Empir Software Eng (2006) 11: 523–539
DOI 10.1007/s10664-006-9021-5

Can observational techniques help novices overcome the software inspection learning curve? An empirical investigation

Jeffrey C. Carver · Forrest Shull · Victor Basili

Published online: 18 October 2006

Editor: Murray Wood

Abstract When learning a new software engineering technique, there is a learning curve that must be overcome. In particular, when studies are conducted in a classroom setting, researchers need a method for quickly accelerating the experience of novice subjects to allow the results to be more applicable in industrial settings. In this paper, we propose and test a method to enable novices to gain process experience to allow them to more quickly overcome the learning curve. The method we evaluate allows an inspector to gain experience with the inspection process by observing an inspection performed by someone else. The results of the study show that the proposed method for gaining experience appears to be useful in some limited cases, that is, for low experienced subjects who were inspecting a requirements document from a domain in which they had low knowledge. Based on the results of this study, we are able to propose some new related hypotheses to be tested in future studies.

Keywords Empirical study · Requirements inspections · Software process · Experimental process · Software quality

J. C. Carver (✉)
Department of Computer Science and Engineering, Mississippi State University,
Starkville, MS, USA
e-mail: carver@cse.msstate.edu

F. Shull • V. Basili
Fraunhofer Center for Experimental Software Engineering, University of Maryland,
College Park, MD, USA

F. Shull
e-mail: fshull@fc-md.umd.edu

V. Basili
Department of Computer Science, University of Maryland,
College Park, MD, USA
e-mail: basili@cs.umd.edu

1 Introduction

There is a significant and growing body of studies conducted to evaluate techniques to improve software inspections. Many of these studies, including some of our own studies, have identified individual the experience of the individual inspector to be a major factor affecting the overall effectiveness of the inspection. However, neither relevant measures (e.g., domain expertise, familiarity with the system under inspection, prior experience with inspections) nor magnitude of the effects have been consistently found across studies. Subjectively speaking, however, the body of knowledge suggests that novice inspectors are not as effective as more experienced ones. Furthermore, this same rule of thumb has appeared in our studies with software engineering students. For example, in a classroom study with students having widely varied backgrounds, we found that the subjects' effectiveness when using a specified technique varied based on their level of prior experience (Carver 2003a) (Shull 1998).

Despite the fact that results can vary based on subject experience, we argue that software engineering studies run in a classroom environment, using university students as subjects, are a useful and necessary method of advancing knowledge in the area of software inspections. There are many practical reasons for using this readily available subject pool, such as the reduced expense relative to using professional subjects, the ability to obtain preliminary evidence to confirm or refute a hypothesis, the opportunity to fine tune the details of a study prior to use in an industrial setting, and the freedom to control factors that may affect the outcome of the study (Carver et al. 2003b). Classroom studies are also useful for debugging development technologies and study protocols for later use in industry. For example, we used a series of classroom studies to debug the steps of a design inspection technique (Shull et al., 2001) prior to its use in industry (Melo et al. 2001). In related work, researchers at the Norwegian University of Science and Technology used a classroom study with undergraduates to identify problems related to subject motivation and accurate time estimates for tasks (Arif and Hegde, 2001). Researchers were able to use a classroom study to understand and correct these problems before using expensive professional subjects. Those experiences later led to a successful industrial study (Conradi et al. 2003). Finally, in some cases, a student population may be an adequate representation of the target population. For example, in the case where the study focuses on a new technique with which industry professionals are not familiar, an adequately prepared student population may provide a close approximation to industrial professionals.

To further improve the contributions made by such uses of classroom studies, the issue of training must be taken into account. The process training that subjects receive in a classroom setting is in most cases very dissimilar to that received in an industrial environment. Researchers often do not have time for thorough training of the subjects due to the constraints of a classroom environment. Subjects are often trained on a new technique and then measured almost immediately with little or no opportunity to practice the new technique. This lack of process training introduces threats to validity of the results. The technology is not being measured as it was intended to be applied by professionals (who would typically have more chances to practice with the technology in their day-to-day work) but instead is being measured in the early stages of the learning curve. Students in such studies may need additional time to get comfortable with using the technology, to better understand how to apply the technology, and to work through practical problems—time that is not always available in a classroom environment with a busy curriculum.

This paper extends the original feasibility study (Carver et al. 2003c) by comparing the results of our method to those reported in the literature regarding alternative approaches to accelerating skill development in applying software processes (Section 2). This paper also provides a more detailed analysis of potential influencing factors (Sections 4 and 5) and formulates specific, well-supported hypotheses to be tested in future studies (Section 6).

2 Related Work

In Section 2.1 we will first review the related work to establish the relevant differences between novices and experts when performing inspections. Then, in Section 2.2 we will discuss methods that have previously been used in an attempt to increase the process experience of novices.

2.1 Differences Between Novice and Expert Inspectors

Our original attempts to investigate the role of inspector experience stem from a study evaluating the use of a new inspection technique by software professionals at NASA's Goddard Space Flight Center (Basili et al. 1996). Although we did not explicitly design the study to test for differences between novices and experts, a *post hoc* analysis of the data showed that the most experienced inspectors seemed to gain the least benefit from the new technique when compared with the students who had either medium or low experience. The quantitative and qualitative data suggested that the experienced subjects were resistant to the new approach because over time they had developed their own highly effective techniques. Conversely, less experienced subjects, who lacked any technique of their own, benefited much more from the new approach.

