Does software developers' motivation affect performance?

Author

Wong, Zoe

Published

2004

Conference Title

Pacific Asia Conference on Information Systems (PACIS) 2004 Proceedings

Version

Version of Record (VoR)

December 2004

Does software developers' motivation affect performance?

Yuk-Kuen Wong
University of Technology Sydney

Does software developers' motivation affect performance?

Yuk Kuen Wong
Faculty of Information Technology
University of Technology Sydney
zoewong@it.uts.edu.au

Abstract

Past research suggests that motivation is the critical factors influencing software review performance. However, there is no published evidence investigating the important relationship between motivation and performance. As a result, the aim of this paper is to examine the effect of motivation on software review performance. A total of 205 developers voluntarily participated in a questionnaire survey in Australia in 2003. The main findings include: (1) contingency is the key driver to the performance, (2) confidence also has a positive effect on performance, (3) both commitment and competency has not value to software review performance.

Keywords: Motivation, performance, software review

1. Introduction

While information systems can be considered a well-established discipline, software development projects are still prone to failure. Even if a software project is not classified as a failure, the general level of software quality leaves room for much improvement. As a result, techniques such as software review for improving software quality are important. Boehm and Basili (Boehm and Basili, 2001) stated that one of the most prevalent and costly mistakes made in software projects today is deferring the activity of detecting and correcting software problems until the end of the project. Hence the cost of rework in the later stages of a project can be greater than 100 times of the project costs (Fagan, 1976). Thus about 80 percent of avoidable rework comes from 20 percent of defects (Boehm and Basili, 2001).

Software review (inspection) was originally introduced by Fagan (1976). The review process essentially includes six major steps: planning, overview, individual preparation, group review meeting, rework and follow-up (Fagna, 1976, 1986).

1. Planning - organize and prepare the software review, typically for preparing the review materials and review procedure, forming review team and scheduling review meeting, selecting review participants and assigning roles.
2. Overview - author explains overall scope and the purpose of the review.
3. Individual preparation - individual reviewers analyze and review the software artefact.
4. Group review meeting - find errors, sometimes also called "logging meeting". Review teams correct and the reader summarizes the work.
5. Rework - defect correction, which involves the author in resolving problems by reviewing, revising and correcting the identified defect or by decreasing the existence of errors of the software artefact.
6. Follow up - validate the correction quality and decide if re-inspection is required.

Since Fagan (1976) introduced software review (inspection) as an important technique to assure the quality of software projects, researchers have investigated ways to improve software review performance. Wong (2003) recently suggested that implicit inputs (i.e. developer characteristics) are critical to the software review performance. Also, the Explicit-Implicit Input-Output (EIIO) model also suggests that implicit inputs such as software developers' characteristics (e.g. motivation behavior) are important to a successful software review (Wong, 2004). However, there is no publication empirically investigation the important relationship between motivation and performance. As a result, this paper aim is to develop identifying the important the effect of motivation on the software review performance.

2. Literature review

2.1 Motivation

One of critical factors in software review performance is the human behavior. Software review is driven by its participants, i.e. the members of a project team. Hence, the success or failure of software review depends on human factors (Fagan, 1986). If reviewers are unwilling to perform software review, all efforts will be deemed to fail.

Boehm and Basili (2001) strongly believe that increasing the amount of effort, leads to the higher performance in the review process. This statement is consistent with a number of studies that the more effort of reviewers willing to perform in defect detection process, the higher defects detection rate can be achieved. From the psychological approach, Hultman (1988) claim that that individual performance is based on the 4'Cs of input as shown in Table 1. The 4'Cs can be classified as (Hultman, 1998):

1. Commitment – individual is willing to handle the task. (Is it important to me?)
2. Confidence – individual has self-confidence to deal with the task. (Can I do it?)
3. Competence – whether individual has actual capability to perform a task. (How can I to do it?)
4. Contingencies frustration – whether individual has support from others. (Do I have support?)

Each input is measured by the attributes that include values, belief, competence and favour. Behavioural outputs can be classified into positive and negative behaviour. Individuals with positive behaviour would attain better performance. On the other hand, individuals with negative behaviour will achieve lower performance (Hultman, 1998).

