A partial replication of Thelin et al.'s usage-based reading experiment

Austen Rainer and Are Oppegård Pedersen
School of Computer Science
University of Hertfordshire
College Lane Campus
Hatfield
Hertfordshire AL10 9AB
U.K.
a.w.rainer@herts.ac.uk; arep@uio-pop.uio.no

26th October 2004

Computer Science Technical Report CS-TR-415

In this paper, we partially replicate Thelin et al.'s investigation of usage based reading (UBR). Our study design is very similar to Thelin et al.'s., except we were unable compare the use of UBR with a *second* reading technique. Instead, seven subjects used UBR to inspect the same design document as used in Thelin et al.'s study, under the same experimental conditions. We ask the general research question: How does the performance of the subjects in our study, using UBR, compare with the performance of those subjects using UBR in Thelin et al.'s study? We find that the subjects in our study perform slightly better than the comparable subjects in Thelin et al.'s study. There is some indication of better performing and poorer performing subjects. There is also some indication of easier and more difficult faults to detect.

"What can you learn from one of anything? Why, all you can."
~ Harry Wolcott

1 Introduction

Broadly speaking, the purpose of a software inspection reading technique is to help inspectors identify more faults, and to do so in a shorter period of time. A range of reading techniques have been developed over the years. These include Perspective Based Reading (PBR), Defect Based Reading (DBR), Checklist Based Reading (CBR), Traceability Based Reading, Usage Based Reading (UBR), and ad hoc reading. The ad hoc reading technique is not really a technique, as the principle of ad hoc reading is that inspectors use their own judgement in deciding how to inspect a software artefact. By contrast, the other reading techniques provide guidance, in one form or another, on how to inspect software artefacts. Phrased another way, all reading techniques seek to improve on the 'baseline performance' of the ad hoc reading technique.

Perspective Based Reading has probably been the most researched inspection technique (see [1] for a summary of research on PBR). PBR has been compared with Checklist Based Reading and with ad hoc reading. Most of this research has been conducted in academic environments, using students as subjects. The results from these studies are inconclusive, with some studies finding a significant difference between PBR and other techniques, and with other studies finding no significant difference. Two studies identified by Thelin et al. that have been conducted within an industrial environment have both found a significant difference between PBR and the other techniques[1].

[1] We are not entirely convinced that these two studies were conducted within an *industrial environment*. For some of the subjects in these studies, although they were professionals they were currently attending a programme of academic study. This raises the question as to whether these studies really were conducted within an industrial environment, as opposed to an academic one, and whether results from these studies can be applied to inspections that occur in industry.

A more recent technique to emerge is the Usage Based Reading (UBR) technique. The Usage Based Reading technique is designed to identify those defects that have the most disruptive impact on the users of the software, based on the users' perception [1]. The users' perception of the system's quality is specified by prioritising use cases. Use cases are constructed and prioritised *before* the inspection(s), and the prioritisation should be done by actual or potential users. Inspectors of a software artefact then examine the software artefact by manually executing the use cases against the software artefact [1].

A small number of studies have been conducted on UBR [1-3], the most significant being Thelin et al.'s comparison of UBR with Checklist Based Reading [1]. In this paper, we report on the conduct of a partial replication of Thelin et al.'s study, and we do so in order to provide some independent findings on the performance of the UBR technique. (The data collection and initial analyses were conducted by Pedersen and are reported in [4].) Due to practical limitations, we were unable to compare *two* reading techniques in our study. As a result, we are unable to directly examine the exact hypotheses investigated by Thelin et al. Instead, we have a general research question *viz.*

RQ1: How does the performance of the subjects in our study, when using UBR, compare with those subjects in Thelin et al.'s study?

We provide more detail on the specific comparisons, and our specific hypotheses, later in this paper.

The remainder of this paper is organised as follows. In section 2, we briefly review Thelin et al.'s experiment. In section 3, we describe the design of our study and identify significant differences between our study design and Thelin et al.'s design. In section 4, we characterise the subjects used in our study. These characteristics are useful for our subsequent analysis. In section 5, we provide a summary of the basic results from our study. In section 6, we discuss the effectiveness of individual subjects in our study, and in section 7 we discuss the efficiency of the individual subjects. In section 8, we consider the subjects' opinions of our study. In section 9, we compare some of the findings of our study with those of Thelin et al. In section 10, we consider some implications and consequences of the two studies. Finally, in section 11 we offer some brief conclusions.

2 A review of Thelin et al.'s experiment

In Thelin et al.'s experiment, the Usage Based Reading (UBR) technique was compared to the Checklist Based Reading (CBR) technique to determine which of these two techniques was the more effective and efficient in finding three types of faults in a software design document. In Thelin et al.'s experiment, 23 fourth-year software engineering masters students from Bleking Institute of Technology in Sweden participated as subjects. The experiment was an obligatory part of a software verification and validation course on which these students were enrolled. The course included lectures and assignments related to verification and validation of software products, and evaluation of software processes.

Before the experiment the students were divided into two groups. One group of 11 students was assigned the UBR technique in the experiment, and the other group of 12 students was assigned the CBR technique. The groups were divided in such a manner that the experience factor of the two groups was blocked out.

The experiment was run over two days during spring 2001. On the first day, all of the subjects had a 45 minute general introduction to a taxi management system, which was the system they were going to inspect for faults. After the introduction, each of the two groups had an additional 45 minute introduction to the reading technique that they were going to use in the experiment the following day. During the introduction to their respective reading techniques, the subjects also used their reading technique in a training exercise.

On the second day, the experiment was conducted. The subjects had 2 hours and 45 minutes to find as many faults as they could in a design document of a taxi management system. To aid their inspections, each student had a requirements document. Depending on which reading technique a student was assigned to use, the student also had either a use case document or a checklist.

The requirements document was only supposed to be used as a reference to show how the system was meant to work. The subjects used the use cases or the checklist to guide their reading when trying to find faults in the design document. The design document contained 38 faults, deliberately introduced to the document. The subjects also had an inspection record on which to record information when they found a fault. The subjects received the following instructions:

1. Log the time when you start the experiment.
2. Read the requirements document, and log the time when you are finished.
3. Read the design document and log the time before you start to read and when you are finished reading the design document.
4. Then start to inspect for faults, using either the use cases or the checklist.
5. When a fault is found, log the time it was found, where in the design document it was found, the use case or checklist number which was used to find the fault, and the type of fault found.

Each student was instructed to stop their inspection when they had checked everything, or after 2 hours and 45 minutes. When the experiment was finished, the students handed in their inspection records to be checked for errors or missing data.

After the experiment, the subjects had a 45 minute introduction to the contrasting reading technique i.e. the subjects who used UBR where introduced to CBR and vice versa.

After the data was collected and validated, the time used on preparation, the effectiveness and efficiency of the subjects, and team performance were analysed. The preparation time was the time used by a subject to read the requirements and design document before inspecting for faults. The effectiveness of a student was calculated as the number of faults found. The efficiency of the subjects were measured as faults found per hour. When analysing the team performance, a simulation of the inspection meeting was performed to investigate the performance of the reading techniques if used by a hypothetical group of inspectors. The purpose of the simulation was to find out whether a UBR team, CBR team or mixed team would be the best alternative when inspecting design documents for a software or system.

After the results of the experiment had been analysed, a debriefing session was held with the subjects. The session included a presentation of the results from the experiment and a discussion about those results.

3 Study design

3.1 Overview

Because this study was intended to be a replication of Thelin et al.'s study, the study design is based very closely on the design described in [1]. Where relevant, we identify significant differences in the designs of the two studies.

3.2 Research question

Because we were unable to compare *two* reading techniques in our study, we were unable to examine the exact hypotheses investigated by Thelin et al. Instead, we have a general research question *viz*.

RQ1: How does the performance of the subjects in our study, when using UBR, compare with those subjects in Thelin et al.'s study?

More specifically, we want to:

- Explore whether any subjects are particularly high performing or low performing in the current study. This exploration would help to identify particular sub-groups of subjects, which might be useful for identifying different levels of inspector performance.
- Explore whether there are any particularly easy or particularly difficult faults to find, regardless of their type.
- Compare the effectiveness of subjects in finding faults, and particular types of faults.
- Compare the efficiency of subjects in finding faults, and particular types of faults.
- Examine whether subjects in the current study find the same specific faults as those in Thelin et al.'s study.

Our hypotheses are summarised in Table 1.

Table 1 Summary of hypotheses

#	Statement
$H1_{alt}$	There are no particularly high performing or low performing subjects in the current study.
$H1_{null}$	There are particularly high performing or low performing subjects in the current study.
$H2_{alt}$	There are no particularly easy or difficult faults to find in the current study.
$H2_{null}$	There are particularly easy or difficult faults to find in the current study.
$H3_{alt}$	There is no difference between the effectiveness of Thelin et al.'s UBR group of students, and the effectiveness of the UBR group of students in the current study.
$H3_{null}$	There is a difference between the effectiveness of Thelin et al.'s UBR group of students, and the effectiveness of the UBR group of students in the current study.

