2-1-2003

Personality Types in Software Engineering

Luiz Fernando Capretz
University of Western Ontario, lcapretz@uwo.ca

Available online at www.sciencedirect.com
SCIENCE DIRECT
ELSEVIER
Int. J. Human-Computer Studies 58 (2003) 207–214
International Journal of Human-Computer Studies
www.elsevier.com/locate/ijhcs

Personality types in software engineering

Luiz Fernando Capretz

Department of Electrical and Computer Engineering, University of Western Ontario, London, Canada, N6G 1H1

Received 22 February 2002; received in revised form 27 August 2002; accepted 1 November 2002

Abstract

Software engineering is forecast to be among the fastest growing employment field in the next decades. The purpose of this investigation is two-fold: Firstly, empirical studies on the personality types of software professionals are reviewed. Secondly, this work provides an up-to-date personality profile of software engineers according to the Myers–Briggs Type Indicator.

1. Introduction

Jung's theory of psychological types assumes that much apparently random behaviour is actually quite orderly and consistent. These consistencies result from differences in the ways persons take in information and make decisions. Naturally, this is not "rocket science"; therefore, it causes a great deal of debate among psychologists.

Extroversion and introversion (*E and I*): Some people are oriented to a breadth-of-knowledge approach with quick action; others are oriented to a depth-of-knowledge approach reflecting on concepts and ideas. Jung calls these orientations extroversion and introversion.

Sensing and intuition (*S and N*): Some people are attuned to the practical, hands-on, common-sense view of events, while others are more attuned to the complex interactions, theoretical implications or new possibilities of events. These two styles of information gathering, or perception, are known as sensing and intuition, respectively.

E-mail address: lcapretz@eng.uwo.ca (L.F. Capretz).

1071-5819/03/$ - see front matter © 2002 Elsevier Science Ltd. All rights reserved.
PII: S 1 0 7 1 - 5 8 1 9 (0 2) 0 0 1 3 7 - 4

Table 1
The MBTI types and their distribution among the US adult population

ISTJ 11.6%	ISFJ 13.8%	INFJ 1.5%	INTJ 2.1%
ISTP 5.4%	ISFP 8.8%	INFP 4.4%	INTP 3.3%
ESTP 4.3%	ESFP 8.5%	ENFP 8.1%	ENTP 3.2%
ESTJ 8.7%	ESFJ 12.3%	ENFJ 2.5%	ENTJ 1.8%

Thinking and feeling (*T and F*): Some people typically draw conclusions or make judgments objectively, dispassionately and analytically; others weigh the human factors or societal import, and make judgments with personal conviction as to their value. These two styles of decision-making are called thinking or feeling, respectively.

Judgment and perception (*J and P*): Finally, some people prefer to collect only enough data to make judgments before setting on a direct path to a goal, and typically stay on that path. Others are finely attuned to changing situations, alert to new developments that may require a change of strategy, or even a change of goals. These two styles are called the preferences for judgment or perception, respectively.

The Myers–Briggs Type Indicator (*MBTI*): The MBTI (Myers et al., 1998) describes 16 types which result from the dynamic interplay of these four preferences—EI, SN, TF, JP, and types are denoted by the letters of preferred orientations (such as ISTJ, ENFP, INTP, etc.), as shown in Table 1. It is important to understand that everyone uses all eight preferences, not merely the four which are preferred. The theory describes 16 distinct ways of being normal. No preference is superior over any other preference, and no type is superior over any other type (though in a given situation, the preferences of one type may match the demands of the situation better than those of a different type).

The software industry has become a major force in society. Software engineering is a field that many outsiders and even insiders have wrongly stereotyped. It is a common belief that to be a good software engineer it is necessary to like mathematics or a similar field. People stereotype the behaviour of software professionals, as introverts working alone in a corner of their office, hating interaction with others, a typical *nerd*. However, specialties within software engineering today are as diverse as the medical profession, with software engineers working as systems analysts, interface designers, programmers, testers and maintainers.

Psychologists have traditionally been interested in understanding the factors that predict career choices using a myriad of personality indicators (Blatt, 1986), including MBTI. Software engineering is a domain that has grown in popularity, yet the degree of job satisfaction has been scarcely investigated among software professionals. This is of paramount importance because nowadays software permeates almost all activities of modern society, which makes the software industry a very broad field of study, as opposed to specialized scientific programming of a few decades ago; consequently, the software engineers' profile might have changed.

