An Evaluation of Checklist-Based Reading for Entity-Relationship Diagrams

Claes Wohlin
Dept. of Software Engineering and Computer Science
Blekinge Institute of Technology
Box 520, SE-372 25
Ronneby, Sweden
Claes.Wohlin@bth.se

Aybuke Aurum
School of Information Systems, Technology and Management
University of New South Wales
Sydney NSW 2052
Australia
aybuke@unsw.edu.au

Abstract

Software inspections have attracted much attention over the last 25 years. It is a method for static analysis of artefacts during software development. However, few studies look at inspections of entity-relationship diagrams (ER-diagrams), and in particular an evaluation of the number of defects, number of false-positives and the types of defects that are found by most reviewers. ER-diagrams are commonly used in the design of databases, which makes them an important modelling concept for many large software systems. This paper presents a large empirical study (486 subjects) where checklists were used for an inspection. The goodness in the evaluation is judged based on the type of defects the reviewers identify and the relationship between the detection of real defects and false positives. It is concluded that checklist-based inspections of entity-relationship diagrams is worthwhile.

1. Introduction

Software inspections were first formalized more than 25 years ago [7]. The area has since been researched and further developed over these years; an overview is provided in [1]. Despite these efforts, there are still a number of open research issues in relation to software inspections.

Another area that has been developed over the years is methods for documenting and describing software. Entity-relationship diagrams [13] are one example of such a method that is used in the design of databases. However, little research has studied inspections in the context of Entity-Relationship diagrams (or ER-diagrams). Amongst the work most closely related to this type of diagram is research done on inspecting UML diagrams [12].

The need for researching the effectiveness of checklists for inspections is stressed in [5], and derivation of checklists is discussed in [6]. The objective of this paper is to evaluate checklist-based inspections for ER-diagrams. In particular, the focus is on the individual phase of software inspections and checklist-based reading. During individual preparation, the participants may either become acquainted with the artefact being reviewed, or they may focus on defect detection. In this case, the latter is the focus. An evaluation was carried out through a major empirical study. 486 subjects were asked to use a checklist to find defects in an ER-diagram. Evaluations as a research method are further described in [17]. Here, we ran the evaluation as a controlled experiment without a control group. As far as we are aware, this is by far the largest single empirical study in this area of software engineering.

The researchers developed the checklist for the study manually. However, given that we used ER-diagrams, it would be possible to generate the checklist automatically. In this case, we refer to a tailored checklist for the specific design. In most studies, the checklists used have been generic and in some cases on a fairly high level, see Section 2.. This is natural when reviewing, for example, requirements specifications, but it is not necessary when reviewing a more formal notation.

The effectiveness of the inspection was evaluated by studying the number of defects found by the reviewers. The number of identified defects was also compared with the number of false positives identified by the reviewers. Defect classifications are used in the study to enable the evaluation of the number of different types defects found. In particular, we evaluated whether defects found in the inspections would be hard to find during implementation. We were also interested in whether the actual effect of leaving the defect in the system would be major or not.

The remainder of the paper is outlined as follows. Some related work is presented in Section 2. The method used for the evaluation is described in some detail in Section 3. Section 4. presents the results from the study, which is then further analyzed and interpreted in Section 5. Finally, a discussion and some conclusions are presented in Section 6.

2. Related work

Since to the best of our knowledge, there has been limited research on inspections of ER-diagrams, we have widened our literature review to include the inspection of requirements documents overall. By doing so, we have been able to compare our results to those of earlier research.

It is important to have a solid software requirements specification for the development of high quality systems. Although this is a well-known fact, a large number of defects occur during the early phases of software development. These defects may have serious consequences if they are not identified and corrected in that particular stage of the development process.

It has been shown that reading techniques have a great influence on the defect detection rate of the requirements specification document. The ability of inspectors to find defects varies with the technique used. Additionally, their attention can be directed towards particular inspection targets [3]. Therefore, many researchers have focused on experiments that give statistically validated information on the effectiveness of different reading techniques applied to the requirements inspection. Porter *et al.* [16] conducted an experiment with 48 graduate students that compared the scenario-based reading technique to the approaches of ad hoc reading (no support) and checklist-based reading. They concluded that the scenario approach was superior to both the ad hoc and checklist techniques, with respect to the defect detection rate. Additionally, it was found that the checklist technique was no more effective than the ad hoc technique. In a replication of Porters et al's experiment Miller *et al.* [14] compared scenario-based reading solely to the checklist approach, finding that the defect detection rate of the scenario approach was once again superior to that of the checklist technique. On the other hand, Fusaro *et al.* [8] obtained different results than those of the original experiment [16]. They conducted an experiment with 30 undergraduate students and could not find any empirical evidence of better defect detection when using scenarios. In accordance with [8], Sandahl *et al.*, [18] could not support the superiority of the scenario technique either. In a later replication of the same experiment Halling *et al.* [10] received results that were quite different from the other studies. Their large-scale experiment (150+ undergraduates) led to the conclusion that checklist-based reading was overall more effective on the individual level, whereas the scenario approach gained effectiveness when applied to a certain target focus, in their case, specific parts of the document.

