Fraunhofer Institut
Experimentelles
Software Engineering

Empirical Investigation of the Success Factors of Scenario Based Reading

Authors:
Christian Denger
Marcus Ciolkowski
Filippo Lanubile

Supported by the BMBF under the grant number VFG0004A ("QUASAR") and by Provincia Autonoma di Trento in the "ForPICS" project

IESE-Report No. 115.03/E
Version 1.0
Oktober 24, 2003

A publication by Fraunhofer IESE

Fraunhofer IESE is an institute of the Fraunhofer Gesellschaft.
The institute transfers innovative software development techniques, methods and tools into industrial practice, assists companies in building software competencies customized to their needs, and helps them to establish a competitive market position.

Fraunhofer IESE is directed by
Prof. Dr. Dieter Rombach
Sauerwiesen 6
D-67661 Kaiserslautern

Executive Summary

In industrial settings many success stories can be found about the effectiveness and the efficiency of software inspections. In order to optimize the inspection approach different reading techniques such as checklist-based reading and scenario-based reading have been proposed. Various experiments have been performed to evaluate which of these techniques produces better inspection results; that is finds more defects with less effort. Scenario-based reading approaches performed better then ad-hoc or checklist-based approaches in some experiments but failed to improve the checklist-based approaches in others. Thus, the success factors of scenario based reading approaches need to be further analyzed.

In this report we describe the design and the results of an empirical study that evaluate the impact of the detailed descriptions provided by scenario-based approaches; that is, the impact of active guidance of the inspectors on the inspector's efficiency and effectiveness. In detail, we analyzed the following hypothesis:

Hypothesis H_1—Team Effectiveness: Inspection teams find more defects with active guidance than without active guidance

Hypothesis H_2—Team Efficiency: Inspection teams find more defects per time with active guidance than without active guidance.

In addition, to the hypothesis, we analyzed the inspector's subjective perception regarding the support provided by the PBR approach; that is the support provided by active guidance.

We conducted a quasi experiment (a controlled case study) to compare a focused checklist-based reading approach and perspective-based reading (as an instantiation of an scenario based approach) with respect to the influence of active guidance provided by the PBR approach. The study was conducted as part of a practical course at the technical university of Kaiserslautern that lasted over 14 weeks. 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 few experience with the application domain. Also, the students had few experiences in performing sys-

tematic 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. The system consist of three sub-systems: The graphical user interface (GUI) that offers an interface to control the system. The light control system (Light) that switches lights on and off depending on the presence of people in a room and a floor. The temperature control system (Temp) that controls the room temperature, depending on the presence of people in a room and the actual day-time. A group of 4 students was responsible for the development of each subsystem. We analyzed our hypothesis within these sub-systems.

The results of this preliminary study provide tendencies that active guidance results in more effective inspections and is perceived as very helpful to support individual defect detection. In Detail the preliminary study showed that:

- PBR finds between 23% and 40% more defects than CBR for the Temp and GUI subsystems
- for the subsystem Light CBR finds about 32% more defects than PBR.
- PBR is significant more effective than CBR for the GUI system (p=0.08) and that the other results are not significant

	Temp	GUI	Light
# defects with CBR	27	21	25
# defects with PBR	35	35	17
Total defects in sys.	54	48	30
CBR Effectiveness	0,50	0,44	0,83
PBR Effectiveness	0,65	0,73	0,57

- CBR is more efficient than PBR for the subsystems Temp and Light
- PBR is more efficient for the sub-system GUI
- PBR is perceived as applicable (easy to use) as CBR
- PBR is perceived as harder to understand than CBR
- PBR is perceived as more useful than CBR
- especially active guidance (i.e. the reading scenarios are perceived as very useful):

  The following figure shows that eight out of eleven students (72.2%) agree that active guidance is helpful in performing the inspection.

In summary, the results indicate the tendency that active guidance provides a benefit to the inspectors. However, the results are only first indications and need to be further investigated in controlled experiments.

Moreover, based on the results of the survey we were able to identify some important research questions that need to be addressed in the future. One of the most important is that we need to investigate the influence of document size and complexity on the effectiveness and efficiency advantage of PBR; that is, we need to find out for which documents the "overhead" of PBR pays off, and when we should rather use CBR. More generally, in further research the question needs to be addressed in which context which reading technique is best suited to optimize the inspection process.

Keywords: Software Inspections, Perspective-based Reading, Checklist-based Reading, Empirical Evaluation.

Table of Contents

1 Introduction

After the success of inspections right after their invention by Fagan [Fag76] a lot of effort was put into the optimization of the inspection process. According to Gilb [GG93], the most essential step in each inspection is the defect detection phase, where the inspectors try to identify as many defects as possible in the document under inspection. However, it is a matter of fact that this process step is also the most problematic step. Often the inspectors do not know how to search for defects in a document or what to check, resulting in a high defect slippage rate and a poor inspection efficiency. To overcome this problem, reading techniques were defined that help the inspectors to identify more defects with less effort.

A survey performed by the Fraunhofer IESE [IEEE] shows that the most used reading technique is checklist-based reading (CBR) [AB89, Che96, Fag76, GG93, Hum89], where the inspectors use a list of questions to detect defects in a document. More recent approaches are scenario based reading techniques (SBR) [Lai00, Bas97, BG96, PV95]. Within these approaches the inspectors follow detailed instructions (given by means of a scenario) that prescribe activities the inspectors should perform during the defect detection step. While and after performing the tasks the inspectors should answer questions on the quality of the document under inspection.

Various experiments that compare CBR and SBR have been conducted in recent year. The aim of these results is the empirical evaluation of which reading technique is the better one in terms of improved effectiveness and efficiency of the inspection. However, the results of these experiments cannot give a definite answer to the research question. Some experiments showed that SBR is more effective and efficient then CBR [BG96, CD97, LA00, LE01, PV95, PV98], while other experiments failed to show any improvement [DR01, DR02, DR03, FL97, MW98, SB98, SM+02].

On major drawback of the conducted experiments is that they are not suitable to compare the effectiveness and efficiency of the reading techniques, as they are not "fair". Most of the above mentioned studies compared a standard checklist to a set of SBR scenarios, thus varying two 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 any factor in isolation. The question that arises is which of these factors really improves the outcome of an inspection.

This question needs to be further investigated. The crucial factors of SBR that are considered to result in an improved inspection result are: (1) giving more guidance on how to perform the inspection through actively working with the document (*active guidance*), (2) providing different reading scenarios for individual analysis whose results are combined at the team level (*composition*), and (3) restricting the focus of a reviewer to a specific aspect of interest (*separation of concerns*). The factors active guidance and separation of concerns are those factors, that might have a direct impact on the outcome of an inspection. It is obvious, that the factor separation of concerns (focusing the inspector) can also be realized with a CBR approach. Thus, the only decisive difference between SBR and CBR is active guidance.

We performed a quasi experiment (a controlled case study) to examine the influence of active guidance on inspection outcome. In contrast to earlier experiments we designed the checklists and the reading scenarios similar to each other. By doing so, we isolated the factor of active guidance. Thus, we are able to analyze the influence of this factor on the inspection outcome. Note that in this paper, the term inspection is used in the sense of a structured review process. Other static quality assurance methods like walkthroughs, management or team meetings are not considered.

In Chapter 2 we describe the empirical study in more detail and analyze the related work. In Chapter 3 we describe the context of the empirical study and the experimental design. The results of the study are presented in Chapter 4 and in Chapter 5 these results are further discussed. The report closes with a conclusion and future research activities. Note that in Appendix A a detailed description of the results of a questionnaire is presented, which was part of the study described in this report.

In this report, the term inspection is used in the sense of a structured review process. Other static quality assurance methods like walkthroughs, management or team meetings are not considered as an inspection in this report.

2 Context of the Empirical Study

This chapter gives background information regarding the empirical study. First a general introduction to inspections and different reading techniques is given in Section 2.1. Related work, especially experiments regarding the comparison of checklist based reading and scenario based reading approach, is discussed in Section 2.2. Also the hypothesis analyzed in our empirical study is stated on an abstract level.

2.1 Introduction

The empirical study described in this report investigates the impact of active guidance provided by SBR reading approaches on the outcome of a requirements inspection. The overall goal of the empirical study is to analyze the most important factors of the SBR approach on the effectiveness and efficiency of an inspection. First of all we will introduce software inspection and the main reading techniques to support the individual review activity: checklist-based reading and perspective-based reading.

An inspection is a static quality assurance method that allows the identification and correction of defects early in the software development cycle. In this report, the term inspection is used in the sense of a structured review process. Other static quality assurance methods like walkthroughs, management or team meetings are not considered as an inspection in this report.

The definition of an inspection as a static analysis method to detect defects in software life cycle products is very abstract and therefore, this definition needs to be enhanced. In addition to the definition, the following aspects characterize an inspection:

1. It follows a defined process
2. The participants of an inspection have defined roles
3. The inspectors are supported by reading techniques
4. The participants in an inspection are trained
5. The results of an inspections are documented

These characteristics are explained in more detail in the following sections.

As depicted in Figure 1 the inspection process consists of four basic steps, which are essential for a good inspection result. One or more roles participate in each step and one of these roles is responsible for the correct performance of the related step. Finally, the figure shows that in each step certain documents are produced or serve as input document. In the following paragraphs each step and the related roles are explained in more detail.

In the planning step, the organizer of the inspection is responsible for planning the whole inspection process. This activity includes the scheduling of the different process steps, to provide the document under inspection and all the other important document, e.g. checklists, reading scenarios, to the inspectors, reserving rooms for the meeting.

Figure 1: The Inspection Process

In the detection step, each inspector searches for defects in the document under inspection. The inspectors are supported by reading techniques. During the detection step, all the issues, i.e. errors, questions, improvement suggestions, raised by the various inspectors are logged in a defect report document.

In the collection step (inspection meeting) the issues raised by the individual inspectors are merged into a defect list during the inspection meeting. The moderator is responsible for leading the meeting into the right direction. The aim of the meeting is to decide whether an issue raised by an inspector is a defect or not. Therefore, the author of the document under inspection shall participate in the meeting to answer questions or to clarify vague aspects in the document. The moderator must assure that the issues are not discussed too long (the duration of the meeting shall not exceed 2 hours, otherwise a second meeting should be scheduled) and that the inspectors evaluate the product not the author of the product.

Finally the author is responsible for the correction of the defects that the inspection team agreed upon in the meeting.

In order to assure successful inspections it is essential that all the results of the inspection are documented. For example, defect logs and effort sheets give valuable input for the evaluation of the effectiveness and the efficiency of an inspection. Moreover, certain roles in an inspection should be trained in specific skills. For example, the moderator needs special social skills to efficiently lead the inspection meeting.

After the introduction of inspections various kinds of reading techniques were proposed to optimize the defect detection step; that is, to support inspectors during the defect detection step of a software inspection. CBR [AB89, Che96, Fag76, GG93, Hum89] is the most frequently applied reading technique [CLB03] while SBR is a more recent reading approach. Examples of SBR include defect-based reading (DBR) [PV95, PV98, MW98], perspective-based reading (PBR) [BG96, Lai00] and traceability-based reading (TBR) [TS+99]. More recently, usage based reading was introduced that focuses the inspectors with use cases [DR01, DR02, DR03, TR03]. In this report, we use CBR and PBR (as an instance of the SBR approach) to analyze the impact of active guidance.

Beside ad-hoc reading (that is, just reading a document without any guidance) **Checklist based reading** is the most frequently used reading technique. Within this approach the inspectors are supported by a checklist that contains questions each inspector has to answer during the defect detection phase. These questions focus on certain quality aspects of the document under inspection, for example completeness, correctness, etc. The checklist approach tells an inspector *what* to check; that is what to look for in the document. However, it provides only poor guidance *how* to check whether a certain quality aspect is fulfilled.

According to different sources, e.g. [Wie02, Lai00, GG93], a checklist shall adhere to the following criteria:

- paraphrased as precise as possible
- not longer than one physical page
- structured so that the quality aspect the questions are focused on is clear to the inspectors
- kept up to date
- focused on questions that reveal major defects
- a checklist question that is answered with “no” points out a potential defect
- derived from guidelines, rules, quality aspects used in the project context in which the artifact under inspection was created.

The basic idea of the **scenario based reading (SBR)** technique is that the inspectors are guided by different reading scenarios that tell them *what* to look for during the inspection and *how* to perform the defect detection. Furthermore, the reading scenarios provide active guidance; that is, the inspectors have to perform activities on the document under inspection while searching for defects.

Active guidance forces the inspector to produce real work. Doing that, inspectors gain a better understanding of the document than by "passive" reading alone, and are thus able to detect more subtle defects. Active guidance is assumed to be especially useful in inspections of complex systems, where the active work helps to gain insights that would have been missed in passive reading. Finally, the attention of the inspectors is focused on the essential parts of the document under inspection and thus avoids a cognitive overhead of the inspectors [Lai00].

The ideas of the scenario based approach are implemented in reading scenarios. A reading scenario gives a step-by-step description of the activities an inspector should perform during the defect detection; for example, an inspector might produce test cases or initial design diagrams during defect detection [Lai00].

Following [LK01] A reading scenario consists of three main parts:

1. Introduction
2. Instructions
3. Questions

In the *introduction*, the goal of the scenario is described and the quality aspects that are most important in the particular scenario are defined. Thus, the focus of the inspector is set; that is, it is clarified what should be inspected. In the *instruction*, an inspector gets concrete guidance how to work with the document under inspection in order to detect defects and to gain a profound understanding. Furthermore, the instructions focus the attention of the inspectors to the essential information in the document. For example, the instruction part of a scenario that supports defect detection in a requirements document can state that an inspector shall derive a high level statechart diagram from the use cases. Finally, the *questions* focus on common defect sources in a particular document or entity under inspection and thus, help the inspector to detect defects related to these questions while working with the document and to decide whether or not the document under inspection fulfills certain quality criteria [LK01]

Examples of scenario based approach include defect-based reading [PV95, PV98, MW98] and perspective-based reading [BG96, Lai00]. More recently, usage based reading was introduced that focuses the inspectors with use cases

[DR01, DR02, DR03, TR03]. In the empirical study described in this report, we used perspective based reading as an instantiation of the scenario based approaches

In the **perspective based reading approach** (PBR), which is a special form of the SBR approach, the scenarios are defined according to different stakeholders of the document under inspection. The approach is based on the fact, that different stakeholders have different perceptions of the quality of a particular document. For example, a customer of the future system has other needs on the requirements documents (understandability, completeness) than a tester who is responsible for deriving test cases from the requirements (testability, correctness). The PBR approach assures, that all necessary views on the document are considered during the inspection and thus, a maximum of possible defects can be detected. Due to this characteristic of PBR one of the essential steps in this approach is the identification of all the important stakeholders of the document under inspection. If the perspective of one stakeholder is not considered in the inspection, the inspectors might miss essential quality criteria during the defect detection phase and thus critical defects might remain undetected [LK01].

