Investigating the Active Guidance Factor in Reading Techniques for Defect Detection

Christian Denger[1], Marcus Ciolkowski[2], Filippo Lanubile[3]
[1]*Fraunhofer IESE Sauerwiesen 6, 67661 Kaiserslautern , Germany; denger@iese.fhg.de*
[2]*University of Kaiserslautern, PO-Box 3049, 67663 Kaiserslautern, Germany; Marcus.Ciolkowski@informatik.uni-kl.de*
[3] *University of Bari, Dipartimento di Informatica, Bari, Italy; lanubile@di.uniba.it*

Abstract

Inspections are an established quality assurance technique. In order to optimize the inspection performance, different reading techniques, such as checklist-based reading and scenario-based reading have been proposed. Various experiments have been conducted to evaluate which of these techniques produces better inspection results (i.e., which finds more defects with less effort). However, results of these empirical investigations are not conclusive yet. Thus, the success factors of the different reading approaches need to be further analyzed. In this paper, we report on a preliminary empirical study that examined the influence of the active guidance factor (provided by scenario-based approaches) when inspecting requirements specification documents. First results show that active guidance is accepted with favor by inspectors and suggest that it is better suited for larger and more complex documents.

Introduction

Following the initial success of inspections right after their invention by Fagan[12], considerable effort was put into optimizing the inspection process [19]. Optimization concerned different aspects of the inspection process, such as the number of inspectors, the optimal reading rate and the optimal size of the document under inspection.

According to Gilb [14], the most essential step in each inspection is the defect detection phase, where the inspectors try to individually identify as many defects as possible in the document under inspection. To optimize individual defect detection, reading techniques were defined that help the inspectors to identify more defects with less effort; that is to make the inspections more cost-effective.

A recent survey among the German industry showed that checklist-based reading (CBR) [1,5,12,14,15], is the most frequently applied reading technique [21,7]. More recent approaches are scenario-based reading (SBR) techniques [2,3,23]. The main claim of SBR is that it improves inspections through two key factors: (1) focusing the inspectors on certain aspects, and (2) giving active guidance to inspectors on how to perform their task. That is, inspectors follow a detailed reading scenario that provides a specific viewpoint on the document and focus the inspectors on specific quality aspects of the document. This ensures that the inspectors exactly know what to check and how to perform the checking. Further, the inspectors are guided to produce work products while searching for defects. To create these work products, the inspectors have to actively work with the document. Doing that, they gain a better understanding of the document than by "passive" reading alone. Therefore, they are able to detect more (subtle) defects.

Various experiments comparing these techniques against each other have been conducted with the goal to determine which reading technique is better in terms of effectiveness and efficiency of the inspection. However, the results of these experiments do not give a conclusive answer to the research question. Some experiments showed that SBR techniques are more effective and efficient than CBR [3,8,16,20,23,24], while other experiments failed to show any significant difference between the techniques [6,11,13,22,25,26].

Most of the existing studies compared a generic checklist to a set of SBR scenarios, thus varying both potential success factors of SBR: giving the inspectors a special focus and providing active guidance. None of the existing empirical studies has investigated the influence of the active guidance factor in isolation. The question that arises is which of these factors really improves the outcome of an inspection.

We performed a repeated case study to examine the influence of active guidance on the inspection outcome.

Proceedings of the 2004 International Symposium on Empirical Software Engineering (ISESE'04)
0-7695-2165-7/04 $20.00 © 2004 **IEEE**

In contrast to earlier experiments we designed the checklists and the reading scenarios similar to each other (i.e. we gave the checklists the same focus as the SBR reading scenarios and asked the same questions), thus attempting to isolate the factor of active guidance.

Section 2 gives a more detailed introduction into the area of reading techniques and defines the basic concepts of the techniques. Section 3 presents the empirical study in detail and the results are presented in Section 4. The results are discussed in Section 5 and threats to validity are described in Section 6. Finally, conclusions and future research activities are presented in Section 7.

2. Background

In general, a reading technique defines a strategy for the individual analysis of a document to achieve the understanding needed for a particular task. Reading techniques for defect detection are a means to support the inspectors during the discovery or preparation stage of an inspection. By guiding the inspectors, reading techniques can help to reduce the problem that inspectors often do not know *what* to look for in a document and *how* to perform the discovery task. The two reading techniques that are the object of investigation in this paper are CBR and SBR.

2.1. Checklists and scenario-based reading

CBR approaches support the inspectors with a set of questions regarding potential quality flaws in the document under inspection. Thus, the checklist provides guidance on *what* to look for during the defect detection. However, there is no guidance on *how* to look for defects. That is, the inspector does not get any guidance on how to check whether a checklist question should be answered with yes or no. Moreover, each inspector is required to check each aspect in the checklist on the whole document. In complex documents this may cause a cognitive overload, as too many aspects need to be considered. This often results in low inspection effectiveness

SBR has been designed with the following two key factors in order to overcome the drawbacks of the CBR approach:

(1) giving guidance on *how* to perform the inspection through actively working with the document (*active guidance*)
(2) restricting the focus of a reviewer to a specific aspect of interest (*separation of concerns*); that is, to guide on *what* to inspect [17].