Several empirical studies investigated the relationship between MBTI and computer programming. Sitton and Chmelir (1984) list some stereotypes about what programmers are like and what attracts them to the field. They painted a picture of creative professionals merrily and irreverently solving complicated problems, untrammelled by routine and humdrum details; however, they gave no specific statistics about their findings.

Bush and Schkade (1985) tested 58 professionals in one high-tech aerospace company involved with scientific programming. They found ISTJ (25%) to be the most common type. Further, the second most frequently reported type was INTJ (16%), and ENTP (9%) to be the third; they also found thinking (74%) and judging (70%) to be abundant. Buie (1988) takes a sample of 47 scientific programmers employed by a private company under contract with NASA performing work on orbit-related software, ISTJ (19%), INTP (15%) and INTJ (13%) were the most frequent, with those three types collectively accounting for nearly half the sample. On the other hand, ESFJ (0%), ISFP (0%) and ENTP (0%) were particularly underrepresented.

Nevertheless, there is more to software engineering than programming. It comprises systems analysis, design, programming, testing and maintenance of software systems. Lyons (1985) surveyed 1229 people from more than 100 companies, including insurance companies, financial institutions, utilities and hardware manufacturers. He too found ISTJ (23%) to be the most common type, INTJ (15%) to be the second, and INTP (12%) to be a close third. He also found thinking (81%) and judging (65%) types to be in the majority; furthermore, he found 67% of his subjects to be introverts. He was the first to observe that R&D organizations and companies that do a lot of state-of-the-art development attract and hire more Ns than Ss. The opposite occurs in large organizations where the bulk of the work involves maintaining and enhancing production systems.

Smith (1989) deals with 37 systems analysts at a large insurance company. The most frequent types in the sample are ISTJ (35%) and ESTJ (30%). From the results, there were slightly more introverts (57%), but there was a heavy bias towards the sensate (81%), thinking (89%) and judging (86%) types. Interestingly, the four NF combinations were not present at all in this small sample. Larger and diverse samples would allow more comprehensive data and definitive conclusions.

The common thread running through the results of these studies is the prevalence of introverts, thinking, judging, and almost as many sensing as intuitives among software professionals. In the past, it seemed reasonable to think of computer work as a practical application of mathematical concepts, as in the aerospace industry, but not anymore. Nowadays software permeates all aspects of life, including finance, administration and games, for example. Software developers can act in occupations without knowing or using mathematics.

Moreover, software engineering is becoming a very broad field of study; consequently, some skills necessary to successfully work in this area 30 years ago may not apply anymore. For instance, software development is much more than manipulating formal or semi-formal notations. It has everything to do with interactions between designers and users, i.e., the designer's perception of what the user wants, and the user's perception of what he/she really needs, and vice versa. Nowadays, successful software artefacts are those developed after a tremendous amount of time has been spent with the user, in the form of prototyping, experimenting and feedback. This is the proper development life cycle of any useful software system. Therefore, further research is still needed to establish an up-to-date profile of software engineers.

2. Method

Our sample comprises a group of 100 software engineers (80% male and 20% female) who study in private or public universities, or work for the government, or work for software companies. They are all productive and motivated software engineers, and were selected to participate in this study based on their occupation, and were administered the MBTI (Form G) to determine their personality types. This study is relevant because it considers both professionals in a job setting and students in upper level university classes, as well as graduate students. Some students are already working as software engineers. This supports a greater generality of the results reported here.

In the tables below, the letter R refers to the ratio known as the self-selection index in the selection-rate-type table. The ratio is computed based on the percentage of the observed frequency to the expected frequency. When the ratio is greater than 1.00, there are more people in that cell of the table than we expected from their numbers in a general population. If the ratio is less than 1.00, there are fewer people in that cell than expected in a general population. A quick inspection of the 16 types shows that all NT and ST types have indices (R) greater than 1.00 showing the trend that NT and ST are overrepresented among software engineers. On the other hand, all SF and NF types have index (R) much smaller than 1.00 indicating that SFs and NFs are underrepresented among software engineers.