One major advantage of PBR is that the inspection results can be evaluated. As real work products are created in the defect detection step, it is possible to have a detailed look at these documents. A quality manager or the inspection champion can then evaluate the quality of these documents and gets hints about the quality of the inspection. Created documents of poor quality indicate a poor quality of the inspection. However, the verifiability of process conformance is not related to the efficiency and effectiveness of the defect detection step performed with the reading technique. Thus, it is not further evaluated in the experiment described in this report.

In the following we analyze the characteristics of SBR approaches in general and the related characteristics of PBR approaches in particular

Considering the characteristics of the SBR approach (and more generally the SBR approach) providing *active guidance* is the most decisive factor of these approaches. That is, the instructions of the reading scenarios give concrete guidance on how to perform certain activities while searching for defects. In the case of PBR, the reading scenarios are tailored particular perspectives of stakeholders of the document under inspection. The instructions describe activities that are typically performed by these stakeholders. For example, from the perspective of a tester, the inspector has to develop test cases from the document under inspection.

The second factor of SBR is that the document is inspected from different points of views. In the PBR approach, these views correspond to the stakeholders of the document under inspection. Thus, the inspectors are not responsible for all possible quality aspects of the document but are focused on special aspects that are most relevant for a certain perspective. We refer to this factor as *separation of concerns*.

Finally, more than one reading scenario is used to inspect the document. Thus, the overall inspection result is a composition of inspection results gained by different scenarios. This factor is referred as *composition*. However, the composition of different individual inspectors to a team result has no direct impact on the inspection result. Therefore, only the factors separation of concerns and active guidance are further investigated.

Only active guidance is special for SBR approaches when comparing them to the CBR approach. Of course, in CBR, checklists are normally used in a way that all the inspectors involved in the inspection process use a single checklist. However, the checklists are not restricted to this way of usage. It is also possible to use different checklists in an inspection and to give the inspectors a specific focus with the assigned checklist (*separation of concerns*) by focusing the checklist questions on certain quality aspects. Thus, active guidance is the only decisive factor of SBR that differentiates this reading technique from others. The research question that arises in this context is whether active guidance has an impact on the outcome (effectiveness and efficiency) of the inspection process.

Of course there are additional influencing factors on the efficiency and the effectiveness of an inspection such as the experience of the inspectors, the document under inspection, the effort for preparation and the project and company environment in which they are used. However, these aspects are not explicitly evaluated in this experiment.

2.2 Related work

In recent years, several experiments were performed to compare the reading techniques with respect to their effectiveness and efficiency. The first family of scenario-based reading techniques, Defect-Based Reading (DBR), was defined for detecting defects in requirements documents written in a state machine notation for event driven-process control systems [PV95]. Each DBR scenario is based on a different class of requirements defects and requires a different model to be built before answering to specific questions. In order to empirically validate their proposal, controlled experiments were first run with graduate students at University of Maryland [PV95] and then with professional software developers from Lucent Technology [PV98]. Both the experiments showed that DBR was significantly[1] more effective than ad hoc and checklist reading. Replications were performed by other researchers [FL97, MW98, SB98] who reused

[1] Here and in the following the significance level, for which we summarize results, is set at the conventional level of 0.05.

the same experimental material and slightly changed the experimental design. External replications did not measure any improvement in inspection effectiveness.

Perspective-Based Reading (PBR) is another family of scenario-based reading techniques which was initially proposed to improve the inspection effectiveness for requirements documents expressed in natural language [BG96]. The idea behind PBR is that the stakeholders of a product should read the document from their particular point of view. For PBR, the different viewpoints are the various roles within the software development process (e.g., designer, tester, user). To support the inspector throughout the reading process, operational descriptions, i.e., scenarios, for each role are provided. PBR has been first empirically validated with software developers of NASA/Goddard Space Flight Center [BG96]. The results showed that teams using PBR achieved better coverage of general documents and were more effective at detecting defects in requirements documents than teams which did not use PBR. These results were confirmed in a replicated experiment conducted by other researchers with undergraduate students [CD97]. PBR has been also tailored and applied for reviewing design documents and source code. Results from two empirical studies with practitioners as part of training courses show that PBR is more effective and less costly than using checklists for defect detection of UML design documents [LA00] and C code [LE01].

From the empirical studies of PBR, an unexpected effect was that individuals applying PBR, rather than just teams, were more effective for defect detection than individuals applying less systematic reading techniques. This effect seems to provide evidence that one of the properties of PBR, being systematic, might be a sufficient cause for improved inspection effectiveness.

A study involving undergraduate students inspecting OO code [DR01] compared the performance of a systematic reading technique, derived from the Stepwise Abstraction approach [LM79], with an ad hoc approach to code reading. The systematic technique required inspectors to reverse engineer an abstract specification for each inspected class method. No significant differences were found between the systematic technique and the ad hoc approach in terms of defects individually discovered. A follow-up study [DR02, DR03] replaced the ad hoc approach with checklists, and introduced a new scenario-based reading technique, called use-case reading technique. Use-case inspectors had to start from system use cases, generate a number of scenarios from them, and create sequence diagrams for examining how code under inspection deals with these scenarios. The experiment failed to reveal any significant difference between the defect detection capability of the three reading techniques. Checklists allowed inspectors to find defects at a quicker rate.

Usage-based reading [TR03] is reading technique that mimics operational-profile testing to focus verification activities on a set of use cases as test cases focus the test effort. It assumes that use cases and related scenarios have been defined earlier in the development process and are made available to inspectors. An experiment with design documents checked whether asking inspectors to develop detailed use cases while reviewing could improve defect detection [TR02]. Results showed that reviewers performed better when uses cases were ready from the beginning of the inspection.

The results of these experiments are not conclusive. They indicate that there is more research necessary to analyze in which cases PBR, and more generally SBR, is more efficient and effective than CBR. This implies the necessity to analyze the success factors of perspective (scenario) based reading; that is, to analyze the impact of those characteristics of perspective based reading on the inspection result that are special for the technique.

Most of the past experiments have some drawbacks. Often one generic checklist was compared against several reading scenarios containing more detailed and different questions than the checklists [e.g.PV98, SB98LA00, LE01]. Thus, two factors were varied in the experiment: Separation of concerns (i.e., focusing the inspector), and active guidance. In other cases, reading scenarios did not contain detailed instructions; thus, the active guidance was not implemented, and they were almost similar to the checklists [PV98, SM+02]. This makes it almost impossible to analyze which of the factors of SBR lead to the improved performance of the reading technique or why other experiments failed to show a difference. The basic problem is that existing studies do examine active guidance and separation of concerns only in conjunction.

Thus, further research is necessary to evaluate the influence of active guidance on the outcome of the inspection. Our hypothesis is that the active work on documents induced by scenarios produces better inspection results; that is, more defects can be detected and the efficiency of the inspectors is improved. In the case that the hypothesis is true, the higher degree of guidance (essentially instructions on what to do with the document) provided by a scenario based reading approach would have a significant influence on the performance of the inspectors

3 The Empirical Investigation

In this chapter the details of the empirical study are described. In Section 3.1 the hypothesis analyzed in the empirical study are described in detail. In Section 3.2 the context of the experiment is described; that is, the project in which the study was conducted and the subjects participating in this project. The experimental material and the approach of the empirical study is described in Section 3.3 and Section 3.4, respectively. The detailed experimental design and the most important variables are outlined in Section 3.5.

3.1 Hypothesis of the empirical study

In order to analyze the impact of active guidance, we examined the influence of this factor on inspection teams that use PBR (i.e., getting active guidance) and compared the results with inspection teams that use CBR (i.e., not getting active guidance). In detail, we analyzed the following hypothesis:

Hypothesis H_1—Team Effectiveness: Inspection teams find more defects with active guidance than without active guidance

Hypothesis H_2—Team Efficiency: Inspection teams find more defects per time with active guidance than without active guidance.

Note that in this context a defect is defined as a problem in the document under inspection; this includes, beside others, incomplete, inconsistent, incorrect, hard to understand, over-specified and not feasible aspects.

Beside this hypothesis we are interested in the general perception of the students regarding the different reading techniques; that is, we are interested in the question whether the subjects perceive the support provided by active guidance is helpful and useful when performing an inspection. These qualitative results support the analysis of our hypothesis.

3.2 Subjects/Participants

The empirical study was performed in the practical course "Software Engineering 1" at the technical university of Kaiserslautern. The practical course lasted over 14 weeks. In the practical course the students had to develop a building automation system that regulates the temperature and the light in rooms and floors of a university building. Thereby, the system controls 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. The students had to develop the whole system following the V-Model, starting from a general problem description of the system and ending with the acceptance testing of the complete 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. Within this development process the students had to develop use cases and scenarios based on a problem description of the system. After the creation of the use cases and the scenarios the students performed inspections of the created documents.

In the working description of the practical course the system was separated into three sub-systems, namely, the "Temperature Control" (Temp), "Light Control" (Ligh) and "User Interface" (UI) system.

In the working description of the practical course, 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 actual day-time.

12 subjects were involved in the case study and a group of 4 people was responsible for the development of one sub-system. During the inspection phase, each group had to inspect the two sub systems of the other groups. As part of the inspection process the students had to document the detected defects and each defect had to be classified according to a defect classification scheme. In addition the subjects had to document the effort they spent on the use case inspections.

The results of the evaluation of another case study that was performed with the same subjects shows that all of the subjects participating in the case study are students, enrolled at the university of Kaiserslautern in computer science, applied computer science, or industrial engineering with computer science as their major subject. All of the subjects have passed their intermediate exams and are studying for at least 6 semesters. The evaluation of the subject's experience with embedded systems and building automation systems showed that all of the subjects are rather inexperienced in the domain of embedded systems and none of the subjects is familiar with the domain of building automation systems. Furthermore, the subjects are quite familiar with requirements engineering in general and use cases in particular.

The subjects have different experiences with inspections. Most of the subjects have none or few experiences with inspections in general (7 out of 12) as well as with requirements inspections in particular (8 out of 12). The other subjects state that they have average (1 answer) or advanced experiences of inspections in general (3 answers). as well as advanced experience in requirements inspections (4 answers). However, a brief introduction to software inspection at the beginning of practical course showed that none of the subjects has performed systematic inspections. The reading technique and the process they applied in earlier inspections were rather ad-hoc.

Half of the subjects have an average knowledge of the general inspection process from their lectures. 3 subjects have fewer knowledge and 3 subjects advanced knowledge of the inspection process. Thus it can be assumed that the subjects have rather similar knowledge of the inspection process but have different experiences in its practical application.

3.3 Experimental material

The first task in the practical course was the derivation of use cases and scenarios from a given problem description. In addition for each sub system the more detailed user requirements written in natural language were used as an input document for this task. Thus, each inspector got the use case document and the related scenarios for the sub systems he or she had to review. The documents were verified against the problem description and the detailed user requirements for the sub systems.

To support the inspector in the defect detection step they used checklists and perspective based reading scenarios. The reading techniques were tailored to support the inspection of use cases in the requirements engineering phase. To develop the reading scenarios, the most relevant stakeholders of the use cases were identified. These stakeholders are the tester, the designer and the user of the system. Beside these perspectives other can be thought of such as a project manager perspective, a marketing perspective, etc. Again in the context of the practical course the perspectives of the tester, the customer, and the user are the most important ones as these roles directly work with the use cases to create other work products such as test cases or the high level design. The designer and the tester perspective need to make sure that it is possible to perform their tasks with the use cases. The user perspective need to make sure that all his or her needs are correctly and completely considered in the use cases.

In order to support the verification of these quality aspects, the reading scenarios focus on the quality needs and the most frequent activities of the different stakeholders. The tester scenario focuses on testability and completeness aspects and requires the inspector to derive acceptance test cases from the use cases. The designer scenario verifies correctness, consistency and feasibility aspects and requires the inspectors to create statecharts from the use cases. Finally the user scenario focuses on completeness and correctness aspects and requires the inspector to design use cases based on the problem description and to compare these use cases with the use cases under inspection. The original scenarios have been developed in German and were translated to English for the practical course. In Appendix B all the scenarios are described in detail.

In order to isolate and investigate the influence of active guidance we created checklists that were comparable to the reading scenarios; that is, we created three checklists that were related to the reading scenarios. Thus, the focus of each checklist was the same than the focus of the related reading scenario; i.e., the checklist and the reading scenarios provided the same separation of concerns. Finally, the questions of the checklists were similar to those in the reading scenarios; that is, in the case that the reading scenario was asking to check the completeness of the actors, the checklist contained the same question. Of course it was not possible to map all of the scenario questions to a similar checklist question as some of the scenario questions are directly based on result of the active work of the inspectors with the document under inspection. For example, in the tester scenario some questions refer to the created test cases. As the inspection with checklists does not create these work products it is not possible to use related question in the checklist. However, with this design of the checklists and the reading scenarios the only difference between the reading techniques was the active guidance provided by the reading scenarios of the PBR approach.

Note that in the following chapters we will use the term perspective to refer to the perspectives of the scenarios **and** the checklist. Also, we use the term focused CBR to differentiate our checklist-based approach from more traditional checklist based approaches-

The subjects followed the general phases of the inspection process as described in Section 2.1. Note that beside the reading techniques the subject's used a predefined issue-list to document the detected issues during the individual inspection of the documents. The individual lists were then transformed into one defect list for each sub-system in the inspection meeting.

3.4 Procedure

In the experiment, subjects were first asked to complete a questionnaire regarding their experiences with software inspections in general and with the different reading techniques in particular. The results of the evaluation of this questionnaire were used to describe the subject's background in Section 3.1.

Before the subjects started the use case inspections a brief introduction to the topic of inspections was given to them. This introduction showed the basic steps of an inspection (planning, defect detection, meeting, and rework) and the essential aspects of these steps. Afterwards the subjects were given the inspection materials; that is, the documentation of the two sub-systems they had to review and the supporting documents for the defect detection step (see Section 3.3). Each subject received only the reading scenario of the perspective he or she assumed in the inspection. The same holds for the checklist. All the subjects were told not to talk to other team members about their perspectives to ensure that the subject's only use the assigned perspective.

