Empirical Software Engineering, 3, 327–354 (1998)
© 1998 Kluwer Academic Publishers, Boston. Manufactured in The Netherlands.

An Extended Replication of an Experiment for Assessing Methods for Software Requirements Inspections

KRISTIAN SANDAHL
ZeLab, Ericsson Radio Systems AB, Box 12 48, S-581 12 Linköping, Sweden

OLA BLOMKVIST
Quality Technology and Management, Department of Mechanical Engineering, Linköping University, S-581 83 Linköping, Sweden

JOACHIM KARLSSON
Focal Point AB, Teknikringen 1E, S-583 30 Linköping, Sweden

CHRISTIAN KRYSANDER
Department of Computer and Information Science, Linköping University, S-581 83 Linköping, Sweden

MIKAEL LINDVALL
Department of Computer and Information Science, Linköping University, S-581 83 Linköping, Sweden

NICLAS OHLSSON
Department of Computer and Information Science, Linköping University, S-581 83 Linköping, Sweden

Received November 18, 1996; Revised April 30, 1998

Abstract. We have performed an extended replication of the Porter-Votta-Basili experiment comparing the Scenario method and the Checklist method for inspecting requirements specifications using identical instruments. The experiment has been conducted in our educational context represented by a more general definition of a defect compared to the original defect list. Our study involving 24 undergraduate students manipulated three independent variables: detection method, requirements specification, and the order of the inspections. The dependent variable measured is the defect detection rate. We found the requirements specification inspected and not the detection method to be the most probable explanation for the variance in defect detection rate. This suggests that it is important to gather knowledge of how a requirements specification can convey an understandable view of the product and to adapt inspection methods accordingly. Contrary to the original experiment, we can not significantly support the superiority of the Scenario method. This is in accordance with a replication conducted by Fusaro, Lanubile and Visaggio, and might be explained by the lack of individual defect detection skill of our less experienced subjects.

Keywords: Controlled experiments, inspections, replicated study, method evaluation

1. Introduction

Inspections of various life-cycle documents are perceived as one of the most effective practices for quality assurance by many large-scale software developers, such as Ericsson and Lucent. The importance of inspecting requirements specifications is a logical consequence

of the well-supported fact that the earlier a defect is detected, the less it costs to fix it (Boehm, 1987).

Despite the effectiveness it must be realised that several costs are associated with inspections counted both in staff-hours and elapsed calendar time. This situation calls for intensive research and development for more efficient inspection methods.

This article contributes to this research by a replication of an experiment originally performed at the University of Maryland (UMD) and Lucent (Porter and Votta, 1994; Porter et al., 1995). In the article we will use the term *originators* when refering to Adam Porter, Larry Votta and Victor Basili, who made a laboratory kit for replications of their experiment available for us in 1995.

Replicated studies as ours and others (Fusaro et. al., 1997) are sometimes disregarded but are critical contributions since it may be possible to further validate findings. The originators observed that their Scenario method was 35% more effective than Ad Hoc and Checklist methods, and we were eager to see if we could replicate the same behaviour in our environment. As scientists we are also motivated by the fact that a replicated study makes more data from the experiment and its instrumentation available, which focus and enrich the discussion in the area of inspection methods.

The hypothesis of the original experiment was:

> "We believe that an alternative approach which gives individual reviewers specific, orthogonal detection responsibilities and specialized techniques for meeting them will result in more effective inspections." (Porter et al., 1995)

The replicated study described in this paper uses the same background reading material, the same checklists and the same specifications. This was possible by retrieving the laboratory kit from the originators. The defect detection rate has been recorded and analysed using two master defect lists: the one provided by the originators and one which extends the list to include other defects we considered to be real. Our list is a superset of the original one and is provided in Appendices A&B.

Our reason for extending the experiment was that in our education and industrial practice, the inspection should not only find functional defects, but also account for anomalies leading to problems for the users of the document.

Since checklist-based methods are the standard methods in our education practice, we chose to compare only the Checklist and Scenario methods. Including also the Ad Hoc method provides more control when interpreting the result, but it comes at a cost of requiring more data for a definite result.

1.1. Methods

The Scenario method and the Checklist method both re-use experience from earlier inspections by drawing the inspectors' attention to various defined topics. In performing the inspection with the Checklist method two phases are carried out:

1. Each inspector uses a Checklist to individually detect defects. The Checklist in the experiment is a two-page document of questions, for example "Are all inputs necessary

defined?". The Checklist is designed to cover a fault taxonomy developed by Basili and Weiss.

2. The team of inspectors gather in a *collection meeting* to discuss individual findings and compile a defect detection report.

An inspection using the Scenario method is conducted in a similar way with the exception that each inspector uses a unique *scenario* to detect the defects in the first phase. The scenario is documented in a single page of imperative instructions, for example "Identify at least one function that uses each output data object" . The Scenarios should be designed to complement each other to secure detection coverage while at the same time distributing the responsibility among team members, thus avoiding redundant work. In the experiment three scenarios are defined. For details, we refer to the originators (Porter et al., 1995).

1.2. Contexts

When reviewing the defect lists from the originators it became clear that they had focused entirely on functional errors and logical inconsistencies in the specifications. In our context, such as academic courses given and industry experience, a defect in a requirements specification is more broadly defined by also counting shortcomings in the specification that impede subsequent work as defects. We have thus created a longer list of defects present in the specifications. Examples of defects added are (no defects are deleted from the originators' lists):

- unclear statements or omitted information that is judged to significantly increase the risk of impeding the work of a designer;
- serious typing mistakes of tokens of the SCR-language (Heninger, 1980), for example a missing delimiter which can cause trouble if a designer uses a search command or performs automatic parsing; and
- omission of domain knowledge that is judged to trap a developer into designing an unusable system, for example, specification of unrealistic control algorithms.

We have not classified spelling errors and errors regarding font, layout and style as defects. Neither have we regarded deviations from standards as a reason alone for classifying a finding as a defect. For the complete lists, see Appendix A and Appendix B.

The final judgement of what should be counted as a defect was made by a two-person committee. Two persons were judged to optimise the competence they used, while reducing the complexity of internal communication. In the judgement the committee had to draw on personal experience. In total this included about 7 years as professional system designers and about 15 years as university researchers. This arrangement is in line with the results by Sauer et al. (1996) suggesting that a two-person expert team is the most cost-effective way of discriminating a true defect and a false positives. Sauer et al. draws on an extensive review of behavioural research findings.

By using the defect lists when assessing the results of the subjects we were able to measure the proficiency of the methods in both our and the original context.

1.3. Hypothesis

The major hypothesis is the same as the originators':

> "We believe that an alternative approach which gives individual reviewers specific, orthogonal detection responsibilities and specialized techniques for meeting them will result in more effective inspections." (Porter et al., 1995)

For this experiment this means that we expect the teams using the Scenario method to show a higher rate of detected defects in comparison to the teams using the Checklist method. The null hypothesis are thus

> H_0: Teams using the Checklist method have as high defect detection rates as teams using the Scenario method.

Similarly, there exists a set of null hypotheses all assuming that no independent variables and their interactions explain the variance in the dependent variable.