Both factors are not part of the CBR approach in its original definition [12].

Active guidance forces the inspector to actively work with the document under inspection by guiding him or her to produce real work products while searching for defects. Doing that, it is likely that inspectors gain a better understanding of the document compared to just "passive" reading alone. It is expected that such an improved understanding of the document under inspection can foster a reviewer to detect more defects than reviewing the document by pure reading. Moreover, an improved understanding is a pre-requisite to detect more complex and subtle defects, which are in most cases the most expensive ones. Thus, active guidance is assumed to be especially useful in inspections of complex documents or systems, where the active work helps to gain insights that would have been missed by passive reading and where a profound understanding is important.

The second factor, separation of concerns, ensures that the inspectors are focused on specific areas or quality aspects of the document. This increases the coverage of addressed issues and reduces the overlap between the different inspectors, as not all inspectors focus on the same aspects.

These ideas of the SBR approach are implemented in reading scenarios. A reading scenario consists of three basic parts.

(1) The introduction defines the focus of a scenario; that is, on which (quality) aspects the inspector should focus.
(2) The instruction part gives a step-by-step description of the activities an inspector should perform during the defect detection; for example, an inspector might create test cases or initial design diagrams based on a requirements document under inspection. [17].
(3) Questions, then, focus the inspector on quality aspects that need to be considered while performing the instructions.

Examples of SBR include defect-based reading [23,24,22] and perspective-based reading [3, 17]. More recently, use-based reading [11] and usage-based reading [27] were introduced that respectively guide the inspectors with use case scenarios and a prioritized, requirements-level use case model.

Considering the factors of SBR, it becomes obvious that the factor *separation of concerns* can also be assigned to a set of checklists. Providing different checklists to different inspectors where each checklist focuses on specific aspects leads to the same effect: the inspectors do not focus on the same questions. Thus, the only decisive difference between CBR and SBR is that

SBR gives the inspectors *active guidance* during the inspection. The important question is now: Can active guidance alone improve the performance of inspectors?

2.2. Empirical studies on SBR and CBR

In recent years, several experiments were performed to compare different reading techniques with respect to their effectiveness and efficiency. In a study performed by Dunsmore et al. [11] CBR was compared against use-case reading and abstraction-driven reading. In the use-case reading approach the subjects were guided to derive scenarios and sequence charts from use cases to examine how the code under inspection deals with these scenarios. In the abstraction-driven reading strategy subjects were asked to create natural language abstract specifications for each method while they read them. The results of the experiment showed that subjects using CBR found more defects than those who used the two scenario-based reading techniques. Other empirical studies comparing CBR against scenario-based approaches failed to show any significant difference between the reading techniques [13,22,25,26].

In contrast, several studies showed that SBR is more effective and efficient than either ad-hoc reading or CBR. Basili et al. analyzed the perspective based reading approach (PBR) with software developers of NASA/Goddard Space Flight Center [3]. The results showed that teams using PBR achieved better coverage of general documents and were more effective and efficient in detecting defects in requirements documents than teams that did not use the systematic approach. Other experiments [8,16,20,23,24] confirmed these results. Thelin at al. [27] found that inspectors following the usage-based reading technique detected more critical defects and were more efficient than inspectors using CBR.

However, these past experiments varied both the separation of concerns and the active guidance factor. Usually, one generic checklist was compared against a number of reading scenarios containing more detailed and different questions [e.g., 16,20,24,25]. Thus, SBR implemented the separation of concerns (i.e., focusing the inspector with different reading scenarios) and active guidance, while CBR did not. These findings make it almost impossible to analyze whether active guidance, separation of concerns, or their combined effect lead to the improved performance (if any) of a reading technique. None of the existing studies has investigated the influence of any factor in isolation. The question that arises is: Which of these factors, if any, really improves the outcome of an inspection?

3. The empirical study

In this section, we describe the empirical study that we conducted at the University of Kaiserslautern to examine the influence of active guidance on the inspection outcome.

In order to analyze the impact of active guidance in isolation, we compared inspections providing active guidance by using perspective based reading (PBR) as a special form of SBR against inspection teams using "focused" CBR (i.e., a version of CBR that implements separation of concerns without any active guidance). Our research questions are the following:

- RQ1: Does active guidance improve inspection effectiveness? In other words, does PBR detect more defects than focused CBR?
- RQ2: Does active guidance improve inspection efficiency? In other words, does PBR require less effort per defect than focused CBR?
- RQ3: Do the inspectors perceive the active guidance as a valuable means? In other words, is PBR better appreciated than focused CBR by the inspectors?

Note that in this context, a defect is defined as any problem in the document under inspection; this includes, beside others, information which is incomplete, inconsistent, incorrect, hard to understand, over-specified.