For the first inspection the subjects had two and a half days to perform the inspection then the inspection meeting was held. At the end of the meeting each inspector got his new reading technique. Those subjects who inspected the use cases with the checklist now got a perspective based reading scenario and vice versa. Also a different perspective was assigned to the subject's, e.g. one subject used the tester-checklist in the first run he or she used either the designer or the analyst perspective based reading scenario in the second run. For the second inspection the subjects had again two and a half days for the defect detection phase. This second run also ended with an inspection meeting after the assigned time.

In addition to the effectiveness and efficiency of the reading techniques we were interested in the subject's perception of the different reading techniques. To evaluate this question, we used a debriefing questionnaire to ask the students about how applicable and how useful the reading techniques after they had finished the two inspections. The qualitative results gained with this questionnaire support the interpretation of our empirical evaluation of the hypothesis.

3.5 Experimental Design

The study was conducted in two runs (see Table 1) In the first run, one group inspected the subsystem Temp with PBR, while the other two groups inspected Temp and GUI using focused CBR. Thus, factors in the design were the reading technique, focused CBR and PBR (and thus, active guidance), and the subsystem under inspection (GUI, Temp, and Light).

Due to the restrictions of the practical course we were not able to create a balanced design. All the sub-systems (Temp, Light, GUI) had to be inspected and each group had to inspect the sub-systems of the other groups.

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 inpsects Sys_GUI Group_GUI inspects Sys_Light	Group_Light inspects Sys_GUI

Table 1: Experimental Design of the study

All subjects were told not to talk about the systems and the detected defects within these systems to the members of the other groups in order to avoid that the second inspection of the light system is biased, Also, we performed a blind experiment; that is, the subjects were not told about the hypothesis.

We choose this design with the goal to reduce learning effects, and to control the influence of the subsystem. Learning effects could occur because the subjects learn more about how to do inspections, and are better the next time. In that case, it is hard to determine whether an improvement in the dependent variables was caused by the reading technique or by the learning effect. In addition, the characteristics of the inspected artifact can influence the inspection results. For example, the complexity and the size of the artifact can vary and thus it is again difficult to determine whether the artifact or the reading technique caused an improvement in the dependent variables.

In order to control these threats 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). In consequence, we can be confident that the inspected artifact did not cause differences in the result of the teams while inspecting that artifact. Moreover, for the GUI and Temp system, the learning effect was similar for all subjects, as all comparisons between techniques referred to the same run. One could argue that the learning effect may be different if you applied PBR in the first run and FOCUSED CBR in the second one, because you learn to actively work with the document (how to make sure that the subjects forget that in the second run?). To minimize this threat, we started with two groups reading FOCUSED CBR in the first run and just one group reading PBR in the first run. Also we assigned different perspectives (foci) in each run, so that the subjects might remember the principles of PBR but not the concrete scenario relevant for the perspective they adopted in the second run; 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.

The design allowed us to do two unbiased comparisons: In run 1, between the teams that inspected the subsystem Temp, and in run 2, between the teams that inspected subsystem GUI. 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. With this design we could analyze the influence of the reading technique on the team level; that is, whether the reading technique improves the performance of an inspection team. We were not able to analyze the influence of the different foci provided by the checklists and the reading scenarios. However, we were only interested in the impact of the reading technique on the inspection teams.

Moreover, we are not in a controlled situation as the students had to perform each inspection over a period of two and a half days. Thus, we have a lower internal validity but also a higher external validity as this represents more a realistic situation. Thus, we cannot talk of a controlled experiment but of a quasi experiment

To investigate the hypothesis in an empirical study, one has to deal with different kinds of variables. Dependent variables represent the result of the empirical study while independent variables may influence the dependent variables; that is, changing the value of an independent variable may have an impact on the value of a dependent variable. In Table 2 the most important independent variables we identified for our study are summarized. Moreover, actions to control the independent variables are described.

Independent Variables	
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:	Characteristics of the artifact can influence the outcome of an inspection. In our study, we used the requirements documents (use cases and scenarios) of the three different subsystems (GUI, Light, and Temp) as artifacts for inspection.
Subject's experience with software development and inspections:	We captured the subjects' previous experience in creating use cases and performing requirements inspections to be able to check that the experience is comparable across the different experimental groups.
Quality of the reading techniques	The quality of the checklists and reading scenarios can influence the inspection result. Several experts reviewed the reading with respect to their understandability and other quality concerns.
Quality of the inspection	If the inspectors do not use the checklists and the reading scenarios the inspection outcome cannot be interpreted with respect to the reading techniques. In a debriefing questionnaire we asked how our subjects behaved during inspection, to be able to check that they applied the checklists and reading scenarios.

Table 2: Independent Variables in the empirical study

The dependent variables are listed in Table 3.

Effectiveness of a reading technique: The number of defects found in a subsystem

Efficiency of a reading technique: The number of defects found per hour.

Dependent Variables	
Effectiveness of a reading technique	The number of defects found in a subsystem in relation to the total number of defects found in the subsystem
Efficiency of a reading technique	The number of defects found per hour

Table 3: Dependent variables in the empirical study

The variable "inspected artifact" is confounded as the inspections were performed in a realistic situation, that is, we could not introduce defects and we could not see the documents before the inspection. We minimized this effect by reducing the influence of the document to one group over the two runs. The two other documents were used in one run thus, limiting the document influence to the involved groups.

As we were in a realistic situation, we could not control how good the subjects performed the defect detection step. This might have an influence on the inspection result. However, the evaluation of the debriefing questionnaire indicates that the subject's followed quite well the checklists and the reading scenario's as they stated to have used most of the checklist questions and reading scenarios. For the reading scenarios the subjects stated to have developed the required work products (test cases, statecharts) which also indicates a thorough defect detection.

Finally, the way of how the questions in the checklists and the reading scenarios are written might have an influence on the inspection result. To reduce the influence of this factor the checklist questions and the perspective based reading scenarios were compared by several experts with respect to the question whether they are similar to each other or not.

To measure the impact of the active guidance of PBR we collected data about how many defects were found during an inspection (effectiveness) and how much effort was spent to detect the defects (efficiency). These are the most frequently used metrics to evaluate the success of an inspection.

3.6 Threats to Validity

This section discusses the threats to validity that are important in the setting of the empirical study. Several threats were already mentioned in the discussion of the experimental design and in the description of the different variables used in the study. (see Section 3.5) Therefore, we describe the different threats in a narrative way rather describing them in a technical way.

Threats to validity are factors that influence the dependent variables but are — unlike controlled independent variables— beyond the experimenters' control [Camp63]. Moreover, the experimenters may not even know about some of these factors. One purpose of the experimental design is to minimize their effect on the result (by selecting an appropriate design) and to capture them, if possible, to be able to explain unexpected results. 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

The following list contains the most important internal threats we identified: Learning effects, instrumentation effects, and process conformance.

- *Learning effects,* As explained in detail in Section 3.5. the experimental design minimized this thread, as CBR was used in two cases in the first run and perspectives were changed in each run.
- *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 minimized this threat.
- *Process conformance*: Students may not have followed the checklists and the scenarios completely. We tried to control this threat by asking the students to what degree they had followed the instructions.
- *Selection effects*: Results can be caused by variations in human performance. Usually, this threat is controlled by assigning subjects randomly to tasks. In our case, selection of the participant was restricted by the practical course. Thus, we were not able to randomly select and randomly assign the participants to the different groups. The risk is that the better students form one group. However, we could not observe such an effect.

The most important threats to external validity are the following:

- *Subject representativeness*: Using student in the experiment reduces the external validity, as they are not representative of the typical developer. However, recent studies have shown that the difference between students and "real" developers may not be as large as assumed [Hoe00]. 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 system comprises of about 16.000 lines of code and about 80 Java-classes. Thus, the system is not exactly tiny, but we cannot claim that it is comparable to typical real-life systems.

4 Analysis of the Results of the Empirical Study

In this Chapter we present the descriptive results of our empirical study. In Section 4.1 we present the results of the comparison of the effectiveness and efficiency of the teams using the different reading techniques. Note that the statistics are presented without further comments. A detailed discussion of the results is given in the next chapter. In Section 4.2 the results of the qualitative survey (questionnaire) are summarized.

4.1 Comparing the Efficiency and Effectiveness of PBR and Focused CBR

The results of the effectiveness of the different reading techniques are described in the following paragraphs. We start with the data we collected during the study regarding defects found by the inspection teams, time the teams spent for defect detection etc. The results are presented according to the different sub-systems (Temp, GUI, Light) that were part of the inspection process.

In order to analyze our hypothesis we collected data regarding the number of defects found and the effort that was necessary for the inspection (only defect detection). To back up the results, we conducted a statistical test to compare the individual inspectors' results. We analyzed the difference in the number of defects found with PBR and CBR (effectiveness). Because we only had few subjects, we chose to conduct a Mann-Whitney U Test. Note that due to the small number of subjects the presented results only show tendencies but need to be further investigated.

Table 4 shows the results regarding the hypothesis H_1 (team effectiveness). The table shows the number of defects detected with a reading technique, the total number of defects detected in the document, and the effectiveness of the reading techniques. Effectiveness is defined here as the number of defects found with a technique in relation to the total number of defects found.

	Temp	**GUI**	**Light**
# defects with CBR	27	21	25
# defects with PBR	35	35	17
Total defects in sys.	54	48	30
CBR Effectiveness	0,50	0,44	0,83
PBR Effectiveness	0,65	0,73	0,57

Table 4: Number of defects found by and effectiveness of the reading techniques

As can be seen, PBR finds between 23% and 40% more defects than CBR for the Temp and GUI sub-systems, respectively. However, for the sub-system Light, where we expected an even larger difference, CBR finds about 32% more defects than PBR.

Table 5 shows the result regarding the hypothesis H_2 (team efficiency). Efficiency is defined as the number of defects found per hour, where "hour" is the sum of hours used by the team members.

	Temp	GUI	Light
Effort CBR (in h)	14,00	18,50	12,50
Effort PBR (in h)	23,00	18,00	13,00
Efficiency CBR	1,93	1,14	2,00
Efficiency PBR	1,52	1,94	1,31

Table 5: Efficiency of the reading techniques

Regarding the efficiency the results show that focused CBR is more efficient in two sub-systems (Temp, Light) while PBR is more efficient in the sub-system GUI.

In the following, Box-Plots are presented that visualize the comparison of the number of defects found with the different reading techniques in the three sub-systems

In Figure 3 the Box-Plots for the sub-system Temp is shown. The graphic indicates that the mean value of defects found with PBR is higher than with focused CBR. Also the extreme values of the number of found defects is higher in the case of PBR. However, the variance of the number of defects is higher in the case that PBR is used.

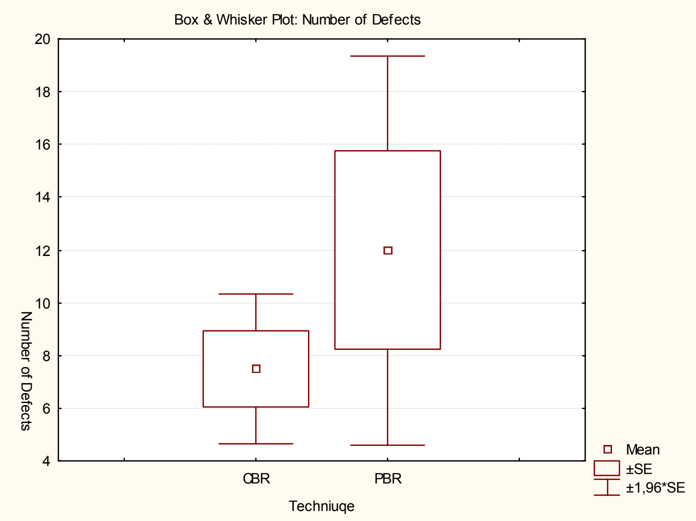


Figure 2: Box-Plots of the Number of Defects found in the Temp-System

In Figure 3 the results for the sub-system GUI are presented. It can be seen that the mean value of defects found with PBR is higher than the mean value of focused CBR.. Thus, the subjects found more defects with the PBR technique. Again the variance is higher in the case that PBR is used. However, in this case the difference between the variances of the two techniques isn't that big compared to the Temp system.

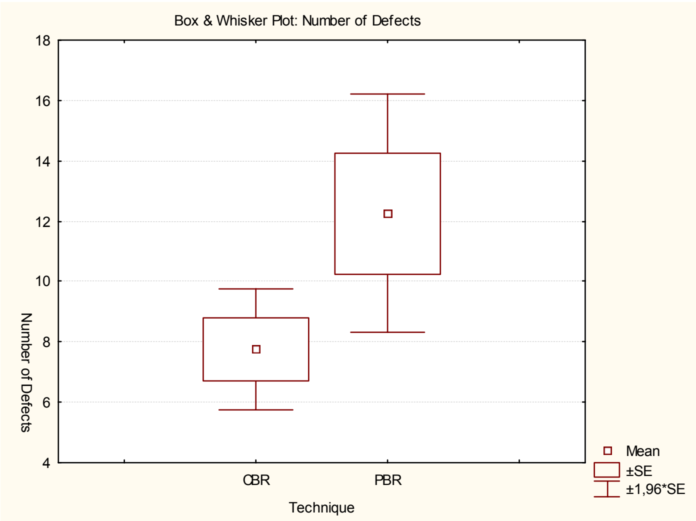


Figure 3: Box-Plots of the Number of Defects found in the GUI-System

Finally, in Figure 4 the results regarding the defects found in the Light subsystem are described. Here, the focused CBR technique found more defects than the PBR approach. The mean value of the focused CBR technique is higher than the mean value of the PBR approach. Also the variance is now higher for the focused CBR technique but the differences regarding the variance between the two techniques is small.

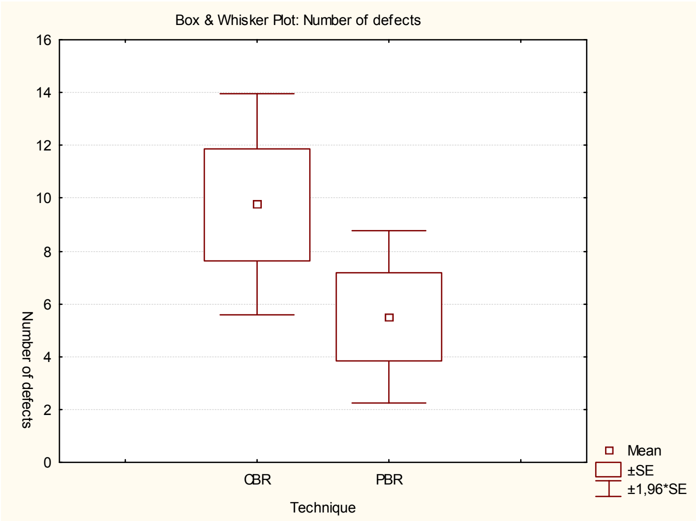


Figure 4: Box-Plots of the Number of Defects found in the Light-System

In order to support the descriptive statistics we performed a Mann-Withney-U test to evaluate the differences in the effectiveness of the different reading techniques. As it is questionable in how far the results of the statistical test are significant (due to the fact that we had only few subjects) we decided to perform this statistical test only for the effectiveness.