3.3 Variables

Table 2 presents the variables used in Thelin et al.'s study, and indicates the differences between Thelin et al.'s study and the current study.

Table 2 Summary of variables used in Thelin et al.'s study and the current study

Variable type	Details of variable(s)	Comments on differences between Thelin et al.'s variables and the current study's
Independent	Reading technique	As Thelin et al. investigated UBR and CBR, their independent variable took two values. The current study only investigates one reading technique (UBR).
	Type of fault	Thelin et al. did not recognize this as a variable in their study. As these faults are seeded by the experimenters, we have treated them as an independent variable. They could be treated as a control variable.
Control	Experience	Information on the subject's experience was collected in both Thelin et al.'s and this study. Thelin et al. allocated subjects to experimental groups to control for experience. With only one group in the current study, we could not control for experience. But this is part of the reason for the research question and hypotheses that we have chosen.
Dependent	Time spent on preparation by each reviewer, measured in minutes	The same types of data have been collected for both studies.
	Time spend on inspection by each subject, measured in minutes	The same types of data have been collected for both studies.
	Break time	Time used as a break from the experiment. Not collected by Thelin et al.
	Clock time when each fault was found by each subject	The same types of data have been collected for both studies.
	Number of faults found by each subject	The same types of data have been collected for both studies.
	Number of faults found by each experimental group	The current study has only one experimental group (UBR), in contrast to Thelin et al.'s two experimental groups.
	Efficiency, measured as: $60 \times \frac{\text{Number of faults found}}{\text{Total time (min.)}}$	Thelin et al. investigated the efficiency of the group. The current study investigates the efficiency of the *individual subjects*. For the current study, efficiency is measured both in terms of *inspection time* only and *total experimental time*.
	Effectiveness, measured as: $\frac{\text{Number of faults found}}{\text{Total number of faults}}$	Thelin et al. investigated the effectiveness of the group. The current study investigates the effectiveness of the *individual* subjects.

3.4 The inspection material

The inspection material for the experiment is the same material that was used by Thelin et al. during their experiment. The material consists of three documents: one requirements document, one design document and one use case document. The requirements document is written in English and shows the requirements for a taxi management system. It was used as a reference document to show the subjects how the system is meant to work.

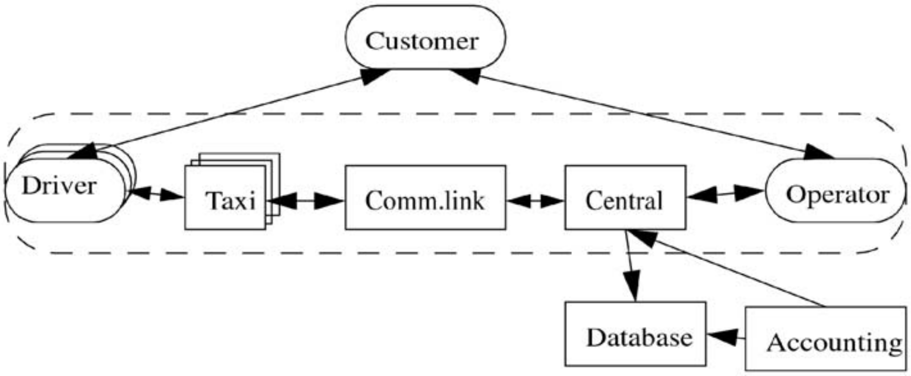

Figure 1 The taxi management system. The rectangles represent software modules and the ovals represent users. The software for the database and accounting system is not described in the complementary requirements document (figure from [1]).

The design document is written in Specification and Description Language (SDL) and consists of two message sequence charts (MSC) [5] (see Figure 1). The MCSs show the signal between the modules in two different cases, one for order handling and one for voice communication. In the design document there are 38 faults deliberately inserted into the document by the experimenters. Out of these 38 faults, 28 of them were made during the development of the document or later found in inspection, 8 of them were planted in by the developer of the system and the last two faults were found during the experiment at Bleking Institute of Technology. In the use case document, there are 24 use cases. These use cases are written in task notation [6] and have been prioritized by using the analytic hierarchy process (AHP) [7] from the users' point of view. In the use cases document, the first use case is the most important and the last use case is the least important. Figure 2 shows an example of a use case.

Taxi: Driving a customer

Purpose: The driver has a customer in the car and is about to transport the customer to the
designated address.

Tasks:

1. Car in state “Available”.
2. Driver receives order. See receive order (use-case 1.5).
3. Car in state “Waiting for customer”.
4. Drive to the pick-up location.
5. Wait until customer arrives.
6. Start meter and transportation.
7. Car in state “Driving”.
8. When driver is confident in time of arrival, send arrival zone and time to central.
9. Car in state “Soon available”.
10. Arrival at destination. Charge customer and print out receipt.
11. Car in state “Available”. The car sends the position (zone).

Variants:

2b. The driver picks up customer without order. Use case starts at step 6.

6b. Customer does not show up. Car is put in state “Available”.

9b. Driver does not use the “Soon available” function. Step 9 is skipped.

Figure 2 An example of a use case written in task notation. This use case describes transporting a customer to a destination (taken from [2])

3.5 Fault classification

Thelin et al. divided the 38 faults in the design document into three types, depending on how important the fault was to the user. Importance is a combination of the probability that the fault would manifest as a failure, and the user’s opinion on the severity of the failure. The three types of faults are summarised in Table 3.

Table 3 Types of faults (defined by Thelin et al.)

Type	Count	Description
A	13	These faults have affected functions that are very important for the user and often used. An example of such a fault is that the user cannot log on to the system.
B	14	Theses faults have affected functions that are important for the user but rarely used or not so important to the user. An example of such a fault is that the user cannot log out of the system.
C	11	These faults have affected functions that are not important for the user. An example of such a fault is that when a user logs off a system, the system do not give a “logging of the system” reply on the screen, but still logs of the user.

Out of the 38 faults in the design document, 13 of them were type A faults, 14 type B faults and 11 type C faults. Syntax and grammatical errors were not counted as faults. If such errors were found by the subjects, they were not included in the analysis.

3.6 The subjects

The people participating as subjects/reviewers in the experiment were seven Computer Science students from the University of Oslo. Of these seven students, five were MSc students and two were BSc students. The MSc students were on their first or last year of their MSc studies, and two of these students also worked in industry during their MSc studies. For the two BSc students, one had just completed the second year of computer science, and the other the third year of computer science. All of the subjects had used use cases, UML and sequence charts [8] in courses before. However, none of the subjects had used the Specification and Description Language (SDL) [9]. Characteristics of the subjects are considered in more detail in section 4.

3.7 Threats to validity

Four sets of threats are considered: conclusion validity, internal validity, construct validity and external validity.

The threats to the conclusion validity are considered to be under control. In this study, we will use some of the same statistical techniques and methods that were used in Thelin et al.'s experiment. One risk to the conclusion validity is that the experiment was conducted on two separate occasions (see section 3.8 for more detail). During the second occasion it is possible that the experimenter was more experienced in explaining and conducting the experiment. However, as a pilot study of the experiment was conducted before the proper experiment was first conducted, the experimenter had gained experience in conducting the experiment.

With regards to the threats to internal validity, there is a risk that some of the subjects could have a lack of motivation doing the experiment. As the experiment was conducted during the summer vacation, some of the subjects could have less motivation for doing the experiment than if it had been conducted during the spring or autumn semester at the University of Oslo (or vice versa). However, all of the subjects have volunteered to participate in the experiment. Therefore, they should have the motivation needed to conduct the experiment in a proper way. The time spent by the subjects to do their inspections and the fact that most of the subjects used all of the available time (see Table 7) suggests that the subjects were motivated.

Another threat to the internal validity of this study is the currency of the subject's knowledge. All of the subjects had gained knowledge of the modeling languages used in the requirements and design documents by attending courses. Some of these subjects, however, could have taken these courses some time ago and, therefore, not remember or effectively use all that they learned about these modelling languages in these courses. In an attempt to address this, all of the subjects were reminded of the modeling languages used in the experiment, in the presentation that occurred the day before the experiment. The subjects also went through a training exercise, in which all of the modeling languages were used.

During the presentation and the training exercise, the subjects could ask any questions. Therefore, we do not think that the subject's knowledge of the modelling languages used is a significant threat to the internal validity of this study.

With regards to threats to construct validity, the requirements document, design document, and use cases are all the same as were used in Thelin et al.'s experiment. However, the requirements document was made after the use cases were made. Therefore, the use cases may have affected the requirements document to be more or less suitable for the use cases. However, the object that was inspected was the design document. The subjects only used the requirements document as a reference.

