

Association for Information Systems AIS Electronic Library (AISeL)

AMCIS 1998 Proceedings

Americas Conference on Information Systems
(AMCIS)

December 1998

Computer-Mediated Group Support, Anonymity and Software Inspection Process

Padmal Vitharana

University of Wisconsin-Milwaukee

K. Ramamurthy

University of Wisconsin-Milwaukee

Follow this and additional works at: <http://aisel.aisnet.org/amcis1998>

Recommended Citation

Vitharana, Padmal and Ramamurthy, K., "Computer-Mediated Group Support, Anonymity and Software Inspection Process" (1998).
AMCIS 1998 Proceedings. 307.

<http://aisel.aisnet.org/amcis1998/307>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 1998 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Computer-Mediated Group Support, Anonymity and Software Inspection Process

Padmal Vitharana

K. Ramamurthy

School of Business Administration
University of Wisconsin–Milwaukee

Abstract

While considerable research in both software inspection and anonymity in Group Decision Support Systems (GDSS) exists individually, anonymity within the context of software inspection has not been explored. Anonymity becomes an important issue as researchers and practitioners advocate the use of computer-mediated inspection over traditional manual-based ones. Moreover, the emergence of new paradigms such as distributed and asynchronous environments for inspection only add to the importance of understanding the effects of anonymity on inspection outcomes. This research examines and proposes possible influences of group member anonymity on the outcome of computer-mediated software inspection.

Introduction

The importance of inspection in verifying software artifacts such as requirements specifications, design documents, and program code is well documented. Software inspection is a formal, structured method for examining software artifacts. The objective of inspection is to verify that the artifact being examined satisfies any conditions imposed on it. Fagan (1986) defines quality of inspection as its ability to detect all instances in which the work-product being inspected does not meet its requirements.

Theory Development

Software Inspection Process

According to Ackerman et al. (1989), software inspection process as defined by Fagan involves the following interacting components: 1) inspection product, 2) stages of inspection, 3) supporting infrastructure, 4) inspection team and its members' roles, and 5) data collection.

1. **Inspection Product.** Inspection product could be any work-product (e.g., specifications, design, code, documentation) that is an intermediary outcome of the software development process.
2. **Stages of Inspection.** The main goal of the inspection process is to identify and eliminate defects from a given work-product. To achieve this goal, the following six stages have been identified (Fagan 1986):
 - a. *Planning:* Establish schedules, choose inspectors, and distribute material.
 - b. *Overview:* Author of the work-product makes a presentation of the product to be inspected.
 - c. *Preparation:* Inspectors individually study the work-product, identify and record defects.
 - d. *Meeting:* Inspectors, as a group, examine the work-product, discuss identified defects, detect further defects, and arrive at an agreement.
 - e. *Rework:* Revise the work-product to correct the defects found earlier.
 - f. *Follow-up:* Certify revisions and re-inspect, if necessary, again following the above process.
3. **Supporting infrastructure.** The software inspection process must be planned and supported by a software inspection coordinator.
4. **The Inspection Team and Its Members' Roles.** The roles and responsibilities of team members are:
 - a. *Moderator* directs the inspection process.
 - b. *Author* is usually the person who produced the work-product being inspected.
 - c. *Reader* leads the inspection team members through the work-product during meeting.
 - d. *Recorder* classifies and records defects detected during meeting.
 - e. *Inspectors* inspect the work-product individually and as a group.
5. **Data Collection.** Inspection related data are collected throughout the software inspection process.

A primary factor that contributes to the success of inspection process is its participants. In order for the inspection process to be effective, the inspection team has to function cohesively with the ultimate goal of finding defects in the inspection product.

Automating the Inspection Process

Traditional inspection is an intensive manual process. With increasing pressures to reduce the amount of manual involvement, researchers and practitioners have recently focused their attention to computer-based support for inspection. There is some existing research on computer-based tools for software inspection (Brothers et al. 1990; Mashayekhi et al. 1993; Johnson et al. 1993; Knight and Myers 1993; Gintell et al. 1993). These tools look at how a computer-based inspection system can automate inspection and thus reduce the level of manual human intervention. Macdonald et al. (1995), in their meta-review of tool support for software inspection, note that by automating some parts of the inspection process and providing computer support for others, the inspection process can be made more efficient and also effective. They note that the support for inspection provided by most of the above-noted computer-based tools include document handling (e.g., line numbering and comments-linking), individual preparation (e.g., support checklists, display standards), meeting support (e.g., track the time spent by each inspector, audio support for distributed meetings), and data collection (e.g., gather metrics on comments). Although these tools do indeed automate some tasks of the inspection process, they do not address the effects of inspection team member anonymity on the success of software inspection, a key factor in GDSS research.

Anonymity in GDSS Research

One of the consequences of computer-mediated GDSS is its ability to provide anonymity for the participants and their inputs. Most past research on anonymity in GDSS has focused on “idea generation tasks”. It has examined the effect of anonymity on communication of participants in group collaborative tasks and whether this influences the effectiveness of the group. Jessup et al. (1990, 1991) found that in idea-generation tasks, groups working anonymously generated more solution clarifications, critical comments, questions about the solution, and total comments than those of identified groups. Connolly et al. (1990), in studying an idea-generation task found that anonymous groups generated more different (rare), critical, and total comments. However, Valacich et al. (1992) in studying an idea-generation task found that anonymous groups are no more effective than identified group while identified groups were more satisfied than anonymous groups. Moreover, George et al. (1990) who studied problem-solving (more analytical) type tasks did not find anonymous groups generating more comments than those of identified groups. Thus, past research findings on the influence of anonymity have been quite inconsistent.