A replication of this study at the University of Maryland (Shull 1998) allowed this hypothesis to be tested on a more diverse subject sample. The subjects ranged in experience from those with no industrial software development or inspection experience, to those who were software professionals returning for advanced degrees who had many years of experience in both topics. In the replication, experience was measured explicitly, and tested using MANOVA tests with an alpha-level of 0.1. Results showed that the effectiveness of the inspection technique varied with the experience level of the subjects:

- Subjects with medium experience (i.e., who had previous experience with software engineering tasks related to the inspection technique, gained from one or two projects) experienced a statistically significant improvement of about 7% (9 subjects, F=4.01, p=0.08).
- Subjects with low experience (i.e., who had no experience with related software engineering tasks) experienced a significantly decreased effectiveness of about 4% when switching to the new inspection technique (49 subjects, F=4.59, p=0.04).
- As in the original study, for subjects with high experience (i.e., who had experience from multiple projects in industry) there was no statistically significant difference observed when they applied the new inspection technique (8 subjects, F=0.134, p= 0.73).

Comparable results in the literature are difficult to find since there is not a commonly accepted manner in which experience is reported. Furthermore, most studies (especially in industrial environments) tend to not even report the experience levels of individual subjects. However, some studies in university environments are helpful for shedding more light on the subject.

In an early study (followed by 2 replications) comparing the effectiveness of code reading to the effectiveness of two types of testing (functional and structural), Basili and Selby showed that students behave differently than professionals. The first study drew its subjects from a population of professional programmers (*study P*) while the other two studies drew their subjects from student populations (*studies S1* and *S2*). In professional *study P*, code reading was more effective than either structural or functional testing. In student *study S1*, code reading and functional testing were similar, but both better than structural testing. In student *study S2*, there was little difference among the three techniques (Basili and Selby, 1987). This same study was replicated by two independent groups of researchers using students as subjects. In these replications the inexperienced subjects saw little difference between the three techniques (code reading, functional testing, and structural testing) (Kamsties and Lott, 1995), (Wood et al. 1997).

Although often assumed, it is not always clear that the direction of any effect due to experience would necessarily be for professional subjects to outperform student subjects. For example, Porter and Votta discuss a pair of studies to compare a scenario-based requirements inspection technique to a checklist and to an ad hoc inspection. The first study had graduate students as subjects and the second had professional subjects. Overall the graduate student subjects were more effective than the professional subjects. While the statistical results showed the same relative effectiveness among the techniques in both studies, the graduate students were overall more effective than the professionals (Porter and Votta 1998).

Based on the studies discussed above, it is still unclear whether the results of studies conducted with students as subjects can be directly applied to professional environments. In most cases it appears that the conclusions drawn from students will not be directly applicable to professionals. Certainly that seems to be a common belief in the literature: For example, in a very thorough literature review (Laitenberger and DeBaud 2000) the authors speculate that inspectors with more experience require decreased effort for an effective review, but could find no data to support this relationship. Similarly, some attempts to build process simulators for inspection results (Münch and Armbrust 2003) attempt to account for the relationship by codifying that increased experience of an individual inspector reduces effort and increases the detection rate, but could again cite no empirical basis.

Based on these somewhat confusing indications from the literature, we believed:

- Students and professionals are likely to behave differently regarding software inspection effectiveness;
- The difference in behavior need not be caused necessarily by intrinsic differences within the subjects themselves;
- We could at least address the part of the problem caused by the likelihood that subjects in professional and classroom environments have different methods for gaining process experience about development practices.

2.2 Methods for Accelerating Process Experience

The overall goal of this work is to evaluate a method for quickly and effectively increasing the process experience of inexperienced users of a process. If the process experience of novices can be quickly increased (to a level that approaches that of an industrial professional) then results obtained from studies using these novice populations, especially those run in university classrooms, can become more generalizable to a target population of industrial professionals. Therefore, our search of the literature focuses on techniques that have shown some promise for improving the effectiveness of novice subjects.

Several prior inspection studies identified the potential benefits of performing software engineering tasks in pairs. In an early study on software inspections, subjects who worked together as a pair had a significant increase in productivity, although not always efficiency, over subjects who worked alone (Bisant and Lyle 1989). In a user interface inspection study, researchers had some subjects work alone and some subjects work in pairs with one subject ensuring that the process steps were accurately followed. In this study the subjects working in pairs reported both that the process was easier to use and that they had higher process conformance (Zhang et al. 1999). In eXtreme Programming, which attempts to generate the benefits of software inspections constantly by having two programmers work together at the same computer to write code (i.e., so that every line written is subjected to immediate review), researchers have also noted the learning that takes place by the programmer who is observing (Muller and Tichy 2001), (Williams 2001).

In order to take advantage of the potential benefits of observation, our previous experience led us to believe that *observational studies*, studies where a subject who is doing a procedure is observed by someone else, would be an effective method for doing this observation. Observers in a previous study (Shull et al., 2001) were able to gain useful insight into the execution of a technique, so we believed that a subject could gain inspection experience by observing another subject performing an inspection. This approach not only allows the subjects to gain experience by observing, but through the notes taken during observation, it also helps the researchers better understand the application of the process. These observational studies provide a level of detail about individual process steps and their usefulness that is difficult to collect using traditional post-study questionnaires (Singer and Lethbridge 1996). Another goal of using the observational approach, similar to the interface inspection study discussed earlier, is for the observer to act as a “process guide” to ensure that the inspector was following the procedure and that any deviations from the process are conscious decisions by the inspector rather than simple oversight.

