Identifying Implicit Process Variables To Support Future Empirical Work

Jeff Carver[1]

[1]Department of Computer Science
University of Maryland
{carver,basili}@cs.umd.edu

Victor Basili[1,2]

[2]Fraunhofer Center for Experimental
Software Engineering, Maryland

Abstract

The most basic questions that researchers must address when introducing a new process or technique are what is the intended effect of that process and can that effect be demonstrated empirically. As the understanding of a process progresses, researchers become interested in more sophisticated questions about a process or technique, such as studying the relationship between a particular type of variable and the outcome of the process. Quite often, researchers will find few, if any, studies in the literature that explicitly identify and analyze the effects of potential variables on the process. This paper proposes a methodology to aid in performing a literature search to be used as a basis for new research into these types of variables. The methodology provides guidance on making use of a large range of studies from which to extract potential variables. Throughout the paper, the methodology is illustrated with a specific example. The example focuses on searching for variables that deal with individual variations among software inspectors that affect their performance during an inspection. At the end of the example, after following the steps of the methodology, a list of potential variables among software inspectors is identified. The paper concludes with the next steps to be taken concerning the identified variables and hypotheses.

Keywords

Software Process Variables, Literature Search, Empirical Software Engineering, Software Inspections

1. INTRODUCTION

The most basic questions that researchers must address when introducing a new process or technique are:

1) What is the intended effect of that process?

2) Can that effect be demonstrated empirically?

These questions form the basis for most early empirical studies on a process or technique. But, once researchers have demonstrated the potential success of the technique certain applications, more sophisticated questions arise that address possible variations in the results achieved and the identification of other variables that impact the effectiveness of the technique, e.g., process experience or domain knowledge of the technique user, or tool support for the technique.

To address these more sophisticated questions, researchers are often interested in studying the relationship between a particular type of variable and the outcome of the process. Before proceeding with formal experimentation, a literature search should be done to identify potential variables and what effect they might have. The literature search might reveal a set of studies that have the goal of directly analyzing a variable of interest. If a sufficient number of studies are found, the researcher can use meta-analysis techniques to combine the results of the prior studies to gain a better overall understanding of the variables.

More often, the researcher will find few, if any, studies that explicitly identify and analyze the effects of these potential variables. However, they might find studies where a variable was discussed, was studied but did not show a statistically significant effect, or was hypothesized, *a postiori*, to have affected the result. So, before proceeding with their own research into these variables, the researcher can use this literature as a basis upon which to build effective hypotheses for more focused investigations. This paper provides a methodology to follow when doing a literature search as a basis for study of variables that were not the primary target of interest in prior studies. An assumption is made in this methodology that some body of literature exists describing studies on the process of interest, but little or no literature exists describing studies on the effects of the specific set of variables of interest on the chosen process.

2. SOFTWARE INSPECTION EXAMPLE

The search for variables affecting the software inspection process will be used to illustrate the steps of the methodology. A software inspection is a static analysis of a software artifact (requirements, design, code, etc…) to ensure the presence of some quality characteristic. Inspections are normally geared towards finding defects, i.e. places where the artifact does contain the information that it should. An inspection was originally defined to consist of a team of inspectors who spend some time individually reviewing the artifact and preparing for an inspection team meeting. After each inspector has prepared, the team meets together to inspect the artifact and record any defects that are found. That list of defects is then returned to the document author for further action [13]. This original idea has been modified in various ways, such as putting more emphasis on the individual preparation, or holding multiple team meetings, but the basic idea remains the same.

Because many studies have been conducted by multiple researchers to investigate various aspects of the software inspection process, there is a body of knowledge in the literature on the process. Furthermore, inspections have been shown to be a useful tool in reducing defects in software artifacts [6], so further study seeking to improve process is warranted.