The Mann-Withney-U test is a nonparametric test to compare two independent groups; that is to compare one group that gets a certain treatment (in our case the group using PBR and thus getting active guidance) and a control group that does not get the treatment (in our case the group using focused CBR).

The analysis of the test showed that:

- For the GUI subsystem, there is a significant difference between the effectiveness of CBR and PBR at the 0.1 level (p=0.08).
- For the subsystems Temp there is no significant results (p=0,39).
- For the subsystems Light there is no significant results (p=0.15)

That is, for the GUI subsystem, we can conclude that there is a significant difference between PBR and CBR within our study. For the subsystems Temp and Light, we cannot draw such a conclusion. However, the results indicate a weak support for our hypothesis that PBR is more effective.

4.2 Results of the Qualitative Survey

As mentioned in Section 3.4 the subjects had to participate in a survey after they completed the inspection of the use cases. In this section, the results of th survey are briefly summarized. A detailed analysis of the results can be found in Appendix A of this report.

In order to evaluate the usefulness of the reading scenarios and thus the active guidance, the subjects had to answer a debriefing questionnaire. This questionnaire was filled in after the subjects completed the inspection of the different sub-systems. This questionnaire aims at the concrete perception of the subjects regarding the usefulness of the reading scenarios provided by the SBR approach and to compare the usefulness and applicability of focused CBR and SBR. In order to evaluated the overall usefulness of the guidelines each of the questionnaires was designed following the model recommended by Davis [Dav89] that evaluates the general usefulness of a certain technique by means of three basic elements:

1. **Perceived usefulness** "the degree to which a person believes that using a particular technique would enhance his or her job performance"

2. **Perceived ease of use (applicability)** "the degree to which a person believes that using a particular technique would be free of effort"

3. **Self-predicted future use** "the degree to which a person would use a particular technique again in the future"

The original definitions of these elements are focused on the use of tools. However, according to [LD98] it is also possible to use the model to evaluate the usefulness of software engineering techniques when the statements are adapted accordingly.

The basic idea of the model is that beside other factors, people are one of the most important factors when talking about the use of tools or certain software engineering techniques such as the reading techniques. The people tend to use a certain technique when they perceive it as helpful to perform their task. But according to [LD98] this factor might be out-weight if the people perceive the technique as too difficult to use; that is, they think that the application of the technique costs to much effort. Thus, it is important to consider the perceived ease of use (*applicability*) of the technique in combination with its *usefulness*. The model is complete by the subject's *self-predicted future use*. This gives hints whether or not the subjects will use a certain technique in a similar situation in the future.

As there are no objective measures for the usefulness and the applicability of a technique subjective measures have to be used. In the questionnaire performed after the subjects used the reading techniques, the subject's are asked to respond to certain statements in terms of their degree of agreement or disagreement to the statement. In this evaluation the subjects had to select one of six statements (extremely likely, likely, rather likely, rather unlikely, unlikely, extremely unlikely) expressing their agreement or disagreement to a certain statement. This scale is a Lickert-scale, which is perceived as similar to an interval scale. Thus, it is possible to evaluate a meaningful average value [LD98]. Note that the questionnaire was designed in a way that allows the comparison of focused CBR and SBR. The main results of the survey are summarized in the following paragraphs. A detailed description can be found in Appendix A.

Understandability of the Reading Technique

The first aspect we analyzed is the understandability of the different reading techniques. To evaluate the subject's perception regarding the understandability the subjects had to:

1. Evaluate the degree of detail of the checklist questions and the reading scenarios

2. evaluate the understandability of each technique

3. directly compare the understandability of focused CBR and PBR

The results regarding the understandability show that:

- most of the subjects perceive both the checklist and the reading scenarios on an appropriate level of detail

- except one subject all of the subjects perceive the focused CBR approach as rather comprehensible or comprehensible. 7 out of 12 subjects (58%) perceive the PBR approach as rather comprehensible, comprehensible or plain.

- there is a tendency that focused CBR is perceived more comprehensible than PBR. 7 out of 12 subjects (%(%) state that they perceive focused CBR better to understand compared to PBR

Applicability of the Reading Techniques:

In order to evaluate the applicability of the reading techniques, the subjects had to state their agreement to the following statements for each reading technique:

1. It was easy to remember the checklist questions / the scenario steps

2. It was easy to apply the checklist / the reading scenario

The results indicate that:

- in general, the subjects perceived no difference regarding the applicability of the different reading techniques. Both techniques are perceived as rather easy to apply. However, there is a slight tendency in favor of focused CBR.

- in both cases the main problem was to remember the details of the technique while using it; that is the checklist questions (CBR) and the scenario steps (PBR) were hard to remember.

Usefulness of the Reading Techniques

The third element of the evaluation of the different reading techniques is the perceived usefulness of the PBR and the focused CBR approach. Regarding this question the subjects had to state their degree of agreement to the following statements:

1. With PBR more defects can be detected than with focused CBR

2. With PBR defects are detected faster than with focused CBR

3. PBR increases the productivity compared to focused CBR

4. Inspections get easier when performed with PBR

5. The detailed description of the PBR scenarios are useful

The results of the survey are that:

- the number of defects that can be found with the techniques is perceived higher by 8 out of 12 subjects (75%) in favor of PBR

- focused CBR is perceived as slightly faster than PBR (7 out of 12; i.e. 58%) .:

- there is a tendency that the productivity (good quality in reasonable amount of time) is better with PBR. (7 out of 12 subjects, i.e. 58%) Interesting is that one subject completely agrees and another subject completely disagrees to that statement.

- the both techniques can facilitate the inspection process. 6 subjects voted in favor of PBR and 6 subjects in favor of focused CBR

- the reading scenarios, that is active guidance of the PBR approach is perceived as very useful 9 out of 11 subjects are positive or rather positive (81.8%). 3 subjects rather disagree that the reading scenarios are helpful.

In addition to the statements the subjects had to specify the most advantageous and disadvantageous aspects of the PBR approach. Unfortunately only a few subject's answered the question and gave only input of minor value. The most frequently mentioned advantage of the PBR approach is that the separation of concerns (different views) provides a clearly defined focus on certain aspects during the defect detection step. However, in the empirical study this was also true for the checklists. Another benefit of the PBR approach is that the subject's perceive the scenario steps as valuable with respect to getting acquainted with the document.

5 Interpretation of the Results

Our results show a tendency that the reading scenarios and thus the active guidance provided by the PBR approach improve the effectiveness of the inspectors during defect detection. Our analysis showed that there is a significant difference between the effectiveness of PBR and CBR in the GUI system. In that case PBR is significantly more effective than CBR. There is no significant difference between PBR and CBR for the Light and the Temp system. However, in the Temp system PBR found more defects than CBR while for the Light system CBR had a better effectiveness.

There are many different explanations for these results. One is that the difference is caused by the groups; that is, that neither technique nor the subsystem had a significant influence, but only the differences in the performance of the groups. However, there are hints that the differences can be explained by the complexity of the artifacts of the subsystems.

The GUI is the most complex system as it comprises 34 use cases and related scenarios. The Temp sub-system is less complex (21 use cases and scenarios) and the Light sub-system is least complex (15 use cases and scenarios) Although we did not expect CBR to outperform PBR, this finding is in line with our assumption that active guidance is most helpful in complex systems. In particular, this finding may hint that the bigger overhead required by applying PBR (e.g, specifying test cases, writing down a statechart model) only pays off for significantly large and complex documents. In complex documents it is much more important to get a better understanding of the document which is essential for detecting defects. But this understanding is gained when the inspectors are actively working with the document instead of just 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.

One drawback of the PBR approach is its lower efficiency compared to CBR. Again, the additional overhead of the PBR approach explains this result. In the more complex system GUI the overhead paid of as in that while in the less complex systems (Temp and Light) the guidance provided by the focused CBR approach was sufficient.

Another result that is interesting are the different variances in the defects found with the two reading techniques. It is interesting that in the Temp sub-system the variance of the PBR technique is very huge. Also, in each sub-system, the technique that had a better effectiveness (found more defects) also had the bigger variance in the number of defects found.

It is difficult to analyze this finding with such a small number of subjects. However, the effect in the Temp sub-system may result from the different experience of the subjects regarding software inspections. At a closer look this is not the case as in the Temp sub-system subjects with more experience used the PBR approach. Another reason might be the document; that is, that the document complexity of the Temp sub-system might have an influence of the variance. However, this could not be further analyzed as the document complexity was not in the focus of our study. In general, the reason for the different variances needs to be further analyzed.

From the qualitative analysis (i.e., interviews and questionnaires), we also found a tendency that active guidance in the inspection is helpful for the inspectors. Summarizing the results of these statements there is a tendency in favor of PBR (and thus active guidance); that is, PBR is perceived as slightly more useful than focused CBR. However, the results also indicate that PBR has some drawbacks. For example, focused CBR is perceived as faster than PBR which supports our claim that the overhead of PBR is time consuming. In general the survey supported the tendency that PBR has some advantages which is supported by the question regarding the self-predicted future use of the techniques. The subjects answered this question and 6 out of 9 subjects stated that they would rather use PBR than focused CBR in a future inspection.

6 Discussion and Future Work

In this paper, we have described an empirical study that examines the influence of active guidance on the outcome of an inspection. 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 was the active guidance given by PBR.

The results of this preliminary study show tendencies 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 indicate that active guidance is perceived as very useful to perform defect detection and can help to further improve the inspection process.

The results show a certain tendency but are not conclusive. Another drawback of this study is that it represents only a small study, using a small number of subjects in an academic environment. Therefore, results can only be preliminary. Further investigations have to follow. We encourage researchers to replicate our study in different settings and to perform controlled experiments to analyze the research question in more detail.

We were able to identify some important and interesting research questions that need to be addressed in the future. One of the most important is that we need to investigate the influence of document size and complexity on the effectiveness and efficiency advantage of PBR; that is, we need to find out for which documents the "overhead" of PBR pays off, and when we should rather use CBR. More generally, in further research the question needs to be addressed in which context which reading technique is best suited to optimize the inspection process.

Appendix A: Detailed Results of the Debriefing Questionnaire

In the following sections the detailed results of the pre-questionnaire and the debriefing questionnaire are presented. The pre-questionnaire also clarifies the context settings of the empirical study.

Experience of the subjects of the survey

In the following the experience of the subjects of the survey regarding software engineering, inspections in general, inspections of requirements (use case) documents and their knowledge of the inspection process in general are summarized.

Software Engineering knowledge

12 students participated in the empirical study. The results of the evaluation of another case study that was performed with the same subjects shows that all of the subjects participating in the case study are students, enrolled at the university of Kaiserslautern in computer science, applied computer science, or economic engineering with computer science as their major subject. All of the subjects have passed their intermediate exams and are studying for at least 6 semesters.

As the students had to perform inspections of requirements documents of an embedded system (in particular an building automation system) it is important to analyze their experiences within this domain. The evaluations showed that the subjects are rather inexperienced in the domain of embedded systems and none of the subjects is familiar with the domain of automation systems. Furthermore, most of subjects are inexperienced in the fields of requirements engineering and have even less experiences with use cases.

Inspection Knowledge

In order to analyze the subjects experiences with inspections they were asked to estimate their experience regarding inspections in general and requirements inspections in particular. The results are shown in Figure 5 and Figure 6.

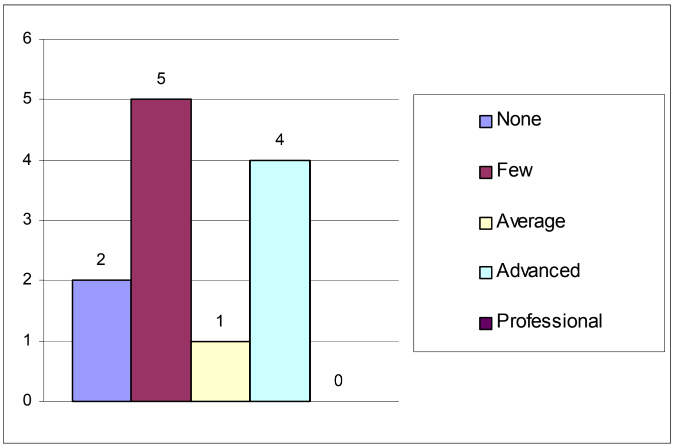


Figure 5: Experience in performing inspections

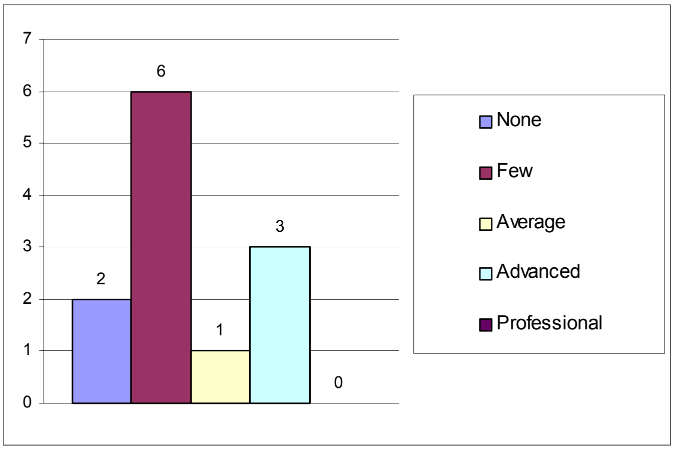


Figure 6: Experience in performing inspections of requirements specifications

The figures show that the subjects have different experiences with inspections. Most of the subjects have none or few experiences with inspections in general (7 out of 12) as well as with requirements inspections in particular (8 out of 12). The other subjects state that they have average (1 answer) or advanced experiences of inspections in general (3 answers) and requirements inspections (4 answers).

The subjects that stated to have advanced knowledge participated already in different kinds of inspections such as code inspections, design inspections or requirements inspections. None of the subjects has inspected use cases so far.

In addition the subjects had to estimate how familiar they are with the inspection process in particular from their lectures at the university.