In synthesizing past research, Er and Ng (1995) identified the following advantages and disadvantages of anonymity in computer-mediated group collaboration:

Advantages

1. Equal participation among participants regardless of their rank.
2. Reduced dominance by some group participants.
3. Criticisms are addressed at idea, not individual levels.
4. Anonymous voting encourages decisions based on merit.

Disadvantages

1. De-individualization of the group meeting process.
2. Controversial views and unworkable ideas may surface.
3. Lack of social emotional support for the participants.

To our knowledge, there has been no research on the effects of group member anonymity on the outcomes of the software inspection. The results of past studies on group member anonymity in idea generation and problem-solving may be reasonably adapted to software inspection tasks and processes. Typically in an inspection process, first, group members work individually and generate a list of potential defects as they review the product being inspected. Second, they work as a group in further identifying defects. Individually identified potential defects are evaluated by both the author and inspection team. Therefore, the evaluative tone interacting with the anonymity of group members as described in past research on idea-generation-tasks (Connolly et al., 1990) could also affect the comprehensiveness and quality of defect identification and their resolution in an inspection process. Thus, despite some of the above-noted disadvantages, we believe that software inspection can benefit from anonymity afforded to group participants.

Jessup and Valacich (1993) note that anonymity may have a weaker effect on outcomes for groups of peers or groups that work together on a regular basis. Members who work on inspection do characterize such groups, and therefore one could argue that anonymity might not be a significant issue in inspection. However, we present the following reasons in favor of studying the effects on anonymity in inspection. Inspection teams are dynamic in that the composition of the team changes depending on what is being inspected. Authors of the work-products being inspected are not direct participants of the inspection process; they may deal with another inspection team later. Inspection teams do not meet regularly, but usually meet only once as a group during the meeting stage. Therefore, although the members of an inspection process may have experience working with each other in the past, the context they work in changes from inspection to inspection (depending on the product being investigated). In addition, even though members are usually peers, depending on their experience, status, as well as the overall composition of the inspection team, there may exist some implicit hierarchical status differences among the inspection group members.

Propositions of the Study

Based on the foregoing arguments, the central theses of this study are that software quality teams would be more effective in detecting errors and providing solutions to these when anonymity can be provided to team members. This will be better facilitated in computer-mediated GDSS environments. Furthermore, such environments will also enable greater efficiency of the process due to more task- focused orientation. Finally, not only will anonymity in computer-mediated GDSS support enable greater effectiveness and efficiencies of team-based software inspection process in a face-to-face setting, it will also function equally well for groups who are asynchronous in time, space, or both.

References

1. Ackerman, A., Buchwald, L., and Lewski, F., "Software Inspections: An Effective Verification Process," *IEEE Software*, May 1989, pp. 31-37.
2. Brothers, L., Sembugamoorthy, V., and Muller, M., "Icicle: Groupware for Code Inspection," *Proceedings from Computer Supported Cooperative Work (CSCW)*, ACM Press, New York, 1990, pp. 169-181.
3. Connolly, T., Jessup, L., and Valacich, J., "Effects of Anonymity and Evaluative Tone on Idea Generation in Computer-Mediated Groups," *Management Science*, 36 (1990), 689-703.
4. Er, M., and Ng, A., "The Anonymity and Proximity factors in Group Decision Support Systems," *Decision Support Systems*, 14 (1995), pp. 75-83.
5. Fagan, M., "Advances in Software Inspections," *IEEE Transactions on Software Engineering*, July 1986, pp. 744-751.
6. George, J. Easton, G., Nunamaker, J., and Northcraft, G., "A Study of Collaborative Group Work With and Without Computer-Based Support," *Information Systems Research*, December 1990.
7. Gintell, J. Arnold, J., Houde, M., Kruszelnicki, J., McKenney, R., and Memmi, G., "Scrutiny: A Collaborative Inspection and Review System," *Proceedings of the Fourth European Software Engineering Conference*, Garwisch-Partenkirchen, Germany, September 1993.
8. Jessup, L., Connolly, T., and Galegher, J., "The Effects of Anonymity on GDSS Group Process in an Idea-Generating Task," *MIS Quarterly*, 14(3) (1990), pp. 313-321.
9. Jessup, L., and Tansik, D., "Decision Making in an Automated Environment: The Effects of Anonymity and Proximity with a Group Decision Support System," *Decision Sciences*, 22 (1991), pp. 266-279.
10. Jessup, L., and Valacich, J., *Group Support Systems: New Perspectives*, Macmillan Publishing Company, 1993.
11. Johnson, P., Tjahjono, D., Wan, D., and Brewer, R., "Experiences with CSRS: An Instrumented Software Review Environment," *Proceedings of the Pacific Northwest Software Quality Conference*, Portland, Oregon, 1993.
12. Knight, J., and Myers, E., "An Improved Inspection Technique," *Communications of the ACM*, November 1993, pp. 51-61.
13. Mashayekhi, V., Tsai, W., Drake, J., and Riedl, J., "Distributed, Collaborative Software Inspection," *IEEE Software*, September 1993, pp. 66-75.
14. Macdonald, F., Miller, J., Roper, M., and Wood, M., "A Review of Tool Support for Software Inspection," *Proceedings of the Seventh International Workshop on Computer-Aided Software Engineering (CASE-95)*, Toronto, Canada, 1995, pp. 340-349.
15. Valacich, J., Dennis, A., and Nunamaker, J., "Group Size and Anonymity Effects on Computer Mediated Idea Generation," *Small Group Research*, February 1992.