3. Results

The type distribution of the software engineers is summarized in Tables 2 and 3. This research found more introverts ($I = 57\%$) than extroverts ($E = 43\%$) types; fairly more sensing ($S = 67\%$) than intuitive ($N = 33\%$); significantly more thinking ($T = 81\%$) than feeling ($F = 19\%$); and slightly more judging ($J = 58\%$) compared to perceiving ($P = 42\%$) type. It can also be noted that TJs comprise 50% of the sample, STs compose 55% and NTs make up to 26% of the subjects (with the highest R ratio). On the other hand, NFs add up to 7% only, and SFs a mere 12% of

Table 2
Dichotomous preferences and pairs temperament ($n = 100$)

Type	%	*R*
E	43	0.87
I	57	1.12
S	67	0.91
N	33	1.24
T	81	2.01
F	19	0.32
J	58	1.07
P	42	0.92
Pair	%	*R*
IJ	34	1.18
IP	23	1.05
EP	19	0.79
EJ	24	0.95
ST	55	1.84
SF	12	0.28
NF	7	0.43
NT	26	2.52
SJ	45	0.97
SP	22	0.82
NP	20	1.06
NJ	13	1.67
TJ	50	2.08
TP	31	1.92
FP	11	0.37
FJ	8	0.27
IN	18	1.61
EN	15	0.97
IS	39	0.99
ES	28	0.83
ET	34	n.a.
EF	9	n.a
IF	10	n.a
IT	47	n.a.

the subjects, with low *R* ratios. As predicted, the results, though not completely, met the expectations of those in previous studies.

More specifically, this investigation has shown that ISTJ, ISTP, ESTP and ESTJ compose over 50% of the sample, therefore, overrepresented in the sample, whereas INFJ, ESFP and ENFJ are all particularly underrepresented in that group. TJs, STs and NTs are abundant among software professionals. On the other hand, SFs and NFs are scarce. It is worth noticing that there are more ISTJ (24%) than any other type. We also found a cluster of sensing, thinking and judging (STJ = 39%). But it is in the thinking/feeling dimension that the gap really widens; part of this concentration may be accounted for by the disproportionate presence of men in software engineering; in our sample 20% are women and 80% are men.

Table 3
Type distribution of software engineers and SRTT comparison with an adult population sample ($n = 100$, R = selection ratio index)

ISTJ 24% $R = 2.08$	ISFJ 2% $R = 0.14$	INFJ 1% $R = 0.68$	INTJ 7% $R = 3.40$
ISTP 8% $R = 1.49$	ISFP 5% $R = 0.57$	INFP 2% $R = 0.46$	INTP 8% $R = 2.46$
ESTP 8% $R = 1.87$	ESFP 1% $R = 0.12$	ENFP 3% $R = 0.37$	ENTP 7% $R = 2.19$
ESTJ 15% $R = 1.73$	ESFJ 4% $R = 0.33$	ENFJ 1% $R = 0.41$	ENTJ 4% $R = 2.23$

The largest single type among the subjects is ISTJ. MBTI data collected over the years show that 11.6% of the US adult population falls into this category (Myers et al., 1998) as compared to 24% in our study. From a statistical perspective, this is a significant finding. Further, the second most frequently reported type is ESTJ, with 15% of the subjects fitting in this category as compared to 8.7% of the US adult population. ISFJs, incidentally, accounted for 2% of the subjects in the sample compared to 13.8% in that general population. There is a dramatic fluctuation between the ESFP percentage in that general population (8.5%) compared to the same type in our sample (1%), which is reflected by the lowest ratio ($R = 0.12$).

4. Discussion

Software engineers and psychological types are clearly related, as suggested by this study. More specifically, the current work suggests that software engineers are most likely to be STs or TJs or NTs. The results are important to employers looking for software professionals and to students looking for careers. However, this investigation has not taken into consideration motivation profile for a career choice such as: money, recognition, autonomy, power and security.

In accordance with the MBTI theory, the NTs tend to be more creative than STs because Ns see possibilities beyond the given facts, and look for patterns and relationships. They are more adept at identifying underlying principles than at memorizing specific data. NTs couple a theoretical framework and the tendency to extrapolate beyond the details, so that new principles can be seen. It may be inferred that projects involving research and state-of-the-art development seemingly attract more NTs, whereas those work concerned with maintaining and enhancing software systems tend to allure more STs due to their practical side.

On the other hand, many NFs and SFs are drawn to fields like psychology and school teaching because of their concern for others; the technical aspects of computing hold little long-term attraction to them. It could be expected that some of them may find their niche in the less technical, more people-oriented aspect of software development. For example, NFs and SFs would possibly be happier as software engineers with direct user contact than they might be developing micro-code for a new micro-processor.