2.1 Context Variables in Software Inspections

While there has been much study done on the software inspection process, there are still some variables that have not been addressed. Most research on software inspections has focused on the organization of the inspection process, including the presence or absence of a team meeting [23, 28], the number of inspectors or the allocation of artifacts among the inspectors [18, 20], and even specific techniques for use by individual inspectors [3, 27]. But, there has been little study focused specifically on the effects of the context within which the inspection is conducted. This context includes both the physical environment and the inspectors themselves. While not studying the context variables formally, some studies have indicated that this context can have an effect on the outcome of the inspection.

For example, in a study conducted to understand if increasing the size of the inspection team or the number of inspections performed could increase the effectiveness of the inspection, the results showed that the variables had little effect on the outcome of the inspection. But, the researchers noted a large unexplainable variation in the performance of the inspectors and teams, which led to further investigation into the inputs to the process. The conclusion was that the inputs to the inspection process, including the artifacts and the inspectors, had the largest impact of any variable on the outcome of the inspection [25].

Researchers who conducted a study to understand whether individual inspectors who were inspecting a requirements document using a specified technique were more effective than those using an ad hoc procedure came to a similar conclusion. A family of scenario-based reading techniques called Perspective Based Reading (PBR), in which each inspector assumed the perspective of one of the stakeholders of the requirements document, was evaluated, in the context of NASA, to determine any benefits a structured and focused technique for reviewing requirements had over the normal technique used by the NASA engineers. The results of this study showed that defect detection effectiveness, i.e. the number of defects found, increased using the PBR technique over the usual technique. In addition, there was some indication, though not statistically significant, that individual variations among inspectors, including their experience, had an impact on the number of defects they found [3].

2.2 Individual Variation among Inspectors

Previous experimental work has been done to understand and improve software review and inspection procedures with respect to defect detection [3, 16, 20, 22, 23]. Inspections have been shown to be an effective tool for defect detection in software artifacts. It has been reported that inspections can help find between 60 and 90 percent of the defects present in software artifacts [6,14]. But, studies often report a large variation between the performance of the least effective inspector and the performance of the most effective inspector.

For example, studies conducted to measure the effectiveness of individual inspectors in finding defects reported wide variations among subjects. In one study, the percentage of defects found ranged from 10% by the least effective inspector to 90% by the most effective inspector [3], and in another it ranged study from 20% to 70% [19]. Furthermore, a study measuring the effectiveness of inspection teams showed the least effective team finding 22% of the defects and the most effective team finding 50% of the defects [24]. These wide variations in effectiveness could not be explained by variations in the inspection process.

2.3 Importance of Studying these Variables

The above studies suggest that there are variables other than the process used that affect inspections. The presence of these variables is shown both by the quantitative variation in performance, discussed above, and by the qualitative discussion often provided by researchers describing their experimental designs and results. Researchers make a, sometimes implicit, identification of variables through the rationale behind the choices made during the planning of studies and/or the explanations given in the discussion of their results. For example, in some studies, to reduce the potential impact of a confounding variable, researchers will group subjects based on an assumption that one or more context variables could affect the outcome, but their goal is not to specifically study the effects of these variables. On the other hand, often when researchers are discussing their results, they explain unexpected results by hypothesizing about the presence or absence of context variables.

Again, these variables are generally not specifically tested in the study, but are implicitly identified by the researcher as having some impact on the results of their study.

3. MOTIVATION

The first step in any new research, including the search for variable about a process, is to search the literature for previous work done on the topic. When such a search uncovers a series of studies that have already been conducted to investigate the set of variables of interest, the researcher has a solid basis on which to build his research. But, when such a literature search yields few or no studies conducted to understand the set of variables of interest, the task of finding a solid basis on which to build the research becomes more difficult. The methodology described in this paper can help the researcher who finds himself in this situation.