The somewhat contradictory results from literature illustrate that it is very difficult to compare reading techniques. The actual reasons for the differences are as yet unknown. It is hard to say whether the results are due to the similarity in the performance of the reading techniques, or if they are due to our inability to capture the context to a sufficient extent. Thus, we have chosen here not to compare two or more reading techniques, but rather to focus on one reading technique (checklist-based reading) and try to evaluate defect detection, identification of false-positives and if specific types of defects are more easily detected in this type of inspection.

Finally, it is worth noting that an extensive overview of checklists for inspections is published in [5], where 117 checklists have been studied. In this paper, it is concluded that many checklists are too generalized. The need for tailored checklists is mentioned, although it seems to primarily be targeted towards organizations or a specific type of system. The paper does not mention the possibility of generating tailored checklists for a specific diagram, for instance, an ER-diagram. Thus, the approach taken in our paper seems to be novel, although the checklist is not actually generated. However, it is clear from the way the checklist was constructed that a tool could be designed to generate a checklist from ER-diagrams.

3. Method

3.1. Introduction

The general objective of the study is to evaluate whether inspections of ER-diagrams using checklist are worth conducting.

The study has conducted within the context of a database course at the University of New South Wales in Sydney, Australia. This means that the subjects in the study were students, which of course is not the same as using professional engineers as subjects, although some studies have shown similar results [11]. However, there is a general need to understand the applicability of results using these types of subjects in real life situations.

On the other hand, we believe that whether that participants have a stake in the study is as important as whether they are students or professionals. However, the latter is a question for further research. Most student experiments are set up with students who do not care about the actual findings, since they neither deal with the defects themselves nor with the artefact afterwards. Thus, this particular study has been conducted as part of the assessment for a database course, such that the students did have a stake in the study. Moreover, they have as part of the course worked with identifying defects in ER-diagram, so they are familiar with the type of task given to them. The students were marked based on their ability to satisfactorily carry out the given task.

The actual ER-diagram used was small scale, but it includes many of the different concepts often included in ER-diagrams. This means that the diagram was fairly realistic, although limited in size. The diagram was adapted from [13].

3.2. Subjects

As mentioned, the subjects in the study were students. The students came from several different educational programs in the area of information systems. In total, 486 undergraduate students participated in the study. The large number of subjects ensured that the individual influence on the results is minimized.

3.3. Problem statement and hypotheses

Inspections are often promoted as a cost-effective way of detecting software defects [9]. Moreover, different reading techniques have been investigated by researchers to determine what is an effective and efficient way of carrying out the individual preparation in software inspections. Some examples of studies evaluating reading techniques can be found in [2, 18].

So far, the research community has been unable to determine when a specific reading technique is superior to others. Different studies come to different conclusions and the main reason is most likely that there is not yet a full understanding of the factors influencing the superiority of a specific technique. This includes a lack of understanding with respect to type of artefact, type of application and the type of people actually performing the inspections. Instead of trying to determine which reading technique is "best" for ER-diagrams, the focus here is on evaluating whether checklist-based reading may be useful. The focus is on checklists for three reasons:

- It is a commonly used technique by software developers [5] and it is often used as a benchmark for new reading techniques [19].
- It has been found to be fairly competitive in general in comparison to other reading techniques [21]. This does not mean that it is superior in general, but it often performs well.
- As mentioned above, it would be easy to automate the generation of checklists from ER-diagrams.

Thus, the problem addressed in this paper is whether checklist-based reading is good at detecting defects in ER-diagrams.

To enable an evaluation of the goodness of checklist-based reading for inspections of ER-diagram, two aspects are tracked in the study:

- The defects are classified according to three categories. The classification is done based on a subjective judgement of the lecturer-in-charge who has more than 10 years experience of teaching ER-diagrams. The first is whether the defect would be hard or easy to detect during the implementation of the ER-diagram. The second is concerned with the actual impact of the defect if it were to stay in the system, i.e. major or minor. The third is whether a defect is viewed as being syntactic or semantic. The classification is further discussed as part of the design of the study.
- Both detection of real defects and false positives is logged in the study. This is also discussed in more detail as part of the design of the study.

Based on the above, two hypotheses are formulated:

1. An inspection of ER-diagrams is effective if the inspection helps in detecting the defects that would be hardest to detect during implementation and if the potential impact on operations is major. The two classifications of defects of particular interest may be combined into four main classes, which are of particular importance when studying the effectiveness or general goodness of an inspection using checklists:
   1) Major and Hard (MajHa)
   2) Major and Easy (MajEa)
   3) Minor and Hard (MinHa)
   4) Minor and Easy (MinEa)

   The order of 2) and 3) may be argued. The ordering above is motivated by that we view the external effect most important, i.e. whether a defect is major or minor.