Based on the success of the observational studies, the methodology proposed below uses the idea of observation to allow subjects to gain process experience and then evaluates the effect of that experience.

3 Proposed Methodology

The previous research showed us that students will likely need some extra training to allow their performance to more closely resemble that of professional developers. Observation has been shown to be one successful method for gaining experience. We therefore developed a methodology that takes advantage of the learning that occurs during the process of observation as a method for increasing the expertise of a student.

Step 1: **Divide the subjects into pairs** This division can be done randomly, or it can be done in order to meet the external constraints of the study.

Step 2: **Train the subjects in the new technology** Training typically occurs during a lecture period in which the subjects are provided with the necessary information in order to use the new technology.

Step 3: **One person in each pair applies the technology while the other observes.** The subject applying the technology is referred to as the *executor* while the subject observing application of the technology is called the *observer*. The job of the observer is not to work together with the executor to apply the technology; rather the observer is tasked with ensuring that the executor faithfully follows the steps of the technology. The observer also takes notes about the executor's use of the technology including places where difficulties are encountered.

Step 4: **Subjects within the pairs switch roles.** This step allows the subject who observed in Step 3 to now perform the task using the technology.

Step 5: **Measurement and analysis of the second treatment** The results obtained from this step should be more accurate because more of the learning curve has been addressed.

An alternative to the approach formulated above would have been to allow subjects to gain experience by performing an inspection rather than observing one. The main drawback to this approach is that an inspector would be likely to repeat mistakes from the first inspection during their second inspection that they made during their first inspection. By allowing a subject to observe an inspection, we believed that they would more easily be able to notice both the successes and the mistakes of another inspector and therefore be more effective when they performed their own inspection.

4 The Study

The remainder of this paper describes the study run to evaluate the methodology described earlier. A more thorough description of the design has already been published elsewhere (Carver et al. 2003c). In this paper we provide a short summary of the study and a more detailed analysis of results, leading to several conclusions and lessons learned for further work.

4.1 Research Questions and Goal of Study

Based on our previous work, we are convinced that process experience is an important variable to study. But, there are still some unanswered questions about process experience that this study addresses:

- What type of process experience is important?
- How can an inspector gain process experience?
- What other types of experience affect process experience?

We explore these questions in the context of a study that evaluated one potential method for gaining process experience. Specifically, of the goal of the study is to:

Analyze the **proposed observational method for gaining process experience** to ***evaluate*** it with respect to **its impact on inspection effectiveness**.

In this study, we measure the effectiveness of an inspection in two ways:

- **Defect detection rate**: the percentage of known defects in a given software artifact that are found during the inspection. (Higher values represent more effective inspections.)
- **False positive rate**: the percentage of issues reported by an inspector that turn out not to represent real quality problems in the artifact. (Since in a real inspection environment, false positives impose a cost on the inspection team to discuss and discard them, lower values of this metric represent more effective inspections.)

4.2 Technology to which the Method is Applied

As a “testbed” for evaluating this new training method, we use our ongoing research on software reading techniques. Software reading techniques are procedurally-based approaches that aim to improve the effectiveness of defect detection in software inspections by providing tailored and focused techniques for inspectors to use during the individual review phase. One specific type of reading technique that has been developed for reviewing software requirements documents written in English is Perspective-Based Reading (PBR) (Basili et al. 1996). A requirements document is used by a series of stakeholders, so it must satisfy the differing needs of each of those stakeholders. To verify this property, PBR provides a method for each inspector to follow to assume the perspective of one of those stakeholders. Each PBR perspective asks its user to create a model that represents an abstraction of the requirements. The model is chosen so that it is relevant to the stakeholder. For example, an inspector assuming the perspective of a tester would create test cases as his or her model, while someone assuming the perspective of a designer would create a high-level design as the model. The technique for each perspective provides a step-by-step procedure that instructs the inspector on how to create the model and then provides a series of questions for the inspector to answer to look for defects. More information about PBR can be found in (Shull et al. 2000).

4.3 Materials

The User perspective of PBR, which has the inspector create use cases as the abstraction of the requirements document, was applied to the requirements documents from two different systems: one for a Loan Arranger (LA) system, and one for an automated parking garage control system (PG). These two documents were chosen so that we would have one from a domain where most subjects would have high domain knowledge (PG) and one where the subjects would have low domain knowledge (LA). The LA system is responsible for organizing the loans held by a financial institution and bundling them for resale to investors. The PG is responsible for managing the open spaces in a parking garage and keeping track of the sales of reserved (monthly) tickets and non-reserved (daily) tickets. The LA requirements document has 8 pages, 26 functional and 4 non-functional requirements, and 18 seeded defects. The PG requirements document has 17 pages, 21 functional and 9 non-functional requirements, and 32 seeded defects.

4.4 Subjects

The subjects in this study were graduate students enrolled in a graduate level Software Engineering class at the University of Maryland in the Fall 2001 semester. The subjects worked in pairs to conduct two inspections, with one subject acting as the *executor* and the other as the *observer*, as defined in Section 3. About half of the subjects (12 out of 26) had industrial experience working with requirements. Subjects were randomly paired up such that two constraints were met. First, each pair consisted of two low experienced subjects (no industrial experience) or two high experienced subjects (some industrial experience). This constraint allowed us to see if the results differed based on the overall experience level of the subjects. Second, each pair had at least one subject who was familiar with the PG domain and one who was not familiar with the LA domain. This constraint allowed us to examine the influence of domain knowledge. The PG document was always inspected by someone who knew its domain well and the LA document was always inspected by someone who did not know its domain well.