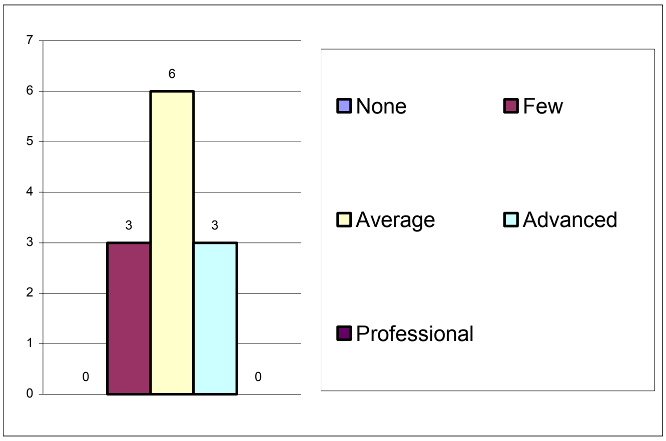


Figure 7: Knowledge of inspection process from lectures

Half of the subjects have an average knowledge of the inspection process from their lectures. 3 subjects have fewer knowledge and 3 subjects advanced knowledge of the inspection process. Thus it can be assumed that the subjects have rather similar knowledge of the inspection process but have different experiences in the practical application of the process.

In addition the subjects had to state how far they were familiar with the other sub-systems of the case study. Most of the subjects state to have an average or little knowledge of the other systems; that is, the system they had to inspect. The results for the different sub-systems under inspection are shown in Figure 8 - Figure 10. A detailed analysis of the answers to this question shows that in case that a subject stated to be “well” or “very well” familiar with a sub-system then it was his or her own sub-system.

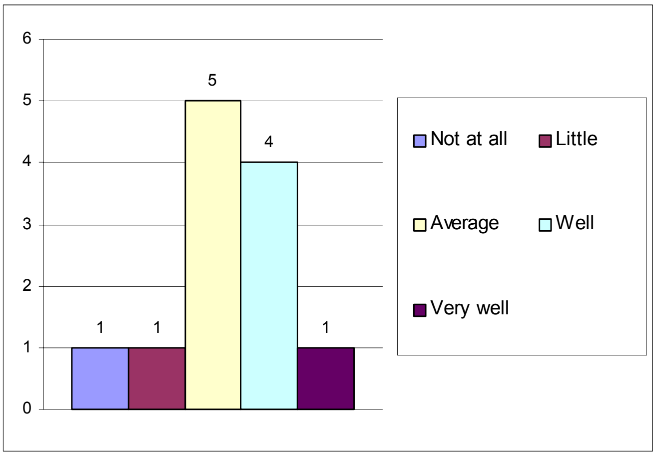


Figure 8: Knowledge of the subsytem "temperature-control"

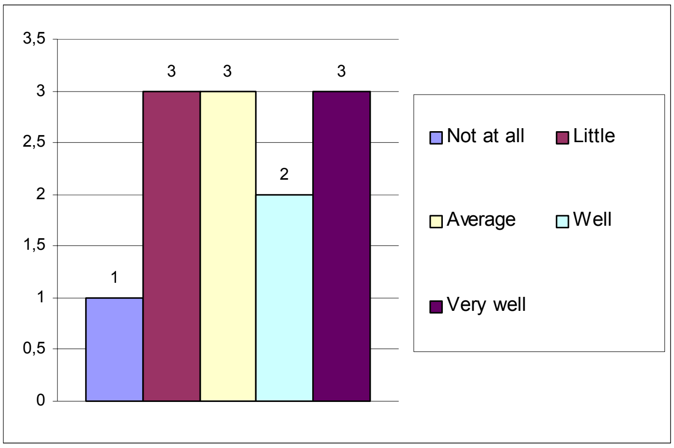


Figure 9: Knowledge of the subsytem "light-control"

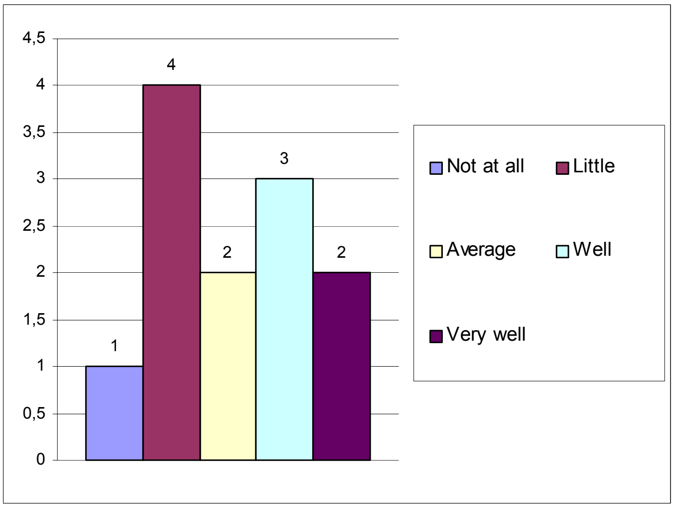


Figure 10: Knowledge of the subsytem "user interface"

In the following section the results of the students perception of the perspective based reading technique in comparison to the checklist based reading technique are summarized.

Evaluation of the subject’s attitude towards perspective based reading

In order to support the evaluation of the impact of the active guidance of perspective based reading scenarios on the inspection result the subjects had to answer several questions in a debriefing questionnaire. These questions focus on the subject’s perception regarding the reading scenarios; that is, whether they perceive the additional scenarios as useful to perform the inspection and easy to apply during the inspection.

The questionnaire consisted of several part each of them will be described in the following paragraphs.

General Questions about the Reading Techniques

First of all the subjects estimated how good they understood the task they had to perform in the practical course; that is how good they understood what to do during the requirements inspection process. All of the subjects stated to understand the task well or very well (see Figure 11).

Furthermore, the subjects had to estimate how many defects they found during the inspection of the use cases of the other sub-systems. As the inspections were performed in two runs, the subjects had to estimate the number of defects for each of these runs separately. Note that each subject performed the inspection with two different reading techniques in the two runs. 8 students used focused CBR and 4 students used PBR in the first run and vice versa. In Figure 12 and Figure 13 the estimations for the different runs are summarized.

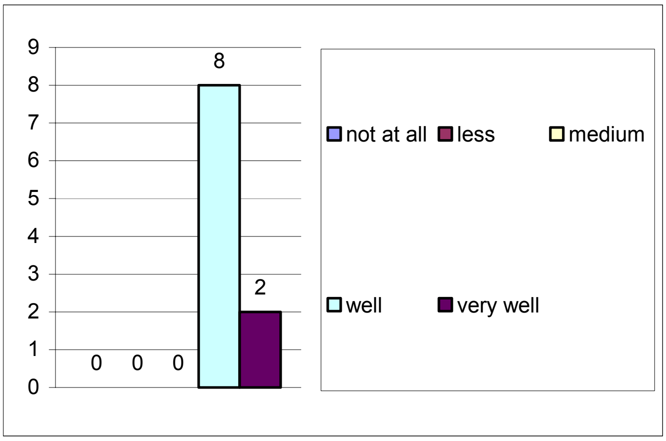


Figure 11: Estimation how good the task was understood

In the first inspection the perception of the students

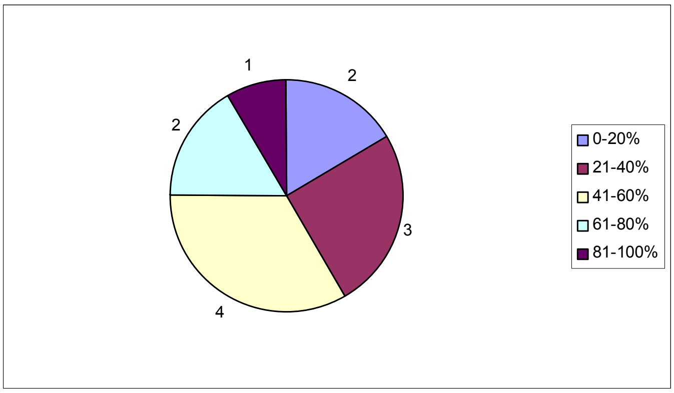


Figure 12: Estimation, how many defects were found in the use-cases and the scenarios (1. inspection)

The perception of the subjects regarding their own effectiveness during the inspection heavily varies. All possible answers were stated at least once. A detailed analysis of the answers of the subjects did not show any relationship between the perceived effectiveness and the used reading technique.

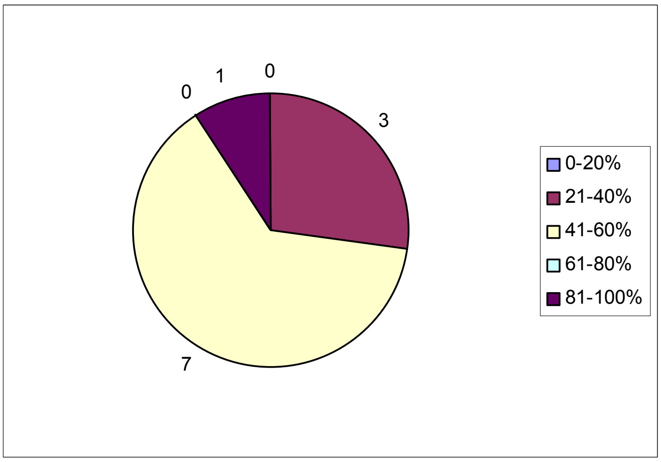


Figure 13: Estimation, how many defects were found in the use-cases and the scenarios (2. inspection)

In the second inspection the subjects estimated their effectiveness on a more average level. Almost all subjects estimated the defect detection effectiveness between 21% and 60%. Compared to the first estimation it is not possible to see a certain trend except that the extreme values (0 – 20%) were not chosen in the second estimation. Again, it is not possible to see a relationship between the reading technique and the estimated effectiveness when analyzing the detailed answers.

Most of the subjects could not see a shift in their effectiveness between the two runs. 8 out of 12 subjects estimated the effectiveness of the second run equal to the first run. 1 subject estimated to find more defects in the second run where he or she used CBR instead of PBR and three subjects estimated to find more defects with PBR than with CBR. Two of them used CBR in the first run and PBR in the second run. Thus, their perception might be biased by a learning effect.

In addition to the perceived effectiveness the subjects compared the time that is needed to perform an inspection with CBR and with PBR. Thus, they had to estimate with which technique they spend more time for the inspection. The results are shown in Figure 14. Again the result indicate that the subject's perception of the reading techniques heavily varies. While 2 subjects perceive both techniques to be as time consuming as the other one half of the remaining subjects perceives CBR as more time consuming and half perceives PBR as more time consuming.

In the context of the given experimental design there are several reasons for this distribution. The checklists and the scenarios of the different perspectives were of different complexity. As the experimental design aims at the evaluation of the impact of the active guidance of PBR approach the subjects had to use different perspectives (for checklist and scenario). As they are of different complexity it is not possible to compare the estimated time.

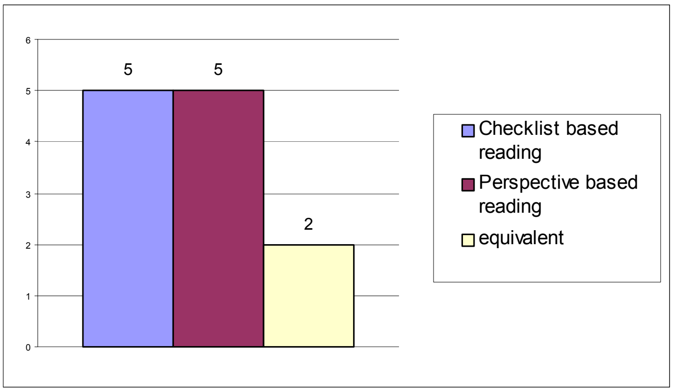


Figure 14: Estimation, which reading technique is more time consuming

Another interesting aspect regarding the time is that in the case that the checklist was perceived as more time consuming the subjects estimated the additional effort to be around 38% on average. In case that PBR is perceived as to be more time consuming, the subjects estimated an additional effort of 58.6%.

Taking these numbers into account the evaluation might indicate that PBR approach is more time consuming then the CBR approach. Again, these numbers must be used carefully as the checklists and the reading scenarios used by each subject in the different runs cannot be compared directly.

In order to evaluate the influence of the different documents under inspection on the inspection result, the subject's stated their impression of the document size with respect to the time they had to perform the inspection. All of the students agreed that the inspected documents had an appropriate size. This indicates that the effect of the document size on the inspection results is on a minimal level. Of course there is still the effect that the different sub-systems are of different complexity, which might influence the inspection result.

Application of the reading techniques

In order to understand how the subjects used the different reading techniques they had to sketch how they performed the defect detection step with the help of the reading technique.

Regarding CBR the subjects followed different approaches to perform the defect detection step:

- Reading the document and using the checklist questions to detect defects during reading
- Reading the requirements and the checklist. Then, for each question of the checklist read the requirements document again (sequential approach)
- Reading the checklist and then reading the requirements document to identify defects.

Most of the subjects used the sequential approach.

Regarding PBR it seems that almost all of the subjects followed the steps of the reading scenarios. In this context the reading scenario of the tester-perspective and the designer perspective seemed to be most valuable as the subject's answers indicate that they exactly followed the instructions of these scenarios. Only two subjects stated that they did not follow the detailed instructions. One subject tried to follow the instructions but stopped and performed an ad-hoc reading approach and one subject did not use the instructions right from the start. However, the evaluation indicates that the subjects performed the perspective based inspection thoroughly. This result is supported by the evaluation of the question how the subjects think they followed the instructions of the scenarios.

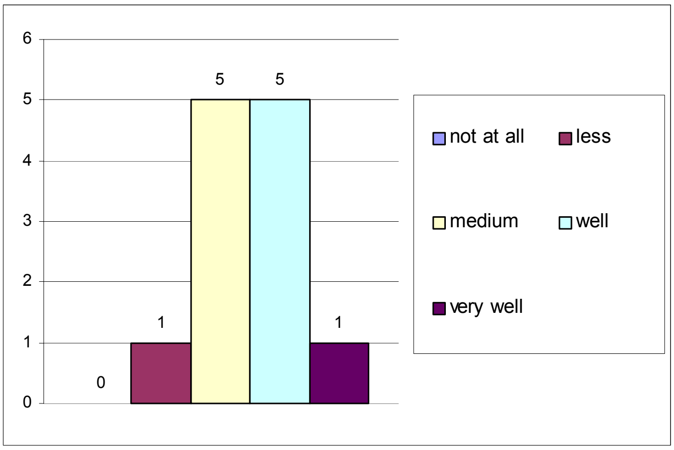


Figure 15: Estimation, how well the subjects followed the instructions of the PBR scenarios

Figure 15 indicates that half of the subjects followed the instructions well or very well. 5 subjects stated that they followed the scenarios to an average degree. Thus, 11 out of 12 subjects followed the scenarios rather thoroughly.