We are also interested in exploring if some factors could affect the detection rate when counting the defects added by us. This is exploratory research with no formal hypothesis, but we will be surprised if the variation of the defect detection rate of our added defects could be significantly explained by other variables than those explaining variance in the original context.

2. Experiment

2.1. Design

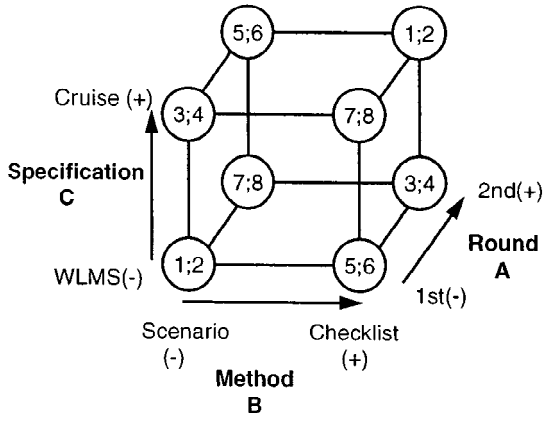
The experiment uses a randomised factorial design and manipulates the following independent variables:

1. Inspection method (Scenario or Checklist)
2. The requirements specification to be inspected (WLMS and CRUISE, see below)
3. The order in which specifications and methods are used.

The main dependent variable measured is the team defect detection rate. This measure is obtained by dividing the number of defects reported by each team by the number of defects known in the context in question.

We also measured the time spent in meeting. All subjects spent 2.5 hour in individual detection. The team meeting was free within the 2.5 hour limit. The check-out time was recorded by the experiment leader with a resolution of 5 minutes.

The experiment was carried out in two rounds, each comprising an individual inspection and a collection meeting with teams of three members each. The subjects of the experiment were students at B. Sc. level. Their average age was 24 years, and only three subjects were women.



Exp.	Team	A	B	C
1	1	-1	-1	-1
2	2	-1	-1	-1
3	3	-1	-1	1
4	4	-1	-1	1
5	5	-1	1	-1
6	6	-1	1	-1
7	7	-1	1	1
8	8	-1	1	-1
9	7	1	-1	-1
10	8	1	-1	-1
11	5	1	-1	1
12	6	1	-1	1
13	3	1	1	-1
14	4	1	1	-1
15	1	1	1	1
16	2	1	1	1

Figure 1. Experiment design. Each corner of the design-plot illustrates a unique combination of independent variables. The numbers in the circles are the numbers of the teams performing the various inspections. The design is also shown in the table which view each team's inspection as an experiment. Since there are two teams of each combination of variables, the total design comprises 16 experiments.

In the design-plot (Barton, 1998) of Figure 1 we show the design by marking the experimental runs with the groups that performed the experiment. For instance, group 3 and 4 in the first round used the Scenario method to detect defects in the CRUISE specification. To make the presentation of the sub-sequent analysis more compact, we introduce abbreviations for the variables and their values:

- A denotes Round and the minus sign denotes the first round and the plus the second;
- B denotes Method and the minus sign denotes Scenario and plus denotes Checklist; and
- C denotes Specification and the minus sign denotes WLMS and plus denotes CRUISE.

The table in Figure 1 contains almost the same information as the design-plot, we think of a design of 16 experiments where a given team performs an experiment with a combination of the independent variables. For instance, experiment number 10 comprise team number 8's inspection in the second round (+1) using the Scenario method (−1) with the WLMS (−1) specification. The reason for introducing the number 1 after the signs is that effects of variables and interactions can be conveniently calculated by multiplication of the vectors in such tables. For instance, to obtain the interaction between A and B, simply build a new column and multiply the values in columns A and B componentwise for each experiment. See Section 3.2 Statistical Analysis for practical usage.

Table 1. Raw data. For each of the 16 experiments, see Figure 1, we give the detected defects, total number of defects and defect detection rate for the original context and our context. We have also provided the time the teams used for their collection meetings to show that no correlation exists between time an performance.

	Detected defects		Total number of defects		Detection rate		meeting time
experiment number	original context	our context	original context	our context	original context	our context	(h:mm)
1	14	18	42	76	0.333	0.237	1:30
2	17	28	42	76	0.405	0.368	1:25
3	10	19	26	73	0.385	0.260	1:50
4	3	15	26	73	0.115	0.205	2:00
5	20	29	42	76	0.476	0.382	1:40
6	7	10	42	76	0.167	0.132	1:40
7	5	10	26	73	0.192	0.137	1:30
8	6	17	26	73	0.231	0.233	1:35
9	15	21	42	76	0.357	0.276	1:10
10	11	20	42	76	0.262	0.263	1:05
11	8	15	26	73	0.308	0.205	1:15
12	4	10	26	73	0.154	0.137	1:25
13	22	33	42	76	0.524	0.434	1:45
14	14	22	42	76	0.333	0.289	1:30
15	6	19	26	73	0.231	0.260	1:30
16	6	20	26	73	0.231	0.274	1:30

2.2. Validity Threats

The internal validity threats to the experiment we identified were:

1. Language. Since all materials obtained from UMD were in English, we feared that difficulties with the language might influence performance. To measure this influence, we allowed only a single dictionary to be used during detection sessions and logged how often it was used during a selected interval. During 5 hours of individual detection and collection meetings the 24 students together only used the dictionary seven times which was far below our expectations. On the basis of this evidence language appear not to have been important.

2. Motivation. Participation in the experiment was a mandatory part of the course examination and replaced a home assignment given in previous years. Since nothing but

the subjects' presence was recorded as far as the course was concerned, we considered the possibility that the students might not work very hard on actually detecting any defects. Some efforts were taken to provide a good working climate, see Section 2.5 Conducting the experiment. As a result we noted that the group as a whole showed active engagement in the task. Afterwards the students explained that the experiment had been taken seriously and that the enthusiasm in finding as many defects as possible was very high in the first round. The second round was performed with less eagerness. Afterwards, some students claimed that the two rounds were scheduled too closely together, and that the increased scheduled time compared to previous years had impeded their enthusiasm. Some were also bored with the SCR-notation after the first round. Our conclusion is that motivation caused no problem in the first round, but can be a problem in the second round. Since motivation is confounded with round it should be possible to detect this in the analysis

3. Elite teams. The originators first tried to limit the effect of accidentally creating an elite team by rating the background knowledge of the subjects. As several drawbacks with this approach were reported (Porter and Votta, 1994), we made a totally random composition of teams in our experiment. With one exception, the teams were the same in both rounds. The effect of an elite team with respect to its influence of the independent variables is balanced by the design. All teams inspect both specifications and use both methods. We are also reducing the effect of an elite team by having two teams for each combination of the independent variables.

4. Maturation. As the experiment proceeds the subjects learn the tasks. We tried to cope with this by completely varying the order in which specifications and inspection methods were used. Maturation is confounded with round and is thus possible to detect. There is a small chance that maturation and motivation will even out the effect of round.