3.1. Subjects and material

The study was conducted as part of a practical course at the University of Kaiserslautern that lasted over 14 weeks, whereby each student invested about eight to ten hours per week. In that course, students were required to change an existing house automation system. After each development phase (requirements, design, implementation), the students had to conduct an inspection of the produced artifacts. Our study focused on the requirements inspection phase, where use cases and related scenarios were inspected.

12 students participated in the study. All students had limited experience with the application domain. Also, the students had some experience in performing systematic requirements inspections.

The software used in the practical course is a reactive system for house automation that was created for and evolved within the course. The system controls a building that consists of an arbitrary number of floors and rooms that have various sensors and actuators. For example, a typical office room contains sensors, actuators to control heaters and light, and a control panel. This panel is used by a person (e.g., a clerk) to set variables of the software

system. Thus, the system was divided into three subsystems.

(1) The graphical user interface (GUI) that offers an interface to control the system.
(2) The light control system (Light) that switches lights on and off depending on the presence of people in a room and a floor.
(3) The temperature control system (Temp) that controls the room temperature, depending on the presence of people in a room and the current daytime.

A group of 4 students was responsible for the development of each subsystem.

During the requirements inspections, the subjects used PBR and CBR approaches to inspect requirements documents. In the case of PBR, the factor separation of concerns was achieved by assigning different perspectives (representing stakeholders of the document) to the inspectors. Each of these stakeholders has different concerns regarding the quality of the requirements document. In the practical course, three perspectives were used in the inspection: a tester (interested in testability of the requirements), a designer (interested in the feasibility of the requirements) and a customer perspective (interested in understandability and completeness of the requirements).

Active guidance was provided by PBR through guiding the inspectors in performing typical activities of the stakeholders (perspectives). Thus, based on the use cases under inspection, the inspector assuming the tester perspective had to derive a set of test cases. The designer perspective had to derive initial statecharts, and the user perspective had to create use cases from the problem description and compare these to the use cases under inspection.

In order to isolate and investigate the influence of active guidance, we created focused checklists that were comparable to the reading scenarios. That is, we defined three checklists, each representing one of the perspectives (tester, designer and user), using similar or the same questions as the corresponding reading scenario. Thus, the focus of each checklist was the same as the focus of the related reading scenario and the checklist provided the same separation of concerns. Note that even though the checklists and the reading scenarios are designed similar to each other, we compare two different reading techniques with one main difference: The active guidance provided by the reading scenarios of the PBR approach. The checklists and the reading scenarios that were used in our study can be found in a technical report [10]

3.2. Experimental design

We chose a fractional factorial design (see Table 1) in which each group participated in two inspections (Run 1 and 2) using a combination of the reading techniques (focused CBR and PBR) and inspected artifacts (the three subsystems). The inspection teams were similar to the teams that build the different sub-systems (e.g. "Group_Light" built the sub-system Light and therefore reviewed the other two sub-systems). Thus, each inspection team included four 4 students. Each sub-system was reviewed by two groups (i.e., 8 subjects), one group using CBR (4 subjects) and one group using PBR (also 4 subjects).

In each run, each inspector used different techniques and different perspectives. For example, the team that used CBR in the first run used PBR in the second run and vice versa. In addition a subject who used the tester perspective in the first run used the designer or the customer perspective in the second run.

Table 1: Experimental design

	PBR	Focused CBR
Run 1	Group_Light inspects Sys_Temp	Group_GUI inspects Sys_Temp Group_Temp inspects Sys_Light
Run 2	Group_Temp inspects Sys_GUI Group_GUI inspects Sys_Light	Group_Light inspects Sys_GUI

The design allows an experimenter to do two comparisons that are not biased by material (i.e., the inspected subsystem) or learning effects: in run 1, between the teams that inspected the subsystem Temp, and in run 2, between the teams that inspected subsystem GUI. However, the comparison between the teams inspecting the subsystem Light is spread onto both runs; therefore, there may be a learning effect that influences the results for this subsystem. In addition, with this design, it is possible to analyze the influence of the active guidance both at the team and individual level. However, the disadvantage of this setting is that it is a between-subject design; that is, observed effects could be caused by differences in performance between the groups.

3.3. Variables

The following are the independent variables we manipulated in our study:

- *Reading technique*: We assigned subjects to different reading techniques: PBR with active guidance and

focused CBR without active guidance. Except this factor the techniques are similar.

- *Inspected artifact*: In our study, we used the requirements documents (use cases and scenarios) of three different subsystems (GUI, Light, and Temp) as artifacts for inspection.

The dependent variables which represent the three main constructs of interest (effectiveness, efficiency, perception of ease of use and usefulness) are the following:

Effectiveness

- Number of defects found by the team (as well as individual inspectors).
- Percentage of defects found by the team (as well as individual inspectors): the number of defects found divided by the total number of defects in the document

The total number of defects in the documents was not known in advance, as the study was conducted in the ongoing process of the practical course. We derived it as follows: First, we listed the defects found by the different inspectors with the different reading techniques. We then removed false positives (i.e., not true defects) and collapsed duplications into unique defects. We finally summed up the unique defects and took that number as the total number of defects in that part of the system.