   Informal hypothesis: Checklist-based reading of ER-diagrams is effective since this technique identifies more defects of the important classes. More formally:

   H_0: There is no difference in the ranking of defects found by the reviewers, where the defect found by most reviewers has rank one and so forth. Let Rank(MajHa), Rank(MajEa), Rank(MinHa) and Rank(Min Ea) be the ranks for respective combinations of the two main defect classifications. The hypothesis can now more formally be stated as: H0: Rank(MajHa) = Rank(MajEa) = Rank(MinHa) = Rank(MinEa).

   H_A. The alternative hypothesis is that the defect classification makes a difference in terms of how difficult a defect is to find (in an ER-diagram). This can also be formulated as that there is a difference between two or several of Rank(MajHa), Rank(MajEa), Rank(MinHa) and Rank(MinEa).
2. An inspection of ER-diagrams is effective if the inspection helps in identifying more real defects than false positives. In addition, a good approach should support the strong reviewers in finding many defects, but few false positives. In particular, we would like to see a high negative correlation between real defects and false positives. The reason being that strong reviewers should find many defects and indicate few false positives while the not-so-strong reviewers (relative to the others) are expected to indicate either few of each or many of each. Hypotheses:

   a) Reviewers identify significantly more real defects than false positives.

   b) There is a high negative correlation between real defects and false positives.

These two main hypotheses are first studied through descriptive statistics (Section 4.) and then evaluated more formally (Section 5.).

3.4. Preparation

The students in the course were taught ER-diagrams as an important modelling tool when designing databases. As part of the course the students were given a lecture and a training session (tutorial) on inspections and checklist-based reading. Some students did not attend the lecture given that it was not mandatory. However, most students attended the training. It should be noted that, overall the students had to attend 80% of the lectures and tutorials. During this specific tutorial, they were provided with training in ER-diagrams in general, as well as ad-hoc and checklist-based reading of ER-diagrams. Thus, a clear majority of students had been exposed to inspections of ER-diagrams prior to the actual study. The study was conducted as part of a course and hence the normal procedure on the course where judged as being more important than making one specific lecture and tutorial mandatory.

In addition to the student training, the actual study was also run in three pilot studies prior to running the study with the 486 subjects. After each pilot study, the design of the study was updated based on the feedback. The subjects in the pilot studies were primarily colleagues. The objective of the pilot studies was, of course, to ensure that the design of the study was as good as possible. This was crucial both from an educational and research point of view, particularly with the large number of subjects involved.

3.5. Design of study

3.5.1. Instructions. At the start of the study, the students were informed about the objective of the study, from both an educational and research perspective. The students were given a story, i.e. business rules, that led to the specific ER-diagram, and the ER-diagram itself. Both the story and the diagram were adapted from [13]. The main reason being that it was felt that it would be better to take an established diagram and adapt it slightly as well as inserting defects, instead of inventing a story and corresponding diagram from scratch. The object of the study is described in more detail below.

The students were given a general instructions indicating that they should study the ER-diagram for defects with respect to notation, cardinality, connectivity, referential integrity and entity integrity. The intention was that this type of description would help the students identifying any omissions or general problems between the two representations (business rules vs. ER-diagram). In addition, they received a specific checklist that was tailored for the specific diagram, i.e. the checklist was generated from the data dictionary forming the basis for the ER-diagram.

3.5.2. Object. The object of the study was a story and an ER-diagram adapted from [13] on pages 112-113. The ER-diagram fit onto one page. It contained both strong and weak entities, relationships, different numbers of connectivity, different cardinalities, optional parts and different types of attributes. The attributes included normal attributes, multi-valued attributes, composite attributes and derived attributes. Thus, while it could be seen as fairly complex, it was not too large.

The story and the corresponding ER-diagram described a company that sold a number of different furniture products. The products are grouped into product lines. Customers submit orders. The products are assembled from one or more raw materials, and each raw material may be used in different products. The company also has different work-centres and a product is only produced at one of them. Thus, the ER-diagram models a fairly complex business that would like to keep track of the business of producing and selling furniture products.

3.5.3. Checklist. As mentioned above, the checklist contained one general part and one specific part. The latter was specific for the ER-diagram and it was developed for this specific study. It was developed manually by the lecturer-in-charge (LIC), but a similar checklist could be generated automatically from a given ER-diagram.

The specific part of the checklist was developed from the entities and relationships in the diagram. The checklist contained 63 items that needed to be checked with respect to the aspects above. The students were asked to check the ER-diagram using the checklist and compare this with the story. The story was assumed to be correct, so all inconsistencies between the story and the ER-diagram were viewed as defects in the ER-diagram. They were asked to tick the box if the item was correct and mark a cross if it was incorrect. If they marked an item as being incorrect, they also had to supply a short description of what was wrong. If they failed to do the latter, it was not regarded as a defect.

