Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2003 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2003

Cognitive Dispositions of IS Personnel and Implications for Pair Programming

Venugopal Balijepally
University of Texas at Arlington

Radha Mahapatra
University of Texas at Arlington

Sridhar Nerur
University of Texas at Arlington

COGNITIVE DISPOSITIONS OF IS PERSONNEL AND IMPLICATIONS FOR PAIR PROGRAMMING

Venugopal Balijepally
University of Texas at Arlington
venugopal@uta.edu

RadhaKanta Mahapatra
University of Texas at Arlington
mahapatra@uta.edu

Sridhar Nerrur
University of Texas at Arlington
snerur@uta.edu

Abstract

The dominance of traditional heavyweight software development methodologies is being challenged by the advent of lightweight or agile methodologies that stress the need to be nimble and adaptive in a world where change is inevitable. A singularly distinctive feature of these new methodologies is their emphasis on people and their interactions. In a development environment that relies heavily on collaborative decision-making in all aspects of software development, an understanding of the cognitive dispositions of developers and their implications is imperative. This paper delineates a framework by reviewing the literature on cognitive dispositions of IS personnel. The implications of these findings for Pair Programming, a core concept in Extreme Programming, and the research issues that emerge as a consequence are also presented.

Keywords: Cognitive styles, programmer team, team diversity, agile methodologies, extreme programming, pair programming

Introduction

The nature of software development has undergone dramatic changes in the last two decades. While a silver bullet still eludes the grasp of the software community, the need to build quality systems that are flexible, scalable, and resilient to change has engendered a host of approaches that challenge the wisdom that has hitherto guided software development. The motivation for such methodological changes stems not only from turbulent business environments and the rapid strides made in technology (for example, new programming languages and tools), but also from a growing realization that existing approaches are woefully inadequate.

A set of methods, collectively called Agile Development Methodologies, is now beginning to capture the imagination of software developers. Extreme Programming (XP), Feature-Driven Development, Crystal Methods, Scrum, Dynamic Systems Development, and Adaptive Software Development are some of the more popular agile methods. What makes these methods philosophically and pragmatically different from traditional software development methods is their emphasis on: 1) People rather than processes and tools; 2) Adaptation rather than optimization; 3) High quality working software rather than extensive documentation; 4) Customer involvement and collaboration in the development cycle; 5) Embracing change rather than making futile attempts to eliminate it; and 6) Adopting an approach that has short iterations of planning, organizing, and coding along with continuous integration rather than following rigid plans that attempt to anticipate exceptional conditions and changes that might arise (Beck 2000, Highsmith and Cockburn 2001; Cockburn 2002). In the words of Cockburn and Highsmith (2001), "Agility requires that teams have a common focus, mutual trust and respect; a collaborative, but speedy, decision making process; and the ability to deal with ambiguity." In addition to increased agility, the vaunted benefits of agile methods are increased productivity, higher quality, and greater satisfaction/enjoyment of developers. The growing interest in these methods is evident from a recent survey of global IT managers by the Cutter Consortium (Charette 2001). One half of the survey respondents expect to use agile approaches in over 50% of their projects by 2003, while 14% anticipate the use of agile methods in all their projects.

At the heart of Extreme Programming (XP), which is the most popular of the agile methodologies (Charette 2001), is the notion of pair programming, a technique in which two programmers jointly plan, strategize, code, and inspect the software they develop. Pair programming also ensures joint ownership of the code, which is another core concept of XP. There is some anecdotal evidence to indicate that collaboration improves both the performance and enjoyment of the whole problem solving process for programmers (Nosek 1998; Williams et. al. 2000; Williams and Kessler 2000; Cockburn and Williams 2001). However, pair programming has been the most controversial and difficult to implement among the XP practices (Glass 2001; English 2002). This could be a result of the traditional view of programming as a solitary activity (Weinberg 1971) and the ensuing mindset of programmers. Unfortunately, very little research has been conducted to understand this phenomenon of two programmers working together as a team.