Efficiency

- Individual Effort: the time (in hours) spent by an inspector for defect detection.
- Team Effort: the cumulative time (in hours) spent by the team inspectors for defect detection.
- Individual Productivity: The number of defects found by an inspector per hour
- Team Productivity: The number of defects found by a team per hour

Subject's Perception

We were also interested in the subject's perception of usefulness and usability of the two reading techniques. The reason to examine these aspects is that, according to [18] the acceptance of a technique is influenced by the perception of its users that the technique is a useful support for their tasks; that is, a benefit is gained when using the technique. Besides usefulness, the second important aspect regarding the acceptance of a technique is that the subjects can easily apply the technique; that is, the technique is easy to learn and easy to apply in their tasks. Both aspects must be considered, as a technique that is useful but hard to apply is worthless, and vice versa.

To evaluate this, we created a questionnaire, based on a 6-point rating scale proposed in [9] and applied in [18] to assess an inspection tool.

In order to measure the construct *ease of use*, we asked the subjects to state their degree of agreement to the following statements:

- The *reading scenario / checklist* is easy to use
- The *steps of the scenario / questions* of the checklists are easy to remember
- The *reading scenarios / checklists* are easy to understand

To measure and compare the *usefulness* of the reading technique, the subjects were asked to state their degree of agreement to the following statements:

- PBR detects more defects than CBR
- PBR detects defects faster than CBR
- PBR increases the productivity of the inspection (good quality in reasonable time)
- PBR makes the inspection easier compared to CBR
- Detailed reading scenarios are useful

4. Data Analysis

Here, we present a first analysis of the quantitative information, which come from the inspection runs and debriefing questionnaires. Results are presented according to the three research questions about effectiveness, efficiency, and users' perception of active guidance.

4.1. Effectiveness

RQ1: Is PBR more effective than focused CBR in finding defects?

Table 2 shows for each of the three documents the total number of defects and the number of defects found by the inspection teams using CBR and PBR. Table 3 shows the percentage of defects detected with focused CBR and PBR on the team level.

Table 2: Number of defects found per team

	# of defects found by CBR-team	# of defects found by PBR-team	Total # of defects
Temp	27	35	54
GUI	21	35	48
Light	25	17	30

Table 3: Effectiveness at the team level

	% of defects found by CBR-team	% of defects found by PBR-team
Temp	50%	65%
GUI	44%	73%
Light	83%	57%

For the Temp and GUI subsystems, the teams using PBR, respectively, found 23% and 40% more defects than the teams using focused CBR. However, for the subsystem Light the team using focused CBR found 32% more defects than the team using PBR.

In Figure 1, the number of defects found by the individual inspectors is shown, using boxplots grouped by reading technique and subsystems. Again, inspectors who used PBR found more defects in the subsystems Temp and GUI while inspectors who used focused CBR found more defects in the subsystem Light.

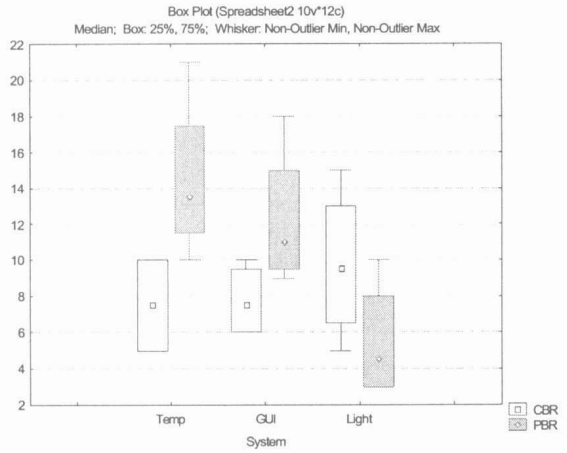

Figure 1: Defects found by individual inspectors

4.2. Efficiency

RQ2: Is PBR more efficient than focused CBR in finding defects?

Table 4 shows, for each document, the team effort (measured in hours). Table 5 shows the team productivity (measured as number of defects found per hour) while using focused CBR and PBR.

Table 4: Required effort at the team level

	Effort in hours for CBR-team	Effort in hours for PBR-team
Temp	14.00	23.00
GUI	18.50	18.00
Light	12.50	13.00

Table 5: Productivity at the team level

	Productivity of CBR-team	Productivity of PBR-team
Temp	1.93	1.52
GUI	1.14	1.94
Light	2.00	1.31

The teams applying focused CBR on the subsystems Temp and Light were more productive than the respective PBR teams (by 27% and 52%, respectively). For the GUI subsystem, the PBR team was 70% more productive (and then more efficient) than the focused CBR team.