The personality type most prominent is a combination of introversion, sensing, thinking and judging. ISTJs assume responsibility readily and tend to persevere. From the data it can be deduced that the majority of software engineers (ISTJ) are technically oriented and prefers working with facts and reason rather than with people. According to the attributes associated with each type by the MBTI theory, we could infer that ESTJs, ESTPs, ENTJs and ENTPs seem to make good systems analysts due to their thinking ability to solve organizational problems and to communicate with other people. ESTPs and ENTPs will often prefer to leave the implementation of their designs to others (*P* factor). On the other hand, the ESTJs and ENTJs are likely to follow a project through the end; thus, achieving the closure that Js seek.

ISTPs appear to be excellent programmers as they have great skills to spot the centre of a problem and seem to find practical solutions, like the *gurus*. From previous results, it is believed that INTPs perform better in scientific programming. In effect, INTP, consistent with their reputation, are likely to be the ones with the skills critical in the early phases of an innovative project or a new field. Their emphasis on problem finding at the expenses of problem solving is similar to that of many academic researchers. INTJs have a high need to achieve, although a low drive to socialize with other people, they are known in the field as *wizards*. ISTJs, meanwhile, make good software engineers as they are perseverant and oriented towards results. As a consequence of the J preference, ISTJs and ESTJs tend to be more organized and to do more planning than the others. In the software industry, ESTJs and ISTJs are more likely to actively seek a management position, while the ISTPs, ESTPs, INTPs and ENTPs would often be very happy pursuing a technical career path.

5. Conclusions

The current study demonstrates that software engineers are a unique group of individuals. Although software engineering attracts people of all psychological types, certain traits are clearly more represented than others in this field. These findings do not mean that career success relates to the number of subjects of a type. The fact that ISTJs outnumber any other type does not mean that they are perceived to be the best in the area. Ackerman (1996) suggests that although interests and personality types may play a role in the selection of a career, they may not predict success in that area. Additionally, Sodan (1999) claims that for personal work as well as relations to be successful, a number of psychosocial qualities are required, and she proposes a model based on the Yin/Yang duality. Due to the diverse nature of software engineering, it is widely believed that no personality instruments will ever accurately predict success in this field.

As a matter of fact the software field is dominated by introverts, who typically have difficulty in communicating with the user. This may partially explain why software systems are notorious for not meeting users' requirements. When software engineers discuss how a task need to be accomplished, the majority tend to be poor at verbalizing how the task affects the people involved. In fact, the greatest difference between software engineers and the general population is the percentage that takes action based on what they think rather than on what somebody else feels. That does not help bring the software engineers closer to the user.

It takes variety to conquer variety. Putting it in software terms, it takes a variety of skill and personalities to solve the myriad of problems related to software development. It might be suggested that organizations would be well served by a conscious attempt to diversify the styles or personalities of its software engineers. Nowadays there are very few solo performers in most software organizations; people have to work together in teams of some sort, and it is almost always good to have some diversity on the team in terms of psychological type. In other words, better software will result from the combined efforts of a variety of mental processes, outlooks and values.

References

Ackerman, P.L., 1996. A theory of adult intellectual development: process, personality, interests, and knowledge. Intelligence 22, 229–259.

Blatt, S.J., 1986. Where have we been and where are we going? Reflections on 50 years of personality assessment. Journal of Personality Assessment 50 (3), 343–346.

Buie, E.A., 1988. Psychological type and job satisfaction in scientific computer professionals. Journal of Psychological Type 15, 50–53.

Bush, C.M., Schkade, L.L., 1985. In search of the perfect programmer. Datamation 31 (6), 128–132.

Lyons, M.L., 1985. The DP psyche. Datamation 31 (16), 103–110.

Myers, I.B., McCaulley, M.H., Quenk, N.L., Hammer, A.L., 1998. MBTI Manual: A Guide to the Development and Use of the Myers–Briggs Type Indicator, 3rd Edition. Consulting Psychologists Press, Palo Alto, CA.

Sitton, S., Chmelir, G., 1984. The intuitive computer programmer. Datamation 30 (20), 137–140.

Smith, D.C., 1989. The personality of the systems analyst: an investigation. ACM Computer Personnel 12 (2), 12–14.

Sodan, A.C., 1999. Toward successful personal work and relations—applying a Yin/Yang model for classification and synthesis. Journal of Social Behavior and Personality 27 (1), 39–71.