factors when evaluating SE methods in empirical research studies.

4. DISCUSSION

It was easy to use the freely available IPIP personality tests and there is little to be lost from administering them over the web. This should make for cheap and simple personality testing. The fact that the 50-item IPIP five-factor scale has only 50 questions compared to the 93 of MBTI can also work to its advantage. When it is combined with other questions or psychometric instruments it is important to keep down the total time needed to complete the survey. Having 50 instead of 93 questions can thus be an advantage. On the other hand, IPIP have larger tests with 100 and 300 items that can be used if the researcher wants to find out more about sub-categories of the main personality factors. A downside with the IPIP is that there are fewer norms to compare to. Over time this problem should vanish.

In the analysis we have done so far we have seen an advantage with the five-factor model and IPIP tests compared to MBTI used previously in SE research. Since the IPIP scales give a numerical value for each personality factor we can apply more powerful statistical methods than if we use the ‘binary’ MBTI types. When looking for other psychometric tests, this is something to remember. A threat to our study is that we only have answers from 47 respondents and that we do not know the answer frequency.

In the presented study we have only focused on attitudes and personality. However, there is a rich literature on psychometrics that can be mined for possible factors and tests to use in our SE studies. We intend to detail the possibilities further in future work. One promising theory is Reversal Theory by Michael Apter. It is a theory of both personality, motivation and emotion that focuses on the way people change between ‘states’. Since it encompasses several different types of psychometric factors in one framework it could prove powerful. There has been some work on evaluating it in the context of technology teams [19].

A problem with applying existing psychometric instruments is that many of them are tied to commercial interests and not freely shared. For example, the MBTI requires that you are a certified assessor. Long-term the IPIP choice seems to be the only plausible one; in analogy to open-source software, researchers in psychology should strive to publish their tests and the analysis procedures needed to evaluate data collected with them freely on the web. Alternatively, SE researchers can utilize that many companies involved in psychometrics have special ‘academic licenses’ for using their tests. This is the case for example with some of the tests for Apter’s Reversal Theory.

Another problem with psychometrics is that there are ethical issues involved. In our study, there are results linking personality with vocational interests and aptitude. Such information is potentially very sensitive. For example, an employee might not want his boss to know his personality profile if that might negatively affect the employees career opportunities. To overcome such fears researchers need to have more rigorous control mechanisms in place. In our case, we ensured we had as much control over the information as possible by hosting the personality test on a computer at the university over which the researchers had full control. No other university employees could access the machine and once the results were gathered they were packed and encrypted and handled with utmost care. Furthermore we informed study participants that their answers would be available only to the researchers and not to anyone else in their company. More exploration of ethical issues are important for wide-spread use of psychometrics in SE.

In recent years the field of medicine has been revolutionized by findings in molecular biology. By mapping the genetic make-up of individuals and understanding how the connections between genes and environmental factors determine both how diseases develop and how they can best be treated, a new type of ‘Individualized Medicine’ is emerging [17, 5]. We think a similar revolution is needed in Software Engineering and that psychometrics could play the role that genes play in the development of individualized medicine. The techniques, methods and processes we propose and evaluate in Software Engineering should be adapted and aligned to the persons that will use them, their group dynamics and the context in which they work. Much new knowledge will

have to be developed in order to realize this, and psychometrics can only be one component in such a development. But it is an important component and one of the easier and more accessible to start with. We hope that many SE researchers will be willing to take part in the development of an ‘Individualized Software Engineering’.

5. CONCLUSIONS

This paper argues that Software Engineering researchers should put a larger focus on the humans involved in software development than what has been done to date. One easy and powerful way to do this would be to collect psychometric measurements. Especially for empirical studies done in a real-world setting where there is less control this is crucial. Without detailed information about the people involved there is a risk that the effects of the change introduced is dwarfed by the effects of how the persons affected by the change accept and adapt to it.

To avoid this problem in future empirical SE studies we propose the use of simple psychometric instruments such as questionnaires to measure personality, attitudes, motivations and emotions. By analyzing these data and relating them to primary data we can build a science of software engineering with equal weight given to techniques, methods and processes as well as to the humans who use them.

This paper presented initial results from an extensive questionnaire taken by 47 software engineers in Swedish software development organizations. The questionnaire had two main parts; 56 questions on attitudes and procedures in SE, and a 50-item personality test. This psychometric instrument is freely available and can be easily administered via the web. It is based on state-of-the-art personality theories and was taken in 10–15 minutes by the subjects of the study. By correlating the results from the two parts we can see that higher levels on the personality dimension ‘conscientiousness’ correlate with attitudes towards work style, openness to changes and task preference.