The general part of the checklist was used to identify some issues. However, they were indirectly connected to items in the specific checklist and hence we have chosen to view all defects as related to the 63 items in the specific checklist, although the students actually identified some of the problems primarily from the general part of the checklist.

The instructions given to the subjects are summarized below for the two parts in the checklist. An excerpt from the specific part of the checklist follows this summary to illustrate the type of issues listed.

General part:

The first task given to the subjects was to examine the ER model in terms of its notation, cardinality, connectivity, referential integrity and entity integrity and compare it with the story. In this task, the students were asked in general to, for example, identify problems in the relation between the diagram and the story, which include identifying missing items or any other omission. The story was assumed to be correct, so all inconsistencies between story and the ER model were viewed as defects in ER-model.

Specific part:

In the specific part the subjects were to study a list of entities and relationships in terms of their notation, cardinality, connectivity, referential integrity and entity integrity. For each component they should indicate correctness with a tick (V) or incorrectness with a cross (X). Further, they should write down the correct solution and use the reference number given in the checklist. An excerpt from the specific part of the checklist is shown below in Table 1. The names of the entities and relationships are given to the subjects from the data dictionary provided together with the story.

Table 1: Excerpt from checklist.

Number	Entity/Relation
...	
16	OrderLine -- QUANTITY
17	QUANTITY
18	OrderLine -- OrderDate
19	OrderDate
20	OrderLine - PRODUCT
21	PRODUCT
22	PRODUCT -- ProdID
...	

3.5.4. Defects. As mentioned briefly above, the defects were classified into three categories with the two first being of particular importance from the objective of evaluating its effectiveness in general terms:

- Impact in operation: major or minor
- Difficulty to detect in implementation: hard or easy
- Defect type: syntactic or semantic

The classification was made by the LIC and each defect was classified into the relevant categories. There was no one to one mapping between any of the three defect classification schemes. Thus, the categories cover three different perspectives of defects.

Out of the 63 items in the checklist, 25 of them contained a defect. Thus, the subjects should optimally have indicated the 25 defects by putting crosses in the boxes in front of the erroneous items and then providing a description of the problem. The other 38 items should have been ticked and no further explanations were needed. The 25 defects were seeded by the lecturer-in-charge who has experience in dealing with these types of diagram and defects that occur in them. Moreover, the defects were studied and discussed as part of the three pilot studies. Thus, the defects were judged to be representative of defects that may occur in this type of diagram, although the defects in this study are seeded into the diagram.

The division of the 25 defects into the two main categories is shown in Table 2.

Table 2: Defects into categories

	Hard	Easy
Major	6	6
Minor	9	4

3.6. Operation of study

The study was run with all subjects performing the study at the same occasion. The subjects were requested to hand in the checklists with their ticks and crosses, along with a short description of the defects that they had identified. After the study was run, several teaching assistants marked the material from the study. However, the LIC coached them and also ensured equal treatment of the material although different assistants handle different subjects.

3.7. Threats

Four different types of threats should be addressed [20]:

- Conclusion validity. This threat is concerned with issues that affect the ability to draw the correct conclusions about the relationship between treatment and outcome. The conclusion validity is believed to be high given the large number of subjects. The main threat here is that some subjects still misunderstood the task. This can be exemplified with reference to the six subjects who had 38 false positives. However, this is very few relative to the total number of subjects and hence the validity should be high.
- Internal validity. This threat is related to issues that may affect the causal relationship between treatment and outcome. All subjects were exposed to the same treatment and the use of the checklist is clearly related to the defects. Thus, there are no threats to the internal valid-

ity of the study except the slight problem of the misunderstanding of the instrument in terms of marking 38 false positives.

- Construct validity. This threat is concerned with issues related to the design of the study and social threats. The study was carefully designed, and the design was piloted three times before actually being run. This means that the threats to the design are limited. The study was run within a limited period of time, and all students were expected to be equally motivated. Hence, no social threats have been identified.
- External validity. This threat is always the most difficult to address and evaluate. It is concerned with the ability to generalize the findings beyond the actual study. The concern is mostly about how representative the subjects are and whether the object of study is realistic. In this case, the subjects were students, which always is criticized in empirical studies. However, the students performed this study as part of a database course of which ER-diagrams were an essential part. This means that the students were quite knowledgeable on this type of diagram and hence their ability to inspect the diagrams should be very good. However, it is hard to say how it actually relates to industrial engineers. The students were however highly motivated, since the study was part of an examination on the course. The actual degree of effectiveness in finding defects also points in this direction. The effectiveness is discussed further in the next section. The object of study is for obvious reasons fairly small, i.e. it fits onto one page. However, the diagram was fairly complex, although it may include more defects than can be expected in general. Anyhow, it is believed that the results provide an indication of what one can expect more generally, although more studies are needed before this can be stated with any certainty.

4. Results

An analysis was conducted, after the teaching assistants finished marking the exam papers and recording the data. This section presents the data, primarily using descriptive statistics. The interpretation with respect to the hypotheses and a discussion are provided in Section 5.