Table 1: Inputs and Outputs of Performance

Inputs	Attributes	Behavioural Outputs	
		Positive Aspects	*Negative Aspects*
Commitment	Values	Motivated to pursue success (drives past fear deciding choosing, embracing persevering, initiating, affirming)	Motivated to avoid failure (succumbs to fear by running away.
Confidence	Beliefs	Rational, logical, reasonable (Positive self-talk, thinks of reasons to try)	Irrational, illogical (unreasonable, negative self-talk, thinks of reason to not try or to give up)
Competence	Present/ Absent	Successful (performs effectively, keeps trying)	Unsuccessful (performs ineffectively, give up or doesn't try)
Contingency	Favour	Supportive (cooperating, sharing, forgiving, helping accepting, forgiving)	Divisive (get even by arguing, blaming, criticizing, condemning, retaliating, sabotaging)

The behavioral outputs have generally theorized two mechanisms whereby extrinsic and intrinsic motivation (Deci, 1985). The extrinsic supports (e.g. rewards) enhance feelings of individuals' competence or self-efficacy; this may increase intrinsic motivation. Further, when extrinsic supports cause individuals to attributes their behaviors to external rather than internal sources, rewards may possible decrease intrinsic motivation (Deci, 1985). Previous researches have appealed this intervening motivation, perceived competence, commitment, contingency and confidence cause of performance (Hultman, 1998; Deci, 1985).

2.2. Performance

In the human performance theory, Campell's theory (1990) suggests that experience, knowledge, and motivation & effort could affect error detection performance (see Equation 1).

Equation 1: Determine of individual task performance

Performance = f (declarative knowledge, procedural knowledge and skills, motivation)

In particular, he proposed that performance is a function of individual's declarative knowledge, procedural knowledge and skill, and motivation. Declarative knowledge is defined as knowledge required to complete a task. Procedural knowledge refers to skill-based knowledge about how effectively to perform a task. Declarative knowledge and procedural knowledge are based on education, training, experience and motivation. Motivation refers to a function of three choices: the choice to expend effort, the choice of the degree of effort to expend, and the choice to persist in task performance.

Campell's theory (Campell, 1990) suggests that motivational influence and experience can affect job performance through changes in, declarative knowledge, procedural knowledge and skill, and/or the three choices.

In the context of a software review, at the completion of defect detection, there are two types of quantitative outputs: the reviewed software artefact, and quantitative outcomes such as defect information recorded in defect forms (e.g. number of defects). There are four possible outcomes of defect detection. These include:

- hit (defect exists and is successfully detected),
- miss (defect exists but is not detected),
- false positive (defect does not exist but is wrongly identified), and
- correct rejection (defect does not exist and is not identified).

The probability of results in each of these cells is determined by the performance of individuals and the interaction between those individuals in a group.

3. Hypotheses

Software review literature suggests that motivation is important to software review performance. Motivations can be classified into four categories that consist of commitment, competency, confidence and contingency.

3.1 Commitment

Psychology theory suggests that commitment related to values (Korman, 1997). This is related to personal beliefs about what is important in life. Value directs the process of choosing among alternatives, and they guide our every action whether we are aware of it or not. When an individual perceives a task is important, they are willing to increase their commitment in performing the task. As a result, we formulate

H1: Commitment has positive on performance.

3.2 Competency

To evaluate actual ability and skills on performance, a simple question can be asked: How can you do a task? In general, competent performance creates positive feelings, reinforcing confidence and commitment. If past experience is the best indicator of future behaviour. The composite measures for competency include effectiveness and ability to do well. Therefore,

H2: Competency has positive effect on performance.

3.3 Confidence

Individuals frequently perform below potential to avoid potential failure and they resist change because of a fear of potential performing poorly, in which avoidance values as seeking to protect their self-esteem and job securities (Hultman, 1998). On the other hand, confident individuals believe they better job than others, resulting attain more benefits or reward from the company. The items measures for the confidence construct comprise confidence in ability, confidence in comparing with others and teaching others.

H3: Confidence has positive effect on performance.