4.5 Procedure

Based on the methodology described in "Section 3", the following steps occurred:

1) **Divide the subjects into pairs**. The pairing was done in such a way that both members of the pair were either low experienced or both were high experienced, according to the constraints in "Section 4.4". The pairing was also done to ensure that assumptions about domain knowledge would be satisfied.
2) **Training**. Subjects were trained in the PBR technique and the observational methods. The PBR training occurred during a 60-minute class lecture that covered the theory behind PBR, the history and evolution of PBR, and the use of PBR (along with some examples). The subjects were given a chance to practice PBR in class and ask questions. The training on the observational methods occurred during a 30-minute classroom lecture consisting of an explanation of the roles of *process observer* and *process executor.*
3) **Office Hours**. Each pair of subjects spent 45 minutes with one of the researchers. During this time the researcher watched the subjects use PBR to perform an inspection on a sample requirements document to ensure that each pair understood both the PBR technique as well as the roles of *observer* and *executor*. Each subject spent part of this time as the observer and part of the time as the executor. The subjects were also given the opportunity to ask questions about PBR and the observer and executor roles.
4) **Inspection 1**. Team Member B gained process experience by observing Team Member A
5) **Team members switch roles**
6) **Inspection 2**. Team Member B performs the inspection

After the second inspection, each pair of subjects wrote a report detailing their experiences during the two inspections. This report was the source of much of the qualitative data for the study. In the assignment description, the subjects were instructed that their report should cover at least the following topics (they could provide additional information):

- How feasible is PBR?
- Is PBR worth using in a practical situation? Which ones?
- What would you do to improve PBR?
- For the team member playing the role of *Observer* during the first inspection, how did that experience affect your performance in the role of *Executor* during the second inspection?

Figure 1 summarizes the quasi-experimental, factorial design of this study (Campbell and Stanley 1963).

5 Results

5.1 Qualitative Data

We first examined the qualitative data from the study. As described in Section 4.5, in their final report the subjects were asked to describe any effect the observation had on the second inspection. The results from the analysis of that data indicated that overall the subjects

Treatments	**Group 1a**	**Group 1b**	**Group 2a**	**Group 2b**
	4 Low Experience Subjects 3 High Experience Subjects	4 Low Experience Subjects 3 High Experience Subjects	3 Low Experience Subjects 3 High Experience Subjects	3 Low Experience Subjects 3 High Experience Subjects
	Training			
	Office Hours – Practice Inspection			
Inspection #1	**Execute Inspection on LA**	**Observe Inspection on LA**	**Execute Inspection on PG**	**Observe Inspection on PG**
Inspection #2	**Observe Inspection on PG**	**Execute Inspection on PG**	**Observe Inspection on LA**	**Execute Inspection on LA**

Fig. 1 Study design. The *rows* represent the sequence of activities undertaken by each group of subjects. Each group participated in exactly one inspection (*shaded boxes*) and observed exactly one inspection (*white boxes*)

found the observation activity to be beneficial, in terms of gaining understanding of or confidence with using PBR, or both:

- All 7 of the ‘low experience pairs’ and 4 of the 6 ‘high experience pairs’ reported that the observation was beneficial.
- Four of the ‘low experience pairs’ (pairs 1, 2, 4, 5) and 3 of the ‘high experience pairs’ (pairs 10, 11, and 13) (7 of the 13 total pairs) reported that observing the inspection process helped the second inspector better understand the overall process.
- Five of the ‘low experience pairs’ (pairs 2, 3, 5, 6, 7) and 2 of the ‘high experience pairs’ (pairs 8, and 13) (7 of the 13 total pairs) reported that observing the inspection process helped the second inspector better perform specific steps of building the use cases or detecting defects.
- Two of the ‘high experience pairs’ (pairs 9 and 12) did not comment at all on this issue.
- None of the pairs reported that the observation process hurt their effectiveness.

5.2 Quantitative Data

Next we conducted a statistical analysis on the quantitative data. Two dependent variables were of interest: 1) the percentage of the known defects detected (DEFECT RATE) and 2) the percentage of the reported defects that were determined to be false positives (FP RATE). For DEFECT RATE a higher value is better while for FP RATE a lower value is better. Independent variables considered were: EXPERIENCE level of the subject (high or low); whether or not OBSERVATION had occurred prior to the inspection (yes or no); and, the ARTIFACT that was inspected (LA or PG).

We used the 3-way ANOVA as the basic test for analyzing the data. This statistical test allows us to examine the effects of multiple factors on the dependant variable and the interactions among those factors. We argue that ANOVA is a more appropriate test than the multiple analysis of variance (MANOVA) test because, although multiple measurements are taken from the same review *pair*, each subject produces only one data point. We consider that the defect detection effectiveness of subjects from the same pair will be independent from one another.

Due to the relatively small number of subjects and the exploratory nature of this study, we considered an alpha-level of 0.10 to be a reasonable tradeoff between having sufficient power and the doubled probability of falsely rejecting H0 (as compared with an alpha of

0.05). Any results therefore may be helpful for identifying possible patterns for further investigation but not for rigorously verifying any relationship between variables.