3.1 Overview of Methodology

The methodology for searching the literature described in this paper is part of a more comprehensive, global methodology used in the investigation of the impact of variables on the outcome of a process. The global methodology is based on concepts that have been useful in other fields of study, such as Sociology and Psychology, from an approach called *Grounded Theory* [17]. In a grounded theory based methodology, hypotheses are formed both top-down, from existing theory, as well as bottom-up, from empirical data. The global methodology is a two-part methodology that has both qualitative and quantitative components. The first part of the methodology is concerned with hypothesis generating. This step uses the grounded theory concepts to identify variables and hypotheses of interest by combining a literature search with qualitative and quantitative analysis to identify variables and then refine those variables into specific well-supported hypotheses. The second part of the methodology is a hypothesis-testing step that uses the more traditional concepts of empirical studies, but draws the hypotheses to be tested from the output of the first step in the methodology. A complete description of the methodology can be found in [7].

3.2 Focus of this paper

This paper focuses on the first part of the first step of the methodology, the identification of a list of variables that can later be refined into hypotheses. While there is much research describing various empirical methods for testing hypotheses, there is little research providing guidance on identifying relevant variables and hypotheses to be studied. The remainder of this paper provides a methodology for identifying the variables using a literature search. The methodology is illustrated throughout with an example.

4. METHODOLOGY

This section describes the proposed methodology for doing the literature search, along with an example. Section 4.1 introduces some terminology and notation used to explain the methodology. Section 4.2 provides the details of the methodology for searching the literature along with an example based on the search for variables concerned with individual inspectors in the software inspection process.

4.1 Notation used in Methodology

This methodology can be used to search the literature for information about a set of variables such that:

(variables x_i) affect the (outcome y_j) of (process p) in a (positive/negative) way

Where X, the set of x_i (i = 1, .. n), is the set variables the researcher is interested in studying, Y, the set of y_j (j = 1, .. m), is the set of measurable outcomes of the process p that have been chosen for study. The set of variables, X, may be initially bounded based on the researcher's expertise to limit the scope of the literature search. In the software inspection example:

p = "software inspections"

y_1 = "number of defects detected during an inspection"

The set X has been bounded to include variables about the variations between inspectors.

4.2 Literature Search Methodology

Goal: Define an initial list of variables and hypotheses

Inputs: The process p, an initial idea of y_j, any bounds on the set X

4.2.1 STEP 1: Search Literature for the Specific Relationship

This step involves searching the literature for studies that were conducted to specifically address the relationship between the x_i and the y_j that is of interest to the researcher. Record any variables identified in the results of those studies.

a. Search the literature from the domain and identify studies that were conducted to better understand process p. Each of these studies can either have as its goal:

   i. Addressing the relationship between the variables x_i and the outcome y_j

   ii. Understanding some other aspect of process p

b. If any of these studies are of the first type, e.g. addressing the relationship between variables x_i and the outcome y_j, then based on the results of those studies, add the appropriate variables x_i to the set X. If there are very few, or no studies that fall into this category, proceed to step 2 after reviewing all the literature.

c. The remaining studies, the second type, e.g. those that are studying another aspect of process p, will be used in step 3 of this methodology.

d. **Example:** The literature was searched for studies conducted on software inspections. Most of the studies found were not specifically designed to understand the relationship between the variation in the individual inspectors and the number of defects found during an inspection. Those studies that were focused on some other aspect of software inspections will be used in Step 3 of the process. One of the few studies that did specifically address the relationship of interest showed that inputs to an inspection process, which included the inspectors, had as much impact on the outcome of the process as did the procedure that was followed. The results of this study showed that there was a large variation in the performance of various inspectors and that the presence or absence of certain inspectors had a large impact on the outcome of the inspection [25].