In Figure 1, the number of subjects detecting a certain number of defects is shown. It should be noted that the first bar shows the number of subjects detecting zero defects and the last bar (number 26) shows the number of subjects detecting all 25 defects. Thus, the categories in Figure 1 and Figure 2 are equal to the number of defects plus one, i.e. category “1” is equal to zero defects. The figure clearly shows that a majority of the subjects found a high number of defects. The median number of defects detected is as high as 19.

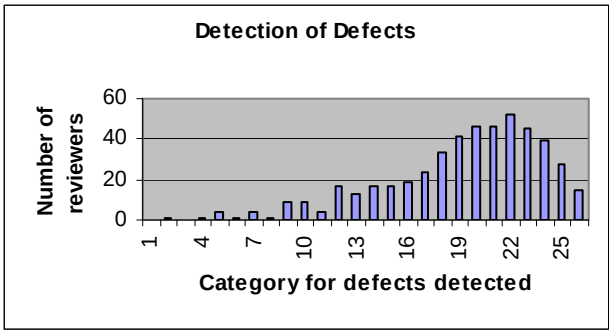

Figure 1: Defects detected

Similarly, the number of false positives is shown in Figure 2. The number of subjects that did not mark any false positives is shown in the bar marked “1”, i.e. category “1”, and so forth. It is clear that the number of false positives in general is fairly low. The median number of false positives marked by the subjects is three. It is, however, noteworthy that six subjects marked 38 false positives (bar number 39). This means that they marked all 63 boxes with crosses. It seems as though these six subjects completely misunderstood their task.

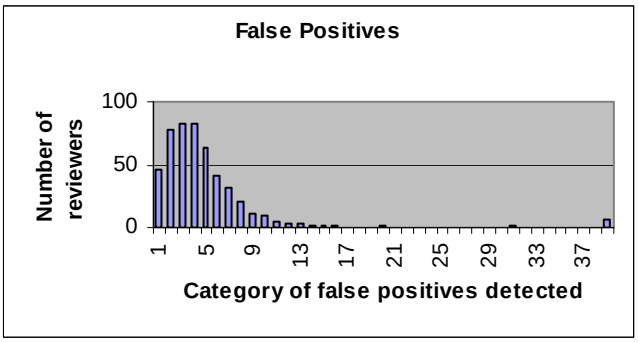

Figure 2: False positives marked

The descriptive statistics show that the subjects identified considerably more real defects than they mark false positives. This indicates that it may be effective to perform inspections of ER-diagrams. The study also shows that the subjects actually found a large number of the defects. The median number of defects found is 19 and hence the effectiveness becomes 19/25, i.e. 76%, which must be regarded as a fairly high effectiveness in an inspection. The proportion of false positives is fairly low. Here, it seems reasonable to compare the median number of false positives with the sum of the medians for real defects and false positives. This would give a proportion of 3/(3+19), i.e. 13.6%.

This indication is further elaborated on in the following section where a more in-depth analysis is conducted, and the hypotheses stated in Section 3. are revisited. The

subjects marking 38 false positives are removed from further analysis since these subjects misunderstood how the marking should have been done. In total this means that 480 subjects remain for the further analysis.

5. Analysis and interpretation

The results in the previous section provide some indications of where the results are heading. However, to strengthen the bearing of results a more detailed analysis is conducted below. In particular, the objective is to provide answers to the hypotheses stated in Section 3..

5.1. Detection of different defect types

This subsection presents the analysis of the first hypothesis in Section 3.. The data used in the analysis is provided in Table 3. The table shows the defect numbers in the first column, and the number of reviewers who found that particular defect in the second column. Based on the second column, the ranks for the defects are provided in the third column. The highest ranked defect is the defect found by most reviewers (defect number 7 is found by 278 reviewers and hence it is ranked as 1) and so forth. The last column shows the classification of the defect according to the three defect classification procedures used. The two classifications of main interest are whether a defect is hard or easy to find during implementation and if a defect is viewed as being major or minor. The latter refers to the effect if the defect was to be carried into the field.

Table 3: Summary of identification of defects.

Defect number	Number of reviewers	Rank	Defect classification[a]
1	106	17	MajEaSyn
2	133	10	MajEaSyn
3	141	8	MinHaSem
4	88	19	MinEaSyn
5	109	15	MajEaSem
6	107	16	MajHaSem
7	278	1	MinHaSyn
8	192	5	MinHaSyn
9	236	2	MinHaSyn
10	209	3	MajHaSem

Table 3: Summary of identification of defects.

Defect number	Number of reviewers	Rank	Defect classification[a]
11	141	8	MajEaSem
12	171	7	MajHaSem
13	85	20	MinEaSyn
14	111	11	MajHaSyn
15	111	11	MinHaSyn
16	51	25	MinHaSyn
17	201	4	MinHaSem
18	111	11	MinHaSyn
19	105	18	MajEaSyn
20	71	21	MajEaSem
21	59	23	MinEaSem
22	57	24	MinEaSem
23	71	21	MajHaSyn
24	111	11	MajHaSyn
25	186	6	MinHaSem

a.Maj-Major, Min-Minor; Ha-Hard, Ea-Easy; Syn-Syntatic, Sem-Semantic.