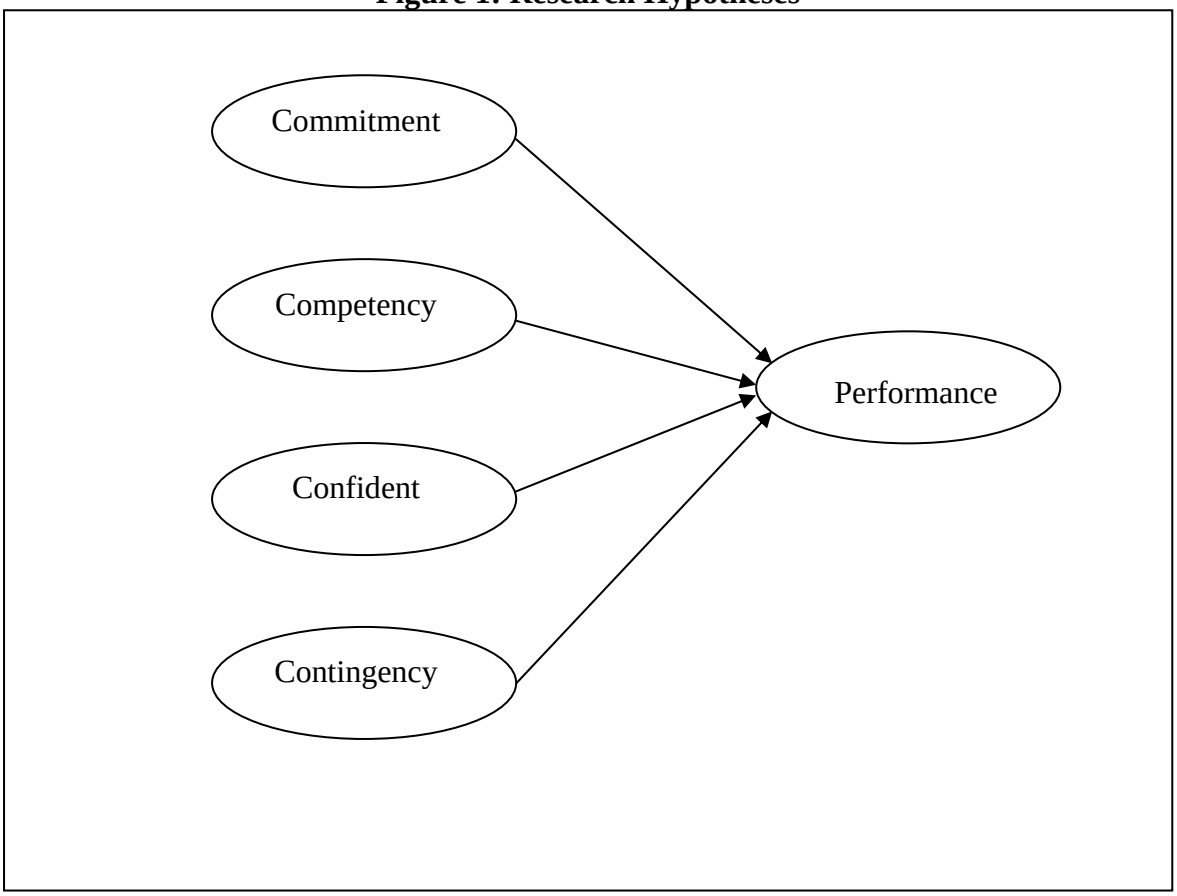
3.4 Contingency

With contingency, it is concerned largely with the availability and location of material and human resources, such as staff, money, time, equipment, supplies, and training (Hultman, 1998). The contingency is concerned largely with individuals' agreements of supports and from the company (Deci, 1985). Favorable contingencies bring about positive feelings and encourage supportive behaviors. As a result, the composite measures of contingency construct include agreement; supports and encouragement from the company.

H3: Contingency has positive effect on performance.

A model shown as Figure 1 presents all the above-formulated hypotheses of the process affect performance.