We know from small group research that the composition of a team affects the team's performance (Gist et al. 1987; Volkema and Gorman 1998). Heterogeneous teams have been found to perform better in unstructured creative tasks, whereas homogeneous teams do well in more structured tasks. With methodologies becoming more people-oriented and team-oriented, there is an urgent need to understand the cognitive styles of IS personnel and their consequences for techniques such as pair programming. In this exploratory study, we provide an overview of the existing corpus of knowledge on cognitive dispositions of IS personnel. The insights gleaned from the overview are then used to examine the implications for programming teams, particularly in the context of pair programming.

Profiling Cognitive Dispositions of Individuals

There are several personality theories in vogue in the field of human psychology. Many instruments have been developed based on these theories to assess the cognitive dispositions and to map the personality profiles of individuals. For the purpose of this study we have categorized the instruments into two categories: Jungian and non-Jungian. The Jungian typology, operationalized by the Myers-Briggs Type Indicator (MBTI) and other instruments, has gained most popularity among IS researchers.

Jungian Typology and Instruments

The Jungian typology of human cognitive dispositions consists of eight factors that are arranged as bi-polar preferences along four dimensions (White 1984a):

Field of View:	Extroversion (E)....Introverson (I)
Perceptual dimension:	Sensing (S)....Intuition(N)
Judging dimension:	Thinking (T)....Feeling (F)
Way of Life:	Judging (J)....Perceiving (P)

The first dimension deals with the orientation of a person. Extroverts tend to be people oriented while Introverts are more comfortable with the world of ideas. The second dimension reflects how a person perceives information. A Sensing type person prefers hard facts while searching for information, whereas an Intuitive person looks for relationships and patterns among data. The third dimension deals with decision-making. A Thinking Type person makes decisions based on logic and analysis, while the Feeling type tends to rely on emotions and personal values. The last dimension relates to how a person deals with the world. The Judging type person tends to prefer a planned, orderly life, while the Perceiving type tends to prefer a more spontaneous and flexible life. Although a given individual's cognitive disposition may include all eight factors to some degree, usually one factor in each dimension tends to be dominant. Thus an individual's cognitive disposition is characterized by one of the sixteen cognitive types listed in Table 1.

Table 1. Jungian Cognitive Types

ISTJ	ISFJ	INFJ	INTJ
ISTP	ISFP	INFP	INTP
ESTP	ESFP	ENFP	ENTP
ESTJ	ESFJ	ENFJ	ENTJ

Several instruments have been developed to identify a person's cognitive disposition based on Jung's typology. Here we briefly describe three such instruments that are popular among IS researchers.

Myers-Briggs Type Indicator (MBTI)

MBTI is the most widely used instrument for studying cognitive dispositions and personality characteristics of the non-psychiatric population. Its reliability and validity have been extensively studied and found to be adequate (Murray 1990). It is available in two versions. A long version, called Form G, includes 126 items, and an abbreviated version contains 50 items. Form G is considered to be the most appropriate instrument for measuring Jungian type. The Abbreviated Version was designed as a self scoring workshop instrument. It is easy to use but may lead to respondent bias. A limitation of MBTI is that it is not suitable for use in blind mail surveys (Mawhinney and Lederer 1988). MBTI has been used in understanding the learning styles of students (Rosati et. al. 1988), managerial cognition (Gardner and Martinko 1996), and group communication effectiveness (Cunningham 2000). Dafoulas and Macaulay (2001) have advocated using MBTI to determine how to organize software engineering teams. IS researchers have used MBTI to study profiles of IS personnel (see for example, Bostrom and Kaiser 1981; Capretz 2003) and to understand the effectiveness of software teams (Kaiser and Bostrom 1982; White 1984a; White 1984b; Rutherford 2001).

Personal Style Inventory (PSI)

PSI contains 32 items equally distributed along the four Jungian dimensions. It is considered to be inferior compared to MBTI with regard to validity and reliability measures (Mawhinney and Lederer 1988). However, its main advantage is that it can be used in mail surveys. IS researchers have used PSI to understand cognitive dispositions of IS personnel and to characterize the profiles of heavy users among end users (Mawhinney and Lederer 1986).