5. Instrumentation. The differences in the requirements specification will affect the result. We tried to alleviate this by having all teams inspect both specifications and analyse the document setting as an independent variable.

The originators identified three external validity threats hampering generalisation of the results:

1. The subjects may not be representative of software programming professionals. This is, of course, a real threat that can never be removed. Some of the students had a professional background but for the majority this was their first inspection of requirements specifications. This fact has to be attained in the interpretation. The original experiment used graduate students with more experience, while the replication from the University of Bari used undergraduate students with a similar level as ours. Comparing our results with these two replications may give us some knowledge about how serious this threat really is.

2. The requirements specifications may not be representative of real software problems. This threat is also difficult to remove, but the documents are of about 30 pages each and the defects are naturally occurring, not inserted by the originators. A major obstacle

is the SCR notation, which is rarely used for requirements specifications in Sweden. Producers of, for instance, traffic control applications tend rather to use predicate logic and state diagrams. The students are used to various graphical interfaces and several students commented that they strongly disliked the SCR notations including tabular SCR notation. More training is probably needed to reveal the intuition and rationale behind SCR.

3. The inspection process is not representative of software development practice. The originators testify that the methods are comparable to the ones used at Lucent. We can add Ericsson to the list. Furthermore, inspections are frequently deployed in the large project courses given at Linköping University.

2.3. Instrumentation

The following instruments were copied from UMD:

1. Requirements Specifications:
   a. Heat Control describes a small controller for indoor climate. This was used for training only and contains a vast number of errors. (12 pages)
   b. Water Level Monitoring System (WLMS) describes a controller for a simulator of a steam generation system. (25 pages + figures)
   c. Automobile Cruise Control System (CRUISE) describes a system controlling the speed of a car. Features such as service indication and fuel consumption are included (30 pages + figure)
2. Method documentation (published in (Porter et al., 1995))
   a. The Checklist method. (two pages of instructions)
   b. The Scenario method. (three sets of instruction for the Scenarios: Ambiguities & Missing Functionality, Incorrect Functionality and Data Type Consistency)
3. Defect list. The list of defects found was not compatible with the pagination we got when printing the requirements specifications on an A4 format. We re-wrote it to comply with the new pagination and with our standard of localising defects in the specifications.

Adapted material:
In addition we created instructions in Swedish and defect report forms from the laboratory kit provided by UMD, and made copies of the references to the background training material (Heninger, 1980; Fagan, 1976; IEEE, 1984).

2.4. Preparation

After having received the material from UMD, four of us performed two inspection rounds for familiarisation, debugging and detection of new defects.

The students (25 persons at B. Sc. level) taking a course in system development were prepared in two lectures of 2×45 minutes each. The first lecture introduced the experiment, the IEEE-standard and the SCR-notation. A general document inspection tutorial of 45 minutes concluded the first lecture and the Heat Control specification was given as a home assignment. We told the students that no more than 60 minutes should be spent on the specification and that everything that was conceived as a defect should be reported. No special method for defect detection was required. The three articles were handed out.

The first half of the second lecture was devoted to collectively gathering in defects found in the Heat Control specification. Further clarification of SCR and the use of report forms was made. The second half introduced the Checklist method with examples from the Heat Control specification.

The teams were formed by drawing lots. One student was granted leave from the first round for personal reasons, so there were exactly eight teams with three students.

2.5. Conducting the Experiment

Two days were scheduled for the two rounds: September 15, 1995 and September 19, 1995. Four of the teams (12 students) started at 8.15 a.m. with a 45 minute lecture on the Scenario method. All of the three Scenarios (Ambiguities & Missing Functionality, Incorrect Functionality and Data Type Consistency) were handled. In the end of the lecture, each student was assigned one of the three scenarios to follow in the inspection right after lunch. The students not having the class in the morning had earlier been instructed to use the Checklist method. Thus, an inspection round could be completed during a single day. The students changed both method and specification in the second round and consequently a second Scenario method lecture was given to the ones using the Checklist method in the first round. Large efforts were taken to make the lectures as similar as possible. See Figure 1 for a summary of the experiment design.

Immediately after lunch the detection session was opened. All students received their specifications and performed an individual detection of 2.5 hours in a single lecture room. The students were advised to spend 30 minutes reading the material through and then 2 hours on the actual detection. Two of us were present to answer questions, prohibit unwanted communication and to log the use of the single dictionary. Only a few questions were asked.

Following the individual detection there was a short break with food and beverages. The subsequent collection session began in two separate rooms where teams inspecting the same specification were in the same room. Groups using the same method were located in opposite corners. One of us was present in each room. Overhearing other teams' discussions was very difficult. The students were advised to spend the first 30 minutes finishing their coffee and organising their work, and the rest of the time in collecting defects. No team worked more than 2.5 hours in total. The check-out time was recorded.

The second round was similar, but the methods and specifications were changed. One absent student was replaced by the student who was on leave from the first round.

Immediately after the two rounds a preliminary analysis was performed and the results were fed back to the students both individually on a special form and by a presentation

at a discussion seminar. In April 1996 the students were invited to a large exhibition in Stockholm with travel and lunch provided by our sponsors as a reward for their participation.

3. Analysis

3.1. Policy for Defect Classification

We collected all defects found by our own inspection of the documents, which amounted to 76 defects for WLMS and 73 for CRUISE. The originators' figures were 42 and 26 defects, respectively. Some defects found by students but not by us were included in our defect lists.

The policy for determining whether a defect had been detected was that, given the student's information, it would take less than a couple of minutes for the author of the document to correctly identify the defect. We used three rounds for the classification: a preliminary classification to provide rapid feed-back, a thorough classification made by two of us, and, finally, a classification made by a single person to ensure a consistent classification for all groups. Three classes were used:

1. A true defect. If a defect is reported several times, it still counts as one defect found.
2. A false positive—an obviously wrong statement of the document.
3. Unclassified comments, which are neither wrong nor a true defect.

3.2. Statistical Analysis

The defect detection rates for each team were calculated by dividing the number of defects found by the total number of defects found for the specification inspected in the given context. See Table 1. For example, the team of experiment number 1 found 14 defects occurring in the originators' list for the WLMS specification in the first inspection round. As the total number of defects was 42 the defect detection rate is 0.333.

Table 1 lists the defect detection rate in the two contexts for all experiments. We also provide meeting time to show that there is no correlation between performance and time. See Figure 2.

The average detection rate for each combination of the independent variables is shown in the Design-Plots (Barton, 1998) in Figure 3. Each corner represents a unique combination of the independent variables round number, specification and detection method. The number in each corner in Figure 3 is the average defect detection rate for the two teams with identical settings of independent variables. The design-plot in Figure 3 shows the result in the originators' context and the design-plot in Figure 4 shows the result in our context.

As can be seen already in the design plots, the type of specification is the most influential factor in both contexts. Almost all teams, regardless of method and round, had a higher defect detection rate in the WLMS specification. We can furthermore observe that the patterns are very similar in both contexts, with only one exception. As regards the detection method, the Scenario method is favoured the first round and the Checklist method is favoured the second round.