The next step is to test whether certain types of defects were relatively easier to detect. This means that we would like to use statistical inferences to evaluate the differences. We have chosen to evaluate the differences in ranks based on the number of reviewers that found a particular defect. In general, it is assumed that a defect found by more of the reviewers is actually an easy defect to find, which should not be confused with the classification “easy”. The latter is a judgement of how easy the defect is found in later development. The ranks from Table 3 are mapped to the two main defect classifications in Table 4. For example, it can be seen that the easiest defect to find (defect number 7, which is ranked as number one) appears in the cell with “hard” and “minor”, since “1” can be found in this cell. Similarly, it can be seen that the defect found by the fewest reviewers can be found in the same cell, since “25” appears in that cell.

Table 4: Ranks and relation to the two main classifications.

	Hard	Easy
Major	3, 7, 11, 11, 16 and 21	8, 10, 15, 17, 18 and 21
Minor	1, 2, 4, 5, 6, 8, 11, 11 and 25	19, 20, 23 and 24

From Table 4, it is noticeable that the ranks in the cell "easy" and "minor" are fairly high. This indicates that these defects have been the hardest to find. The next question is whether this observation is statistically significant.

First, a non-parametric test is applied. In this case a Kruskal-Wallis test is appropriate. This type of test is capable of detecting that differences exist, however it does not provide any means for finding out between which types of defects the differences exist. We have therefore chosen to apply an ANOVA test if the Kruskal-Wallis test gives a significant value.

The ANOVA test is a parametric test, but it is generally robust, see, for example, [4]. A significance level of 0.05 is used for all tests. If the ANOVA test turns out to be significant, a Fisher PLSD (Protected Least Significant Difference) test is performed to evaluate the pairwise significance between the measures [15]. The test is called "protected" since it is only used after a significant ANOVA test. An example of a question that may be evaluated with the Fisher's PLSD (after a significant ANOVA) is whether there is a significant difference between "major and hard" defects and "minor and easy" defects?

A plot of the average ranks for the four classes generated by the two main defect classes can be found in Figure 3 in the following order: Bar 1: Major/Hard; Bar 2: Major/Easy; Bar 3: Minor/Hard and Bar 4: Minor/Easy. Here, it is worth noting that a lower rank is viewed as better (as in Tables 2 and 3), and once again it is possible to see that minor and easy defects comes out worst in the ranking, which is shown by the fourth bar.

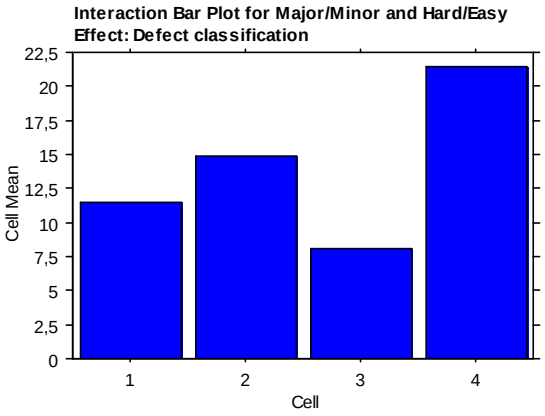

Figure 3: Bar plot for the defect classification with respect to Major/Minor and Hard/Easy.

Based on the four combined classes in Figure 3, a Kruskal-Wallis test can be performed. The p-value for this test was 0.026 and for the ANOVA test the p-value was 0.0098. Both tests are highly significant. Both these p-values are significant with the chosen significance value (the p-values are actually considerably lower than the chosen significance level). Using Fisher's PLSD test means that the significant differences between the four classes can be identified. From the Kruskal-Wallis and the ANOVA tests, we only know that there are significant differences. To study where the significant differences are, the Fisher PLSD test or a similar test is needed. The results from Fisher's PLSD test are shown in Table 5.

Table 5: p-values for the four defect classes created.

Defect classifications	p-value
Major and Hard versus Major and Easy	0.35
Major and Hard versus Minor and Hard	0.30
Major and Hard versus Minor and Easy	0.018
Major and Easy versus Minor and Hard	0.047
Major and Easy versus Minor and Easy	0.10
Minor and Hard versus Minor and Easy	0.0013

From the table, it can be seen that three differences are significant with the chosen significance level (0.05). The significant differences are:

- Major and Hard (bar 1) vs. Minor and Easy (bar 4): This is good because it shows a statistically significant difference between the most important type of defects (major and hard) and the least important type (minor and easy).
- Major and Easy (bar 2) vs. Minor and Hard (bar 3): This result is not that important, since the order of importance between these two may be discussed (as mentioned in Section 3.). In addition, the difference is in the "wrong" direction, i.e. the minor and hard defects are easier to find.
- Minor and Hard (bar 3) vs. Minor and Easy (bar 4): This result is also strong since it shows that minor defects that are hard to find in implementation are easier to find than those that are easy to find in implementation.