In Figure 2, the individual effort spent by inspectors for defect detection is presented (measured in hours) with boxplots. The inspectors using the PBR approach needed more effort (48%) than those using focused CBR for the subsystem Temp, while the effort for the subsystems GUI and Light is approximately the same.

In Figure 3 the individual productivity (number of defects individually found per hour) is shown with boxplots. It can be seen that especially for the subsystem Light, the variation of the individual productivity is extremely high for the PBR technique.

4.3. Subjects' perception

RQ3: Is PBR better appreciated than focused CBR by inspectors?

Here we present the results of the subjective evaluation of usability and usefulness of the reading techniques.

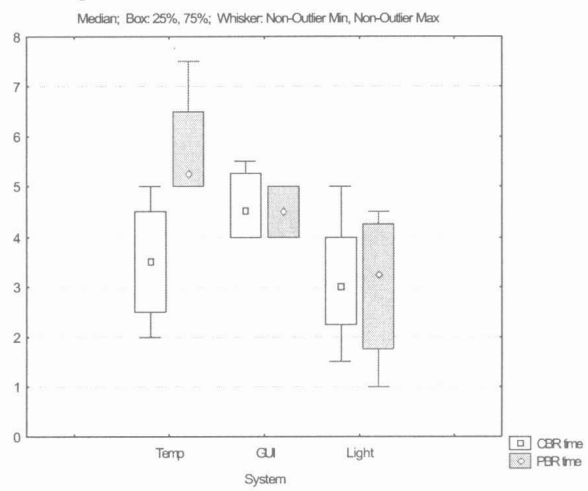

Figure 2: Individual Effort

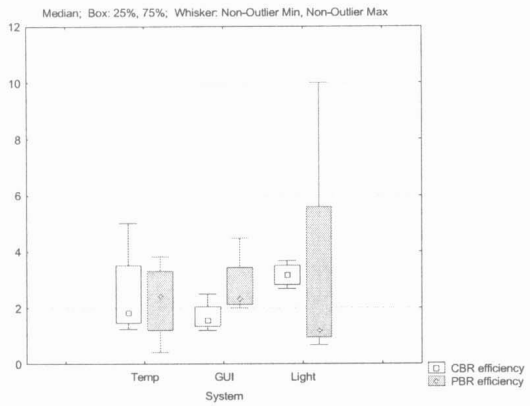

Figure 3: Individual productivity

4.3.1. Usability Figure 4 shows the degree of agreement to the statement that *the checklist / the reading scenarios* are easy to use, while Figure 5 shows the degree of agreement to the statement that the *checklist questions / reading scenario steps* are easy to remember. The subjects perceived both, checklist questions and scenario steps, as rather easy to use and hard to remember.

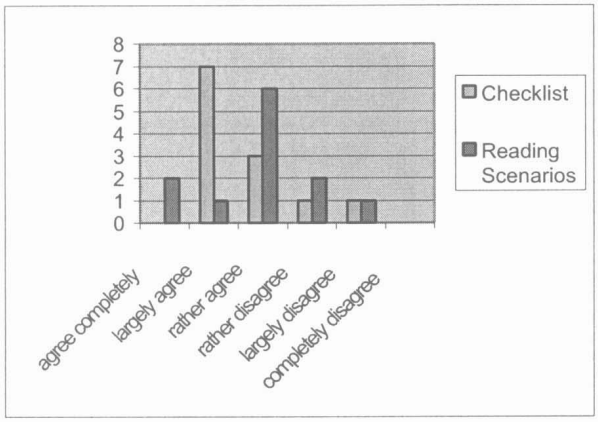

Figure 4: Answers for "easy to use"

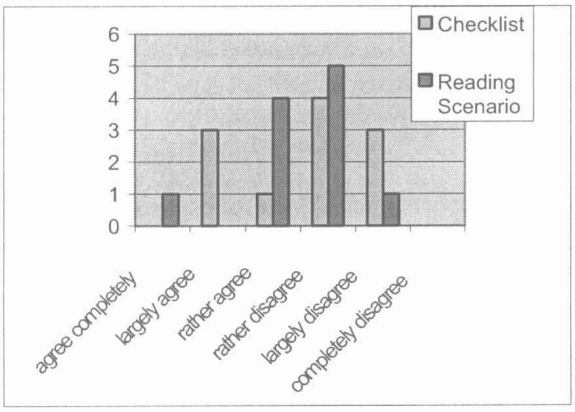

Figure 5: Answers for "easy to remember"

Regarding understandability, we asked the subjects to directly compare the perceived understandability of the CBR technique with the PBR technique. In detail we asked how the students perceive the understandability of CBR in comparison to PBR. Figure 6 shows that there is a weak tendency in favor of CBR (7 out of 12 subjects perceive CBR as easier to understand).