Kiersey Temperment Sorter (KTS)

KTS uses the Jungian typology to classify individuals into four intelligence types called temperaments. These are Rationals (NT), Ideals (NF), Artisans (SP), and Guardians (SJ). Thus, its classification scheme is somewhat different from the one advocated by Jung. More information on KTS can be found at **www.keirsey.com**, which also provides an online assessment of a person's cognitive profile. It was used by Rutherfoord (2001) to assess the cognitive profiles of software engineering students.

Non-Jungian Approaches

Non-Jungian approaches used by IS researchers include Minnesota Multi-Phasic Inventory (MMPI), Strong Vocational Interest Bank (SVIB), 16 Personality Factor Test (16PF), and Five Factor Model (FFM). These are briefly discussed in this section.

Minnesota Multiphasic Personality Inventory (MMPI)

MMPI was first published in 1943 and is extensively used for clinical diagnosis. It measures ten factors about an individual's personality: hypochondriasis, depression, hysteria, psychopathic-deviate, masculinity-femininity, paranoia, psychasthenia, schizophrenia, hypomania, and social introversion. Scores are obtained on four validity scales: Question, Lie, Validity, and Defensiveness. Personality scores are plotted on a profile sheet reflecting standard deviations from the mean (Sweetland and Keyser 1986). Of the 10 dimensions measured by MMPI, the social introversion dimension can be related to the Jungian dimension of introversion. The usefulness of MMPI to IS researchers stems from the fact that it provides an objective assessment of the personality traits of an individual that influence his/her personal and social adjustment skills (Barnes 1975).

Strong Vocational Interest Blank (SVIB)

SVIB measures the inclination of an individual to a particular profession. It consists of 400 questions that measure the respondent's preference for various occupations, jobs, recreational activities, types of people. This test can be used to identify the inclination of a person towards the programming profession. SVIB is known to be free of age bias and has been extensively used in counseling services (Mayer and Stalnaker 1968). It has been used by IS researchers to study the personality profile of IS personnel (Barnes 1975).

16 Personality Factor Test (16PF)

16PF test measures the personality profile of an individual using sixteen bipolar factors, such as, reserved vs. outgoing, trusting vs. suspicious, secure vs. apprehensive, etc. (Sweetland and Keyser 1986). The personality profile is created using a set of secondary bipolar factors such as, introversion vs. extraversion, low anxiety vs. high anxiety, etc., that are derived from the primary factors. The 16PF test has been extensively used and validated, primarily in occupational and clinical psychological studies. It has been used for clinical evaluations, personnel selection and placement, vocational and educational guidance and marriage counseling. IS researchers have used it to understand the personality profiles of IS personnel (Larkin 1983; Moore 1991).

Five Factor Model (FFM)

FFM, also called Big Five Model, represents personality traits along five dimensions – *Extraversion* representing the degree of sociability and gregariousness, *Conscientiousness* representing amount of organization, commitment and persistence, *Agreeableness* representing degree of friendliness and trust in others, *Negativity* representing level of tolerance for stress and personal adjustment and *Openness* representing level of openness to new ideas and experiences. There appears to be wide consensus emerging in psychology literature that these five factors are useful in categorizing personality attributes (Digman 1990). FFM has been found to be stable over time (Costa and McCrae 1997) and generalizable across cultures (McCrae and Costa 1997). Adjective Check List (ACL) is one of several instruments that operationalize FFM and has been found to have strong construct validity (Gough and Heilbrun 1983). While FFM has been extensively used in psychology, its use in IS research is relatively new. Clark et. al. (2003) used FFM to identify the characteristics of exceptional software developers.

Review of IS Research on Cognitive Disposition

We have categorized research on cognitive dispositions of IS personnel using two dimensions: instrument type (Jungian vs. Non-Jungian) and the context of study (individual vs. team). Figure 1 lists IS research papers classified into four quadrants using this framework. Some papers that address a broader set of issues are appropriately listed in more than one quadrant in Figure 1. This list does not include research papers that investigate cognitive issues outside the IS context. Also excluded from the list are papers on cognitive dispositions of end-users, which sometime ago formed a vibrant research stream in the context of MIS and DSS design (see for example, Bariff and Lusk 1977; Benbasat and Taylor 1978; Zmud 1979). While we have made an attempt to create a comprehensive list of research papers we do not claim this to be an exhaustive list.