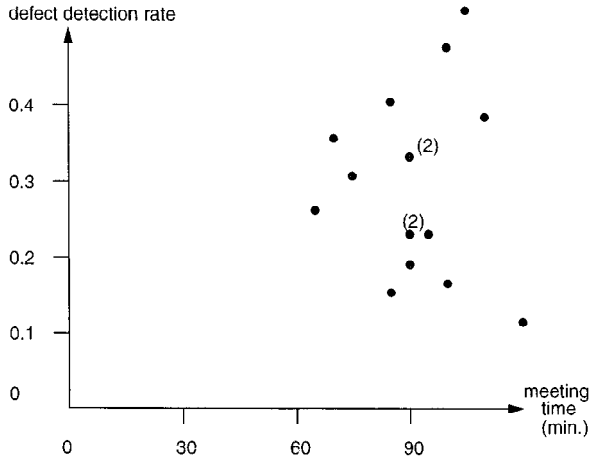


Figure 2. The defect detection rate in the original context is plotted against the collection meeting time. As can be seen, there is no correlation between time and performance.

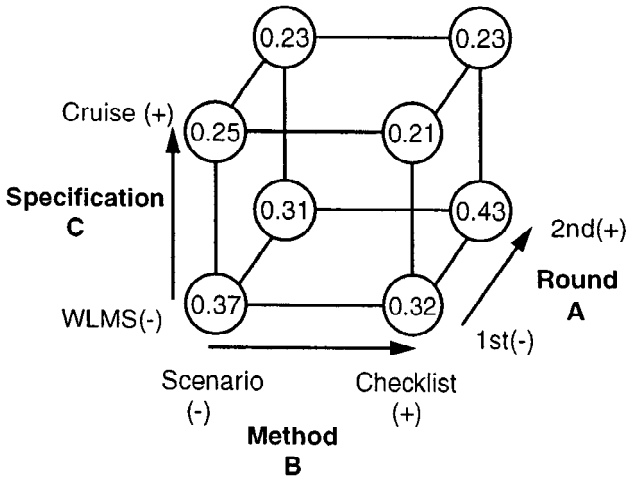


Figure 3. Result in the original context. Each corner of the design-plot illustrates a unique combination of independent variables. The numbers in the circles are the average values of defect detection rate for the teams which had the same combination of independent variables. Refer to Figure 1 for team numbers.

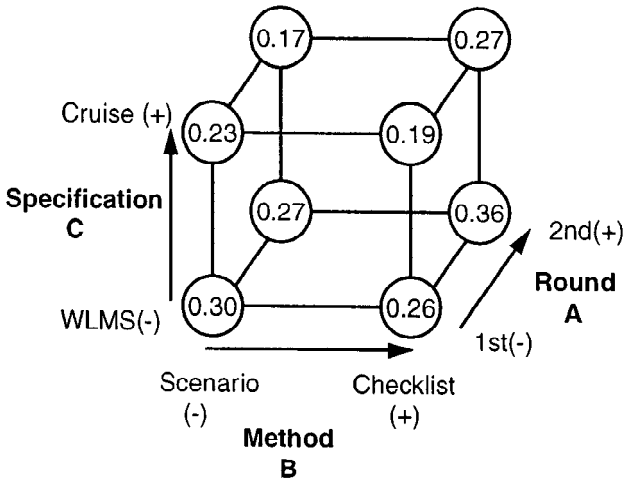


Figure 4. Result in our context. Each corner of the design-plot illustrates a unique combination of independent variables. The numbers in the circles are the average values of defect detection rate for the teams which had the same combination of independent variables. Refer to Figure 1 for team numbers. We observe that the numerical relations between the values in two corners connected with an edge are similar to the ones in Figure 3. The only exception is the teams inspecting the CRUISE specification in the 2nd round. The have equal defect detection rate in Figure 3. Here there is a higher value for the Checklist method.

To establish if these patterns were statistically significant or not, a statistical analysis of the experiment was performed in the original context and with the "added defects" from our context.

The statistical properties of the dependent variable, defect detection rate, is not quite clear. One possible strategy of analysis could be to apply the binomial distribution, $Bi(n, p)$, and transform the dependent variable to equal variance (Bisgaard and Fuller, 1994). This approach implicates that each defect has the same probability of being detected which is an assumption that can be questioned. The fact that the total number of defects (n) differ both between specifications and contexts make the situation even more complicated. Furthermore the n's are not predetermined constants, but obtained as a result of the experiments.

Resolving these issues is not in the scope of this article, but ought to be addressed in future research to take full advantage of the results.

The approach chosen here is an exploratory compromise. First an ANOVA will be applied to investigate the main effects and interactions of untransformed data. This corresponds to the columns 2–8 in the analysis matrix shown in Table 2. Columns 2–4 correspond to the table in Figure 1, and interactions are given in columns 5–8. We will use the fact that the design allows us to estimate all main effects and interactions independently. Since we suspect that the error distribution is not normal and also recognise that there are no degrees of freedom available to estimate pure error, the p-values should only be regarded as indicators. As a follow-up analysis we will utilise all the degrees of freedom in the experiment and

Table 2. Analysis matrix. For each experiment we list the value of 16 different contrasts. M is the intercept. A, B and C corresponds to the independent variables Round, Method and Specification respectively. The frame corresponds to the experiment design in Figure 1. AB, AC, BC and ABC are the possible interactions between independent variables. D_{ij} correspond to the difference between teams using the same combinations of independent variables. DD_{ij} are the change in performance difference between two teams using the same combinations of independent variables.

column		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
exp.	team	M	A	B	C	AB	AC	BC	ABC	D_{12}	D_{34}	D_{56}	D_{78}	DD_{12}	DD_{34}	DD_{56}	DD_{78}
1	1	1	−1	−1	−1	1	1	1	−1	1	0	0	0	1	0	0	0
2	2	1	−1	−1	−1	1	1	1	−1	−1	0	0	0	−1	0	0	0
3	3	1	−1	−1	1	1	−1	−1	1	0	1	0	0	0	1	0	0
4	4	1	−1	−1	1	1	−1	−1	1	0	−1	0	0	0	−1	0	0
5	5	1	−1	1	−1	−1	1	−1	1	0	0	1	0	0	0	1	0
6	6	1	−1	1	−1	−1	1	−1	1	0	0	−1	0	0	0	−1	0
7	7	1	−1	1	1	−1	−1	1	−1	0	0	0	1	0	0	0	1
8	8	1	−1	1	1	−1	−1	1	−1	0	0	0	−1	0	0	0	−1
9	7	1	1	−1	−1	−1	−1	1	1	0	0	0	1	0	0	0	−1
10	8	1	1	−1	−1	−1	−1	1	1	0	0	0	−1	0	0	0	1
11	5	1	1	−1	1	−1	1	−1	−1	0	0	1	0	0	0	−1	0
12	6	1	1	−1	1	−1	1	−1	−1	0	0	−1	0	0	0	1	0
13	3	1	1	1	−1	1	−1	−1	−1	0	1	0	0	0	−1	0	0
14	4	1	1	1	−1	1	−1	−1	−1	0	−1	0	0	0	1	0	0
15	1	1	1	1	1	1	1	1	1	1	0	0	0	−1	0	0	0
16	2	1	1	1	1	1	1	1	1	−1	0	0	0	1	0	0	0

use the normal probability plotting technique (Box et al., 1978). This way we can eyeball the information in the experiment in a more complete way and avoid problems with the multiple test effect.