Figure 1: Research Hypotheses



4. Methodology

The objective of this study is to test the four hypotheses of a model of software review, postulating relationships between motivation and performance. To ensure sufficient variance in the data, a random sample of data from companies in six Australian states (i.e. New South Wales (NSW), Victoria (VIC), Queensland (QLD), South Australia (SA), Western Australia (WA) and Tasmania (TAS) was investigated.

4.1 Questionnaire design

The questionnaire was originally developed by Deci (1985) and modified by Wong1 in 2003. All questions were five-point-scale, close-ended questions, and reflective indicators except the measurement of performance construct measures with formative indicators.

The measurements of commitment construct include:

- Motivation - I am motivated to participate in software reviews
- Effort - I put a lot of effort into finding defects
- Willingness - I am willing to participate in software reviews

The measurements of competency construct include:

- Effectiveness - I can find defects in an effective way
- Do well - I do well in software reviews

The measurements of confidence construct include:

- Confident in ability - I am confident that I am capable of finding defects
- Confident in performing better than others - I am confident that I can find more defects than other team members
- Confident in teaching others - I am confident that I can teach my defect detection skills to others

The measurements of contingency construct include:

- Agreement - I agree with the way my company conducts software reviews
- Support - I participate in software reviews because I have support from my company
- Encouragement - I participate in software reviews because my company encourages us to do so

The measurements of performance construct include:

- True defects (TR) - defects that actually exist and have been successfully detected
- False positive (FA) - defects that do not exist but were wrongly identified
- Net defects (NE) - true defects minus false positive.
- Total issues (TL) - true defects plus false positive.

1 The questionnaire can be obtained from Y. K. Wong, zoewong@it.uts.edu.au/ wongyukkuen@hotmail.com.

Note that the measurement of performance is based on the average number of defects found by an individual respondent compared with the average number of defects found by review teams in their company. Individuals were asked to rate themselves in comparing their group members.

4.2 Questionnaire Pre-tests

To ensure the reliability and validity of the study, the mail survey instrument pre-tests were performed. There are four stages of the pre-test – experts review, focus groups, reconfirming review and a pilot study of the self-administrated questionnaire survey.

In stage one, five academic experts evaluated the questionnaire. Based on their recommendations, a few items were added, deleted and modified in the initial questionnaire.

In stage two, the revised questionnaire was tested in three focus groups. The purpose of the focus group meeting is to ensure the quality of the instrument (i.e. make sure it is understandable and readable). The communication medium was face-to-face. Two academics information systems, two PhD students in software engineering, five honours students in information systems and eleven undergraduate students in Information Technology Management were randomly assigned in a group of size of four, ten and six. They all have experience in software review. The focus group provided useful feedback about the perceived length of the questionnaire, the time required to complete the questionnaire, and the content of the questions. From the results of the focus groups, the suggestion to improve the wording of few items and the questionnaire appearance were implemented. In order to ensure reliability of the instrument, test re-test method of testing the reliability of indicators was also employed in three focus groups. The high correlations were found (from 0.81-0.95) and the high degree of significance gives considerable support for the survey instrument as a reliable measure. It is suggested the high correlations (rule of thumb is 0.8 or above). Upon completion of the focus group studies, three academic and two PhD students evaluated the revised the questionnaire before the final stage. From the comments and feedback, the wording of few items was changed.

In the final stage of the pre-test, the questionnaire was a pilot test on the practitioners from the selected software companies. Sixteen completed questionnaires were received from six companies. Based on the feedback from the pilot study test, a few questions on company information that was not strictly relevant for testing the hypotheses were deleted. Since the pre-test sample size is small, only frequency and descriptive analysis are conducted which indicated that all measures have reasonable range and variance.

In summary, pre-testing the questionnaire in three stages provided useful guidelines for designing the final survey instruments that is simple, well-presented, and has high content validity.

4.3 Samples

The main goal of the sampling process chosen was to capture as wide a range of software development companies in the study as possible. Software firms from the computer services category2 and from the top 500 companies listed in the Australian stock exchange were identified for this research.

---

2 **Categorized size of firms can be based on Australian Bureau of Statistics**.