One crucial aspect when talking about different reading techniques is whether the users of the techniques perceive them as applicable, useful and easy to understand. Thus, the subjects in the empirical study where asked about their perception of CBR and PBR regarding these aspects of a reading technique. In the following paragraphs the results are presented. Note that there are no objective measures available for usefulness, understandability and applicability. Thus, in most of the cases we used subjective measures where the subjects had to state their agreement or disagreement to certain statements. Then we compare the answers of the subjects regarding the different reading techniques.

Perceived understandability of the reading techniques

The first aspect we analyzed is the understandability of the different reading techniques. To evaluate the subject's perception regarding the understandability the subjects had to:

- Evaluate the degree of detail of the checklist questions and the reading scenarios
- evaluate the understandability of each technique
- directly compare the understandability of CBR and PBR

Based on the answers to these questions the understandability of the two techniques is comparable. In Figure 16 and Figure 17 the results regarding the perceived degree of detail are shown.

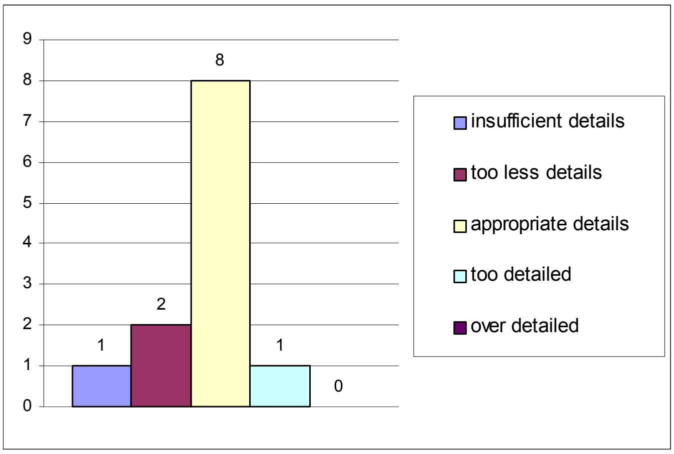


Figure 16: Perceived degree of detail of the checklist questions

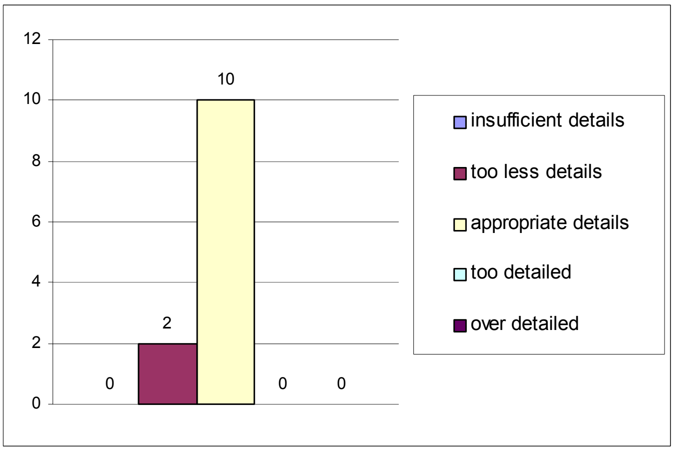


Figure 17: Perceived degree of detail of the perspective-based scenarios

Most of the subjects perceive both the checklist and the scenarios on an appropriate level of detail. Only three subjects perceive the checklist as not sufficiently detailed and 2 subjects perceive the reading scenarios to be not detailed enough. These results indicate that the subjects had only minor problems with the different techniques regarding the degree of detail.

More differences can be seen in the perceived understandability as shown in Figure 18 and Figure 19.

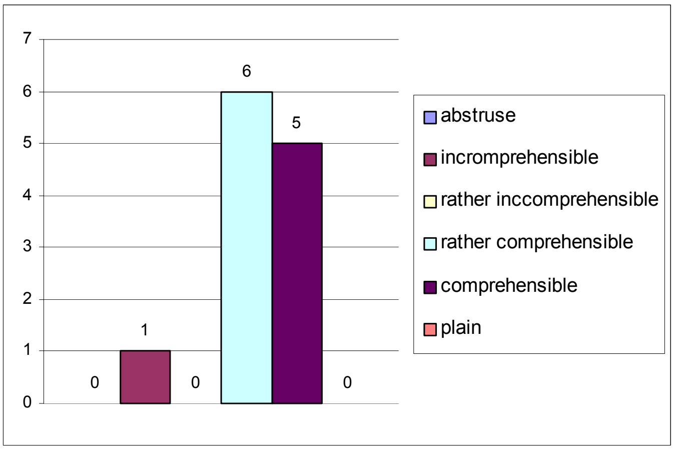


Figure 18: Rating of the understandability of checklist-based reading

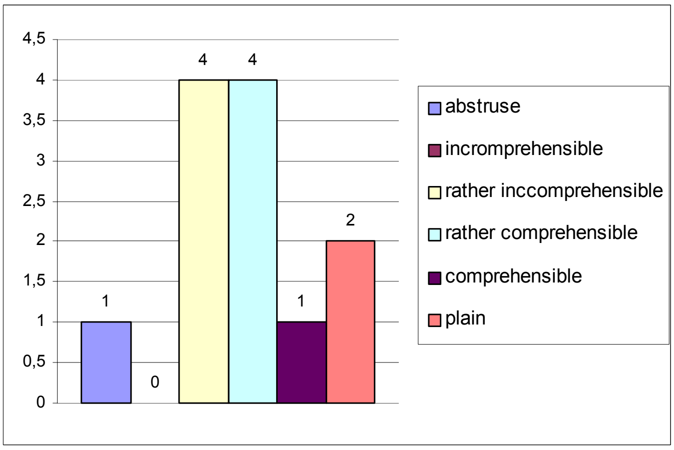


Figure 19: Rating of the understandability of perspective-based reading

The results show that except one subject all of the subjects perceive the CBR approach as rather comprehensible or comprehensible. Compared to PBR approach it gets obvious that more subjects choose the extreme values when evaluating the understandability of PBR. Compared to CBR only 7 out of 12 subjects perceive the PBR approach as rather comprehensible, comprehensible or plain. 5 of the subjects perceive PBR as rather incomprehensible or abstruse. These results indicate that CBR seems to be perceived as easier to understand than PBR.

This impression is supported by the results presented in Figure 20. The subjects should directly compare the understandability of CBR to PBR; that is, they had to state whether CBR is comprehensible or incomprehensible when compared to PBR. The evaluation indicates a tendency that CBR is perceived more comprehensible than PBR. 7 out of 12 subjects state that they perceive CBR better to understand while 5 out of 12 perceive PBR as better to understand. These results are not significantly supporting that CBR is easier to understand but in combination with the results presented in Figure 18 and Figure 19 there is in indication in that direction.

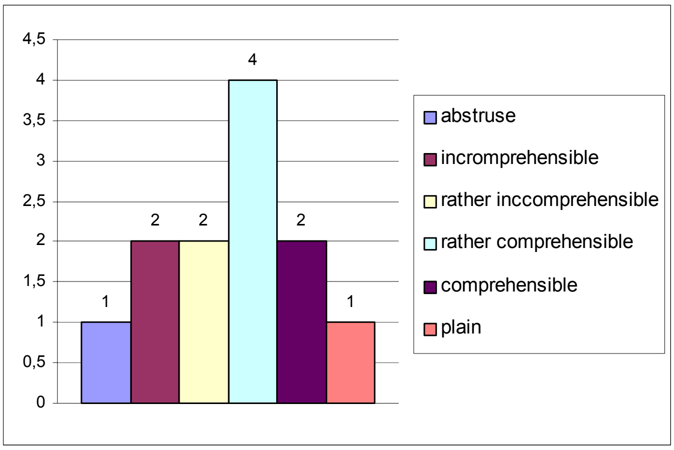


Figure 20: Evaluation of the understandability of CBR in comparison to PBR

Beside the answers to the perceived understandability the subjects had to state improvement suggestions for the different techniques. To this question the subjects made only minor remarks ranging from "the checklist questions are good as they are" and "no improvement suggestions, the designer scenario for PBR is very good" to "questions should be defined in more detail", "a consistent language should be used", and "the checklist and the scenarios should be adapted more to the project". Thus, it is not possible to analyze the reasons for the different perception of the different reading techniques, as the answers do not provide valuable input for such an evaluation.

Applicability of the reading techniques

The second aspect that should be compared is the applicability of the different reading techniques. Beside the estimation how many checklist questions and scenario steps were used during defect detection the subjects had to state their agreement to the following questions for each reading technique:

- It was easy to remember the checklist questions / the scenario steps
- It was easy to apply the checklist / the reading scenario

The agreement or disagreement towards these statements is correlated to the perceived applicability of the reading techniques. In Figure 21 and Figure 22 the percentage of the used checklist questions and scenario steps are presented.

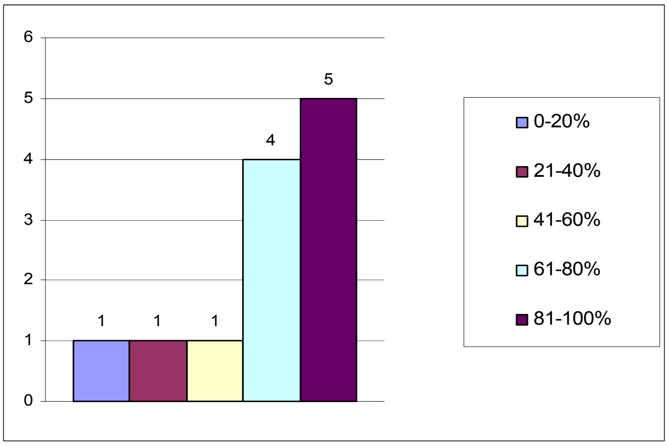


Figure 21: Estimated number of used checklist questions (in percent)

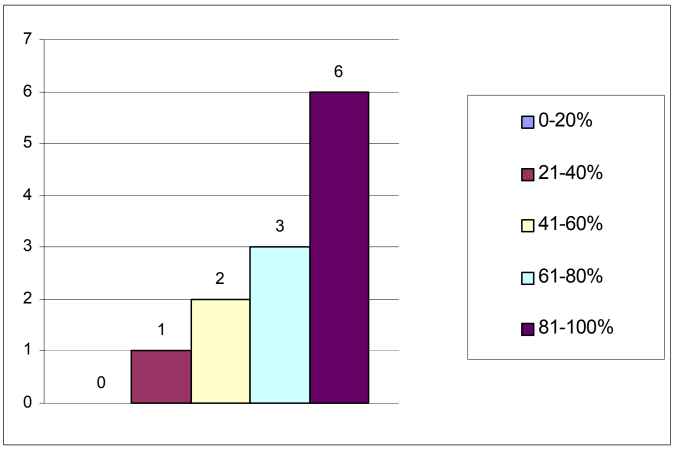


Figure 22: Estimated number of used scenario steps (in percent)

The evaluation indicate that the students followed the checklist and the perspective based reading scenarios quite thoroughly. For both techniques 9 out of 12 subjects stated that they used 61% to 100% of the checklist questions and the scenario steps. This supports the earlier results how the students used the reading techniques to support the defect detection in the inspection.

In Figure 23 and Figure 24 the degree of agreement to the above mentioned statements are presented for the CBR approach. In the case that the checklist questions are easy to remember it can be concluded that the checklist can be easily applied.

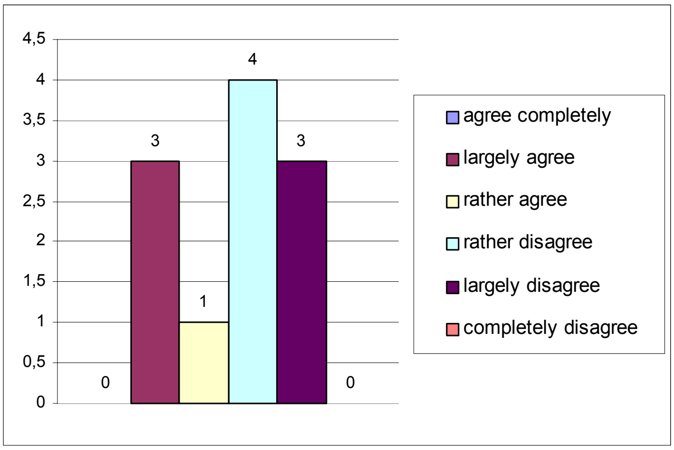


Figure 23: Degree of agreement to the statement "It was easy to remember the checklist questions"

As a result of the evaluation the subject's perceive the different checklist questions as hard to remember. 7 subjects would rather disagree or disagree that the different checklist questions are easy to remember. Only 3 subjects agreed that the questions are easy to remember.

These results are in contradiction to the degree of agreement whether the checklist was easy to apply. In Figure 24 it is shown that 10 out of 12 subject's agree to the statement that the checklist is easy to apply during the defect detection step. Only two subjects do not agree to the statement.

One reason for these results might be the fact that most of the subject's used the checklist in an iterative manner; that is, they read one checklist question and then analyze the document under inspection with respect to this question, then they repeat this process for each checklist question.

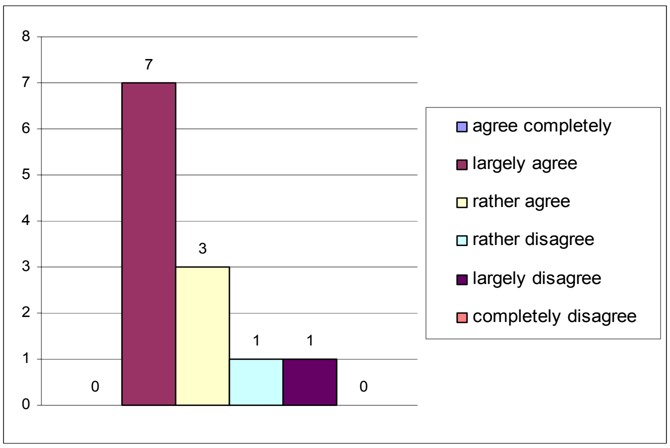


Figure 24: Degree of agreement to the statement "It was easy to apply the checklist"

In Figure 25 and Figure 26 the degree of the subject's agreement to the same statements are summarized.