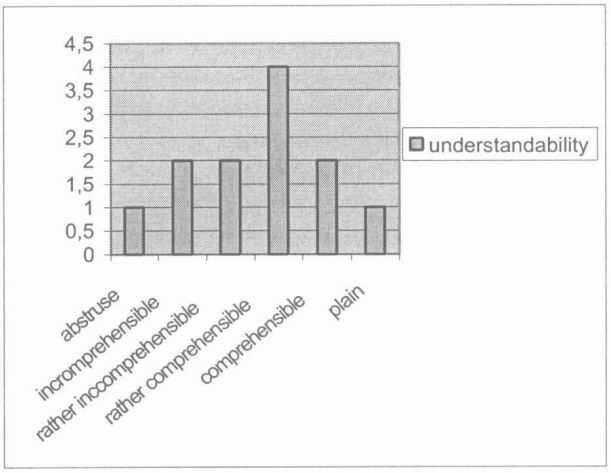

Figure 6: CBR easier to understand than PBR

This makes sense, as a checklist does not contain instructions, and the subjects only have to understand the checklist questions.

In summary, the subjects do not perceive the usability of the two reading techniques differently.

4.3.2. Usefulness Regarding usefulness, the survey shows that PBR is perceived slightly more useful than CBR. In detail, the subjects perceived the number of defects found with PBR higher than with CBR (8 of 12 subjects). This indicates that the inspectors have a higher trust into PBR than into focused CBR. Also, the productivity of PBR is perceived slightly better (7 of 12 subjects). Again this shows a slightly higher trust into PBR.

However, for the other statements explicitly comparing PBR to focused CBR with respect to usefulness, the degree of agreement of the subjects is not visibly in favor of PBR:

- 7 out of 12 subjects perceived the checklists as the faster approach (that is they did not agree to the statement that PBR helps to detect defects faster than with CBR);
- 6 subjects agreed that the reading scenarios facilitate the inspection and also 6 subjects perceived the checklist as facilitating the inspection.

In the last statement regarding usefulness we asked the students of the usefulness of being actively guided. Active guidance is perceived as highly valuable

compared to the checklists. Figure 7 shows that nine out of eleven students (81.8%) agreed that active guidance is helpful in performing the inspection.

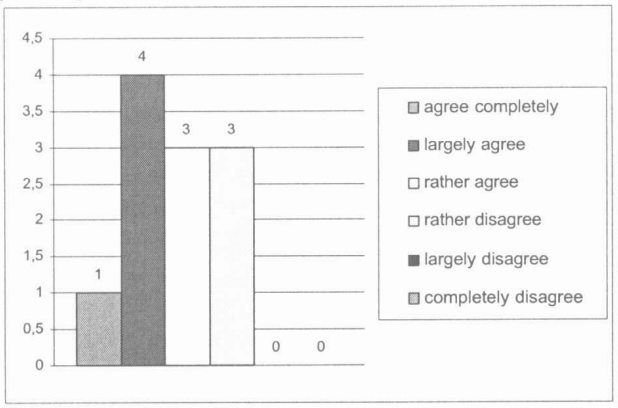

Figure 7: The detailed description of the reading scenarios is useful

Thus, because of the contradictory answers to the statements, we can only find a weak tendency in favor of PBR, mainly based on the agreement about the usefulness of the detailed reading scenarios; that is the active guidance.

5. Discussion

The results for different combinations of the two factors, reading technique and subsystem, do not follow a common pattern. Our analysis showed that active guidance provided by PBR improved the effectiveness of defect detection in the GUI system but not in the Light and the Temp system. Our results do not appear conclusive mainly because the very small sample size of the study (12 inspectors organized in 3 teams with 4 inspectors each) hampers running reliable tests for differences.

However, there are some hints that the complexity of the subsystems can explain the discrepancy in the results. The GUI is the most complex system, as it comprises 34 use cases and related scenarios. Temp is less complex (21 use cases and scenarios) and Light least complex (15 use cases and scenarios). Although we did not expect CBR to outperform PBR, the findings are in line with our assumption that active guidance is most helpful in understanding complex systems. In particular, our findings may hint that the overhead required by applying PBR (e.g., specifying test cases, writing down a statechart model, or re-specifying a use case model) only pays off for significantly large and complex documents. In complex documents, it is much more important, and at the same time more difficult, to get a thorough understanding of the document. As mentioned before, the assumption behind PBR is that this understanding is gained when actively working with the document instead of passive reading. In less complex systems, it is easier for the inspectors to understand the document by pure reading and thus, the overhead of actively working with the document may not pay off. Therefore, this may explain why focused CBR is more effective than PBR for the least complex subsystem, while PBR is strongest for the most complex one. However, this explanation needs to be further investigated, as it is based on the hypothesis that active guidance pays off only for larger (more complex) documents.

One drawback of the PBR approach seems to be its lower efficiency compared to CBR. The additional overhead of the active guidance factor embedded in the PBR approach may explain this result. In the more complex system GUI, the overhead needed for PBR paid off (i.e., PBR was 70% more efficient for this subsystem than CBR) while in the less complex systems (Temp and Light) the CBR approach was sufficient.