4.2.2 STEP 2: Verify the worth of studying the relationship between x_i and y_j

In the cases where there are few or no studies found that focus on the specific relationship between x_i and y_j, a second literature search should be done to verify that the relationship is worth studying.

a. This verification can be done by searching the literature for studies that have shown that one or more variables x_i have influenced the outcome of some process, p', which is similar in nature to the chosen process p. The main goal here is to understand whether or not the type of variable of interest is capable of having the type of impact the researcher is interested in.

b. If one or more processes p' are found, then the researcher has some evidence that variables in the set X can have an impact on the outcome of a process and therefore can continue to study the impacts of variables in the set X on the outcome of process p. If no such p' are found, then the researcher should proceed with caution and spend more time in Step 3 of this methodology.

c. **Example**: In Step 1, few studies were found specifically studying the relationship between individual differences among inspectors and the number of defects found during an inspection. Therefore, it was necessary to gather more support for studying this type of relationship before continuing with the research. In this case, the software engineering literature was searched for studies that showed that individual variations among people had an effect on other software engineering processes. Two, well known examples were found showing that variations in individuals can have an effect on the outcome of a process. First, the COCOMO cost-modeling tool uses a series of metrics to predict the cost of a software system being developed. Among the characteristics used in the model are: 1) Analyst Capability; 2) Programmer Capability; 3) Applications Experience; 4) Platform Experience; and 5) Language and Tool Experience [5]. By taking into account these variations in the individual software developers in predicting the overall cost of the project, the COCOMO model has shown that individual differences have an impact on a software engineering process. Second, in a study done to identify metrics useful in project productivity estimation, some of the important metrics included: 1) Overall personnel experience and qualifications; 2) Percentage of programmers doing development who had participated in the design of the functional spec; 3) Previous experience with the programming language; and 4) Previous experience with applications of similar or greater size and complexity [29]. By identifying these metrics dealing with individual differences, this study also shows a case where a software engineering process, productivity estimation, is influenced by individual variations among people.

4.2.3 STEP 3: Review general literature on the chosen process

This step uses the studies from the literature that were found but not used in Step 1. Those are studies on the chosen process p that were not specifically aimed at studying the relationship of x_i to y_j. In these studies, potential x_i are identified implicitly by the researchers in the choices made in the design of the study, and in the explanations given for unexpected results. Because the researcher often implicitly indicates these variables, identifying variables in these studies is not as straightforward as identifying the variables in the studies from Step 1. The following procedure should be followed to find potential variables in:

a. The Study Design

   i. In the design of the study, one or more variables are chosen as the independent or treatment variables to be studied. These variables often deal with the type of technique or process being studied. In addition to the main independent variable, there are often other variables, which are related to the context, that are implicitly identified, but not specifically controlled for. These potential variables can be found in the assumptions made about the context of the study or about the subject population. For example, if it is assumed that all the subjects are native English speakers, then implicitly a potential variable of Native Language has been identified.

   ii. The second place to find variables is in the selection of subjects for the study. For example, if the subjects for the study are only selected from a subset of all possible subjects, and the subjects in that subset can be categorized as having (or not having) a particular type of knowledge or skill, then that type of knowledge or skill has been implicitly identified as a potential variable.

   iii. **Example:** In a study conducted at NASA, an assumption was made that the subjects did not have had exposure to inspections prior to the study. This assumption was somewhat counter to what would be expected in industry, but it revealed that the researchers believed that having some subjects who were experienced and some who were not would create difficulty when analyzing their results [12]. This assumption indicated that **Process Experience** was a potential variable. In many cases the study designs pointed towards some potential variables. There were a series of studies that identified multiple groups of potential inspectors based on their knowledge of the application domain. Then subjects for the study were drawn only from those groups that had high domain knowledge [11, 22]. These choices indicated that **Domain Knowledge** was a potential variable. Additionally, in the same studies subjects were also chosen from the group of inspectors who had high knowledge of the software domain for the project. This choice indicated that **Software Development Experience** was a potential variable.

b. The Discussion of Results

   i. Often in the discussion of the study results, one or more variables are implicitly identified. When the results of a study are not the expected ones, the researcher will often identify a potential variable to explain the results. For example if the results of subjects in one treatment, which were hypothesized to be similar, can be logically split into two groups because of a large variation in the performance, such that the subjects in each group can be characterized similarly, then this characterization is a potential variable.