	Jungian Instruments	Non-Jungian Instruments
Individual	Barnes (1975) Bostrom and Kaiser (1981) Bush and Schkade (1985) Capretz (2003) Hardiman (1997) Kaiser and Bostrom (1982) Lyons (1985) Mawhinney and Lederer (1988) Sitton and Chmelir (1984) Teague (1998)	Barnes (1975) Clark et. al. (2003) Larkin (1983) Mayer and Stalnaker (1968) Moore (1991)
Team	Kaiser and Bostrom (1982) Rutherfoord (2001) White (1984a) White (1984b)	

Figure 1. Research on the Cognitive Dispositions of IS Personnel

Instruments based on the Jungian typology seem to be most popular among IS researchers, with MBTI being the preferred instrument of measurement. Much of the research on the cognitive characteristics of IS personnel investigate typical cognitive profiles that are likely to predict success of individuals in the field of computing. Research in team context, on the other hand, try to unravel the differences between successful and unsuccessful IS teams in terms of the cognitive profiles of their members. The notion that personality and cognitive styles of members can impact team performance (Driskell et. al. 1987) seems to have influenced the stream of research on IS teams.

IS personnel are found to be predominantly Introverted (Barnes 1975; Lyons 1985; Bush and Schkade 1985; Capretz 2003) and Thinking (Barnes 1975; Lyons 1985; Teague 1998; Capretz 2003) types. Programmers tend to be mostly Thinking types with near exclusion of Feeling types. While some studies suggest no difference among IS personnel on the Perceptual dimension (Sensing vs. Intuition) (Lyons 1985; Kaiser and Bostrom 1982), others claim Sensing to be a dominant characteristic (Teague 1998; Smith 1989; Capretz 2003). With regard to Judging vs. Perceiving dimension, some studies suggest IS personnel to be predominantly Judging type (Lyons 1985; Teague 1998; Smith 1989) while others indicate no dominant characteristic on this dimension (Kaiser and Bostrom 1982; Sitton and Chmelir 1984). It is unclear whether the conflicting results among these studies reflect a change in the profile of IS personnel over time. With regard to combination of dimensions, programmers tend to be predominantly introverted and thinking types (Lyons 1985; Teague 1998; Bush and Schkade 1985). Systems analysts, on the other hand, while predominantly being thinking types tend to include both introverts and extroverts (Lyons 1985). It is also reported that people with NT and SJ profiles make the best programmers (Hardiman 1997; Teague1998).

Studies using non-Jungian instruments suggest that programmers are defensive (Barnes 1975) and conservative (Moore 1991). They share interests with persons engaged in scientific and professional occupations (Mayer and Stalnaker 1968) and with persons whose vocations include mathematics, nature, and music (Barnes 1975). Their interests differ from those engaged in occupations associated with gregariousness such as public speaking or politics (Barnes 1975). IS managers were found to be less warm and outgoing, more aggressive and assertive, less adventurous, less trusting and accepting of conditions, more self-sufficient and resourceful, more impulsive and more tense than other business executives (Moore 1991). They do not share characteristics traditionally associated with management but have profiles similar to programmers and systems analysts. This could be explained by the fact that IS managers tend to rise from the ranks of programmers/analysts (Moore 1991). Clark et. al. (2003) found that exceptional IT developers tend to be extroverts, while exceptional IT students in terms of GPA tend to be introverts. Both groups however tend to be conscientious.