To utilise all degrees of freedom we construct eight more *contrasts* that are interpretable and apply an ordinary regression analysis. The regression coefficients of all predictors are then estimated using ordinary least squares. The new eight contrasts measure noise under the assumption that the difference between teams is random with an expected value of zero.

In Table 2 columns 9–12 correspond to D_{ij} which measures the difference in performance between teams i and j where i and j replicate the same experimental conditions, see Figure 1. If, for instance, D_{12} is significant, this means one of the teams 1 or 2 has a significantly better performance than the other.

The columns 13–16 correspond to DD_{ij} which measures the change in performance difference between the teams i and j comparing the two experimental conditions. If, for instance, DD_{12} is significant, this means that the difference in performance between team 1 and 2 significantly changes between the experimental conditions. Informally, this can also be viewed as an interaction between team performance and experimental conditions. The DD_{ij}'s are very unlikely to be active.

This choice of contrasts is not unequivocal since there are many octets of linear combinations, mutually orthogonal that could be constructed by the chosen contrasts. Our choice is guided by the ease of interpretation.

Relying on the Central Limit Theorem and an assumption that the variance is approximately constant between the experimental runs, the regression coefficients are normally distributed. Since all columns in Table 2 are orthogonal, the regression coefficients can

Table 3. ANOVA for the original context. The P-value column in the bottom table shows the significance levels at which each factor explains the variance in defect detection rate. For instance there is a probability of 0.075 that the effect from specification, C, is explained by pure chance. Normally a level under 0.05 is required to show significance.

Regression statistics	
Multiple R	0.636
R Square	0.405
Adjusted R Square	−0.116
Standard Error	0.123
Observations	16

ANOVA

	df	SS	MS	F	Significance F
Regression	7	0.083	0.012	0.777	0.624
Residual	8	0.122	0.015		
Total	15	0.205			

	Coefficients	Standard Error	tStat	P-value
M (Intercept)	0.294	0.031	9.524	0.000
A	0.006	0.031	0.193	0.852
B	0.004	0.031	0.134	0.897
C	−0.063	0.031	−2.047	0.075
AB	0.026	0.031	0.831	0.430
AC	−0.006	0.031	−0.193	0.852
BC	−0.014	0.031	−0.445	0.668
ABC	−0.016	0.031	−0.519	0.618

be estimated independently. After a standardisation of the regression coefficients to equal variance (here experimental error) they can be plotted on normal probability paper. From the pattern in the plot, we can judge if any of them differs significantly from the rest (Box et al., 1978).

The ANOVA-table associated with the original context are displayed in Table 3. As is easily seen, none of the independent variables and their interactions seems to explain the variance in defect detection rate.

The complementary normal probability plot is displayed in Figure 5. Here three of the standardised regression coefficients might be considered as deviating from the reference distribution, which correspond to a straight line through $(0, 0)$ aligning to a majority of points in Figure 5. Only one of these is associated with an independent variable of the experiment, C-Specification. The other two detect differences between team 5 and 6 and team 3 and 4 respectively.

It should be noted that the specification, C, which has the largest effect, is suggesting a result that is in agreement with the originators. In contrast, the effect of the method,

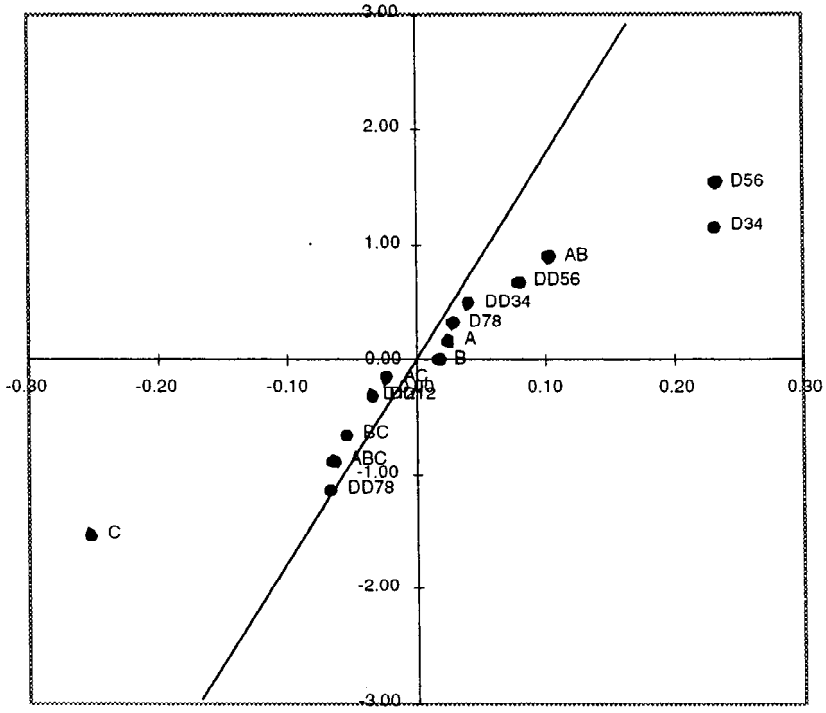


Figure 5. Normal probability plot of standardized regression coefficients corresponding to the contrasts in Table 2. The vertical axis shows the number of standard deviations. The horizontal axis shows the regressions coefficients in the original context. The reference variance is a straight regression line through (0, 0). The effects of contrast far off such a line can not be explained by chance. In the figure contrasts C, D_{56} and D_{34}, have a significant effect.

B, does not stick out at all, which is in disagreement with the originators. The result also suggests that there are differences in performance between the groups that are not negligible compared to the experimental error.

The lack of agreement between the ANOVA and the normal probability plotting technique can be explained by the fact that the ANOVA utilises an error term which contains variation that can be explained by difference in group performance. An ANOVA utilising only the DD_{ij}'s in the error term gives results in agreement with the normal probability plot. An analysis of the residuals showed no anomalies.

To study the effects of our context, we made a normal probability plot of the defect detection rate of the defects that are exclusively counted in our context. The plot is displayed in Figure 6 and, as can be seen, the regression coefficients are almost on a straight line. From a statistical point of view, no factor can explain the variability in the detection rate of the added defects, which makes it less interesting to make a complete analysis in this context.