From the analysis of debriefing questionnaires, we found a weak preference of inspectors in favor of active guidance in PBR over the lack of it in CBR. Again, also this finding is not conclusive, as PBR is perceived as more useful than CBR but CBR is perceived as easier to apply than PBR. This may mean that the reading scenarios need to be improved.

6. Threats to validity

Threats to validity are factors that influence the dependent variables but are — unlike controlled independent variables — beyond the experimenters' control [4]. This section lists the threats to validity imminent in our design, and some of the measures we took to control or limit them.

We distinguish between threats to *internal* and to *external* validity. Internal validity concerns the interpretation of data collected in an experimental run. If the level of internal validity is too low, data of this experimental run cannot be interpreted properly. Threats to external validity endanger the generalization of the results; that is, if the level of external validity is too low, the results cannot be transferred to different environments.

In the following we list the most important internal threats we identified.

- *Learning effects:* Learning effects occur because the subjects learn more about how to do inspections, and are better the next time. The experimental design minimized this threat: We assigned the groups in such a way that, for each run, we were able to compare focused CBR and PBR on the same

subsystem (Temp in run 1, GUI in run 2, Light in run 1 and 2). Thus, for each comparison, the subjects had the same amount of accumulated experience: None in the first run, and one previous inspection in the second run.

One could argue that the learning effect may be different if you applied PBR in the first run than if you applied CBR in the first run, because you learn to actively work with the document (how to make sure that the subjects forget that in the second run?). Therefore, we assigned different foci to the subjects in the different runs; that is, if they had used the designer perspective (or designer checklist) in the first run, they would use, for example, the tester checklist (or tester perspective) in the second run. Thus, they could not "reuse" their previous experience with the other focus.

- *Instrumentation effects*: Instrumentation deals with the problem that differences in the results may be caused by differences in experimental material. As we compared only results of teams that used the same subsystem, we have reduced this threat.
- *Process conformance*: Students may not have followed the checklists and the scenarios completely. We tried to control this threat by not giving grades tied to the inspector's performance and asking the students to what degree they had followed the instructions.
- *Quality of the reading techniques*: Several experts reviewed the reading techniques with respect to their understandability and completeness.
- *Random heterogeneity*: This refers to the risk that the variation due to individual differences is larger than due to the treatment; that is, observed effects may be caused by differences between the groups rather than by the reading techniques. Usually, assigning subjects randomly to tasks controls this threat. In our case, selection of the participants was restricted by the practical course; random assignment was not possible. In addition, our design is a between-subject design, so we cannot control this threat. However, we found no hints of significant differences between the results of the different groups.
- *Isolation of active guidance*: Even though we aimed at isolating the factor of active guidance by designing the checklists and the scenarios as similar as possible to each other, we still compare two different reading techniques. We might have, therefore, additional influencing factors that are related to the reading techniques themselves and that might have an impact on the results.

The most important threats to external validity are the following:

- *Selection effects / Subject representativeness*: Using students in the experiment reduces the external validity, as they are not representative of typical developers. Moreover, the subjects were graded on the course. To avoid that they "cheated" on the results of the inspections to get better grades, we made sure that the students knew that the inspection results had no influence on their grade.
- *Setting representativeness*: This threat is concerned with not having a setting or material representative for industrial environment or material. In our case, the question is whether the requirements models of our system are comparable to industrial systems. The requirements document of the different subsystems comprises 34 use cases and scenarios for the GUI subsystem, 21 use cases and scenarios for the Temp subsystem, and 15 use cases and scenarios for the Light subsystem. Thus, the system is not a pure toy-example, but we cannot claim that it is comparable to typical real-life systems.

7. Conclusion and future work

In this paper, we have described a preliminary empirical study that examines the influence of active guidance on inspection outcome. In contrast to earlier experiments, we designed the checklists and the reading scenarios to be similar to each other. By doing so, the only distinguishing factor between the reading techniques is the active guidance given by PBR.

We observed some weak evidence that active guidance can help to improve the effectiveness of inspections but seems to have drawbacks regarding the efficiency of the inspection process. Qualitative results, based on a survey among the subjects, indicate that active guidance is perceived as useful to perform defect detection and can help to further improve the inspection process.

However, the results are far from being considered conclusive, as we obtained different results across subsystems. In addition, the study was too small to achieve statistically significant results because there were too few subjects. Therefore, further investigations have to follow. We encourage researchers to analyze the research questions by replicating our study in different settings and performing controlled experiments. One replication of this study has already been carried out at the University of Bari in December 2003. At the moment, we are analyzing collected data and comparing results.

Despite the limited findings we could obtain from this current study, we were able to find hints that active

guidance pays off only if the document under inspection exceeds a certain complexity or size. We need to investigate the influence of document size and complexity on the effectiveness and efficiency of PBR and reading techniques in general. That is, we need to find out for which documents the "overhead" of active reading pays off.