Research on IS teams has almost exclusively relied on the Jungian typology. A significant finding in this research stream emphasizes the value of diversity in team composition as a predictor of success. Inclusion of Feeling types to supplement Thinking types in the project team is found to be critical for the success of programmer teams (Kaiser and Bostrom 1982; White 1984a). Heterogeneous teams with diverse cognitive dispositions have been found to employ more open and varied approaches to problem solving compared to homogeneous teams (Rutherfoord 2001). Heterogeneity is found to be a useful team characteristic while solving unstructured tasks whereas homogeneity in team composition is desirable for structured tasks (White 1984b). Team members with complementary cognitive profiles have been found to make up for one another's limitations, thus further reinforcing the importance of diversity in the context of IS teams (Shneiderman 1980).

Implications for Pair Programming

The review of IS research presented in the previous section suggests that programmers tend to be a homogeneous group characterized by introversion and thinking types with near exclusion of extroverts and feeling types. Traditionally, the programmer's job involved building systems based on specifications. It placed heavy demand on logical thinking with little need for inter personal communication and interaction. Therefore, it is not unusual that the profession attracted people of a particular disposition. The introduction of agile methods with their emphasis on iterative development and programmer teams is adding new dimensions to the programming task. The programmer is now required to work in a team with joint ownership of code. He/she also has to work closely with the users, thus making additional demand on his/her inter personal and communication skills. One issue that needs to be investigated is how to cope with this changing demand on the programmer's skill set and how to enhance the diversity in the profiles of programmers. At the organization level, this may require planned interventions by the human resources department to recruit and retain a diverse group of individuals for the software developer job. To meet this recruiting need, effort may be required at the University level to counsel college students about the changing role of programmer with a view to enhance the diversity among IS majors with regard to their cognitive dispositions.

Research on IS teams has demonstrated the importance of diversity in the composition of software project teams. It has also revealed the existence of interaction effect between task characteristic and team composition. While we can learn from this stream of research, further investigation is required to evaluate these conclusions in the context of pair programming. This is due to the fact that the group dynamics within a typical 2 person pair programming team (a dyad) is likely to be different from that within a larger software team.

Since team composition (in terms of the cognitive disposition of its members) and task characteristics affect team performance, a critical research issue in the context of pair programming is how to form effective teams. Guidelines need to be developed to assist IS managers in forming programming teams.

The Jungian typology is most widely used in IS research with MBTI being the instrument of choice. The intuitiveness of the Jungian typology and the widespread use of MBTI in prior research make it appealing to use in the context of research on pair programming. At the same time new measurement methods such as FFM need to be evaluated to assess its usefulness vis-à-vis MBTI.

Conclusion

The need to deliver high-quality software in a cost-effective and timely manner has ushered in a new breed of methods called agile or lightweight methodologies that are as refreshingly different as they are revolutionary. The emphasis that they place on people and their interactions rather than on processes grew out of the firm belief that through communication and collaboration competent people can achieve superior performance and derive more enjoyment. However, the ability to effectively communicate and collaborate on problem-solving strategies, planning, coding, reviewing code, and the like, depends largely on the cognitive leanings of the team members. The fact that most programmers and many analysts are little accustomed to sustained collaborative efforts in traditional software development makes it all the more necessary for managers to understand the implications of cognitive dispositions of IS personnel for determining the composition of the team.

This study reviews the various instruments that have been used in the past and presents a framework that provides the foundation for comprehending the implications of cognitive dispositions for collaborative software development. Studying the cognitive dispositions of IS personnel in the context of agile methodologies, particularly XP, would provide insights that might have a significant influence on their adoption and performance. Cockburn (1999) reiterates that people should be investigated as first order drivers of success more than any other process factors in any IS context, and the new paradigm of XP and pair programming should be the perfect opportunity to bring people back to the center stage.

References

Bariff, M. L. and Lusk, E. J., "Cognitive and Personality Tests for the Design of Management Information Systems," Management Science (23:8), 1977, pp. 820-829.

Barnes, P., "Programmer Paranoia Revisited," Proceedings of the Thirteenth Annual SIG/CPR Conference, Toronto, Ontario, Canada, 1975, pp. 114-131.

Beck, K., Extreme Programming Explained: Embrace Change, Addison Wesley Longman Inc., Reading, MA, 2000.

Benbasat, I. and Taylor, R. N., "The Impact of Cognitive Styles on Information System Design," MIS Quarterly (2:2), 1978, pp. 43-54.