   ii. **Example:** In a study done to compare an inspection done using an ad hoc technique to one done using a more procedural technique, the results showed that the performance of the subjects who used the ad hoc method varied based on their experience with computing or information systems [8]. This result indicated that **Software Development Experience** was a potential variable. In another study, conducted at ORACLE, the results showed a wide variation between the performance of the least effective and the most effective subject. Upon further analysis, the researcher reported that the inspector who was least effective was the

least familiar with the inspection technique, while the inspector who was most effective was the most familiar with the inspection technique [21]. This result pointed towards **Process Experience** as a potential variable.

   c. The Flaws or Weaknesses of a Study

      i. If a discussion of the flaws or weaknesses of a study is provided, then potential variables are often identified in this discussion. For example, if the researchers identify the presence of a heterogeneous subject population as a threat to the validity of the study, then the characteristic that was used to measure that heterogeneity is a potential variable.

      ii. **Example:** When researchers replicate a study, they sometimes alter the design of the original study to address some of its flaws or weaknesses. In a replication of a study on the N-fold inspection process, the researchers corrected some flaws in the initial study by collecting more data about the software development experience of the subjects. Also, the researchers altered the training so that all of the subjects had the same level of expertise in the N-fold inspection process before participating in the study [24]. These two alterations indicated that **Software Development Experience** and **Process Experience** were potential variables.

   d. The Qualitative Data of a Study.

      i. The qualitative data of a study, collected either through a questionnaire or interview at the conclusion of the study, often indicates some potential variables. For example, if the subjects report in a post experiment questionnaire that a particular type of knowledge or experience was needed to do the task assigned, then the presence or absence of that knowledge or experience is a potential variable.

      ii. **Example:** In a study to compare a checklist based inspection technique to a more procedural technique, the subjects were assigned one of the two techniques to use on a series of two inspections. Although the researchers do not report a statistical analysis comparing the performance of the subjects in the first inspection to their performance in the second inspection, they do report the results of a post-study questionnaire that the subjects felt more comfortable during the second inspection when using the more procedural technique [15]. This result suggested that **Process Experience** was a variable to consider when studying a procedural technique. In the same study, the results of the questionnaire showed that the subjects desired to have equal amounts of time during the training sessions to practice both types of techniques. This result indicated that **Training** was a variable that should be considered.

4.2.4 STEP 4: Review Literature from Other Related Fields

When searching for variables about a process, insight can often be gained from studies conducted in other fields on processes that are similar to p, and on processes in general. A good starting point is the literature from the Education and Psychology fields.

   a. When searching this literature, look for one of the following:

i. Studies that deal with the relationship between a more general version of the chosen process p and the chosen outcomes y_j or even a more generalized outcome. Look for variables that are identified in these studies as being important. Determine if the variable can be transferred to the specific process p being studied.

ii. **Example:** In a paper discussing the differences between novice and experienced computer programmers, it is noted that experts have at least two beneficial kinds of knowledge that the novices do not have. First experts have a better understanding of commonly used code fragments that can be reused. Secondly, experts better understand how to communicate with other programmers through programming conventions [26]. The process of writing code involves the understanding and translation of a description of a system from one representation to another; similarly the process of inspection involves understanding and verifying a representation of a system. Therefore the type of knowledge identified here, **Process Experience**, is a variable to be considered when studying inspections.