With regards to threats to external validity, the use of students with such different experiences could affect the results of the study. However, having subjects with such differences may lead to a better contrasting study group with which to compare to Thelin et al.'s group. Another threat to external validity is the design document used in the experiment. The size of the inspected document is rather small range for a real-world problem, even if it describes a real-world problem.

3.8 Operational details of the experiment

The experiment was conducted on the 4th, 5th 28th and 29th of July 2004. The experiment had to be conducted on two occasions because four of the subjects did not have the time to participate on the 4th and 5th of July. These four subjects did the experiment on the 28th and 29th of July. All subjects went through the same introduction and training exercise using the same amount of time. All subjects had the same amount of time (2 hours and 45 minutes) to do the experiment. The timetable for the experiment is shown in Table 4.

Table 4 Timetable for the experiment

Day / hour	Event
Day 1 (The first hour)	An introduction to the taxi management system
Day 1 (The last hour)	Introduction to UBR
Day 2 (15 minutes)	Information about the experiment
Day 2 (2 hours and 45 minutes)	The experiment
Day 2 (20 minutes)	Questionnaire about the experiment and UBR

On the first day of the experiment, the subjects received an introduction to the taxi management system that they were going to subsequently inspect. In this introduction, the subjects were introduced to the documents they were going to use the next day. These documents were the requirements document, the design document, the use case document and the inspection records. The introduction to the inspection records included: a description of the different types of fault classes; an explanation of where to log the time used to read the requirements and design document, and where to log the time used on the inspection; and a fault log where the subjects could log the faults they found during the inspection. The different modelling techniques which were used in the documents were also explained for the subjects.

The subjects were also introduced to UBR, the reading technique they were going to use to find faults in the design document. In this introduction, the subjects also went through a training exercise. In this exercise, the subjects used UBR to find faults in a small software system for a “can machine”. The subjects had about 35 minutes to finish the exercise. When the exercise was finished, the experimenter went through all the faults in the design document for the “can machine” with the subjects, so that they could see how well they had done.

On the second day of the experiment, before the experiment started, the subjects all received the same instructions on how they were supposed to conduct their inspections. These instructions were:

“ 1. The requirements in the requirement document are assumed to be correct. If you [the inspector] find any inconsistency between the requirement document and the design document, the fault is in the design document.
2. Log the time when you start.
3. Read the requirement documents first, maximum 20 minutes.
4. Log the time used on reading the requirement documents.
5. Read the design documents, maximum 20 minutes.
6. Log the time used on reading the design documents.
7. The start inspecting the design documents by using the Use cases.
8. For each fault found, log it in the inspection record.
9. Log the time when you are finished inspecting. The inspection experiment is finished either after 2 hours and 45 minutes or when everything is checked.”

When the introduction was complete, the subjects could start their inspection. They then had 2 hours and 45 minutes to inspect the design document for faults. When all the subjects were finished with the inspection, all the inspection records were handed in to be checked for errors or missing data. After this had been done, all the subjects filled in a questionnaire about UBR, the reading technique they used during the inspection, and the different documents they used during the inspection. They had 20 minutes to answer all the questions in the questionnaire before they handed them in. When the questionnaire was filled out, the experiment was finished. Feedback from the questionnaire is discussed in section 8.

4 Characteristics of the subjects

Prior to commencing the experiment, the subjects each completed a short questionnaire that asked them about their experience. Table 5 and Figure 3 summarise the subjects' responses to the questions asking them about their experience. Table 6 provides an explanation of the responses in Table 5.

None of the seven subjects have used SDL before (question 7). All of the subjects have used use cases (question 5). Subject 4 is the only subject with any industrial experience of testing software (question 6), and is the only subject with any experience of developing taxi systems (question 9). Subjects 2 and 4 are the only subjects with any industrial experience of programming. Overall subject 4 appears to be the most experienced of all the subjects (see Figure 3). Subject 2 is the only subject with *no* experience in inspecting requirements documents (question 3), and subjects 2 and 7 are the only subjects with *no* experience of inspecting design documents (question 4). Subjects 6 and 7 have the least experience in programming (question 1), and subject 7 has never used a taxi (question 10). Overall, subject 7 appears to be the least experienced of all the subjects (see Figure 3).

Table 5 Characteristics of each of the subjects

					Subject			
#	**Characteristic**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	General knowledge of programming	2	2	2	2	2	1	1
2	Industrial experience in programming	1	2	1	2	1	1	1
3	Experience in software requirements inspection	2	1	2	2	2	2	2
4	Experience in software design inspections	2	1	2	2	2	2	1
5	Experience in developing use cases	2	2	2	2	2	2	2
6	Experience in software testing	2	1	2	3	2	1	1
7	Experience in SDL	1	1	1	1	1	1	1
8	Experience in UML, sequence charts and MSC-diagrams	2	2	2	2	2	2	2
9	Experience of developing Taxi Systems	1	1	1	2	1	1	1
10	Experience of using taxis	2	2	2	2	2	2	1

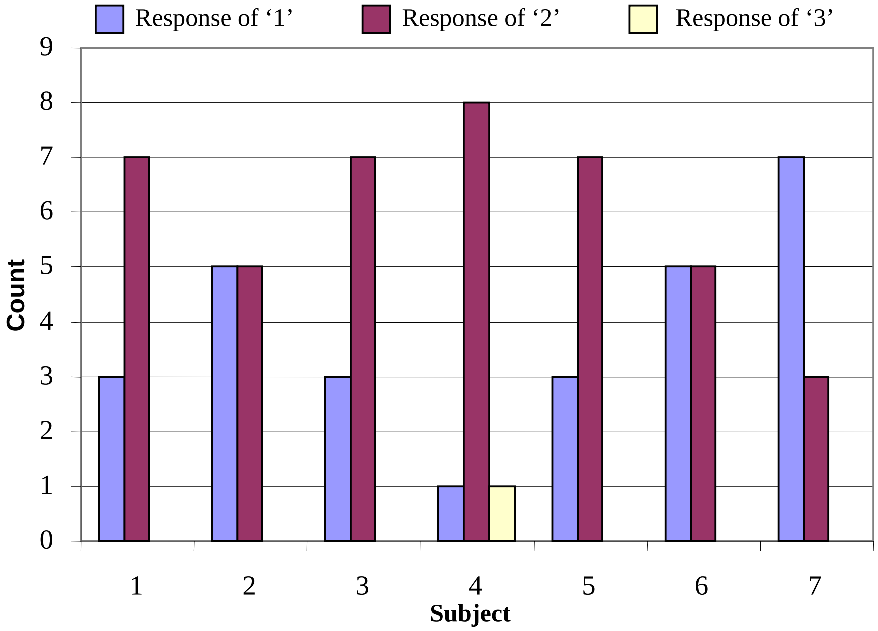

Figure 3 Counts of the subjects' responses

Given the experience of each of the subjects, and noting hypotheses H1, we can speculate that:

- Subject 4 should perform the most effectively and efficiently.
- Subject 7 should perform the least effectively and efficiently.

Table 6 Explanation of the subject's responses

Question	Responses	Explanation of responses
1	1	1 – 2 courses in programming
	2	3 or more courses
2	1	No industrial experience
	2	One year or less industrial experience in programming
	3	More than one year of industrial experience.
3	1	Never inspected a requirements document before
	2	Inspected requirements documents in courses before
	3	Industrial experience in inspecting requirements documents
4	1	Never inspected design documents before
	2	Inspected design documents in courses before
	3	Industrial experience in inspecting design documents
5	1	Never used use cases before
	2	Used use cases in courses before
	3	Industrial experience in using use cases
6	1	Never tested any software before
	2	Tested software in courses before
	3	Industrial experience in testing software
7	1	Never used SDL before
	2	Used SDL in courses before
	3	Industrial experience of using SDL
8	1	Never used UML, sequence diagrams or MSC-diagrams
	2	Used UML, sequence diagrams, or MSC-diagrams in courses
	3	
9	1	No knowledge of the taxi domain
	2	Developed a taxi system 1 – 2 times
	3	Developed a taxi system 3 times or more
10	1	Never used a taxi
	2	Used a taxi at least once
	3	Worked as a taxi driver

5 Basic summary of the results

Table 7 Times taken by each subject, in the experiment

	Subject						
	1	**2**	**3**	**4**	**5**	**6**	**7**
Preparation time, of which:	33	40	30	36	36	35	37
Requirements time	20	20	15	18	17	20	19
Design time	13	20	15	18	19	15	18
Inspection Time	132	122	135	129	105	112	117
Total experimental time	165	162	165	165	141	147	154
Breaks	0	3	0	0	0	18	8
Total time	165	165	165	165	141	165	162

Table 7 presents a summary of the times taken by each subject to do the experiment. If one includes the time spent on breaks during the experiment, then five of the seven subjects used the entire time available. In other words, two hours and 45 minutes appears to be an arbitrary duration for inspecting for these faults, because most of these subjects would have *continued* to look for faults if further time had been available. Against that, there may be a Hawthorn-type effect operating here, where subjects continue to perform precisely because they are in an experiment and are being observed.