Bostrom, R. P., and Kaiser, K. M., "Personality Differences within Systems Project Teams: Implications for Designing Solving Centers," Proceedings of the Eighteenth Annual Computer Personnel Research Conference, Washington D.C., 1981, pp. 248-285.

Bush, C. M and Schkade, L. L., "In Search of the Perfect Programmer," Datamation (31:1), 1985, pp. 128-132.

Capretz, L. F., "Personality Types in Software Engineering," International Journal of Human-Computer Studies (58:2), 2003, pp. 207-214.

Charette, R., "The Decision is In: Agile versus Heavy Methodologies," Cutter Consortium Free Research Articles (2:19), 2001, Agile Project Management Update, **http://www.cutter.com/freestuff/epmu0119.html**, accessed on 03/14/03.

Clark, J. G., Walz, D. B., and Wynekoop, J. L., "Identifying Exceptional Application Software Developers: A Comparison of Students and Professionals," Communications of the Association for Information Systems (11:2), Article 8, 2003, pp. 137-155.

Cockburn, A. and Highsmith, J., "Agile Software Development 2: The People Factor," IEEE Computer (34:11), 2001, pp. 131-133.

Cockburn, A., "Agile Software Development Joins the "Would-Be" Crowd," Cutter IT Journal (15:1), 2002, pp. 6-12

Cockburn, A., "Characterizing People as Non-Linear, First Order Components in Software Development," Humans and Technology Technical Report, TR 99.05, Oct 1999 @acm.org accessed on 02/20/03.

Cockburn, A., and Williams, L., "The Costs and Benefits of Pair Programming," in Extreme Programming Examined, G. Succi, and M. Marchesi, (eds.), Addison Wesley, 2001, pp. 223-248.

Costa, P. T. Jr., and McCrae, R. R., "Longitudinal Stability of Adult Personality," in Handbook of Psychology, R. Hogan, J. A. Johnson, and S. R. Briggs (eds.), Academic Press, San Diego, CA, 1997, pp. 269-290.

Cunningham, P. B., "Improved Communication and Teamwork Through Use of Myers Briggs Type Indicator," Proceedings of the 1st Austin Workshop on Engineering Management in Technology-Based Organizations, Austin, TX, 2000, pp. 31-36.

Dafoulas, G. A., and Macaulay, L. A., "Facilitating Group Formation and Role Allocation in Software Engineering Groups," Proceedings of the ACS/IEEE Conference on Computer Systems and Applications, Beirut, Lebanon, 2001, pp. 352-359.

Digman, J. M., "Personality Structure: Emergence of Five-Factor Model," Annual Review of Psychology (41:1), 1990, pp. 417-440.

Driskell, J. E., Hogan, R. and Salas, E., "Personality and Group Performance," in Group Processes and Inter-group Relations, C. Hendrick, (ed.), Sage Publications, Beverly Hills, CA, 1987, pp. 91-113.

English, A., "Extreme Programming: It's Worth a Look," IT Pro (4:3), 2002, pp. 48-50.

Gardner, W. L., and Martinko, M. J., "Using the Myers-Briggs Type Indicator to Study Managers: A Literature Review and Research Agenda," Journal of Management (22:1), 1996, pp. 45-83.

Gist, M.E., Locke, E. A., and Taylor, M. S., "Organizational Behavior: Group Structure, Process, and Effectiveness," Journal of Management (13:2), 1987, pp. 237-257.

Glass, R. L., "Extreme Programming: The Good, the Bad, and the Bottom Line," IEEE Software (18:6), 2001, pp. 111-112.

Gough, H. G., and Heilbrun, A. B., The Adjective Check List Manual, Consulting Psychologist's Press, Palo Alto, 1983.

Hardiman, L. T., "Personality Types and Software Engineers," Computer (30:10), 1997, pp. 10-10.

Highsmith, J., and Cockburn, A., "Agile Software Development: The Business of Innovation," IEEE Computer (34:9), 2001, pp. 120-122.