In another study, the differences in the design approaches of application domain experts and application domain novices were discussed. First, the experts tended to make mental models before creating a design. Second, the experts tended to all have the same level of abstraction at different points in the procedure. Third, the experts took notes about issues that needed to be addressed later. Finally, the experts did mental simulations of their partially completed designs [1]. Again, the specific activities are not important, but the fact that application domain experts behaved in similar ways that were beneficial to their task shows that **Application Domain Knowledge** is a useful variable and should be studied in the context of inspections.

iii. Secondly, studies that deal with any process p_i and either the specific outcomes y_j or more general outcomes can provide variables that can be useful in the study of the process p.

iv. **Example:** Effectively training subjects in a new process is recognized by many researchers as being important. In the Education and Psychology literature, a series of papers describe different methods for training subjects along with the proposed benefits of each. One method argues for abstracting specific cases to general principles and then applying general principles to new situations [9]. Another method proposes using an apprenticeship model where the expert works with the novice on using a new process and slowly gives the novice more responsibility until the expert is no longer needed [10]. Finally, a third method suggests that clear goals be set for the subjects during training. The subjects should have their activities monitored and be given positive or negative feedback on their progress [4]. While all three of theses methods proposed different approaches, they all indicate that **Training** is an important variable.

In a paper that discusses how people acquire knowledge, it was argued that for someone to effectively learn something, they had to have some interest in what they were learning [2]. This idea of being interested in learning the new process indicated that **Motivation** was an important variable to consider. Interestingly, this variable showed up only in the Education and Psychology literature.

b. For each variable identified

 i. If the variable has already been identified by step 2 or 3, then the variable has more support and it should be noted that the variable has support from multiple sources.

 ii. If the variable has not appeared previously, then record it as a new variable, which is related to the more general process, that should now be evaluated on the specific process p.

4.2.5 STEP 5: Create Hypotheses

For each identified variable, an initial high-level hypothesis should be generated that relates that variable to the outcome of the process. The hypotheses will take one of the following forms:

a. (Variable x_i) $\Rightarrow$ (Outcome y_j)

 i. A positive effect, indicating that an increase in x_i will result in an increase in y_j. Likewise, a decrease in x_i will result in a decrease in y_j.

b. (Variable x_i) $\Rightarrow \neg$ (Outcome y_j)

 i. A negative effect, indicating that an increase in x_i will result in a decrease in y_j. Likewise, a decrease in x_i will result in an increase in y_j.

c. (Variable x_i) $\neg \Rightarrow$ (Outcome y_j)

 i. No effect, indicating that an increase or decrease in x_i will not result in any change to y_j.

d. **Example:** Based on the review of the literature discussed above, an initial list of relevant variables was created. Some of these variables came only from the software engineering literature, some came only from the other literature, and some were found in both sets of literature. The list of variables along with their source of discovery and high-level hypothesis is:

x_1 = Application Domain Knowledge (Both sets of literature)

 (Application Domain Knowledge) $\Rightarrow$ *(More defects found)*

x_2 = Software Development Experience (Software Engineering literature)

 (Software Development Experience) $\Rightarrow$ *(More defects found)*

x_3 = Process Experience (Both sets of literature)

 (Process Experience) $\Rightarrow$ *(More defects found)*

x_4 = Training (Both sets of literature)

 (Proper Training) $\Rightarrow$ *(More defects found)*

x_5 = Motivation (Other literature)

 (Proper Motivation) $\Rightarrow$ *(More defects found)*

Output: A list of variables that have been identified in various sets of literature and a set of high-level hypotheses related those variables to the outcomes of process p.

4.3   Refining the Variables and Hypotheses

This process of developing a list of potential variables is only the starting point of the research process. Each one of these variables and hypotheses must be further refined to be more useful. As the variables are refined, two questions arise:

1) How should each variable be measured?

2) What should be the potential values for each variable?

For instance, a variable like *software development experience* is very broad and abstract. This variable, along with the other ones, must be refined using more concrete metrics. Another issue that must be addressed during this refinement is the measurement scale of each variable and metric. In most cases it makes sense to define the variables in terms of ordinal values, but in some cases it may be necessary to use only nominal values. As each variable is refined into these more specific metrics, a decision must be made as to the scale that is appropriate for that variable. Details on how to do these steps, along with the complete methodology can be found in [7].