In addition to the five subjects that used all the time available, subject 7 used almost all of the time available. It is not clear whether this subject elected to finish slightly early in the belief that they would not find any additional faults in the remaining three minutes of the experiment. Subject 5 appears untypical in that the subject finished considerably earlier (over 20 minutes) than any of the other subjects.

If one excludes the time spent on breaks during the experiment, then three subjects (subjects 1, 3 and 4) still used the total time available to inspect for faults. The subjects with the *most* experience (subject 4) used all of the time available, whilst the subject with the *least* experience (subject 7) used almost all of the time that was available for the experiment.

Table 8 Faults identified by each subject for each fault type

	Subject						
Fault type	**1**	**2**	**3**	**4**	**5**	**6**	**7**
A	8	6	7	7	5	3	2
B	9	3	8	4	7	6	5
C	3	4	5	3	6	3	4
A+B	17	9	15	11	12	9	7
A+B+C	20	13	20	14	18	12	11

Table 8 summarises the faults found by each subject. As with Thelin et al.'s study, the design document contains 13 faults of Type A, 14 faults of Type B, and 11 faults of Type C. So, for example, subject 1 found 8 of the 13 type A faults. Interestingly, the subject who used the least amount of time (subject 5), and hence *chose* to finish early, found the second highest number of faults.

Surprisingly, the subject with the *most* experience (subject 4) did not find a high number of faults. Also surprisingly, the two subjects with industrial experience (subjects 2 and 4) did not find a high number of faults. But the subject with the least experience (subject 7) did find the least number of faults.

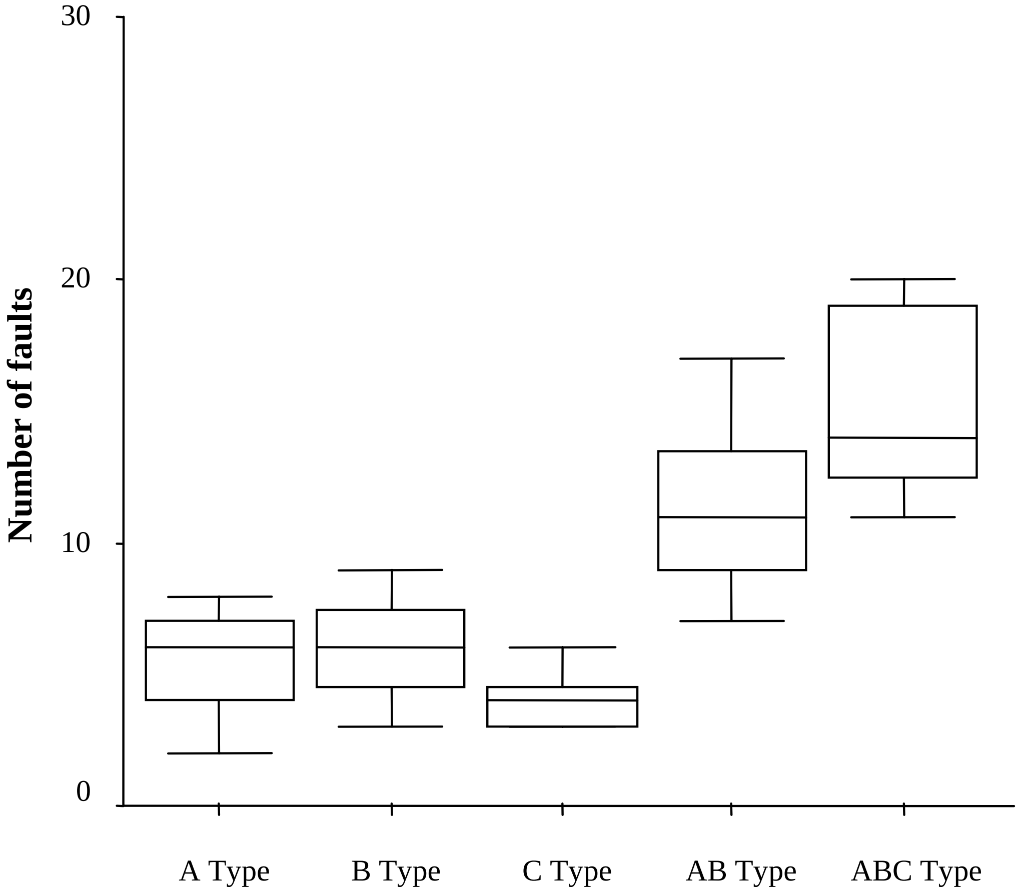

Figure 4 Box plots of faults

Figure 4 presents box plots of the distribution of the effectiveness of subjects when finding different types of faults. In general, subjects appear to be more effective at finding A type and B type faults, compared to C type faults. (There was a slightly smaller number of C type faults seeded into the design document.) The greater effectiveness of subjects for finding A type and B type faults is consistent with UBR’s focus on specifically identifying those types of faults (cf. Table 7).

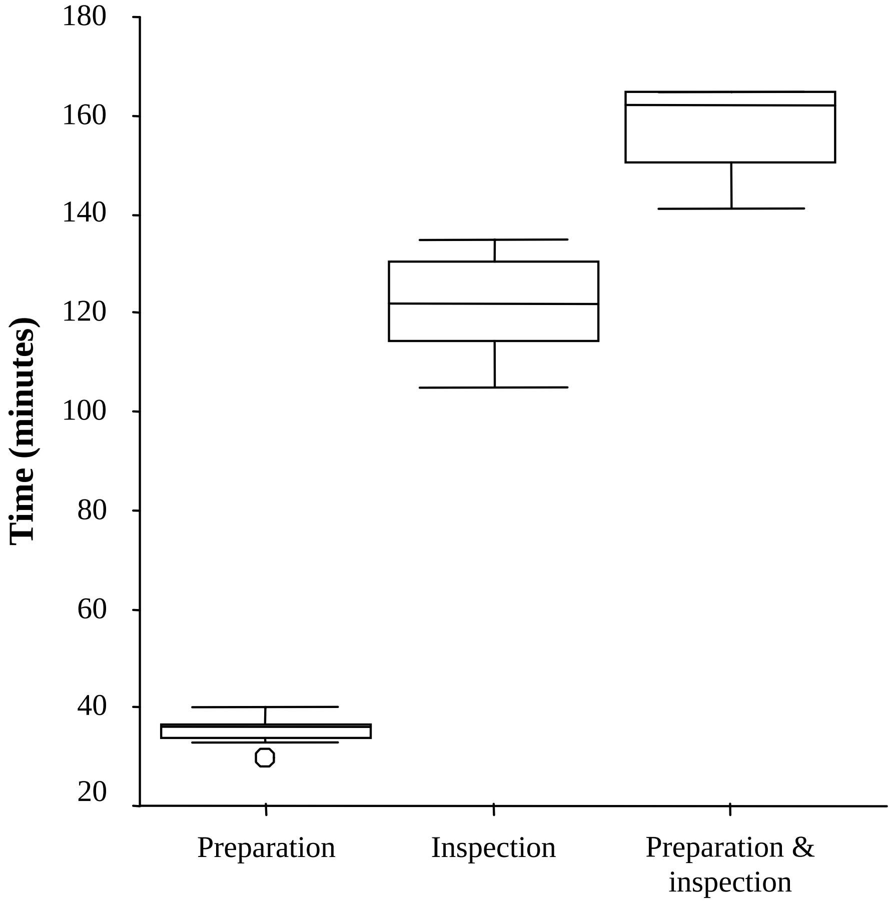

Figure 5 Box plots of time taken

Figure 5 presents box plots of the distribution of time taken by subjects for the preparation and inspection parts of the experiment. One subject (subject 3) takes an un-typically short amount of time to complete the preparation part of the experiment. Interestingly, this subject then took the *longest* amount of time to complete the inspection part of the experiment. This student was also one of the students who used all of the time allocated to the experiment.

6 Effectiveness of individual subjects

Table 9 Percentage effectiveness of each subject

	Subject						
Fault type	**1**	**2**	**3**	**4**	**5**	**6**	**7**
A	62%	46%	54%	54%	38%	23%	15%
B	75%	25%	67%	33%	58%	50%	42%
C	30%	40%	50%	30%	60%	30%	40%
A+B	68%	36%	60%	44%	48%	37%	29%
A+B+C	56%	37%	57%	39%	52%	34%	32%

Table 9 summarises the effectiveness of each subject in finding the seeded faults. Recall that the design document contains 13 faults of type A, 14 faults of type B, and 11 faults of type C. For the first four subjects (subjects 1 – 4), all of these subjects find more A type faults than C type faults. The UBR reading technique is intended to direct inspectors at finding more A type and B type faults. For the other three subjects, all three subjects find more C type faults than A type faults.