We see increased use of psychometrics as an important step towards more rigorous, scientific theories and knowledge in SE. They will be an important step towards individualizing the advice and choice of SE methods so that they are better adapted to the particular humans, groups and contexts and lead to better results. We hope that other SE researchers will be interested in this quest for a more individualized Software Engineering.

6. REFERENCES

- [1] R. L. Atkinson, R. C. Atkinson, E. E. Smith, and D. J. Bem. *Introduction to Psychology*. Harcourt Brace Jovanovich, Florida, USA, 11th edition, 1993.
- [2] L. F. Capretz. Personality Types in Software Engineering. *International Journal of Human-Computer Studies*, 58(2):207–214, 2003.
- [3] J. Chao and G. Atli. Critical Personality Traits in Successful Pair Programming. In *AGILE*, pages 89–93. IEEE Computer Society, 2006.
- [4] B. Curtis. A Review of Human Factors Research on Programming Languages and Specifications. In *Proceedings of the 1982 Conference on Human Factors in Computing Systems*, pages 212–218, New York, NY, USA, 1982. ACM Press.
- [5] W. E. Evans and M. V. Relling. Moving towards individualized medicine with pharmacogenomics. *Nature*, 429(6990):464–468, 2004.
- [6] L. R. Goldberg. A Broad-Bandwidth, Public-Domain, Personality Inventory Measuring the Lower-Level Facets of Several Five-Factor Models. *Personality Psychology in Europe (Selected Papers from the Eighth European Conference on Personality held in Ghent, Belgium, July 1996)*, 7:7–28, 1999.
- [7] L. R. Goldberg, J. A. Johnson, and H. W. Eber. The International Personality Item Pool and the Future of Public-Domain Personality Measures. *Journal of Research in Personality*, 40(1):84–97, 2006.
- [8] C. G. Jung. *Psychological Types*, volume 6 of *Collected Works of C. G. Jung*. Princeton University Press, 1971.
- [9] J. Karn and T. Cowling. A Follow Up Study of the Effect of Personality on the Performance of Software Engineering Teams. In *ISESE '06: Proceedings of the 2006 ACM/IEEE International Symposium on Empirical Software Engineering*, pages 232–241, New York, NY, USA, 2006. ACM Press.
- [10] M. I. Kellner, B. C., T. DeMarco, K. Kishida, M. Schlumberger, and C. Tully. Non-Technological Issues in Software Engineering. In *ICSE '91: Proceedings of the 13th International Conference on Software Engineering*, pages 144–146. IEEE Computer Society / ACM Press, May 1991.
- [11] R. Likert. A Technique for the Measurement of Attitudes”. *Archives of Psychology*, 23(140):1–55, 1932.
- [12] J. Michell. *Measurement in Psychology*. Cambridge University Press, Cambridge, 1999.
- [13] I. B. Myers, M. H. McCaulley, N. L. Quenk, and A. L. Hammer. *MBTI Manual (A Guide to the Development and Use of the Myers Briggs Type Indicator)*. Consulting Psychologists Press, 3rd edition, 1998.
- [14] R. R. McCrae and P. T. Costa Jr. Reinterpreting the Myers-Briggs Type Indicator From the Perspective of the Five-Factor Model of Personality. *Journal of Personality*, 57(1):17–41, 1989.
- [15] J. Sabini. *Social Psychology*. W. W. Norton & Co Inc., 2nd edition, 1995.
- [16] M. Samuelsson. Personality Types and Attributes in Software Engineering. Master thesis, Dept. of Informatics, University West, Trollhättan, Sweden, June 2005.
- [17] Individualized Medicine Emerging From Gene-Environment Studies. <http://www.sciencedaily.com/releases/2005/01/050123213425.htm>, January 2008.
- [18] D. C. Smith. The Personality of the Systems Analyst: An Investigation. *ACM SIGCPR Computer Personnel*, 12(2):12–14, 1989.
- [19] J. Tucker and H. Rutledge. Shaping Motivation and Emotion in Technology Teams. *CrossTalk - The Journal of Defense Software Engineering*, November 2007.
- [20] G. M. Weinberg. *The Psychology of Computer Programming*. van Nostrand Reinhold Company, 1971.