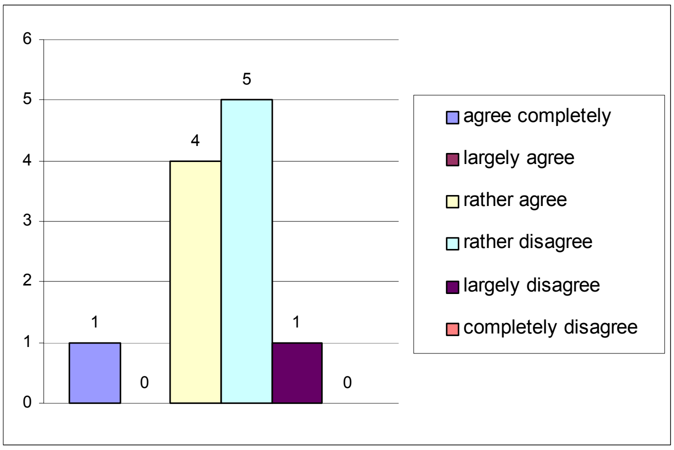


Figure 25: Degree of agreement to the statement "It was easy to remember the steps of the perspective-based scenarios"

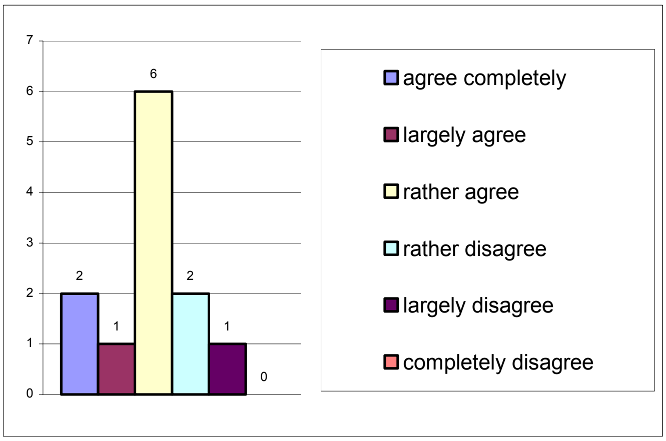


Figure 26: Degree of agreement to the statement "It was easy to apply the perspective-based scenarios"

The results are similar to the results for the CBR approach. Again the subject's state that it is hard to remember the different scenario steps (6 out of 11 answers) but would rather agree that the PBR scenarios are easy to apply during the defect detection.

Based on these results there is no difference between the two reading techniques regarding the perceived applicability. This is a surprising result as our hypothesis was that the CBR approach is perceived easier to apply for inexperienced inspectors. The reason for this finding might be that the subjects had problems with the application of the checklist due to the focus of the checklist to a certain perspective.

The usefulness of the reading techniques

The third element of the evaluation of the different reading techniques is the perceived usefulness of the PBR and the CBR approach. Regarding this question the subjects had to state their degree of agreement to the following statements:

- With PBR more defects can be detected than with CBR
- With PBR defects are detected faster than with CBR
- PBR increases the productivity compared to CBR
- Inspections get easier when performed with PBR
- The detailed description of the PBR scenarios are useful

Again, all these statements are related to the overall aspect of the usefulness of PBR compared to CBR. In the following figures the subject's responses to the statements are presented.

Figure 27: Degree of agreement to the statement "With PBR scenarios more defects can be detected than with CBR"

Figure 27 indicates that 8 out of 12 subject's think that more defects can be detected with PBR compared to the CBR approach. However, this is not a significant result but shows a certain tendency in favor of the PBR approach.

Regarding the time that is needed to perform an inspection with PBR the subject's disagree that PBR accelerates the inspection (defect detection) step. As shown in Figure 28, only 5 out of 12 subjects agree that PBR is faster than CBR and 7 subjects perceive CBR to be faster. Also this result is not significant as around half of the subjects voted for CBR and the other half for PBR. However, the result is in line with the results presented in Figure 14 as there also half of the students perceived CBR more time consuming and the other half PBR. Thus, there seems to be no significant difference between the two techniques regarding the time needed to perform the inspection.

This result is surprising, as in earlier experiments PBR was in almost all cases more time consuming. This is reasonable as during the defect detection with PBR the inspectors have to perform additional activities to work with the product. One reason for our findings might be that in our study 3 detailed checklists. In earlier experiments the PBR scenarios were always compared to one single high level checklist.

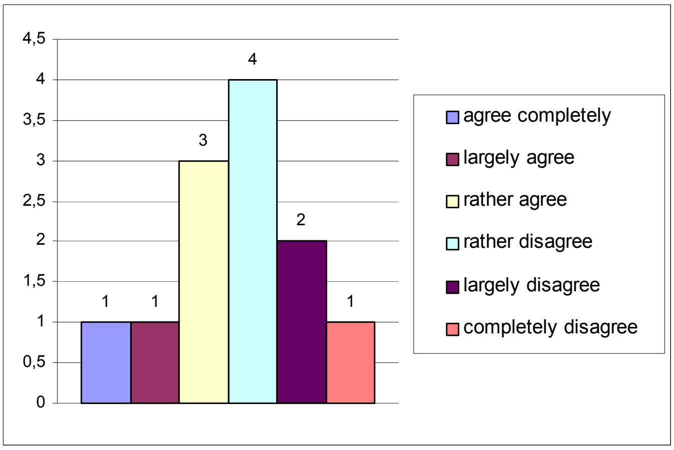


Figure 28: Degree of agreement to the statement “With PBR scenarios defects are detected faster than with CBR”

Figure 29: Degree of agreement to the statement “PBR scenarios improve the productivity of the inspection”

The third statements the subjects had to evaluate is the whether they perceive PBR as a technique that increases their productivity of the inspection; that is, that they produce good inspection results in an adequate amount of time. In

Figure 29 the answers to this statement are collected. Again the answers vary over the complete domain of possible answers. While one subject completely agrees that PBR improves the productivity another subject completely disagrees to this statement. However, slightly more subjects (7 out of 12) perceive the PBR approach as more productive and thus there is a slight tendency in favor of PBR.

The fourth statement to which the subjects had to state their degree of agreement is whether inspections are easier to perform with PBR then with CBR. In Figure 30the responses to this statement are collected. Also the answers to this question are heavily varying. One subject completely agrees that the PBR approach makes the inspection easier another subject completely disagrees. Considering the sum of the subjects, half of them would rather agree or agree that PBR improves the productivity the other half of the subject's votes in favor of the CBR approach. Thus, it is difficult to identify a tendency toward one of the techniques.

Figure 30: Degree of agreement to the statement "It is easier to accomplish inspections with PBR scenarios than with CBR"

Finally, the usefulness of the detailed scenarios during the inspections was evaluated. In Figure 31 the answers are summarized. 9 out of 11 answers indicate that the detailed description of the scenarios is perceived as useful while 3 subjects rather disagree to this statement. This result supports our hypothesis that the detailed scenarios give better guidance to the inspectors during the defect detection as a checklist.

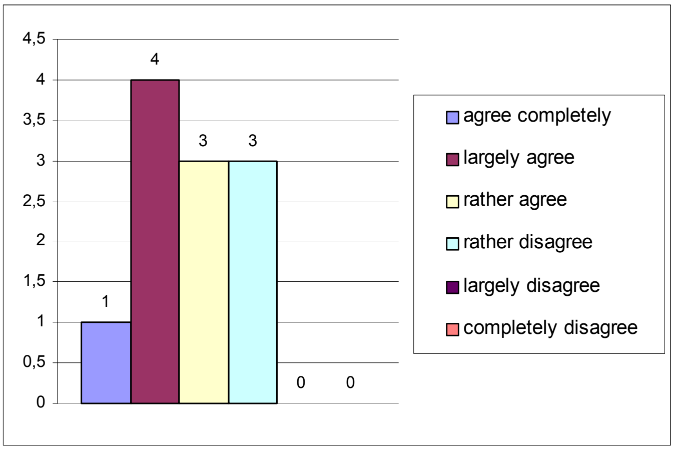


Figure 31: Degree of agreement to the statement “The detailed description in the PBR scenarios is useful”

Assuming that the five statements evaluated in the last paragraphs all contribute in the same way to the perceived usefulness of the PBR approach in comparison to the CBR approach we summarized the answers of the subjects. The result of this summarization is shown in Figure 32

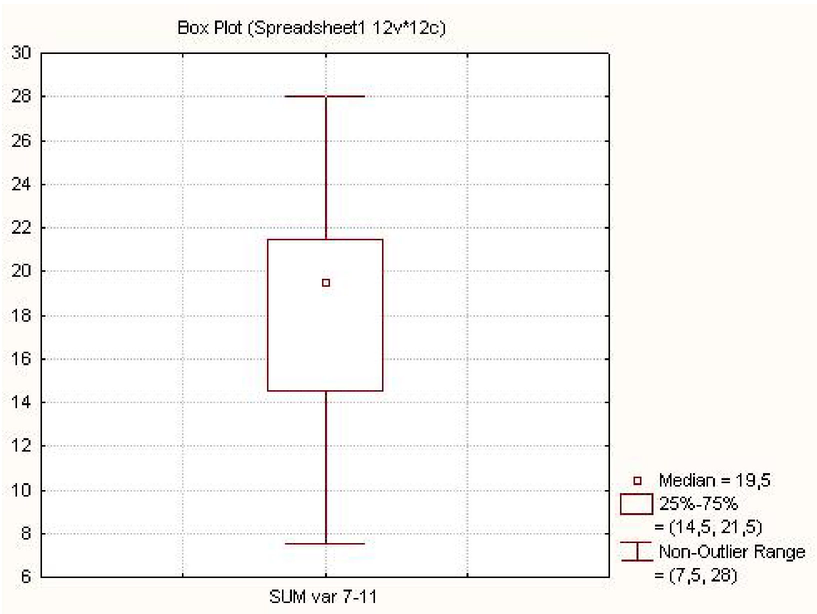


Figure 32: Summarized result for usefulness

The summarized results show that there is a general tendency that PBR is perceived as more useful as CBR. The maximum value is 30 while the median of all answers is 19.5. Thus, there is a slight tendency in favor of PBR; that is PBR is rather useful than CBR.

Self-Predicted future use

In addition to the statements regarding the usefulness and the applicability the subjects had to vote for one technique that they would use in a future inspection. Only 9 subjects answered this question. However, the result indicates that the 6 out of 9 subjects would rather use PBR in a future inspection 3 subjects stated to use CBR.

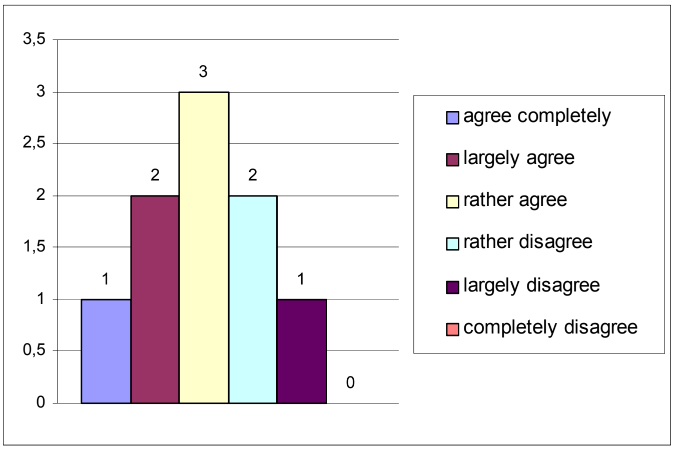


Figure 33: Degree of agreement to the statement "In future inspections I would use PBR rather than CBR"

Again, these results show a slight tendency of the subjects toward the PBR approach.

In order to support the evaluation of the subject's agreement and disagreement to the various statements they were asked to specify the most advantageous and disadvantageous aspects of the PBR approach. Unfortunately only a few subject's responded to this question and gave only input of minor value.

The most frequently mentioned advantage of the PBR approach is that the separation of concerns (different views) provide a clearly defined focus on certain aspects during the defect detection step. However, in the empirical study this was also true for the checklist based approach. An other benefit of the PBR approach is that the subject's perceive the scenario steps as valuable with respect to getting acquainted with the document. The subjects did not specify any important drawbacks of the PBR approach that are not already described in the suggested improvements for PBR and CBR earlier in this chapter.

Appendix B: Reading Scenarios and Checklists

Reading Scenario "Customer Perspective"

Envision to be the customer of the System described in the System Requirements Document (Benutzeranforderungen). As part of your work you have to ensure that all of your requirements are described in the System Requirements Document as intended. For you, the most important quality aspects of the System Requirements Document are the completeness and correctness of the requirements.

No	Step	Validation
1	Identify all concrete people who will use (interact with) the system as described in the problem description and combine them into user roles. Compare this list with the actors in the Use Case Diagram and with the actors who are listed in the single textual Use Cases.	1. All different roles occur as actors in the Use Case Diagram. They also appear with exactly the same description and relation to the single Use Cases, in the textual description of each Use Case and the related scenarios of the Use Cases.
2	Sketch concrete tasks, which you conceive for each single Use Case, relative to the Use Case name in the Use Case Diagram. Compare your tasks with the event flow described in the Use Case and the scenarios of the Use Cases.	1. All the tasks you came up with are covered through Use Cases and the related scenarios of the Use Cases. 2. The Use Case name reflects the perceived intent of the Use Case. 3. Each Use Case in the Use Case diagram is related to a textual description and vice versa. 4. Each Use Case is related to (at least) one related scenarios and vice versa.
3	Based on the problem description, construct a table that includes the names of all Uses Cases in the first row. List the names of all monitored and controlled variables that should be used in the single Use Cases in the first column, independently from the problem description. Add in each cell of the table an "M" for monitored variable, a "C" for controlled variable or a "M/C" for monitored and simultaneously controlled variable. If a variable does not occur in a Use Case, mark out that space.	1. The generated table corresponds to the list of monitored and controlled variables in the single Use Cases. 2. The controlled and monitored variables described in the template elements "monitored "/ "controlled variable" are part of the event flow of the Use Case. They are also consistently used in the related scenarios of the Use Cases.

4	Based on the problem description construct a table with the name of the Use Case in the first row. Add all quality criteria that are relevant for single Use Case in the corresponding column of the table. Compare this list to the section "Quality Criteria" of the Use Cases.	1.All quality requirements of the system that refer to specific Use Cases are listed in the field "Quality Criteria" of the Use Case.

Questions:

1. Which event flow (especially exceptions) can you envision that are not covered by the Use Case and the related scenarios ?

2. Which steps of the event flow of the Use Cases and the related scenarios are unclear and difficult to understand?

3. Which monitored or controlled variables are being used against your conception?

4. Which terms are unclear; that is not understandable?

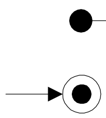
5. To what extent is the intent of the Use Case not reachable by the Use Case and the related scenarios?