Interestingly, one subject (subject 2) seems to find a low number of B type faults (25%). The box plot in Figure 6 does not, however, identify this subject as a statistical outlier. The subject with the least experience (subject 7) finds a very low number of A type faults (15%). But again, the box plot in Figure 6 does not identify this subject as a statistical outlier. Finally, three of the seven subjects have the modal average for finding C type faults (30%). The modal average also represents the lower limit of the number of faults found i.e. no subject finds less than 30%. This suggests a skewed distribution.

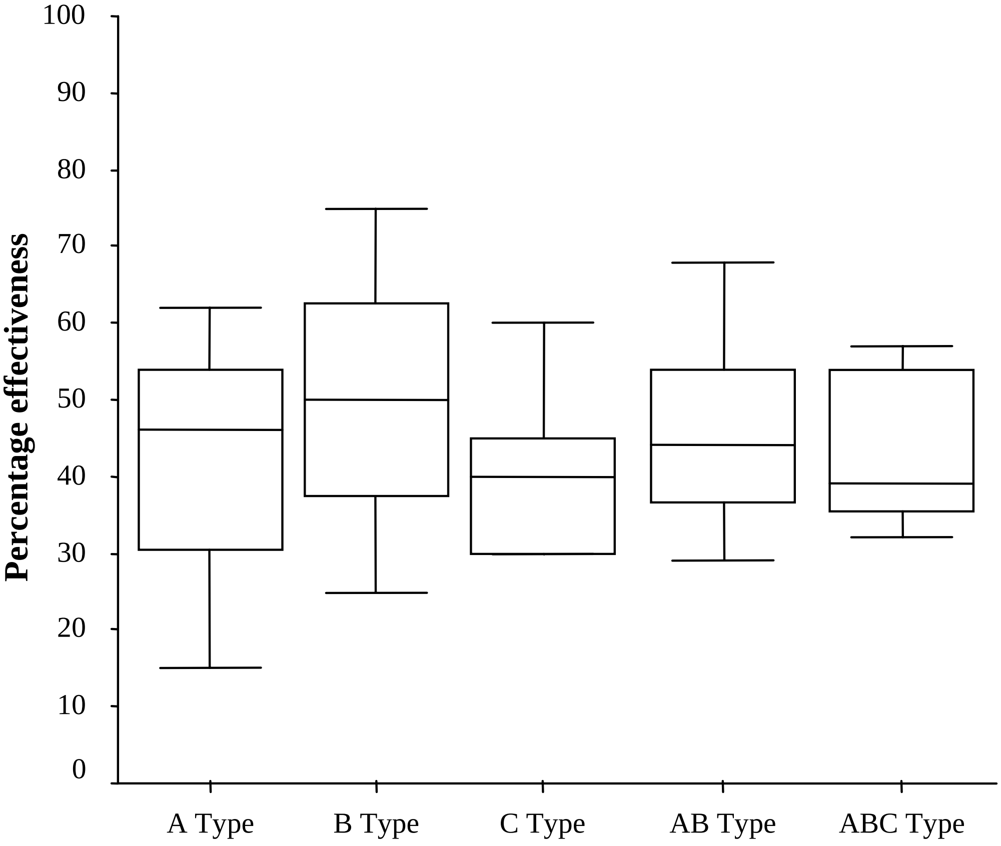

Figure 6 Box plot of subjects' individual effectiveness

Figure 6 presents box plots of the distribution of subjects' effectiveness in finding faults of different types. (The figure is a graphical representation of the data in Table 9.) The median average for finding A type and B type faults is higher than for finding C type faults. As with Figure 4, this is consistent with UBR's focus on specifically identifying faults of type A and B. The distribution of subjects' effectiveness in finding faults of type C is clearly skewed toward the lower percentages. Table 9 indicates that three of the seven subjects are finding only 30% of the type C faults.

6.1 Effectiveness at finding particular types of faults

Table 10 Breakdown of faults found per subject, for A type faults

		Subject							
Fault	**Type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Count**
4	A	44		39		51			3
5	A	92			22				2
6	A		40	49					2
8	A	17			99	5			3
14	A	45	7	25	93		20		5
15	A		88	52	107				3
17	A	76	63	85	78	38			5
20	A	45					65	107	3
22	A						40		1
23	A	54		61	111	52			4
26	A							58	1
30	A		100						1
36	A	60	58	38	30	57			5

Table 10 provides a breakdown of when (in minutes) each A type fault was found by a subject, and the number of subjects that found each fault. Four faults (numbers 14, 17, 23 and 36) are found by more than half the subjects. Subject 2 is the only subject to find fault 30. This subject is also only one of two subjects to find fault 6. Similarly, subject 7 is the only subject to find fault 26. Subject 6 is the only subject to find fault 22.

Table 11 Breakdown of faults found per subject, for B type faults

		Subject							
Fault	**Type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Count**
1	B	100	56	105	39	79	77	31	7
3	B								0
7	B								0
10	B	71	69	82	124	70	35	99	7
11	B	39					80		2
12	B	36						115	2
13	B	124		85		92			3
16	B		106	60	111	53		55	5
18	B	128		85		92		62	4
21	B				87	43			2
27	B	6		5			110		3
29	B			35			111		2
31	B	24		134		18			3
35	B	82					74		2

Table 11 provides a breakdown of B type faults. Two faults (numbers 1 and 10) are found by all the subjects. These are not, however, the first faults found by any of the subjects. Also, there is wide variation between the subjects in when this fault was found. For example, subject 7 found the fault after 31 minutes, whereas subject 3 found the fault after 105 minutes. There are only two other faults (number 16 and 18) that are found by more than half the subjects. Two faults (faults 3 and 7) are not found by any subject.

Table 12 Breakdown of faults found per subject, for C type faults

		Subject							
Fault	**Type**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**Count**
2	C	20	110	30	14	59	85		6
9	C			76					1
19	C	116	48				113	31	4
24	C					54			1
25	C		98					48	2
28	C					30			1
32	C				46	26		82	3
33	C			65		75			2
34	C								0
37	C			10	90				2
38	C	15	20	73		8	13	12	6

Table 12 provides a breakdown of C type faults. Three faults (numbers 2, 19 and 38) are found by over half of the subjects. Three faults (faults 9, 24, and 28) are only found by a single subject (subjects 3, 5, and 5 respectively). One fault (fault 34) is not found by any subject.

7 Efficiency of individual subjects

Table 13 Efficiency of each subject during *inspection time* only (faults found per hour)

	Subject						
Fault type	**1**	**2**	**3**	**4**	**5**	**6**	**7**
A	3.6	3.0	3.1	3.3	2.9	1.6	1.0
B	4.1	1.5	3.6	1.9	4.0	3.2	2.6
C	1.4	2.0	2.2	1.4	3.4	1.6	2.1
A+B	7.7	4.4	6.7	5.1	6.9	4.8	3.6
A+B+C	9.1	6.4	8.9	6.5	10.3	6.4	5.6

Table 13 summarises the efficiency of each subject in finding types of faults during only the *inspection part* of the experiment (cf. Table 7).

Table 14 Efficiency of each subject during *total experiment time* (faults found per hour)

	Subject						
Fault type	**1**	**2**	**3**	**4**	**5**	**6**	**7**
A	2.9	2.2	2.5	2.5	2.1	1.2	0.8
B	3.3	1.1	2.9	1.5	3.0	2.4	1.9
C	1.1	1.5	1.8	1.1	2.6	1.2	1.6
A+B	6.2	3.3	5.5	4.0	5.1	3.7	2.7
A+B+C	7.3	4.8	7.3	5.1	7.7	4.9	4.3

Table 14 summarises the efficiency of each subject in finding types of faults during the *entire* experiment (cf. Table 7). There is a slight decrease in efficiency which is to be expected as the preparation time is now included in the calculation of efficiency. For both tables, it is clear that the subject with the least experience (subject 7) is the least efficient subject. Curiously, this subject is particularly inefficient at finding Type A faults. For both tables, it is clear that tthe subject with the most experience (subject 4) is not one of the most efficient.