In summary, results are significant and indicate that the hardest defects to find in this type of inspection are those that are both minor and easy to find during implementation.

Thus, the results show that inspections of ER-diagrams with tailored checklists most likely are worthwhile. It is also worth emphasizing that the type of checklist used could be generated automatically from the diagram, and hence the results are based on a tailored checklist. With respect to the first research hypothesis, it is concluded that these types of inspections are effective in finding defects. This is concluded based on the large number of defects actually detected in average.

As an aside, we have also analyzed if there are any significant differences in the identification of syntactic versus semantic defects. In this case, the non-parametric Mann-Whitney test is applied, given that we only have two levels (syntactic or semantic). The p-value is 0.72 and hence there are no significant differences in identification of syntactic and semantic defects respectively.

5.2. Real defects and false positives

For an inspection to be truly effective, it is also important that the inspection does not find too many false positives in relation to the number of true defects found. This issue has two aspects, which relate to the two sub-hypotheses of the second hypothesis in Section 3.. First, it has been studied whether the number of true defects found is significantly higher than the number of false positives identified. Second, it has been studied whether people finding a lot of defects also find few false positives. If the latter were the case then we would have very effective reviewers. The different sub-hypotheses are presented in the order they were introduced in Section 3..

a) The first sub-hypothesis is concerned with whether more real defects are found than false positives. This can be tested using a non-parametric Mann-Whitney test. As already indicated by the descriptive statistics in Section 4., the reviewers found considerably more real defects than false positives. The median number of defects found was 19 out of 25 and the median for false positives was 3 out of a total of 38. It is quite obvious that there was a significant difference, which is also shown by the Mann-Whitney test. The resulting p-value was below 0.0001. In other words, it is clear with a high significance that the reviewers find substantially more real defects than false positives. This is very good, and the median of false positives is fairly low while the same time as the median of real defects found is rather high.

b) The second sub-hypothesis was evaluated using two different approaches that in different ways capture the same point. First, the correlation between real defects and false positives for all reviewers was calculated. If an inspection was really effective, then we would have had very good reviewers that could find a lot of defects and simultaneously mark very few false positives. In the case with really good reviewers and poor reviewers, with the latter marking few real defects and relatively speaking more false positives, we would have a high negative correlation between real defects and false positives. In this case the correlation is -0.10, which basically means that there is no relation between real defects found and false positives marked.

   Second, to study the previous finding in a little more detail, the reviewers were divided into three parts for both real defects and false positives. The objective was to study the pattern of how reviewers mark real defects and false positives. This approach resulted in nine different cells, as shown in Table 6. Ideally, the majority of the reviewers would be located on the diagonal starting from the upper right corner. The cell in the upper right corner of Table 6 indicates the best reviewers, those in the upper third of reviewers with respect to finding real defects and in the lower third when it comes to false positives. From the table, it is clear that no pattern is discernible.

Table 6: Identification of real defects versus false positives.

		False positives		
		Upper third	**Middle third**	**Lower third**
Real defects	**Upper third**	59	67	52
	Middle third	74	57	35
	Lower third	59	41	36

This is the main negative finding, i.e. that the reviewers finding many real defects are not different from the others in terms of marking false positives. Thus, the inspection is not effective when it comes to avoiding the marking of false positives. In other words, there does not seem to be a way to select reviewers that do not mark false positives as well as real defects. On the other hand, the median of the number of false positives is rather low. It is below 10% (3/38) of all possible false positives for this particular study. Thus, this drawback is not viewed as being too critical.

With respect to the second hypothesis, it is possible to conclude that inspecting ER-diagrams is most likely effective given the high number of real defects found in comparison to the false positives. However, it is probably not possible to expect that reviewers who find a lot of defects will find fewer false positives than others.

6. Conclusions

This paper has addressed the general goodness or effectiveness of checklist-based reading when inspecting ER-diagrams. Given that it should be possible to generate a tailored checklist from specific ER-diagrams, means that the use of checklists for inspecting ER-diagrams is an interesting approach. It would mean that we do not have to resort to generic checklists stating general issues like check for consistency, clarity and so forth.

As a starting point, two main hypotheses were stated with respect to the effectiveness. The first hypothesis was concerned with the type of defects found in the inspections. From this study, it is observed that the most important defects are found during the inspections. More specifically, this means that defects classified as hard to find during implementation, and as having major effects if they were to be left in the system are easiest to find. This is certainly a good characteristic of an inspection.

The second hypothesis was concerned with the detection of real defects versus the marking of false positives. No relationship was found between the number of defects found and the number of false positives marked. This is unfortunate, since it would have been better to be able to identify really effective reviewers that find a many defects and very few false positives. On the other hand, the number of false positives marked (median) is low in comparison to the number of defects found (median).