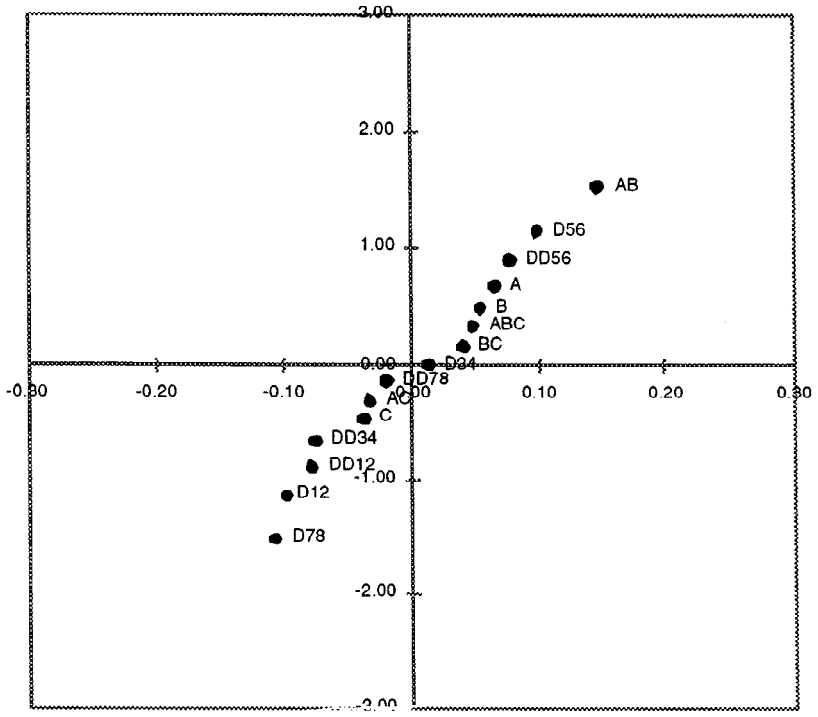


Figure 6. A normal probability plot for the contrasts. The dependent variable is the defect detection rate of our added defects only. As can be seen, no contrast have significant effect. Adding defects only add noise to the data.

3.3. Result of Discussion Seminar

At the discussion seminar following the preliminary analysis the students did not doubt the result and volunteered information about the drop of motivation the second time, and their aversion to the SCR notation. In response to a direct question they stated that they worked differently using the Scenario and Checklist methods, which indicates that the treatment had at least had an intuitive effect. Generally they were more in favour of the Checklist method since it was simpler to use and the individual was in control of the entire task. However, the overall impression was favourable and they appreciated the practical training obtained and contact with research.

4. Discussion

The originators had also observed that the requirements specification explains variation at a significant level, so did also the replication reported from the University of Bari (Fusaro

et al., 1997). Our normal probability plot did also reveal a significant level of specification. In the experiment the explanation can be sought in the nature of the specifications. The CRUISE specification has some major deficiencies that the WLMS specifications avoids:

1. The entire CRUISE system is too naive when it comes to the measurement of physical parameters and control of the throttle. WLMS has a more honest attitude as a laboratory system.

2. The CRUISE system tries to solve several tasks, both throttle control and mileage and fuel reporting. The purpose of WLMS is much more clear and concentrated.

3. Much of the functionality of CRUISE is buried in the use of the recursive nature of the glossary. WLMS is described more in terms of a series of state machines, which is easier to visualise when running the scenarios.

The resulting hypothesis from this is that it is easier to find defects in a good and clear specification. The challenge is to define good and clear precisely enough to allow for a future experiment.

The replication from Bari generated a series of hypotheses about why the Scenario approach did not increase the overall effectiveness of the inspections (Fusaro et al., 1997):

1. Subjects were asked to learn too many new things.

2. Defects in the introductory parts create confusion.

3. Training was unfair.

4. Subjects who had trouble with the Scenario approach used different techniques to execute the task

5. Time limit was too short.

Since we conducted our replication before the Bari replication was published, we could not attain these formally. Our experience corroborates the first three: our subjects did not like the SCR-notation at all; very few had driven a car with cruise control; and no practical training in the methods was done before the experiment. Similarly to the Bari replication, our students had a lower defect detection ration defects than the originators.

Before we ran the experiment, we tried the material ourselves. Some of us had problems in following the incorrect functionality scenario, also reported from the Bari replication. Due to this we spent extra effort in constructing invariants in the training of the Scenario method.

The time limit was never mentioned as a problem from the students. All students used 2.5 hours for individual detection. Some teams ended their collection meetings early. We found no correlation between time spent in meeting and the number of defects found. Even though we did not measure meeting gain, this indicates that a collection meeting is not the most important activity.

Our observations and the Bari replication boils down to the hypothesis that in order to perform a good inspection you need proper training, an understanding of the task of the

system, a readable document, enough time and familiarity with notations and the role of various life-cycle documents. Our students did not meet all of this requirements.

The originators used more experienced graduate students. A brave hypothesis is that you have to reach a certain level of skill before dividing responsibility shows a significant difference. The individual task skill is also put forward as a salient part of inspections by researchers comparing software inspections with behavioural research (Sauer et al., 1996). If this skill is not developed by the subjects, it is not surprising that an easy-to-read specification gives more help than a specialised scenario.

The extension of the defect definition did not reveal any surprises. This is natural since there was no support for detecting added defects in the methods used. This shows that the original instruments are well-focused to its task, which corroborates the assumption that independent replications of this experiment are feasible. Had we seen the opposite we could always claim that the instruments are so sensitive to context that replications can be questioned.

5. Conclusion and Future Work

We have reported a replication of a controlled experiment using the same instruments but in another educational culture. We have extended the experiment with a larger set of specification defects. The added defects focus on the likelihood that defects of the specification as a document will cause problems and delaying the work in the sub-sequent life-cycle phases.

Our design and analysis technique are different from other instances of this experiment. With this in mind, our results indicate that the specification is the most significant explaining factor of the variance amongst the independent variables.

Similarly to the replication from Bari, we could not support the originators' preference of the Scenario-based method. We also note that the average defect detection rate is lower in our and the Bari replication. A hypothetical explanation supported by both replications is that the undergraduate subjects had not developed their individual inspection skills enough to take full advantage of the more sophisticated techniques suggested in the Scenarios.

We believe that we have observed how novice inspectors finds more defects in a a specification which can reveal interesting properties of the product in an understandable and natural way, regardless of inspection method. In the best of worlds the readability of the product and the skill of the inspector coincide.

Applying methods gives a variation which can not be explained by pure chance. This removes the suspicion that the treatments had no effect. The significant difference in performance between two pair of teams shows that it is sometimes hard for novices to understand and take full advantage of a new method.

Of course, the most interesting question is what happens when the experiment is replicated using professional subjects. We hope that our work will demonstrate the feasibility and utility of replicated experiments to professional organisations in order to obtain the necessary interest and resources.

6. Epilogue

Amongst the students participating in the experiment three master thesis projects in related areas were initiated—a positive side-effect of the close contact between senior students and industrial research. Two of the students are now enrolled as graduate students in empirical software engineering. For educational purposes it would be of great benefit if there were a number of "classical" experiments that could be easily replicated and modified. One suggestion would be to rate the severity of the defects in order to estimate the obtained value of the inspection and to let the students participate in the analysis, thus also teaching an experimental approach to their future profession.