The total number of companies selected for the study was 1380. From this a total of 205 developers voluntarily participated in the survey. The distribution of the companies sampled in Australia. The following provides a brief review of characteristics of the survey group: A majority of respondents were in age groups ranging between 20 to 60 years old (age range: 20 or below, 6.3%; 21-30, 13.7%; 31-40, 45.4%; 41-50, 21.5%; 51-60, 10.7%; 61 or above, 2.4%). Approximately 73.4% of these were male and 26.3% female. About 80% of the participants have university degrees. Approximately 79% are working in computer related occupations.

The number of participants have software industry experience are approximately 93%. About 89% had role experience in requirements review; 93% in design review, 91% in code review and about 78% in testing review. All subjects are industry practitioners and currently working in Australia.

5. Results

5.1. Reliability and Validity

In order to validate the content validity, several sources of data were used during the questionnaire development including questionnaire developed by other researchers; The Explicit and Implicit Input-Output "EIIO" model developed from relevant literature but independent from earlier instruments; and feedbacks from pre-tests respondents on the representativeness of questions.

Constructs reliability and validity tests were conducted. Cronbach's (1977) alphas of commitment, competency, confidence and contingency are between 0.846 and 0.934 respectively (see Table 1). Although 0.7 is generally regarded as the lower acceptable bound for alpha (Cronbach, 1977). But some authors suggested that it is not unusual to see published scales with lower alpha. It is suggests that below 0.6 is unacceptable, 0.6-0.65 undesirable, 0.65-0.7 minimally acceptable, 0.7-0.8 respectable and 0.8 or above very good (Cronbach, 1977). While performance is .8854, which shows good reliability based on Nunnally's Criteria A principal components factor analysis with varimax rotation exactly reproduced four factors, explaining 74 percent of the variance indicates both nomological and discriminate validity of the instrument (see Table 2).

Table 1: Reliability tests

	Cronbach alpha
Commitment	.9340
Competency	.9347
Confidence	.6460
Contingency	.9058
Performance	.8854

Table 2: Factor analysis

	Component				
	1	2	3	4	5
Willingness	0.91				
Effort	0.90				
Motivation	0.90				
Support		0.88			
Encourage		0.88			
Agree		0.894			
Effective			0.87		
Do well			0.86		
Con_teach				0.77	
Con_More				0.76	
Con_perform				0.59	
Total defects					0.912
Net defects					0.894
True defects					0.833
False positives					0.766

Extraction Method: Principal Component Analysis.

5.2 Hypotheses Tests

The significance level for all data analyzed was 0.05 (two tailed). Pearson's correlation test was used to test the relationships between four motivation constructs (i.e. commitment, competency, confidence, and contingency) and performance (number of defects found). Table 10 shows the results of the Pearson's correlation test. The results show that there is a strong positive relationship between contingency and performance ($r = .406$, $p < 0.01$); there is a positive relationship between confident and performance ($r = .154$, $p < 0.05$). These results indicate that hypotheses 3 and 4 are strongly supported. Interesting results demonstrate that that is no relationship between commitment and performance ($r = .018$, p= not significant). In addition, we found that there is no relationship between competency ($r = .067$, p= not significant). The results suggest that both hypotheses 1 and 2 are rejected. Figure 2 shows the revised model.

Table 3: Results of correlation analysis on the relationships between motivation and performance

