December 1998

Where's the Beef? In the Process or in the People?

Thomas Rodgers
University of Arizona

Douglas Dean
University of Arizona

Douglas Vogel
University of Arizona

Where's the Beef? In the Process or in the People?

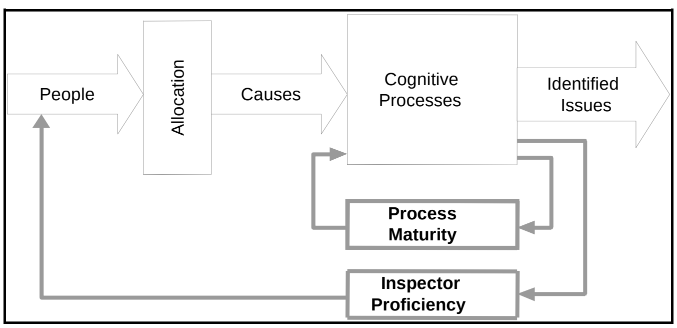
Thomas Lee Rodgers
Douglas Dean
Douglas R. Vogel
University of Arizona

Abstract

Recent research suggests that as much as 50% of explained sources of variation in software code inspections relates to people related issues and less than 30% relates to inspection processes. This paper examines the impact of two feedback mechanisms (process maturity and inspector proficiency) on inspection processes. Results of a survey of thirty-one experienced software developers and some follow-up interviews are presented. Variations appear to exist within organizations at the process level even within a relatively mature development organization. Process maturity and inspector proficiency constructs are difficult to measure. The findings should extend to other formal technical reviews and provide insights into managing different-place, different time inspections.

Background

"Understanding the Sources of Variation in Software Inspections" is the topic of a research article by Porter, Siy, Mockus and Votta (Porter, 1998). They find that about 50% of variation in defect detection relate to input factors (reviewers, coders and code units) and very little variation relate to process related treatment factors (team size, type of session, and repair strategy). Other studies suggest process factors explain less than 30% of the variation in defect detection (Porter, 1997). Such studies suggest that the "real beef" is in people-related and not process-related factors.



This research is based on the cause-and-effect model shown above (Rodgers, 1998). Although similar to that used by Porter, we theorized inclusion of feedback mechanisms. Similar to control mechanisms (such as speedometers or gauges) in state machines, feedback mechanisms are both cause and effect. Two examples are process maturity and inspector proficiency. Process maturity focuses on inspection team practices rather than overall organizational practices. Inspector proficiency focuses on individual abilities and motivations (coding, using support tools, and executing inspection strategies) to work effectively during an inspection. This paper focuses on measurement and use of feedback mechanisms during software inspections.

Survey

Thirty-one experienced software developers representing organizations with different organizational process maturity levels were surveyed regarding inspection process maturity and inspector proficiency. Survey participants represent eleven organizations that range from small software development teams to large international firms.

- 13 represent a single international hardware and software development firm with a reputation for mature processes.
- 6 represent small software development firms based in the United State and selling product in international markets.
- 4 represent European software development firms most of which have multinational development efforts.
- 8 were graduate students who also work for software development firms. 5 of the 8 are employees of a large international hardware and software development firm.

On average, those surveyed represent 12.6 years of software development experience working 12.2 years with their current employer. Individuals were asked to participate based on having software inspection experience. On average, they participated in 19.8 code reviews, 21.6 design reviews and 4.5 other formal technical reviews within the last twelve months. Out of 60 individuals asked to participate 31 responded (52% return rate). This rate was significantly higher for the 23 individuals that were directly asked to participate (with 18 responses and a 78% return rate). This compares to 37 individuals who were asked to participate by a Quality/Process Leader within a single large international firm (with 13 responses and a 35% return rate). Nevertheless, we believe that the individuals surveyed have significant relevant experience. Also, the survey purpose was to explore impacts of process maturity and inspector proficiency on the inspection process and not to validate constructs or generalize findings.