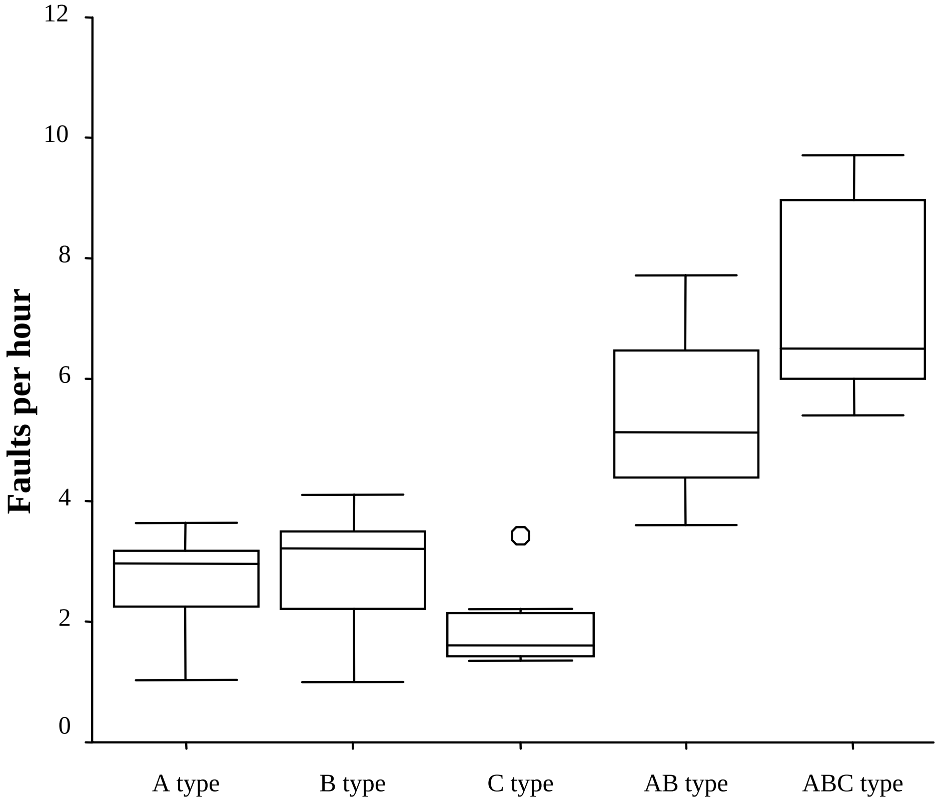

Figure 7 Box plots of subjects' efficiency in finding faults during only the *inspection time*

Figure 7 present box plots of the distribution of subjects’ efficiency in finding faults of different types. The median average for finding A type and B type faults is noticeably higher than for finding C type faults. As with Figure 4 and Figure 6, this is consistent with UBR’s focus on specifically identifying faults of type A and B. One subject (subject 5) is un-typically efficient at finding faults of type C. This subject took the least amount of time during the inspection part of the experiment and was one of the more effective subjects in finding the seeded faults (cf. Table 7 and Table 8).

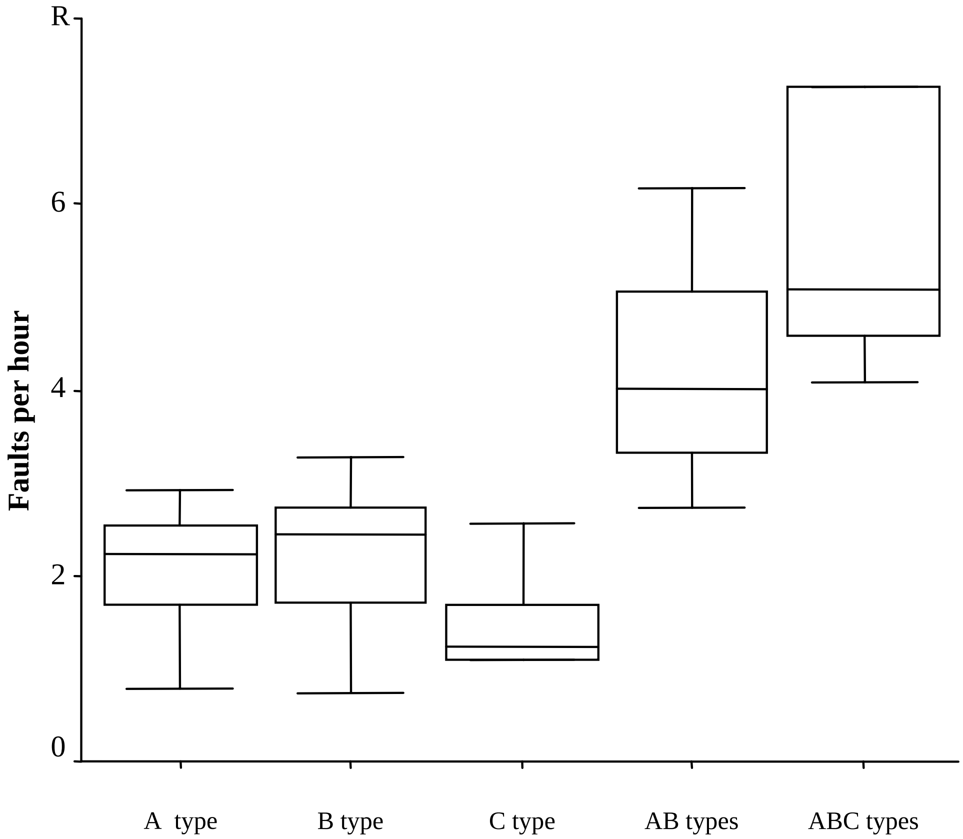

Figure 8 Box plots of subjects' efficiency in finding faults during entire experiment

Figure 8 present box plots of the distribution of subjects’ efficiency in finding faults of different types. Again, the median average for finding A type and B type faults is noticeably higher than for finding C type faults. As with Figure 4 and Figure 6, this is consistent with UBR’s focus on specifically identifying faults of type A and B. Subject 5 is no longer an outlier for finding C type faults, but Table 14 indicates that this subject is still the most efficient in finding Type C faults.

8 Opinions of the subjects

After the subjects completed their inspections, they were asked to provide feedback by completing a questionnaire. Table 15 and Table 16 present some of the subjects' opinions of the materials used in the study. We will first contrast the opinions of the most experienced and least experienced subjects, and then consider the general responses of the subjects.

Table 15 Opinions of subjects

#	Question	Responses	Freq	Subject
1	Was the inspection method easy/difficult to apply?			
		Easy	2	1,5
		Neither easy nor difficult	2	2,4
		Somewhat difficult	3	3,6,7
2	Did you follow the instructions of the inspection method during the inspection?			
		All of the time	4	1,2,4,5
		Most of the time	2	3,7
		Sometimes	1	6
3	How do you rate the quality of the requirements document?			
		Very good	1	6
		Good	3	4,5,1
		Neither good nor poor quality	3	2,3,7
4	How do you rate the quality of the design document?			
		Neither good nor poor quality	6	1–3,5-7
		Poor quality	1	4
5	How do you rate the quality of the use cases?			
		Neither good nor poor quality	3	4 - 6
		Poor quality	4	1 – 3, 7

8.1 Contrasting responses from the most and least experienced subjects

Recall from section 4 that subject 4 appears to be the most experienced subject, having industrial experience of programming and having some experience of developing taxi systems. Subject 7, by contrast, appears to be the least experienced of all the subjects. It is not surprising to find that subject 7 found the inspection method somewhat difficult (question 1). What is surprising is that subject 7 considers the use cases to be of poor quality (question 5), but thinks that the use cases are useful (question 7) and yet makes little use of them (question 10)! It also appears that subject 7 makes more use of the requirements document (question 8) and of the design document (question 9) than of the use cases (question 10). There may be a response bias in subject 7's response to question 7 e.g. having received an introduction to Usage Based Reading and been told that use cases are an important aspect of Usage Based Reading, subject 7's opinion of use cases might be biased. This is of course a threat that affects all of the subjects and all of the subjects have used use cases (cf. section 4). Due to the subject's lack of experience, however, subject 7 may be the most susceptible to this threat.

Subject 4 made little use of the requirements document (question 8) even though they found the requirements document useful (question 6) and thought the requirements document was of good quality (question 3). Of all the seven subjects in the study, subject 4 has the most experience with taxi systems. It may be that this subject does not need to frequently use the requirements document, but recognises that in general requirements documents are useful and/or that this particular requirements document may have been useful for clarifying particular issues. Contrasting the responses of subjects 4 and 7, it seems that documents can be useful during inspections even if they are not often used. This is intuitively sensible, and is also consistent with the experimenters' expectations of the requirements document i.e. that it should be used as a *reference* document (cf. section 2 and section 3.4).

To take a very different example, as readers we find dictionaries useful as a source of reference and clarification, but may not often *use* a dictionary whilst reading. And, to extend this example, the more reading we do the less likely we will use a dictionary (because we will become familiar with an increasing number of words in increasing contexts) even though we may retain an opinion that dictionaries are useful. It is intriguing to note that it is the most experienced and least experienced subjects who make the *least use* of the design document.