5. CONCLUSIONS

This paper has described a process for performing a literature search to establish a basis for research into the relationship between a set of variables and a process. The methodology described in this paper is useful in the situation where there is little or no prior work specifically aimed at studying the relationship of interest. This methodology provides guidance for identifying potential variables from existing literature so that the researcher can base their new research on a solid grounding even in the absence of prior work.

The methodology was illustrated through an example of studying the relationship between the individual variations among software inspectors and their effectiveness during an inspection. Using the methodology, a series of variables were identified for further study.

The generation of this set of variables and their associated high-level hypotheses is not the ultimate goal of the researcher. It is only the first step in a more global research methodology, as mentioned earlier. Once these variables and hypotheses have been generated, the next step is to refine the high-level hypotheses into a set of hypotheses that are more concrete and useful as described at the end of the previous section. One useful method for doing this refinement process is, in the case where it is available, to make use of data from existing studies that may have been designed with a different goal in mind, but collected the right kind of data to be useful. If this data exists, it can be analyzed to determine if any of the specific metrics collected in the study can be mapped to any of the high level variables defined in the above process. Furthermore, the generated hypotheses can find some support in this historical data and be refined to use specific metrics. After refining the hypotheses and finding support from historical data, the final step is to design and run a new study to test a specific hypothesis that is of interest to the researcher.

6. ACKNOWLEDGEMENTS

7. REFERENCES

[1]   Adelson, B. and Soloway, E. "A Model of Software Design." In *The Nature of Expertise*, Chi, M.H, Glaser, R. and Farr, M. (eds.). Lawrence Erlbaum Associates, 1988.

[2]   Alexander, P.A. "Toward a Model of Academic Development: Schooling and the Acquisition of Knowledge." *Educational Researcher* 29(2): 28-33,44. March 2000.

[3]   Basili, V. R., Green, S. ,Laitenberger, O., Lanubile, F., Shull, F., Sorumgard, S., Zelkowitz, M.V., "The Empirical Investigation of Perspective-Based Reading"; *Empirical Software Engineering – An International Journal,* vol. 1, no. 2, 1996.

[4]   Berliner, D.C. and Rosenshine, B. "The Acquisition of Knowledge in the Classroom." In *Schooling and the Acquisition of Knowledge*. Anderson, R.C., Spiro, R.J. and Montague, W.E. (eds.), Lawrence Erlbaum Associates, New Jersey, 1977. pp. 375-396.

[5]   Boehm, B., Clark, B., Horowitz, E., Westland, C., Madachy, R., and Selby, R. "Cost Models for Future Software Life Cycle Processes: COCOMO 2.0." *Annals of Software Engineering: Special Volume on Software Process and Product Measurements*. Arthur, J.D. and Henry, S.M. (eds.). 1995, Vol. 1, pp. 57-94.

[6]   Boehm, B.W. and Basili, V.R. "Software Defect Reduction Top 10 List." *IEEE Software* 18(1): 135-137, Jan. 2001.

[7]   Carver, J. "The Impact of Background and Experience on Software Inspections." PhD Thesis, University of Maryland, April 2003. (University of Maryland, Department of Computer Science Technical Report CS-TR-4476)

[8]   Cheng, B. and Jeffery, R. "Comparing Inspection Strategies for Software Requirements Specifications." In *Proceedings of the 1996 Australian Software Engineering Conference*, p. 203-211.

[9]   Collins, A. "Processes in Acquiring Knowledge." In *Schooling and the Acquisition of Knowledge*." Anderson, R.C., Spiro, R.J., and Montague, W.E. (eds.). Lawrence Erlbaum Associates, New Jersey, 1977, pp. 339-363.

[10]   Collins, A., Brown, J.S., and Newman, S.E. "Cognitive Apprenticeship: Teaching the Crafts of Reading Writing and Mathematics." In *Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser*, L.B. Resnik (ed.), Hillsdale, NJ: Erlbaum, 1989, p. 453-494.