5.2.1 Defect Rate

We began our analysis by conducting the 3-way ANOVA using the defect rate data and the 3 independent variables described above (OBSERVATION, EXPERIENCE, and ARTIFACT). The Kolmogorov-Smirnov test for normality provided no evidence against a normal distribution, therefore the ANOVA test is appropriate. The results of the test showed that there was a significant three-way interaction ($F_{18,1}$=12.7, p=0.002) among the three independent variables. Furthermore, the results showed a significant two-way interaction ($F_{18,1}$=3.844, p=0.066) between OBSERVATION and ARTIFACT. Finally, the main effect of EXPERIENCE was also significant ($F_{18,1}$=5.212, p=0.035). Based on these results, we investigated the interaction effect between OBSERVATION and ARTIFACT in more detail.

In order to do this investigation, we plot the interaction between OBSERVATION and ARTIFACT separately for each experience level (shown in Figure 2). The left chart depicts the results for the 'low experience pairs' and the right chart the 'high experience pairs.' In each chart, the line connects the results before observation and after observation for inspections of the same artifact. Note that the lines connect data points based on the artifact inspected not based on the group of inspectors. These charts clearly show that the interaction between OBSERVATION and ARTIFACT varies with the EXPERIENCE of the subjects. Specifically, for the 'low experience pairs' on the LA artifact, observation shows a large increase over no observation, but the opposite is true for the PG artifact. We analyzed this observation using both a t-test (the Kolmogorov-Smirnov test indicated all samples tested were normally distributed) as well as the non-parametric Mann-Whitney test (due to the small sample sizes involved). The analysis showed that the increase for the LA artifact was significant ($t_5 = -3.26$, p = 0.02; p = 0.03 [Mann-Whitney]), while the decrease for the PG artifact was not significant. Performing a similar analysis for the 'high experience pairs', we see little change on the PG artifact due to observation, but we see a decrease for the LA

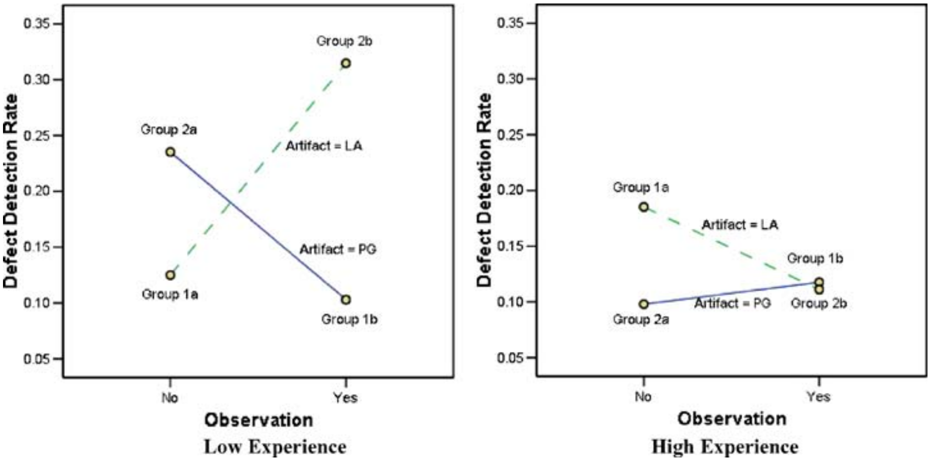

Fig. 2 Defect rates for inspection execution. The *points* represent the mean defect detection rate for each group (see Figure 1). The subjects in Group 1b observed the subjects in Group 1a and *vice versa*. Likewise, the subjects in Group 2b observed those in 2a and *vice versa*

artifact after observation (this decrease was not significant). We performed a similar analysis for major and minor defects separately, but found no significant results. Therefore, we have not included the details of those analyses in this section.

A related question was whether the defect rate was affected by synergistic effects between the observer and executor, or was increased just by virtue of having two people conducting an inspection together. The observer was tasked with recording what was done by the executor and was instructed not to aid the executor in the task of defect detection. Even so, during the course of their observation it was possible that the observer would see defects that had been overlooked by the executor. We wanted to be able to understand these defects. Therefore, the observer was instructed to record any defect he or she saw in their notes, but not to bring them to the attention of the executor until the completion of the inspection. At this point, the observer added any unfound defects to the list that was started by the executor and clearly marked them so that they could be counted separately. (Note: these defects are not included in any of the other analyses presented in this paper.) Table 1 reports the number of defects found by the observers and the effect on the team's overall detection rate. In all but one case, the increase is less than 10%. Having a second person present during an individual inspection will allow some additional defects to be detected, but on the basis of these results we conclude that the defects found by a passive observer do not result in a major increase in the overall defect rate.

5.2.2 False Positive Rate

We used the same approach to analyze the *false positive rate* (FP RATE) that we used for the *defect detection rate*. WE begun by conducting a 3-way ANOVA using the same independent factors described earlier (OBSERVATION, EXPERIENCE, and ARTIFACT). The results of the ANOVA test showed a significant 3-way interaction ($F_{18,1} = 3.93$, $p = 0.063$) among the three independent variables. In addition, the test showed that the main effect of ARTIFACT was significant ($F_{18,1} = 3.03$, $p = 0.099$). Based on this result, we investigated the main effect due to the ARTIFACT in more detail. When the FP RATE of the subjects on the Loan Arranger and the FP Rate of the subjects on the Parking Garage are analyzed using both a *t*-test and a non-parametric Mann-Whitney test we find a significant difference ($t_{24} = -1.84$, $p = 0.078$; $p = 0.072$ [Mann-Whitney]). This result can be seen graphically in the box plots shown in Figure 3. Note that the lines connect data points based on the artifact inspected not based on the group of inspectors. Similar to the analysis for DEFECT RATE, we ensured the normality of the data sets prior to running the ANOVA and *t*-tests. We also include the Mann-Whitney test to compensate for the small sample size.