Appendix A: Anomalies Found in the CRUISE Specification

ID	location(s) page:line	explanation	number in originators list
1	1:1	The case when the car is going in the reverse direction is unspecified	0
2	2:1	Document index is missing	0
3	2:4	The customer is excluded from intended readers	0
4	2:6	The user benefits are excluded	0
5	2:6	The throttle should not be controlled at low speed	1
6	2:6	Automatic gear is assumed	0
7	2:16	"previous speed": ambiguous reference	0
8	2:24	Information about what happens when user accelerates is needed	0
9	3:3	"Average fuel consumption for a tank of gas" is meaningless	0
10	3:4	Missing information about light indicators	2
11	3:5-7, 19:12	RHS's are inconsistent with sub-sequent terminology	0
12	3:5-7, 11:7	Dependencies between maintenance activities are disregarded	0
13	3:11	The h/w examples are inconsistently named and untraceable	0
14	3:20	"ignores the brake" is not an intuitive phrase, maybe wrong	0
15	4:2	The interaction with the external environment is unclear, e.g. how are the constants accessed	0
16	5:1	Mode description missing	0
17	5:6	Inconsistent terminology	0
18	5:11, 7:13	Resume not specified	0
19	5:15	The user can not decrease the speed under system control	3
20	7:2	All input sections have a generic undetailed text	0
21	7:13	The function is not traceable to a need by the user	0
22	7:14	"as soon as": to vague	0
23	7:22	The list of cases is incomplete	0
24	7:22	A control algorithm for keeping constant speed missing	0
25	8:1	Several device failures are missing	6
26	8:4-5	Should be!desired speed!	4
27	8:7	Setting //throttle// to 0 will only have a temporal effect	5
28	8:17	The constraint untestable	7

ID	location(s) page:line	explanation	number in originators list
29	9:17, 11:21, 13:22	The sentence can lead to confusion	0
30	9:13, 11:17, 13:17	Reset should be possible in *off* mode	8
31	9:13, 11:17, 13:17	The case when the car rolls downhill without engine on is not treated	0
32	9:21, 11:25, 13:26	The use of the intermittent value is unclear	0
33	9:22, 11:26, 13:28	Why 4950 instead of 5000 etc.? Design Rationale missing	0
34	10:3, 12:3	Should be /maint reset/	9
35	13:18	"mode*" should be mode	0
36	15:15	How and by whom is !min trip time! set?	0
37	15:18	There is no $fail$ value	10
38	15:28, 16:24	Accuracy inconsistent with 3.2.2.2.1	11
39	16:14	Undefined value immediately after fill-up	12
40	17:11	The processing description lacks information about !const pos!	0
41	17:11	Conditions for remaining in modes are missing	0
42	17:15	Initial values are unspecified	13
43	17:16	"=$true$" is missing	0
44	17:23	What if the gear is shifted manually?	0
45	17:25	The transition from *inactive* to *override* not specified	0
46	17:26	What if the accelerator is released?	0
47	17:26	A non-deterministic transition possible	14
48	17:28	Confusing misspelling in safety-critical part	0
49	19:1	All items lack traces to other parts of the document	0
50	19:9	All switches should be reset to 0	15
51	19:10	The services should be split in three descriptions	0
52	19:21, 20:2	"User override the control with accelerator" does not happen	16

ID	location(s) page:line	explanation	number in originators' list
53	19:26, 23:16	Pedal is sometimes referred to as accelerator	0
54	20:11	The formulation of the text leads to confusion regarding the relation between the system and its impact on the speedometer	0
55	22:2	The name "trip average speed" leads to confusion	0
56	23:16	Missing resolution	18
57	23:16	The relation to //throttle// unspecified	19
58	23:18	Ambiguous angle	17
59	24:17	2^16 for shaft rotation is too little	0
60	24:20	Shaft rotation does not have $fail$	0
61	25:2	Use of /Time/ is not specified	0
62	25:5	The system is likely to exist beyond the year of 2036	0
63	26:4	Should be //oil message//	20
64	27:9	In range of non-failure values	21
65	29:1	Too much functionality in glossary	0
66	29:6	The calculation too naive	22
67	29:9, 24:15	Missing unit, missing calculation	23
68	29:17	Missing unit, missing use	24
69	29:16	Redundant abbreviation !fuel amount!, there is a /fuel amount/ port	0
70	30:14	Should be "+" instead of "-"	25
71	30:25	How to calculate number of miles travelled?	0
72	30:16	!stop incr! is never used	0
73	30:26	Missing unit	26

Appendix B: Anomalies Found in the WLMS Specification

ID	location(s) page:line	explanation	Number in originators' list
1	1:1	Failure handling missing in entire spec	0
2	2:1	Missing index	0
3	2:1	Missing contents description	0
4	2:4	User/Customer benefits excluded	0
5	2:13	The multiple use of "=" for both assignment and equality is not specified	0
6	3:9	Only modes within a mode class are mutually exclusive	1
7	4:16	The relay is not shown in Fig. 3	2
8	5:1	Several h/w devices in Fig. 2 are not specified	3
9	5:2	Figure 2 should be referred	0
10	5:29	The computation of memory failure is not specified, h/w or s/w?	0
11	5:18	"WLMS operation": not clarified	0
12	6:14, 18:18	ALARM referred to as SPEAKER	4
13	6:22	Pump rates can be negative	42
14	6:22	!Max level rate! is wrongly typed	0
15	7:3	There is no timing information, e.g. call-time, duration	5
16	7:5	Several devices are not initialized	7
17	7:16	%watchdog% should be set to $operate$	6
18	8:16	Inputs are missing in the list	9
19	8:17, 14:6, 15:6, 16:6, 17:6, 25:8, 25:9, 25:19	Should be /currlevel/	8
20	8:20	%time% is not used	0
21	9:1	The mode description is too short and unclear, will delay future work	
22	9:4	The table is not complete	0
23	9:5	Vertical bars are missing and tabulation not clear	0
24	9:6	Deadlock can occur	10

ID	location(s) page:line	explanation	Number in originators' list
25	9:7	Deadlock can occur	11
26	9:8	Simultaneous transition possible	12
27	9:9	Pressing /slftest/ does not start h/w test	13
28	9:12	Missing parenthesis	0
29	9:13	Should be /slftest/ = $released4	14
30	9:18	Should be inStnd2	15
31	9:27, 9:28, 9:29	Input devices written as application variables	16
32	10:5	%time% is lacking in the list	17
33	10:5	Watchdog is not specified as input	0
34	10:6, 21:23	//dogcmd// is an input port	32
35	10:9	"The update" is an unclear reference	0
36	10:13	//dogcmd// can not take $init$	18
37	10:16	No timer for watchdog (inconsistent)	0
38	11:10	The time limit is too long (gives an unclear impression)	0
39	11:15	The use of set operator is not defined and inconsistent with rest of the specification	0
40	12:4	Shutdown is not specified as input	0
41	12:8	Timing specifications missing	19
42	12:8	Two values to //shutdown// possible if /reset/ is released and device fails simultaneously	22
43	12:16	Unnecessary computation	20
44	12:17, 12:18	Event((Enter *x*) and Enter(*y*)) should be Event(InMode(*x*) and InMode(*y*))	21
45	14:13, 15:13, 17:13, 17:19	Event(Enter *x*) is missing *x*	23
46	14:19, 17:19	Missing "*" in safety critical part	0
47	15:10, 16:9, 17:10	//high window// wrongly pasted in text	24
48	16:7	register should not be written in *badlevdev* mode	28
49	16:12	Wrong parameter type	26