Results

The results support the importance and need for further research. Significant process variations exist within relatively mature development organizations. The 13 individuals from the same international firm stated that inspection practices for their last inspection process ranged the entire spectrum of process maturity (2 ad hoc, 1 repeatable, 4 defined, 3 managed, and 3 optimized). Interestingly, the organizational process maturity is considered well managed (at CMM level 4). Also interesting is that the volume reviewed was essentially bipolar with 6 reporting that they last inspected less than 50 lines of code or 5 pages and the other 7 reporting that they inspected over 4,000 lines of code or 25 pages. Only 3 had expectations about the number of major issues to be found and only 5 reported the actual number of issues uncovered. Only one reported a significant number of issues uncovered (141 total issues of which 54 were major and 25 majors were anticipated based on 5,000 lines of code).

Of all survey responses, the majority expected to find general defects (29 of 31) and specific types of defects (19 of 31). Other inspection expectations included requirement assurance, product usability, and verification of known defect fix. Similar to the findings reported for the single firm, last inspection processes ranged the entire spectrum of process maturity (6 ad hoc, 9 repeatable, 9 defined, 3 managed, and 4 optimized). For most, inspectors can from the same project (20 of 31) and the same development team (22 of 31). Methods for assigning inspector roles varied widely (8 informally, 14 were assigned the same tasks, 8 were assigned different main responsibilities, 2 involved only one inspector, and 1 used a checklist). For most written standards and guidelines exist (17 of 31); however the formality of the process definition was subject to wide variations (10 informally defined, 11 written processes, 8 provided with defect codes and type classification, 14 checklists, 1 script, and 2 scenarios). For most they worked with individual inspection members on similar reviews numerous times previously (28 of 31 with 7 reporting an average of 2.4 times during the last twelve months).

Inspector proficiency is more difficult to measure and many surveyed expressed concerns about recording or deriving prior individual performance. The following table summarizes assessments about whether inspector proficiency can or should be assessed formally, informally, or not at all. Surveyors were asked to use a 7 point Likert scale (where 1=no importance, 3=some importance, 5=important, and 7=extremely important and expected). The responses are ranked from most important down to least important.

Discussion

We now turn to a discussion of issues based primarily on survey participant responses.

Construct measurement is difficult. We assumed that workgroup processes can be classified similar to the capability maturity model framework which focuses on organization level process practices (Humphrey, 1989). As previously stated participants within relatively mature organizations perceived wide variations in inspection process maturity. It was suggested that the classification wording might imply "goodness" of personal practices.

	Average Rank	**Standard Deviation**
In order to be a proficient inspector, how important are the following?		
programming language proficiency (for code inspections)	6.1	1.0
cognitive ability to find defects	5.9	1.0
experience as an inspector	5.3	1.5
development environment proficiency	5.2	1.4
experience as an author being reviewed	4.1	1.7
Assuming the following information can be record or derived about prior individual performance, how acceptable is it to use the following information to select and assign inspectors to an inspection team?		
experience (number of prior inspections)	5.0	1.4
efficiency (percent of defects found per defects suggested)	4.5	2.2
experience (pages or lines of code inspected)	3.9	1.5
review rate (pages/lines of code reviewed per hour)	3.7	1.9

productivity (number of defects per page/lines of code)	3.7	2.1
productivity (defects found per preparation hour	3.6	1.9
productivity (defects found per meeting hour)	3.5	1.9

Assuming the following information can be recorded or derived about prior individual performance, how acceptable is it to use the following information to select and assign inspectors to an inspection team?

experience with programming language	5.5	1.4
experience as an inspector	5.1	1.5
productivity as a person who can find defects	5.1	1.9
efficiency as a finder of defects	5.0	1.8
experience with the development environment	4.9	1.8
review rate as a through and detailed inspector	4.3	2.0
experience as an author of reviewed materials	4.0	1.7

and is difficult to assess. Inspector proficiency is even more difficult to measure and potentially subject to measurement dysfunction (Austin, 1996) and illustrated in the following participant quotes. "Yes, although this is difficult to quantify" "People will be less open about the inspections and their results." "People will not log if they think management will base performance evaluation of author on the results." "Counting is ridiculous especially if you have to justify what you find is over or under an expected amount."