	Performance
Commitment	.018
Competency	.067
Confidence	.154*
Contingency	.406**

** P< 0.01.

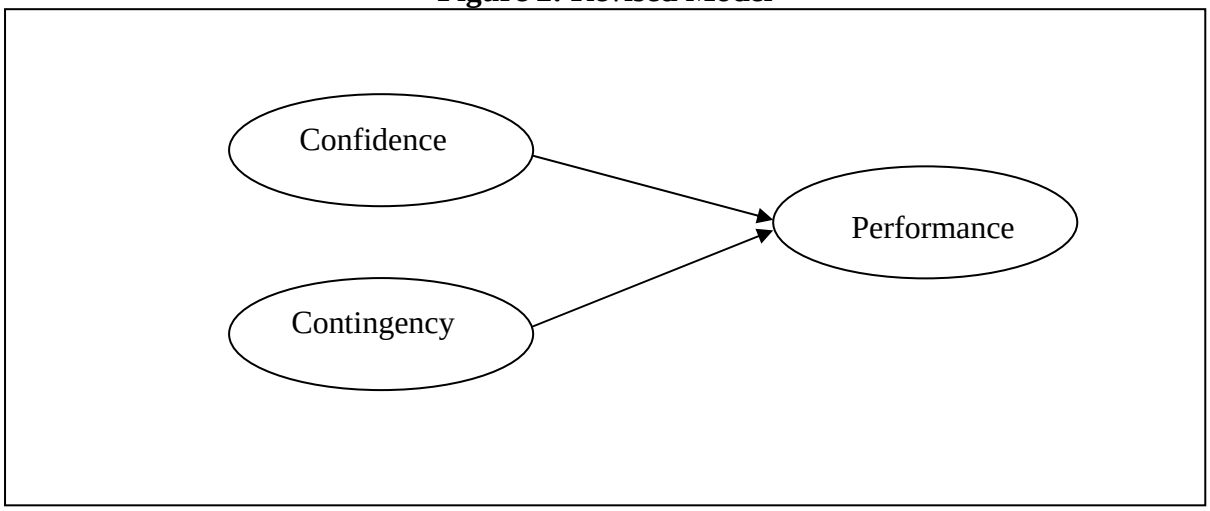
*P < 0.05.

Table 4: Regression Analysis

	Performance
Commitment	R^2 = .0003
Competency	R^2 = .004
Confidence	R^2 = .023*
Contingency	R^2 = .164**

** P< 0.01.
*P < 0.05.

Figure 2: Revised Model



6. Conclusion

The main goal of this study was to validate the relationships between motivation and performance in the software review process. The main findings of this study are summarized below.

- Rejected H1: Commitment has positive effect on performance.
- Rejected H2: Competency has positive effect on performance.
- Accepted H3: Confidence has positive effect on performance.
- Accepted H4: Contingency has positive effect on performance.

The results indicate that contingency is a key driver to the performance. It is suggested that the importance of supports and encouragement from the company have significantly effect on developers' performance. How developers perceive the way of conducting a software review task is critical to their motivation, as well as performance.

In addition, the levels of confidence are significant related to review performance. In particular, the levels of confidence are refer to (1) confidence in ability, (2) confidence in perform better than others and (3) confidence in teaching others.

On the other hand, surprising results show that both commitment and competency do not hold any value to software review performance. The overall results suggest that H3 and H4 are supported, whereas H1 and H2 are rejected. It is recommended that future research should focus on replication works are required to examine the proposed model.

7. References

Boehm, B. W., and Basili, B. R., “Software Defect Reduction Top 10 List”, IEEE Computer, (34:1), January 2001.

Campbell, J. P., “Modelling the Performance Prediction Problem in Industrial and Organizational Psychology”, in Dunnette M. D., Hough L. M. and (Eds.), Handbook of Industrial and Organizational Psychology, 2nd Edition, Consulting Psychologists Press Inc., Palo Alto, CA, 1990, pp.87-732.

Cronbach L. J., Aptitudes and Instructional Methods: A Handbook for Research on Interactions, Irvington, New York, 1977.

Deci, E. L., & Ryan R. M., Intrinsic Motivation and Self-determination in Human Behaviour, New York, Plenum, 1985.

Fagan M. E., “Design and Code Inspections to Reduce Errors in Program Development”, IBM System Journal, (15(\:3), 1976, pp.182-211.

Fagan M. E., “Advances in Software Inspections”, IEEE Transaction on Software Engineering, (12:7), July1986.

Hultman K. E., The psychology of performance management Training and Development Journal, 121, July, 1988, 34-39.

Korman A .K., Organizational Behavior, Prentice Hall, Englewood Cliffs, NJ, 1977.

Wong Y. K., “An exploratory study of software review in practice”, Proceeding of Portland International Conference on Management of Engineering and Technology PICMET’03, 20-24 July 2003.

Wong Y.K., “A Knowledge-based 'Explicit and Implicit Input-Output' Model for Analysis Software Review Performance”, Proceedings of IASTED International Conference on Software Engineering, Innsbruck, 17-19 Feb. 2004.