Table 16 Opinions of subjects (continued)

#	Question	Responses	Freq	Subject
6	How useful did you find the requirements document?			
		Useful	1	1–4,6,7
		Neither useful or useless	6	5
7	How useful did you find the use cases?			
		Very useful	2	5,7
		Useful	4	2,3,4,6
		Neither useful or useless	1	1
8	How much of the requirements document did you use?			
		Much	1	3
		About half	3	2,6,7
		Little	1	4
		Very little	2	1,5
9	How much of the design document do you use?			
		Almost entire document	3	1,3,5
		Much	2	2,6
		About half	2	4,7
10	How much of the use cases did you use?			
		Almost entire document	3	1,3,5,6
		Much	2	4
		Little	1	7

8.2 General responses

Most subjects appear to have followed the instructions of the inspection method (question 2) which implies that subjects followed the prescribed *process*. Generally, the subjects thought that the requirements document was of good quality (question 3) and found it to be useful (question 6). There was greater variation, however, in the degree to which the requirements document was used by subjects (question 8). Almost all of the subjects had a neutral opinion on the quality of the design document (question 4) with the exception of the most experienced subject who thought the design document was of poor quality. But subjects made more use of the design document than the requirements document. This is probably because it is the design document that actually is being inspected.

Most notably, four of the seven subjects thought that the use cases were of poor quality (question 5) but almost all of the subjects found the use cases useful (question 7). Anecdotal evidence suggests that these students, during their courses, learn to make use cases, in task notation, that are more descriptive and with more tasks. This may explain the subjects' opinions of the quality of the use cases.

9 Comparison with Thelin et al.' study

Thelin et al. investigated UBR and CBR. By contrast, this study has only investigated UBR. We can compare the results of Thelin et al.'s study and the current study to gain some insights into the effectiveness and efficiency of the UBR reading technique in identifying A type and B type faults.

9.1 Summary

Table 17 Summary statistics on time used (Thelin's study vs. Pedersen's study)

	Thelin				**Pedersen study**	
	UBR		**CBR**		**UBR**	
Time	**Mean**	**Std. Dev.**	**Mean**	**Std. Dev.**	**Mean**	**Std. Dev.**
Preparation	52.8	20.4	59.3	15.5	35.3	3.15
Inspection	77.1	17.8	81.1	19.2	121.7	11.04
Total	129.9	14.5	140.4	12.4	157	9.75

Table 17 presents summary statistics for Thelin et al.'s study and for the study reported in this paper. Overall, the table indicates that the subjects in the current study took less time in their preparations, but took more time to complete the actual inspections. (This may be because a 40 minute time limit was placed on the preparation time available for subjects of the current study.)

Table 18 Normalised effectiveness

Faults (total)	**Thelin**	**Pedersen**
Number of subjects	11	7
A type (13)	5.5	5.4
B type (14)	4.4	6
C type (11)	2	4
All faults (38)	11.7	15.1

Table 18 indicates that the subjects for the current study were better at finding B-type and C-type faults, but (very) slightly worse at finding A-type faults. The data has been normalized because 11 subjects were used in Thelin's study, buy only seven subjects were used in the current study. The normalised figures are, effectively, an average.

9.2 Group effectiveness at finding each fault

Table 19 Group effectiveness of finding A type faults (normalised)

	Fault												
	4	**5**	**6**	**8**	**14**	**15**	**17**	**20**	**22**	**23**	**26**	**30**	**36**
Thelin UBR	0.6	0.5	0.8	0.2	0.6	0.2	0.3	0.5	0.1	0.6	0.3	0.5	0.3
Pedersen UBR	0.4	0.3	0.3	0.4	0.7	0.4	0.7	0.4	0.1	0.6	0.1	0.1	0.7

Table 19 summarises the respective groups' effectiveness in finding each type A fault. Each group's effectiveness is, essentially, the average effectiveness of each member of that group.

Table 20 Group effectiveness of finding B type faults (normalised)

	Fault													
	1	**3**	**7**	**10**	**11**	**12**	**13**	**16**	**18**	**21**	**27**	**29**	**31**	**35**
Thelin UBR	0.5	0.2	0	0.5	0	0.4	0.3	0.5	0.4	0.5	0.2	0.5	0.3	0.2
Pedersen UBR	1	0	0	1	0.3	0.3	0.4	0.7	0.6	0.3	0.4	0.3	0.4	0.3

Table 20 summarises the respective groups' effectiveness in finding each type B fault. Each group's effectiveness is, essentially, the average effectiveness of each member of that group

Table 21 Group effectiveness of finding C type faults (normalised)

	Fault										
	2	**9**	**19**	**24**	**25**	**28**	**32**	**33**	**34**	**37**	**38**
Thelin UBR	0.5	0.1	0.5	0	0.4	0.1	0.1	0.1	0	0	0.2
Pedersen UBR	0.9	0.1	0.6	0.1	0.3	0.1	0.3	0.4	0	0.3	0.9

Table 21 summarises the respective groups' effectiveness in finding each type C fault. Each group's effectiveness is, essentially, the average effectiveness of each member of that group

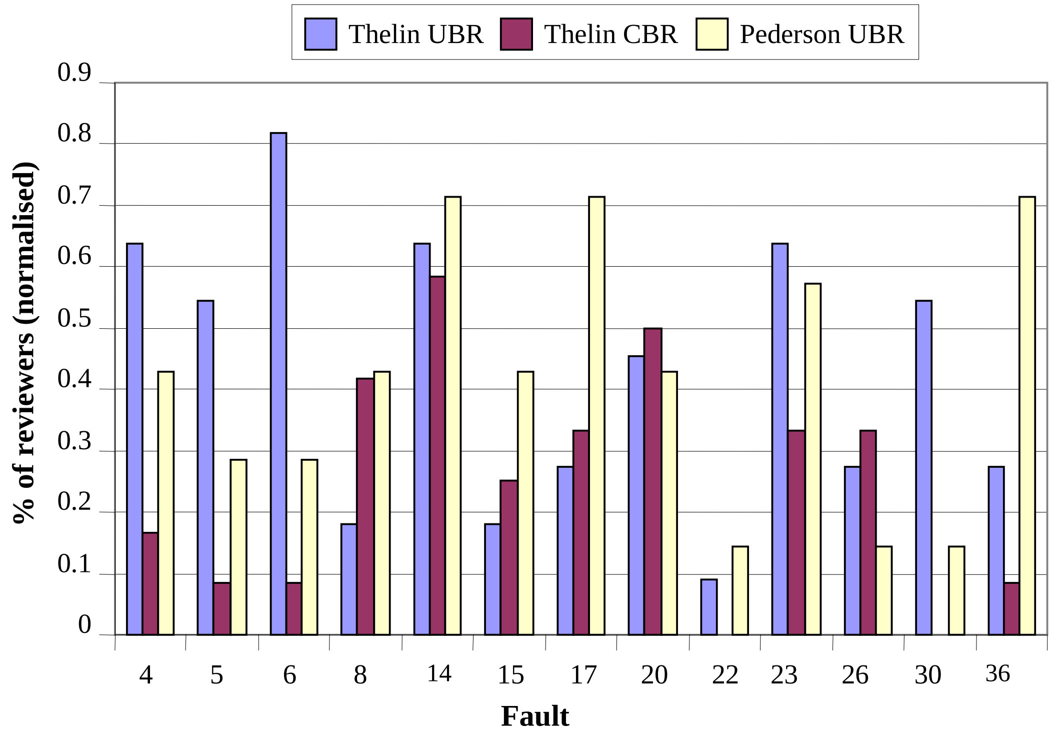

Figure 9 Bar chart of number of subjects finding each type A fault (normalised)

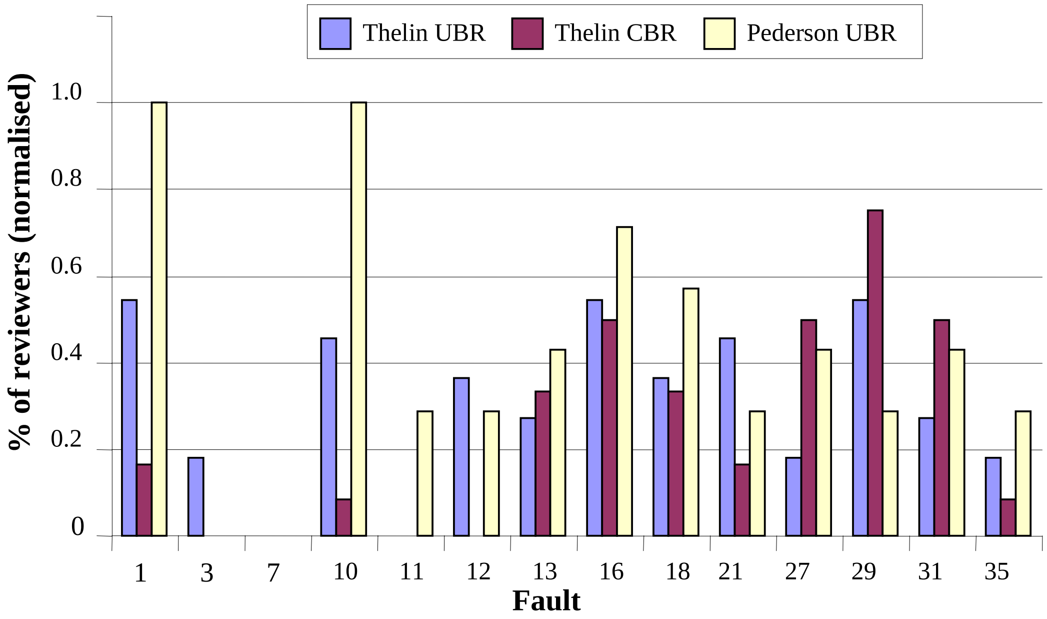

Figure 10 Bar chart of number of subjects finding each type B fault (normalised)