Team ownership of the inspection process should be considered. "Our goal is to find defects ourselves as a development team, not to create professional super-inspectors. Every member should inspect once in a while." "The team typically knows who are the good inspectors. The data about it should be owned by the individual or the team."

Perceptions are preferred over recordings of individual performance. "If you can record the original error maker, you can record who can find the error." But recording "can hurt a person's privacy" "Sure, if I know that they are not checking details, I probably won't use them." "It is valuable information for performance evaluation. But I don't think it needs to be a formal process to record. A close working relationship among the team members would yield better information." "Don't waste time doing recording"

Problem-solving skills are important. "Are they good engineers?" Inspectors need to have a "knack" and "see to the heart of the problem." "At least one person who can find defects is a must to make the review worthwhile." "Only certain people can be considered good reviewers, or can learn to be good reviewers."

Inspection review rates were given little importance. "You want people who spend time and find problems." However, inspection expert, Tom Gilb states, "check at your organization's optimum rates to find major defects. This is the big one!" (Gilb, 1998) Apparently, this is also a major issue many inspectors either do not know or have forgotten the important.

Workload and project commitment is important. "We barely have time for reviews, much less all the formalization." "Our environment is changing given shortened time-to-market constraints. We need to tailor processes to fit the life cycle." "Don't include a person that is not motivated to participate." Do not "increase overhead." "Inspection is not a popular pastime. Many of us find it drudgery even though it is of critical importance to quality. People may decide to find fewer defects if they know the more they find, the more review time they must spend."

Training needs must be considered. "Others ought to be included to improve their skills." "Other members can be in training." "Everybody needs to be in the game."

Process versus people needs to be explored further. "People are more important; however you are given limited resources and you must mange the process." "The process is more important than the people. Don't blame the persons but the process." "The amount of defects that can be found is also dependent on the author."

Future Research

Factor analysis was conducted based on the survey and yields additional insight. The results are the subject of a future paper. The survey is a part of an on-going research stream dedicated to developing a better understanding of process feedback mechanisms. The research stream uses a multi-methodological approach that includes case studies, surveys, instrument validation, action research, prototype building, model building, and field experimentation.

Conclusion

The findings should extend to other formal technical reviews and provide insights into managing different-place, different time inspections. Feedback mechanisms might provide a means of measuring and managing performance at the workgroup level. Doing so in an unobtrusive and effective manner is a challenge. The goal is to enable teams to self-define tasks and assign of responsibilities in a manner that encourages productivity and decreases time to market

References

Austin, R. *Measuring and Managing Performance in Organizations*, Dorset House Publishing, New York, N.Y, 1996

Gilb, T. " Optimizing Software Inspections" in *Crosstalk, The Journal of Defense Software Engineering*, (11:3 pp. 16-19), March 1998

Humphrey, W.S. *Managing the Software Process,* Addison-Wesley, Reading, MA, 1989

Porter, A. A., Siy, H., Mockus, A. and Votta, L.G. "Understanding the Sources of Variation in Software Inspections," *ACM Transactions on Software Engineering and Methodology*, (7: 1 pp. 41-79), January 1998

Rodgers, T. L., Vogel, D. R., Purdin, T. and Saints, B. "In Search of Theory and Tools to Support Code Inspections," *Proceedings of the 31st Hawaii International Conference on Systems Sciences*, (3: pp.370-8), 1998

Tervonen, I., Lisakka, J, and Harjunamaa, L. "Software Inspection – a Blend of Discipline and Flexibility," *ESCOM-ENCRESS 98 Conference Proceedings*, [http://rieska.oulu.fi/~tervo/Rome.pdf], Rome 27-29. May 1998