ID	location(s) page:line	explanation	Number in originators' list
50	16:15	Values out of range	25
51	16:19	Should be >=	27
52	16:19	What is the rationale for this formula?	0
53	17:8	Information about *hardfail* is missing	30
54	17:14	Alarm should sound when not !Within limits!	29
55	18:1	The SRS lacks information about RESERVOIR, P1, P2, SPEAKER, RELAY, PUMP SWITCH	0
56	18:1	All variables lack init-value and cross-references to functional req's	0
57	19:15	Lacks type, resolution, tolerance, units	31
58	19:20	Lacks timing information	0
59	19:23	H/w called "RESET" in overview	0
60	21:28	Missing semantics of "high voltage"	0
61	22:2	Lacks type, resolution and semantics	0
62	22:18	Should be //low window//	33
63	22:20	Lacks specification of the alarm signal	0
64	23:6	The POWER button must be better described	0
65	23:9	%Time% lacks upper limit	0
66	23:16	//DOGCMD// should be written with low-case letters	0
67	25:2	No units specified	34
68	25:2	Inconsistent with glossary notation	0
69	25:3	The argument should be "y"	35
70	25:7	Incorrect formula	36
71	25:8	RHS transposed with next line	37
72	25:12	Inconsistent units	38
73	25:18	!Water tolerance! never used	39
74	25:19	Missing "!"	0
75	25:19	Both formulas can be satisfied simultaneously	40
76	Fig. 2	Several STATION devices missing	41

Acknowledgments

This work has been sponsored by the Department of Computer and Information Science, Linköping University and Ericsson Radio Systems AB. The authors wish to express their

gratitude to all students participating in the study. The English was improved by Ivan Rankin. Invaluable comments were also given by anonymous reviewers of this journal.

References

Barton, R. R. 1998. Design-plots for factorial and fractional-factorial designs. *Journal of Quality Technology* 30(1): 40–54.

Bisgaard, S., and Fuller, H. T. 1994. Analysis of factorial experiments with defects or defectives as the response. *Quality Engineering* 7(2): 429–443.

Boehm, B. W. 1987. Industrial software metrics top 10 list. *IEEE Software* 4: 84–85.

Box G. E. P., Hunter W. G., and Hunter J. S. 1978. *Statistics for Experimenters. An Introduction to Design, Data Analysis, and Model Building*. New York: John Wiley & Sons.

Fusaro, P. F., Lanubile, F., and Visaggio, G. 1997. A replicated experiment to assess requirements inspection techniques. *Empirical Software Engineering* 2: 39–57.

Heninger, K. L. 1980. Specifying software requirements for complex systems: New techniques and their application. *IEEE Transactions on Software Engineering* SE-6: 2–13.

IEEE Std. 830. 1984. *IEEE Guide to Software Requirements Specifications*. New York: Institute of Electrical and Electronic Engineers.

Porter, A. A., and Votta, L. G. 1994. An experiment to assess different defect detection methods for software requirements inspections. *Proceedings of the 16th International Conference on Software Engineering*. Sorrento, Italy, 103–112.

Porter A. A., Votta L. G., and Basili V. R. 1995. Comparing detection methods for software requirements inspections: A replicated experiment. *IEEE Transactions on Software Engineering* 21: 563–575.

Sauer, C., Jeffery, R., Lau, L., and Yetton, P. 1996. A behaviourally motivated programme for empirical research into software development technical reviews. Centre for Advanced Empirical Software Research, The University of New South Wales, Technical report 96/5.

Kristian Sandahl is member of ZeLAB, the systems engineering research laboratory at Ericsson Radio Systems in Linköping, Sweden. He is also part-time associate professor at the department of Computer and Information Science at Linköping university.

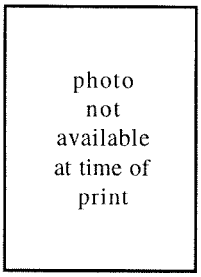
In 1992 he received a PhD degree in computer science at Linköping University and founded a research group with the goal of performing empirical studies in industrial software and knowledge engineering. Since 1995 dr. Sandahl has been employed by Ericsson with a special responsibility of facilitating the cooperation with the academic world in the fields of Computer-supported cooperative work and inspection methods.

His major interests are knowledge management, knowledge engineering, CSCW, industrial software engineering, quality improvement paradigm, inspection methods and empirical research methods.

Ola Blomkvist is a PhD candidate at the Division of Quality Technology and Management, Linköping University, Sweden. His main area of research is Design of Experiments and Robust Design Methodology.

In 1993 he received his MSc degree in Applied Physics and Electrial Engineering and was employed at ABB, ASEA Brown Boveri, in Ludvika, Sweden. Since 1994 he is absent on leave from ABB to finish his PhD degree.

Joachim Karlsson is cofounder and managing director of Focal Point AB, a leading provider of methods and tools for managing and prioritizing software requirements. He works with major companies in Sweden to improve their processes for managing and prioritizing requirements. His main research interests are requirements engineering, product management and project management. Karlsson received his MSc and PhD in computer science from Linköping University, Sweden.



Christian Krysander is assistant professor in computer science at the Department for computer and information science at Linköping university. He has served as tutor in computer science, software engineering, software maintenance, software quality and leadership for 25 years. In his teaching, Christian Krysanders uses a problem-based pedagogy, which have fostered the curiosity of an entire generation of Swedish software developers.

Mikael Lindvall recently left his position as assistant professor in Software Engineering, Processes and Methods, at the Department of Computer and Information Science at Linköping University, Sweden, to become project manager at Sapient corp. in San Francisco.

In 1984 he founded and managed a 20 persons company in CAD applications programming. Increasing interest in principal Software Engineering questions based on experiences with programming projects made him also start an academic career.

In 1991 he received a MS degree in computer science and engineering. In 1997 he received a PhD degree in computer science and especially software engineering. Both from Linköping University, Sweden.

The Ph.D. thesis is based on an long-term empirical study of an industrial object-oriented project at Ericsson conducted over several releases. Dr. Lindvalls interests are Quality aspects of Software Development methods, Software Evolution, Object-Orientation, Requirements Engineering, Impact Analysis and Traceability.

Niclas Ohlsson received the M.S. degree in computer science from Linköping University Sweden, in 1993 and his PhD degree in 1998. His current research is conducted in close collaboration with Ericsson Telecom AB, SAAB Military Aircraft and Swedish Defence Material Administration (FMV), and aims at developing and evaluating methods for software fault prevention. He has especially focused on prediction of fault-prone modules in large software systems. Dr. Ohlson has also focused on development and empirical evaluation of an integrated defect analysis process for software quality improvement. The objective of the research is to move efforts from fault detection to fault avoidance, and also to provide data for more complete quantitative analysis of project results. His interests also include software quality engineering, continuous process improvement, learning organisation and software metrics.