However, it is concluded that inspections using tailored checklists for ER-diagrams is most likely an effective way in finding many defects that otherwise slip through later phases. This is particularly applicable where tool support is developed to automatically generate checklists. Moreover, the large empirical study has shown that it is most likely worthwhile to perform checklist-based inspections on Entity-Relationship diagrams. Further work may include replicating the current study, evaluation of other reading techniques for ER-diagrams and the development of a tool for generating checklists from ER-diagrams.

Acknowledgment

We are grateful to everybody that has been involved in the study, including students, teaching assistants and colleagues that have helped us with the study and provided inspiration in the area.

References

[1] Aurum, A., Petersson, H. and Wohlin, C. (2002): ‘State-of-the-Art: Software Inspections after 25 Years’. *Software Testing Verification and Reliability*, 12(3), 133-154.

[2] Basili, V. R., Green, S., Laitenberger, O.; Lanubile, F., Shull, F., Sörumgård, S. and Zelkowitz, M. (1996): ‘The Empirical Investigation of Perspective-Based Reading’. *Emp. Software Eng.*, 1(2), 133-144.

[3] Biffl, S., (2000): ‘Analysis of the Impact of Reading Technique and Inspector Capability on Individual Inspection Performance’. *IEEE, 7th Asia-Pacific Software Eng. Conf.*, 136-145.

[4] Briand, L., El Emam, K. and Morasca, S. (1996): ‘On the Application of Measurement Theory in Software Engineering’. *Emp. Software Eng.*, 1(1), 61-88.

[5] Brykczynski, B. (1999): ‘A Survey of Software Inspection Checklists’. ACM Sigsoft: Software Engineering Notes 24(1), 82-89.

[6] Chernak, Y. (1996): ‘A Statistical Approach to the Inspection Checklist Formal Synthesis and Improvement’,. *IEEE Trans. on Software Eng.* 22(12), 866-874.

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

[8] Fusaro, P., Lanubile, F. and Visaggio, G. (1997): ‘A Replicated Experiment to Assess Requirements Inspection Technique’’.

Emp. Software Eng. 2(1), 39-57.

[9] Gilb, T. and Graham, D. (1993): *Software Inspection*. Addison Wesley Publishing Company. ISBN 0-201-63181-4.

[10] Halling, M., Biffl, S., Grechenig, T. and Koehle, M. (2001): 'Using Reading Techniques to Focus Inspection Performance'. *IEEE Proc. of Euromicro Conf.*, 248-257.

[11] Höst, M., Regnell, B. and Wohlin, C. (2000): 'Using Students as Subjects - A Comparative Study of Students and Professionals in Lead-Time Impact Assessment', *Emp. Software Eng.* 5(3), 201-214.

[12] Laitenberger, O., Atkinson, C., Schlich, M. and El Emam, K. (2000): 'An Experimental Comparison of Reading Techniques for Defect Detection in UML Design Documents", *Journal of Systems and Software*, 52(2), 183-204.

[13] McFadden, F. R., Hoffer, J. A. and Prescott, M. B. (1999): *Modern Database Management*, Addison-Wesley (5th edition).

[14] Miller, J., Wood, M. and Roper, M. (1998): 'Further Experiences with Scenarios and Checklists'. *Emp. Software Eng.* 3(1), 37-64.

[15] Montmomery, D. (1997): *Design and Analysis of Experiments*, John Wiley and Sons, ISBN 0-471-15746-5.

[16] Porter, A. A., Votta L. G. and Basili, V. R. (1995): 'Comparing Detection Methods for Software Requirements Inspections: A Replicated Experiment'. *IEEE Trans. on Software Eng.* 21(6), 563-575.

[17] C. Robson (2000): *Small-Scale Evaluation*, A SAGE Publications Ltd., London, UK, ISBN 0-7619-5510-0.

[18] Sandahl, K., Blomkvist, O., Karlsson, J., Krysander, C., Lindvall, M. and Ohlsson, N. (1998): 'An Extended Replication of an Experiment for Assessing Methods for Software Requirements Inspections'. *Emp. Software Eng.* 3(4), 327-354.

[19] Thelin, T.; Runeson and Wohlin, C. (2003): 'An Experimental Comparison of Usage-Based and Checklist-Based Reading'. *IEEE Trans. on Software Eng.* 29(8) (to appear August 2003).

[20] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B. and Wesslén, A. (2000): *Experimentation in Software Engineering: An Introduction*, Kluwer Academic Publisher, USA, ISBN 0-7923-8682-5.

[21] Wohlin, C., Petersson, H., Aurum, A. (2003): 'Combining Data from Reading Experiments in Software Inspections: A Feasibility Study'. in *Lecture Notes in Empirical Software Engineering*, edited by N. Juristo and A. Moreno, World Scientific Publishing, ISBN 981-02-4914-4.