Table 1 Defects reported by observers

Subject	Inspection #	Artifact observed	Number of defects (major/minor)	Increase in overall effectiveness	Increase in major effectiveness
2_B	1	PG	2/0	6%	13%
3_B	1	PG	1/0	3%	6%
4_A	2	PG	0/1	3%	0%
4_B	1	LA	0/1	6%	0%
8_B	1	PG	1/3	12%	6%
13_A	2	PG	1/2	9%	6%

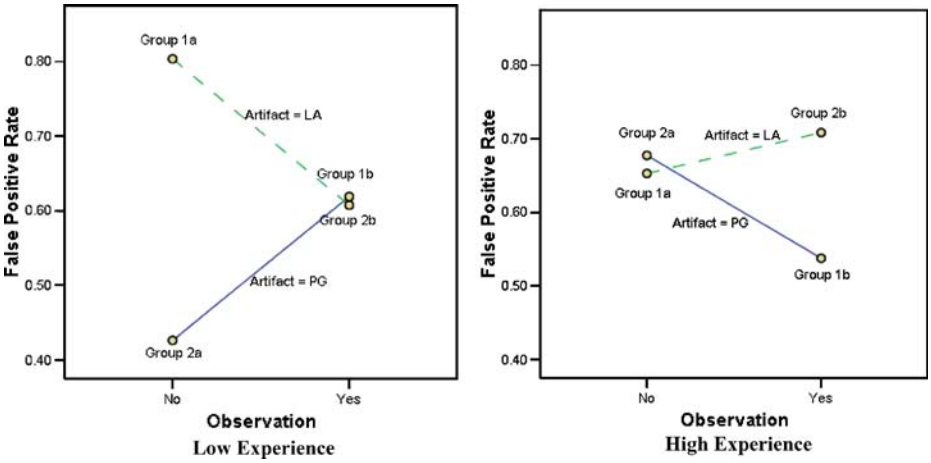

Fig. 3 False positive rates for inspection execution. The *points* represent the mean false positive rate for each group (see Fig. 1). The subjects in Group 1b observed the subjects in Group 1a and vice versa. Likewise, the subjects in Group 2b

6 Discussion of Results

Based on the qualitative data in Section 5, it is clear that the *low experience pairs* felt that there was a benefit from the observation process as all seven of them included this information in their report. On the other hand, only 2/3 of the *high experienced pairs* expressed their belief in the benefits of the observation process. This result makes more sense when taken together with the quantitative data that showed the only significant improvement based on observation came in the *low experience* group.

In addition, the results allow us to pose three new hypotheses. These hypotheses require testing in future studies. For each new hypothesis, we data from the study that provide support in proposing the hypothesis and we give a logical justification for the hypothesis. In the discussion that follows, subject groups are identified by their experience level (Low, High), artifact inspected (PG, LA), and inspection (1, 2).

H1 ***Observation of an effective inspection is beneficial***

- Support:
  - Low LA-2 had a significantly higher defect diction rate than Low LA-1. Low LA-2 observed Low PG-1 which had the highest defect detection rate of any PG group
  - High LA-2 had a significantly lower defect detection rate than High LA-1. High LA-2 observed High PG-1 which had the lowest detection rate of any PG group
- Justification
  - Watching someone who performs a task well should translate into good performance. Researchers in the education domain have translated this concept into a mechanism for training novices. First a novice observes an expert performing some task, then the novice aids the expert in performing the task, then the novice

performs the task with some coaching form the expert, finally the novice can perform the task on their own (Collins et al., 1989), (Pintrich and Schunk, 1996).

H2 ***Observation of an inspection of an artifact from a familiar domain is beneficial***

- Support:
  - Overall LA-2 had a higher defect detection rate than LA-1. LA-2 inspectors observed an inspection in the PG domain (for which they had high domain knowledge)
  - Overall PG-2 had a lower defect detection rate than PG-1. PG-2 inspectors observed an inspection in the LA domain (for which they had low domain knowledge)
- Justification
  - If the observer does not know much about the domain, he or she will have to spend more time understanding the domain and less time gaining from observing the process

H3 ***Inspections of a document from a domain of low knowledge will result in a higher percentage of False Positives***

- Support
  - The FP RATE for the LA document (regardless of experience or inspection number) was significantly higher than the FP RATE for the PG document
- Justification
  - If the inspector is unfamiliar with the problem domain, they will be more likely to report 'defects' based on their ignorance or lack of understanding that someone more knowledgeable of the domain would not report.