8. Acknowledgement

This study was partially founded by the German Federal Ministry of Education and Research (BMBF) under the grant VFG0004A (QUASAR) and by Provincia Autonoma di Trento in the "Forpics" project. Special thanks go to the students who participated in our study, and to Ralf Carbon and Marcus Trapp for their support in carrying out the study.

References

[1] Ackerman A. F., Buchwald L S, Lewski F H. Software Inspections: An Effective Verification Process. IEEE Software, 6-3 (1989), 31-36.

[2] Basili V R. Evolving and Packaging Reading Technologies. Journal of Systems and Software, 38-1 (July 1997) 3-12.

[3] Basili V R, Green S, Laitenberger O, Lanubile F, Shull F, Sorumgard S, and Zelkowitz M. The Empirical Investigation of Perspective-based Reading, Empirical Software Engineering 1 (1996) 133–164.

[4] D. Campbell, J. Stanley, *Experimental Designs for Research* Houghton Mifflin Company, Boston, MA, 1963

[5] Chernak Y. A Statistical Approach to the Inspection Checklist Formal Synthesis and Improvement. IEEE Transactions on Software Engineering, 22-12, (December 1996) 866–874.

[6] Ciolkowski M. Evaluating the Effectiveness of Different Reading Techniques for Informal Requirements Documents. Master's thesis, University of Kaiserslautern, 1999.

[7] Ciolkowski M., Laitenberger O., Biffl S. Software Reviews: The state of the practice, IEEE Software, December 2003, pp. 46-51.

[8] Ciolkowksi M, Differding C, Laitenberger O, and Munch J, Empirical Investigation of Perspective-based Reading: A Replicated Experiment, Technical Report ISERN-97-13 (1997).

[9] F. D. Davis. Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly, pages 319–340, 1989

[10] Denger, C; Ciolkowski, M; Lanubile, F Empirical Investigation of the Success Factors of Scenario Based Reading, IESE-Report, 115.03/E, Kaiserslautern, 2003

[11] Dunsmore A., Roper M, Wood M. The Development and Evaluation of Three Diverse Techniques for OO Code Inspection. IEEE Transactions on Software Engineering, 29-8 (August 2003).

[12] Fagan M E, Design and Code Inspections to Reduce Errors in Program Development, IBM Systems Journal 15-3 (1976) 182–211.

[13] Fusaro P, Lanubile F, and Visaggio G. A Replicated Experiment to Assess Requirements Inspection Techniques, Empirical Software Engineering 2 (1997) 39–57.

[14] Gilb T, and Graham D. Software Inspection. Addison-Wesley Publishing Company, 1993.

[15] Humphrey W. S. Managing the Software Process. Addison-Wesley Publishing Company, 1989.

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

[17] O. Laitenberger; Cost-effective Detection of Software Defects through Perspective-based Inspections; PhD Thesis in Experimental Software Engineering; Fraunhofer IRB Verlag; 2000.

[18] O. Laitenberger, H Dreyer. Evaluating the Usefulness and the Ease of Use of a Web-based Inspection Data Collection Tool, IESE-Report 027.98/E, 1998.

[19] O. Laitenberger, and J.M. DeBaud. An Encompassing Life Cycle Centric Survey of Software Inspection, The Journal of Systems and Software, 50 (2000), 5-31.

[20] Laitenberger O, El Eman K, and Harbich T G. An Internally Replicated Quasi-Experimental Comparison of Checklist and Perspective-Based Reading of Code Documents, IEEE Transactions on Software Engineering, 27-5 (May 2001) 387-421.

[21] Laitenberger, O., Vegas, S., and Ciolkowski, M. The State of the Practice of Review and Inspection Technologies in Germany. Technical Report ViSEK/010/E, ViSEK, 2002

[22] Miller J, Wood M, and Roper M. Further Experiences with Scenarios and Checklists, Empirical Software Engineering 3 (1998) 37–64.

[23] Porter A, Votta L G, and Basili V R. Comparing Detection Methods for Software Requirements Inspections: A Replicated Experiment, IEEE Transactions on Software Engineering, 21-6 (June 1995) 563–575.

[24] Porter A, and Votta L G. Comparing Detection Methods for Software Requirements Specification: A Replication Using Professional Subjects, Empirical Software Engineering 3 (1998) 355-379.

[25] Sandahl K, Blomkvist Ö, Karlsson J, Krysander C, Lindvall M, and Ohlsson N. An Extended Replication of an Experiment for Assessing Methods for Software Requirements Inspections, Empirical Software Engineering 3 (1998) 327–354.

[26] Sabaliauskaite, G., Matsukawa, F., Kusomoto S., Inoue K.; An Experimental Comparison of Checklist-Based Reading and Perspective-Based Reading for UML Design Document Inspection; In Proc. of the International Symposium on Empirical Software Engineering; Nara, Japan; 2002

[27] Thelin T, Runeson P, Wohlin C. An Experimental Comparison of Usage-Based Reading and Checklist-Based Reading. IEEE Transactions on Software Engineering, 29-8 (August 2003), 687-704.