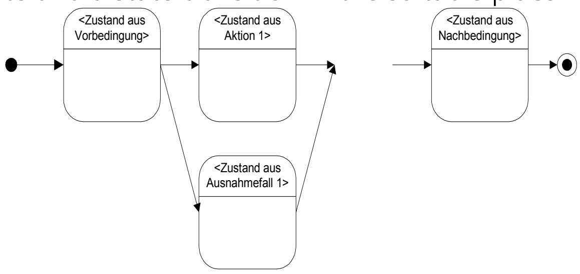
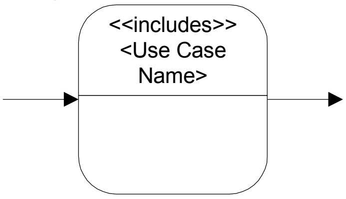
6. To what extend is the desired system not described by the entirety of the Use Cases?

7. To what extend are use interface details that are not requested by in the problem description described in the Use Cases and related scenarios?

Reading Scenario “Designer (System specification document developer)”

Imagine you are the developer of the system specification document (Design). As part of your work you have to gain an overview of the system requirements document (Benutzeranforderungen). It is very important for the success of your activity to be able to derive a state graph from the system requirements document. For you the main quality aspect of the system requirements document is the completeness and the realizability of the requirements.

<table>
<tr><th>No</th><th>Step</th><th>Validation</th></tr>
<tr><td rowspan="4">1</td><td>Build the statechart for the Use Cases. For each Use Case:</td><td rowspan="4">1. All states can be derived from the precondition, event flow, the related scenarios, and post condition of the Use Case
2. All states can be reached and have an outgoing transition.
3. All the states and state transitions of the Use Cases are consistently described in the related scenarios.
4. For each actor action (event) the system reaction is described in the Use Case and the related scenarios
The following statements should be evaluated based on the problem description:
5. All possible states are described in the Use Case and the related scenarios and shown in the statechart.
6. All possible state transitions are described in the Use Case and the related scenarios and shown in the statechart: For each state it is apparent whether and under which circumstances it can be entered, resumed or abandoned.</td></tr>
<tr><td>1.Draw the initial state and the final state with great distance to each other. </td></tr>
<tr><td>2. Find the precondition of the Use Case and draw a state. </td></tr>
<tr><td>3. Find the first actor action (event) of the Use Case within the Use Case description. Starting from the state of the precondition, draw a state transition (arrow) to the state representing the result of the actor action, and from this to the state that represents the result of the next actor action and so forth, until the state of the post condition is reached. Use the event flow and the scenarios of the Use Case to derive the statecharts</td></tr>
</table>

2	For each exception described in the Use Case insert a state and a state transition in the suitable place.  	1. All state transitions and states of the exceptions are described in the Use Case and the related scenarios and shown in the state chart. 2.All imaginable exceptions are described in the Use Case and the related scenarios. 3. The exceptions of the Use Cases are consistently described in the related scenarios.
3	For all includes relationships, draw a state that is labeled with <<includes>> and the name of the Use Case, as shown below.  	1. All <<includes>> relationships are consistently described in the Use Case Diagram and the single Use Cases. 2. The refered to Use Cases are included in the systems requirements document a state chart exists for the included Use Case.

Questions:

1. Which states do not occur in the statechart or the Use Cases (and the related scenarios of the Use Cases), although they could occur?
2. Which event flows are unclear?
3. Which aspects of the Use Cases and the related scenarios cannot be realized (implemented) or only with difficulties?
4. Which events additionally may occur that were not described?
5. Which relevant rules are not considered in the Use Cases?
6. Which rules stated in the Use Cases contradict your statechart?
7. To what extend do the Use Cases or the related scenarios contradict each other?
8. To what extend is the traceability matrix inconsistent to the relationships between the Use Cases and the scenarios?

Reading Scenario "Tester"

Imagine you are the tester of the system requirements document. As part of your work you have to develop a test plan and test cases from the systems requirements document. It is essential for your work that you are able to derive testing processes and test case from separate parts of the system requirement. The main quality aspect of the systems requirements document is the verifiability of the system requirements.

<table>
<tr><th>No</th><th>Step</th><th>Validation</th></tr>
<tr><td>1</td><td>Based on the problem description construct a test plan matrix that represents an initial version of the concrete test plan. Use the following structure to design the matrix:
1. List all the subsystems of the system in the first row.
2. List all functionalities and quality requirements that the system has to comply with from a user point of view in the first column.
3.Note in each cell of the matrix whether the respective combination is to be tested.

Template:
<table><tr><th></th><th><Sub-system1></th><th><Sub-system2></th><th><Sub-system3></th></tr><tr><td><Func-tionality1></td><td></td><td></td><td></td></tr><tr><td><Func-tionality2></td><td></td><td></td><td></td></tr></table></td><td>1. All subsystems of the overall system under test can be derived from the problem description.
2. All essential functions and quality requirements can be derived from the problem description.
3. All functionalities and quality requirements could be assigned to one or more subsystems.</td></tr>
</table>

<table>
<tr>
<td>2</td>
<td>Based on the Use Cases and the related scenarios of the subsystem under inspection, sketch test cases for system and acceptance testing.
1.
For each functionality relevant for the subsystem (see test plan matrix created in step 1) identify the Use Cases that realize this functionality. To do so, create a matrix with the name of the Use Cases in the first row and list the functionalities in the first column. Mark each cell with a cross if the Use Case realizes the functionality.

Template:
<table>
<tr><td></td><td>Use Case 1</td><td>Use Case 2</td></tr>
<tr><td>Functionaliy1</td><td></td><td></td></tr>
</table>

2. Based on the Use Case description and the related scenarios specify for each Use Case one test case for the main flow of events and one test case for each exception.
To do so, specify the Use Case name, the actor, possible actions of the actor, reason for the actor to perform the actions and expected reaction of the system, following the template below.

Template:
<table>
<tr><td>Use Case Name</td><td>Actor</td><td>Action</td><td>Motivation / Cause</td><td>Expected Reaction</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td>
<td>1. For each action of the actors all the expected system reactions are clearly described in the Use Cases. They are also consistently described in the related scenarios.
2. All possible actor actions are considered in the Use Cases. They are also consistently described in the related scenarios.
3. All actors that are interested in a certain system reaction are considered in the Use Case. They are also consistently described in the related scenarios.
4 All functions are realized in at least one Use Case. They are also consistently described in the related scenarios.</td>
</tr>
</table>

Questions:

1. Which functionalities are unclear?

2. In which way are the marks in the test plan matrix unclear to you?

3. Which test case cannot (entirely) be specified?

4. Which invalid actor actions (events) can you think of that are sp far not considered in the Use Cases? Where would you add these exception descriptions?

5. Which functions are not realized in a Use Case and the related scenarios?

Checklist-“Customer”

No.	Question	
Use Case and Use Case Diagram		
1	Does a textual description exist for each Use Case in the Use Case diagram and vice versa?	
2	Does at least on scenario exist for each Use Case and is each scenario related to a Use Case?	
3	Are all the terms used in the Use Case and the related scenarios clearly defined; that is, each single Use Case and each related scenario is clear and understandable?	
4	Does the name of the Use Cases reflect the intent of the Use Case?	
Actors		
5	Are only actors in the Use Case Diagram, the related scenarios and in the textual Use Case description that really use the system?	
6	Based on the problem description, are all the actors that interact with the system considered in the Use Case-diagram and in the textual description of each Use Case?	
7	Are all the actors connected to the right Use Case in the Use Case diagram and mentioned in the textual description of the Use Case?	
8	Are all the actors that are described in the Use Cases consistently used in the related scenarios and vice versa?	
Variables and Quality Aspects		
9	Based on the problem description, are all the necessary monitored and controlled variables considered in the textual Use Case description?	
10	Are all the monitored and controlled variables described in the Use Case elements “monitored variables” and controlled variables“ used in the event flow of the Use Case (including the related scenarios) and vice versa?	
11	Are the quality requirements that refer to a specific Use Cases considered in that Use Case (in the template element “Quality Criteria”)?	
12	Does any of the Use Cases makes prescriptions regarding the user interface that are not re-quired in the problem description?	

Event Flow of the Use Case		
13	Is the intent of the Use Case reachable by the flow of events described in the Use Case as well as by the related scenarios?	
14	Are all the possible event flows, especially exceptions, covered in the Use Cases and consistently described in the scenarios?	
15	Is the flow of events described in the Use Cases and the related scenarios clear and understandable?	
16	Is the desired system (as in the problem description) completely described by the entirety of the different Use Cases and the related scenarios?	

Checklist-"Designer"

No.	Question	
Single Use Cases		
1	Are all possible events (actor actions) and system reactions considered in the textual Use Case description and the related scenarios?	
2	Are all the events and states described in the Use Cases consistently described in the related scenarios?	
3	Are all the events described in the Use Cases and related scenarios completely described?	
4	Is a system reaction specified for each event in the Use Cases and consistently described in the related scenarios?	
5	Are all possible exceptions that might occur in a certain Use Case considered in the textual Use Case description and in the related scenarios?	
6	Is the system reaction in the case of an exception clearly described for all exceptions described in the Use Case?	
7	Are all the exceptions described in the Use Cases consistently described in the related scenarios?	
8	Are all the rules that are important for a Use Case considered in the textual description of the Use Case?	
9	Does each single Use Case as well as each related scenario comply with the rules specified in the Use Case?	
10	Are all the event flows in the Use Cases and the related scenarios clearly described?	
Set of Use Cases		
11	Are all the relationships between the Use Cases described in the Use Case diagram consistently realized in the event flow and the related scenarios and vice versa?	
12	Are all the aspects of the Use Cases and the related scenarios realizable (implementable) in the system design?	
13	Does none of the textual Use Case descriptions contradict an other Use Case?	
Traceability Matrix		
14	Are all the marks set in the traceability matrix correct; that is, are the Use Cases connected to the right scenarios and vice versa?	

Checklist-"Tester"

No.	Question	
Actors		
1	Are all the actor that are interested in the system reaction considered in the Use Cases and consistently described in the related scenarios?	
Event and Reactions		
2	Are all possible actor actions considered in the textual Use Case description and the related scenarios?	
3	Are all the actor actions described in the Use Cases consistently described in the related sce-narios?	
4	Are all possible system reactions clearly described in the Use Cases and the related scenarios for or each actor action?	
Functionalities + Quality		
5	Can you identify the functionalities and quality requirements that need to be tested in sys-tem or acceptance testing in the Use Cases?	
6	Are all the functionalities and quality requirements that are described in the Use Cases con-sistently described in the related scenarios?	
7	Are all the important functionalities and quality requirements of the Use Case considered in the textual Use Case description and the related scenarios?	
8	Are all functionalities and quality requirements of the Use Case and the related scenarios clearly described; that is, are they easy to understand?	
9	Can you derive test cases for system and acceptance testing from the Use Cases; that is, are the Use Cases verifiable?	

References

[AB89] Ackerman A F, Buchwald L S, Lewski F H. Software Inspections: An Effective Verification Process. IEEE Software, 6-3 (1989), 31-36.

[Bas97] Basili V R. Evolving and Packaging Reading Technologies. Journal of Systems and Software, 38-1 (July 1997) 3-12.

[BG96] 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.

[Camp63] Campbell D., Stanley J. Experimental Designs for Research Houghton Mifflin Company, Boston, MA, 1963

[Che96] Chernak Y. A Statistical Approach to the Inspection Checklist Formal Synthesis and Improvement. IEEE Transactions on Software Engineering, 22-12, (December 1996) 866–874.

[CD97] Ciolkowksi M, Differding C, Laitenberger O, and Munch J, Empirical Investigation of Perspective-based Reading: A Replicated Experiment, Technical Report ISERN-97-13 (1997).

[CLB03] M.Ciolkowski, O. Laitenberger, Stefan Biffl. A global Survey on the state of the practice in Software Reviews, submitted to IEEE Software Special Issue on the State of the Practice in Software Engineering, December, 2003.

[DR01] Dunsmore A., Roper M, Wood M. Systematic Object-Oriented Inspection - An Empirical Study. Proceedings of the 23rd Int. Conf. on Software Engineering, 2001, 135-144.

[DR02] Dunsmore A., Roper M, Wood M. Further Investigation into the Development and Evaluation of Reading Techniques for OO Code Inspection. Proceedings of the 24th Int. Conf. on Software Engineering, 2002, 47-57.

[DR03] 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).

[Fag76] Fagan M E, Design and Code Inspections to Reduce Errors in Program Development, IBM Systems Journal 15-3 (1976) 182–211.

[FL97] Fusaro P, Lanubile F, and Visaggio G. A Replicated Experiment to Assess Requirements Inspection Techniques, Empirical Software Engineering 2 (1997) 39–57.

[GG93] Gilb T, and Graham D. Software Inspection. Addison-Wesley Publishing Company, 1993.

[Hoe00] Höst, M., Regnell, B., and Wohlin, C. 2000. Using Students as Subjects – A Comparative Study of Students and Professionals in Lead-Time Impact Assessment. Empirical Software Engineering: An International Journal 5(3): 201-214.

[Hum89] Humphrey W S. Managing the Software Process. Addison-Wesley Publishing Company, 1989.

[LA00] 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.

[LE01] 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.

[LM79] Linger R, Mills H, Witt B. Structured Programming: Theory and Practice. Addison-Wesley, 1979.

[MW98] Miller J, Wood M, and Roper M. Further Experiences with Scenarios and Checklists, Empirical Software Engineering 3 (1998) 37–64.

[PV95] 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.

[PV98] 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.

[SB98] Sandahl K, Blomkvist O, 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.

[SM+02] 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

[TR02] Thelin T, Runeson P, Wohlin C, Andersson C. How much Information is Needed for Usage-Based Reading - A Series of Experiments. Proc. of the Int. Symp. On Empirical Software Engineering, 2002, 127-138.

[TR03] Thelin T, Runeson P, Wohlin C. Prioritized Use Cases as a Vehicle for Software Inspections. IEEE Software, 20-4 (July/August 2003), 30-33.

[TS+99] Travassos, H. Guilherme; Shull, Forrest; Fredericks Michael; Basili, Victor; Detecting Defects in Object Oriented Designs: Using Reading Techniques to Improve Software Quality; In the Proceedings of the Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA); Denver, Colorado; 1999

[Wie02] Wiegers, K. E.; Peer Reviews in Software - A Practical Guide – Addison-Wesley, 2002

Document Information

Title:	The Empirical Investigation of the Success Factors of Scenario Based Reading
Date:	Oktober 24, 2003
Report:	IESE-115.03/E
Status:	Final
Distribution:	Public