7 Threats to Validity

A potential threat to the internal validity of the study was the selection of students into treatment groups. While every effort was made to randomize the groups within the experience levels (e.g., low experience and high experience), after examining the results it appears possible that the students assigned to the Low Experience PG1-LA2 group may have been better skilled overall than those assigned to the Low Experience LA1-PG2 group. This possibility arises from the observation that the Low Experience PG1-LA2 subjects performed better on *both* inspections than their counterparts in the Low Experience LA1-PG2 group. This situation could also explain the decrease in performance from PG-1 to PG-2 (i.e., the group who did PG-2 were simply not as proficient at this task as the group who did PG-1).

A second threat to validity is the relatively small number of subjects who participated in the study. Due to this fact, it is difficult to extrapolate this result. The study was an exploratory study in which we were looking for broad patterns in the data and therefore this threat is not as serious as it would be in a confirmatory study. The hypotheses developed in this study need to be exposed to a more rigorous study aimed at studying them directly.

8 Conclusion

The goal of this study was to understand whether or not subjects of a study run in a classroom setting could gain process experience through observation. If so, this technique would help researchers to make their results more applicable to an industrial setting. This study could also provide some insight to industry concerning training techniques. We examined two independent variables. For DEFECT RATE the results showed a significant impact due to the interaction between EXPERIENCE, OBSERVATION and ARTIFACT. Further analysis showed that OBSERVATION did provide a significant improvement for low experienced subjects inspecting an artifact from an unfamiliar domain. For FP RATE, the results also showed a significant interaction between EXPERIENCE, OBSERVATION and ARTIFACT. Further analysis did reveal that there was a significantly higher rate of false positive reports for the document from the domain of low knowledge.

To further address the research questions posed in Section 4.1, we also examined the qualitative results. The subjects were asked to describe the effect that the observation had. All of the *low experience pairs* and 2/3 of the *high experience pairs* stated that process experience was important and observation was a useful method for acquiring that experience. To answer the question of which other types of experience affect process experience, we turn to the statistical results mentioned earlier. It appears, based on those results, that knowledge of the application domain and requirements experience both influence the effect of process experience. It does appear, at least in some situations, that process experience gained through observation can be beneficial to an inspector.

In addition to these results, some new hypotheses about the usefulness of observation as a method for gaining process experience are proposed based on the results of this study:

1) Observing an inspection of an artifact from a domain where the inspector has high knowledge can be of more benefit than observing an inspection of an artifact from a domain where the inspector has low domain knowledge (hypothesis H2 from Section 6).
2) When inspecting a document from a domain of low knowledge, a higher percentage of the defect reports should be expected to be false positives than when inspecting a document from a domain of high knowledge (hypothesis H3 from Section 6).
3) Observation is not an effective way to gain process experience in general but is effective under certain conditions:

   - Inspectors must observe more than one inspection. This hypothesis arises from the fact that the subjects indicated the observation was helpful, but the data did not support this observation.
   - Inspectors must observe an expert in either the inspection process or the specific technology used or the application domain (hypothesis H1 from Section 6). This hypothesis arises from the fact that the low experienced subjects who observed an inspection on the PG, which was both a familiar domain and, based on the effectiveness, a well done inspection, found more defects when inspecting the LA than any other group. This argument is also made in the context of the learning that takes place in eXtreme programming (Spinellis 2001).
   - Inspectors must take a more "active" role during their observation. This hypothesis is similar to the argument made in the Active Design Review literature that for a design reviewer to fully understand the design, he must do something "active", such as construct a model (Knight and Myers 1993). Likewise, in the study by Bisant and Lyle (Bisant and Lyle 1989) mentioned earlier, subject who worked

> together in pairs were significantly more effective inspectors than subjects working alone. In this study, some of the observers, even though in a more passive role, were able to report defects not seen by the executor. It is likely that if the observers were to take a more active role, they would report additional defects.

In order to continue to understand this issue of the applicability of classroom studies to industrial environments, further studies should be run to test these new hypotheses. It is important to continue investigating methods that allow subjects of classroom studies to gain sufficient process experience in order to allow researchers to have more confidence in the applicability of their results to industrial settings.

The limitations of this study include the lack of a comparison to another method for gaining process experience. While these results do indicate that observation was helpful for some subjects, we have no indication about any of the potential alternative methods, such as performing the inspection rather than observing it as mentioned in " Section 3 ". The next step in this work will be to perform additional studies to gather more information about this question as well as the other hypotheses posed as a result of this study.

Acknowledgments We would like to thank the students of CMSC735 in the Fall Semester of 2001 at the University of Maryland for their participation in this study. We thank the anonymous reviewers for their insightful comments. We would also like to thank the reviewers for their helpful and insightful comments. Finally we would like to thank Dr. J. Edward Swan from Mississippi State University for providing aid on the statistical presentation. We acknowledge support from the NSF Reader's Project (CCR-9900307) and the NSF CeBASE Project (CCR-0088078).