[11]   Doolan, E.P. "Experiences with Fagan's Inspection Method." *Software – Practice and Experience*. Feb. 1992. 22(2): 173-182.

[12]   Dow, H.E. and Murphy, J.S. "Detailed Product Knowledge is not a Prerequisite for an Effective Formal Software Inspection."   In *Proceedings of the 7ᵗʰ Software Engineering Process Group Meeting*, Boston, MA, May 1994.

[13]   Fagan, M. E., "Design and code inspections to reduce errors in program development." *IBM Systems Journal,* 15(3): 182-211, 1976.

[14]   Fagan, M.E. "Advances in Software Inspections."   *IEEE Transactions on Software Engineering,* 12(7): 744-751, 1986.

[15]   Fusario, P., Lanubile, F., and Visaggio, G. "A Replicated Experiment to Asses Requirements Inspection Techniques."   *Empirical Software Engineering: An International Journal* 2(1), 1997. p. 39-57.

[16]   Gilb, T., Graham, D. *Software Inspection*. Addison-Wesley, 1993

[17]   Glaser, B. G., and Strauss, A.L. *The Discovery of Grounded Theroy: Strategies for qualitative research*. New York : Aldine de Gruyter. 1967.

[18]   Knight, J.C., and Myers, E.A. "An Improved Inspection Technique." *Communications of the ACM*, 36(11): 51-61, Nov. 1993.

[19]   Laitenberger, O., Atkinson, C., Schlich, M. and El Emam, K. "An Experimental Comparison for Reading Techniques for Defect Detection in UML Design Documents." *Journal of Systems and Software*, 53(2), August 2000, 183-204.

[20]   Martin, J., and Tsai, W.T. "N-Fold Inspection: A Requirements Analysis Technique." *Communications of the ACM*, 33(2): 225-232, Feb. 1990.

[21]   Melo, W.; Shull, F. and Travassos, G.H., 2001. Software Review Guidelines. Technical Report ES-556/01. Systems Engineering and Computer Science Program. COPPE. Federal University of Rio de Janeiro. September.

[22] Parnas, D.L. and Weiss, D.M. "Active Design Reviews: Principles and Practice**."** *Proceedings of the 8th International Conference on Software Engineering*, 1985, 132-136.

[23] Porter, A.A. and Johnson, P.M. "Assessing Software Review Meetings: Results of a Comparative Analysis of Two Experimental Studies." *IEEE Transactions on Software Engineering*, 23(3): 129-145.

[24] Schneider, G.M., Martin, J., and Tsai, W.T. "An Experimental Study of Fault Detection in User Requirements Documents." *ACM Transactions on Software Engineering and Methodology* 1(2): 188-204.

[25] Siy, H.P. "Identifying the Mechanisms Driving Code Inspection Cost and Benefits." *Ph.D. Dissertation, University of Maryland*, 1996.

[26] Soloway, E., Adelson, B., and Ehrlich, K. "Knowledge and Processes in the Comprehension of Computer Programs." In *The Nature of Expertise*, Chi, M.H, Glaser, R. and Farr, M. (eds.). Lawrence Erlbaum Associates, 1988.

[27] Travassos, G.H., Shull, F., Fredericks, M., and Basili, V.R. "Detecting Defects in Object Oriented Designs: Using Reading Techniques to Improve Software Quality." *Proceedings of OOPSLA '99*, Denver, CO. November, 1999.

[28] Votta, L.G., Jr. "Does Every Inspection Need a Meeting?" *Proceedings of ACM SIGSOFT '93 Symposium Foundations of Software Engineering*. ACM, Dec. 1993.

[29] Walston, C.E., and Felix, C.P. "A method of programming measurement and estimation." *IBM Systems Journal* 16, No. 1, 54-73. 1977.