Kaiser, K. M., and Bostrom, R. P., "Personality Characteristics of MIS Project Teams: An Empirical Study and Action-Research Design," MIS Quarterly (6:4), 1982, pp. 43-60.

Larkin, J. E., "The Psychology of DP Professionals: A Career Planning Tool," Computerworld (17), 1983, pp. 6-15.

Lyons, M., "The DP Psyche," Datamation (31), 1985, pp. 103-110.

Mawhinney, C. H., and Lederer, A. L., "Validation of a Jungian instrument for MIS Research," ACM SIGCPR Computer Personnel (11:3), 1988, pp. 2-9.

Mawhinney, M. G., and Lederer, A. L., "Manager Personality Type and End-User Computing," Proceedings of the Twenty-Second Annual Computer Personnel Research Conference, Calgary, Canada, 1986, pp. 117-128.

Mayer, D. B., and Stalnaker, A. W., "Selection and Evaluation of Computer Personnel – The Research History of SIG/CPR," Proceedings of the 23rd ACM Conference, 1968, pp. 657-670.

McCrae, R. R, and Costa, P. T. Jr., "Personality Trait Structure as a Human Universal," American Psychologist (52:5), 1997, pp. 509-516.

Moore, J. E., "Personality Characteristics of Information Systems Professionals," Proceedings of the Conference on SIGCPR, Athens, Georgia, 1991, pp. 140-155.

Murray J. B., "Review of Research on the Myers-Briggs Type Indicator," Perceptual and Motor Skills (70), 1990, pp. 1187-1202.

Nosek, J. T., "The Case for Collaborative Programming," Communications of the ACM (41:3), 1998, pp. 105-108.

Rosati, P., Dean, R. K., and Rodman, S. M., "A Study of the Relationship Between Students' Learning Styles and Instructor's Lecture Styles," IEEE Transactions on Education (31:3), 1988, pp. 208-212.

Rutherfoord, R. H., "Using Personality Inventories to Help Form Teams for Software Engineering Class Projects," Proceedings of the 6th Annual Conference on Innovation and Technology in Computer Science Education (ITICSE), Canterbury, U.K., 2001, pp. 73-76.

Shneiderman, B., Software Psychology: Human Factors in Computer and Information Systems, Winthrop Publishers Inc., Cambridge, Massachusetts, 1980.

Sitton, S., and Chmelir, G., "The Intuitive Computer Programmer," Datamation (30), 1984, pp. 137-141.

Smith, D. C., "The Personality of the Systems Analyst: An Investigation," ACM SIGCPR Computer Personnel (12:2), 1989, pp. 12-14.

Sweetland, R. C., and Keyser, D. J., Tests: A Comprehensive Reference for Assessment in Psychology, Education and Business, Second Edition, Test Corporation of America, Kansas City, 1986.

Teague, J., "Personality Type, Career Preference and Implications for Computer Science Recruitment and Teaching," The Proceedings of the Third Australasian Conference on Computer Science Education, The University of Queensland, Australia, 1998, pp. 155-163.

Volkema, R. J., and Gorman, R. H., "The Influence of Cognitive-Based Group Composition on Decision-Making Process and Outcome," Journal of Management (35:1), 1998, pp. 105-121.

Weinberg, G. M., The Psychology of Computer Programming, Van Nostrand Reinhold, New York, 1971.

White, K. B., "MIS Project Teams: An Investigation of Cognitive Style Implications," MIS Quarterly (8:2), 1984a, pp. 95-101.

White, K. B., "A Preliminary Investigation of Information Systems Team Structures," Information & Management (7:6), 1984b, pp. 331-335.

Williams, L. A., and Kessler, R. R., "All I Really Need to Know About Pair Programming I Learned in Kindergarten," Communications of the ACM (43:5), 2000, pp. 108-114.

Williams, L., Kessler, R. R., Cunningham, W., and Jeffries, R., "Strengthening the Case for Pair Programming," IEEE Software (17:4), 2000, pp. 19-25.

Zmud, R. W., "Individual Differences and MIS Success: A Review of Empirical Literature," Management Science (25:10), 1979, pp. 966-979.