References

Arif T, Hegde LC (2001) Inspection of Object Oriented Construction: A Study of Reading Techniques Tailored for Inspection of Design Models Expressed in Uml. Prediploma Thesis. Norwegian University of Science and Technology

Basili V, Green S, Laitenberger O, Shull F, Sorumgaard S, Zelkowitz M (1996) The empirical investigation of perspective based reading. Empirical Software Engineering—An International Journal 1:133–164

Basili V, Selby R (1987) Comparing the Effectiveness of Software Testing Strategies. IEEE Trans Softw Eng 13:1278–1296

Bisant DB, Lyle JR (1989) A two-person inspection method to improve programming productivity. IEEE Trans Softw Eng 15:1294–1304

Campbell D, Stanley J (1963) Experimental and quasi-experimental designs for research. Houghton Mifflin, Boston

Carver J (2003a) The Impact of Background and Experience on Software Inspections. PhD Thesis. Department of Computer Science, University of Maryland

Carver J, Jaccheri L, Morasca S, Shull F (2003b) Issues in Using Students in Empirical Studies in Software Engineering Education. Proceedings of Ninth International Software Metrics Symposium (METRICS 2003), pp 239–249

Carver J, Shull F, Basili V (2003c) Observational Studies to Accelerate Process Experience in Classroom Studies: An Evaluation. Proceedings of International Symposium on Empirical Software Engineering, ISESE 2003, pp 72–79

Collins A, Brown JS, Newman SE (1989) Cognitive apprenticeship: Teaching the crafts of reading, writing and mathematics. In: Resnik LB (ed) Knowing, learning, and instruction: Essays in honor of Robert Glaser. Lawrence Erlbaum, Hillsdale, NJ

Conradi R, Mahagheghi P, Arif T, Hegde LC, Bunde GA, Pedersen A (2003) Object-Oriented Reading Techniques for Inspection of Uml Models—An Industrial Experiment. Proceedings of European Conference on Object-Oriented Programming (ECOOP'03) Darmstadt, Germany, 483–500

Kamsties E, Lott C (1995) An Empirical Evaluation of Three Defect-Detection Techniques. ISERN Technical Reports. ISERN-95-02

Knight JC, Myers EA (1993) An improved inspection technique. Commun ACM 36:51–61

Laitenberger O, DeBaud J-M (2000) An encompassing life cycle centric survey of software inspection. J Syst Softw 50:5–31

Melo W, Shull F, Travassos G (2001) Software Review Guidelines. COPPE Technical Reports. ES-556/01

Muller MM, Tichy WF (2001) Case Study: Extreme Programming in a University Environment. Proceedings of 23rd International Conference on Software Engineering, ICSE 2001, 537–544

Münch J, Armbrust O (2003) Using Empirical Knowledge from Replicated Experiments for Software Process Simulation: A Practical Example. Proceedings of Empirical Software Engineering, 2003. ISESE 2003. Proceedings. 2003 International Symposium on, 18–27

Pintrich PR, Schunk D (1996) Chapter 5: Other social cognitive processes. Motivation in education: Theory, research and practice. Prentice Hall, Englewood Cliffs

Porter A, Votta L (1998) Comparing detection methods for software requirements inspections: A replication using professional subjects. Empirical Software Engineering — An International Journal. 3:355–379

Shull F (1998) Developing Techniques for Using Software Documents: A Series of Empirical Studies. PhD Thesis. Department of Computer Science, University of Maryland, College Park

Shull F, Rus I, Basili V (2000) How perspective-based reading can improve requirements inspections. Computer 33:73–79

Shull F, Carver J, Travassos G (2001) An Empirical Methodology for Introducing Software Processes. Proceedings of Joint 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Foundations of Software Engineering (FSE-9). Vienna, Austria, pp 288–296

Singer J, Lethbridge T (1996) Methods for Studying Maintenance Activities. Proceedings of Workshop for Empirical Studies of Software Maintenance. Monterey, California, pp 105–110

Spinellis D (2001) Fear of coding, and how to reduce it. IEEE Computer 34:100–199

Williams L (2001) Integrating Pair Programming into a Software Development Process. Proceedings of Software Engineering Education and Training, 2001. Proceedings 14th Conference on, pp 27–36

Wood M, Roper M, Brooks A, Miller J (1997) Comparing and Combining Software Defect Detection Techniques: A Replicated Empirical Study. Proceedings of 1997 Foundations of Software Engineering, pp 262–277

Zhang Z, Basili V, Shneiderman B (1999) Perspective-based usability inspection: An empirical validation of efficacy. Empirical Software Engineering—An International Journal 4:43–70

Jeffrey C. Carver is an Assistant Professor in the Computer Science and Engineering Department at Mississippi State University. He received his PhD from the University of Maryland in 2003. His PhD thesis was entitled “The Impact of Background and Experience on Software Inspections.” His research interests include: Empirical Software Engineering, Software Inspections, Qualitative Methods, Software Process Improvement, Software Metrics, and Software Engineering for High Performance Computing.

Forrest Shull is a scientist at the Fraunhofer Center for Experimental Software Engineering, Maryland (FC-MD). He received his doctorate degree in Computer Science from the University of Maryland, College Park in 1998. At FC-MD he is project manager and member of the technical staff for projects with clients that have included Fujitsu, Motorola, NASA, and the U.S. Department of Defense. He is responsible for basic and applied research projects in the areas of software inspections, software defect reduction, general technology evaluation, and software measurement. A primary focus of his work has been developing, tailoring, and empirically validating improved techniques for inspections of software artifacts, including requirements and design documents.

Dr. Victor R. Basili is a Professor of Computer Science at the University of Maryland. He was founding director of the Fraunhofer Center for Experimental Software Engineering, Maryland, and one of the founders of the Software Engineering Laboratory (SEL) at NASA/GSFC. He received a B.S. from Fordham College, an M.S. from Syracuse University, and a PH.D. in Computer Science from the University of Texas at Austin. He has been working on measuring, evaluating, and improving the software development process and product for over 30 years. Methods for improving software quality include the Goal Question Metric Approach, the Quality Improvement Paradigm, and the Experience Factory organization.