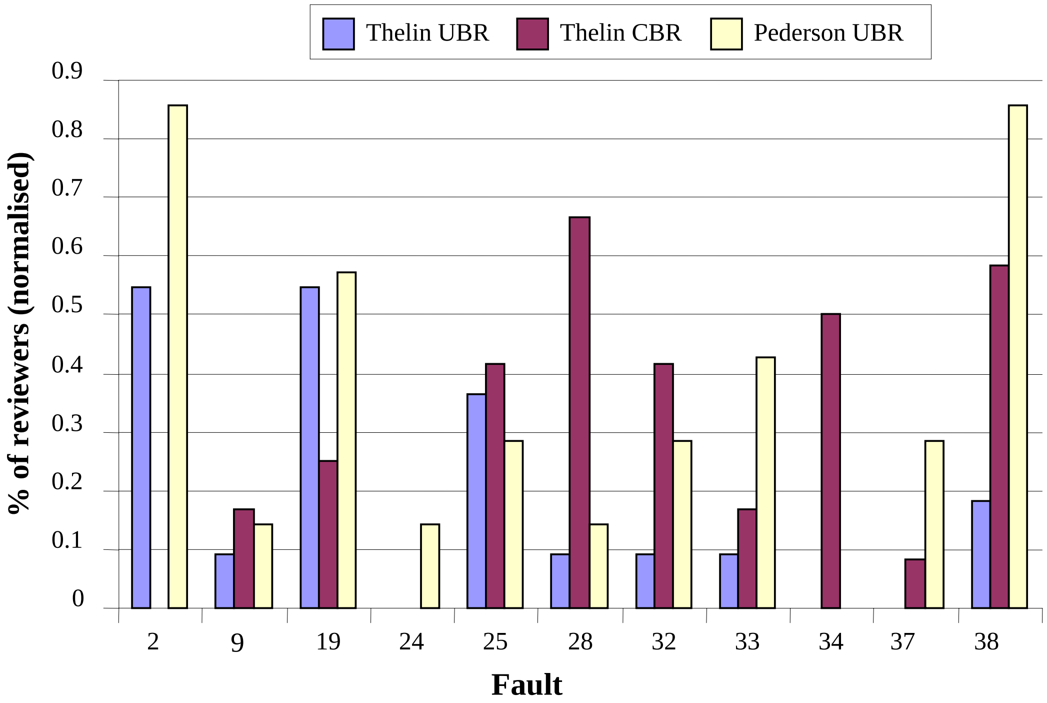

Figure 11 Bar chart of number of subjects finding each type C fault (normalised)

Figure 9, Figure 10, and Figure 11 compare the number of faults found by Thelin et al.’s two groups of subjects with the number of faults found by the subjects in the current study.

Figure 9 indicates that all Type A faults were found by subjects in the two UBR groups. Fault 22 and 30 were not found by the CBR group in Thelin et al.’s study.

For Type B faults (see Figure 10), none of the three groups of subjects found fault 3. The current study's UBR subjects were the only group to find fault 11. The current study's UBR group were also more effective in finding faults 1, 10, 13, 16 and 18. By contrast, Thelin et al.'s UBR subjects were more effective at finding faults 3, 12, and 21. All of the subjects of the current study found faults 1 and 10. Given the poor effectiveness for all groups in finding faults 3, 7 and 11, we wonder whether there may be something particularly complex about these three faults.

For Type C faults (see Figure 11), Thelin et al.'s CBR group were unable to find faults 2 and 24. Neither of the two UBR groups was able to find fault 34. The current study's UBR group appears to be more effective in finding faults 2, 19, 24, 33, 37, and 38. This is surprising given the fact that the UBR technique is not focused on finding Type C faults. As with B type faults, there is a suggestion that certain faults are more difficult to find.

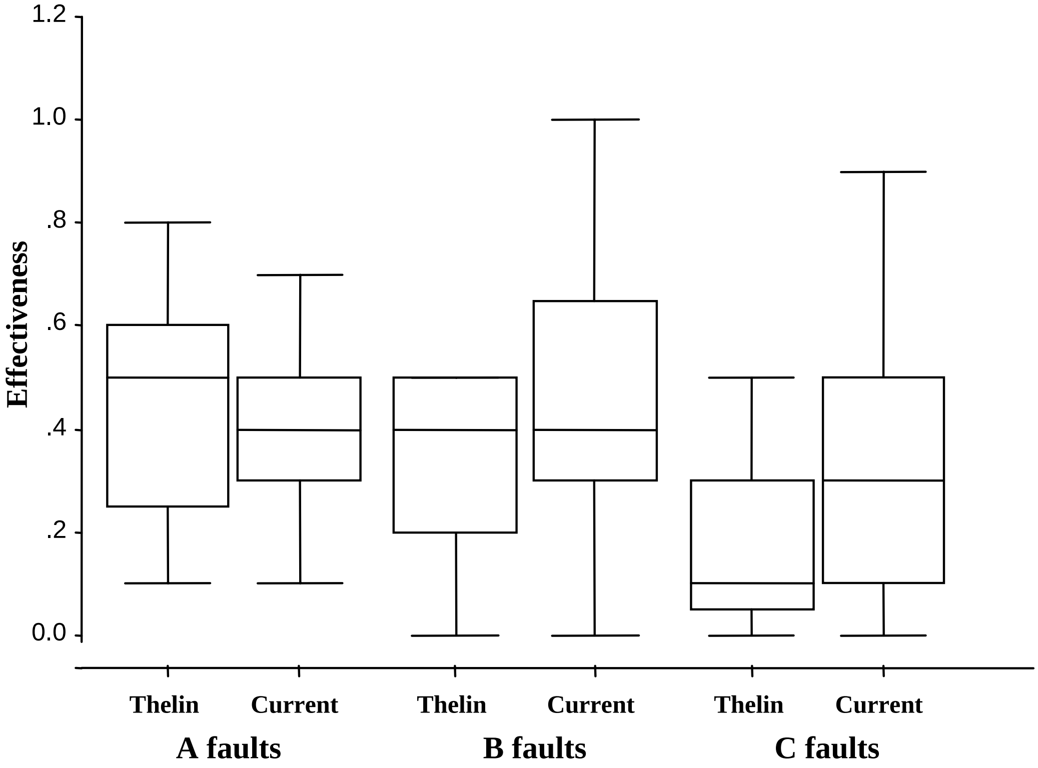

Figure 12 Box plots comparing the group effectiveness of Thelin's and this study's subjects

Figure 12 presents box-plots comparing the distribution of effectiveness in finding faults[2]. For finding type A faults, Thelin et al.'s subjects and the current study's subjects have the most similar distribution of effectiveness. Thelin et al.'s subjects are on (median) average slightly more effective. The distributions in effectiveness at finding B type and C type faults are very different between Thelin's and this study's subjects. For B type faults, both Thelin et al.'s subjects and the subjects of the current study have the same median average, but a rather different distribution. The difference between Thelin et al.'s study and the current study is probably most noticeable when finding C type faults. None of these three pairs of distributions are likely to be statistically different to each other.

[2] In Thelin et al.'s study, Thelin et al. presented box-plots of the efficiency and effectiveness in finding faults of each type. In that study, efficiency and effectiveness are *attributes* of the *entity* fault. By contrast, we are analysing the effectiveness of subjects. For the current analyses, effectiveness is an *attribute* of the *entity* subject. We are currently unable to directly compare our analysis with Thelin et al.'s analyses, because comparable information is not available in Thelin et al.'s paper. For example, Thelin et al.'s paper does not provide information on how long *each* subject took to complete the preparation and inspection parts of the experiment.

10 Discussion

10.1 The research question

The general research question for this study was:

RQ1: How does the performance of the subjects in our study, when using UBR, compare with those subjects in Thelin et al.'s study?

In answering this question, we considered two specific points:

- A comparison of the effectiveness of subjects in finding faults, and particular types of faults.
- A comparison of the efficiency of subjects in finding faults, and particular types of faults.

The findings from the current study support the findings from Thelin et al.'s study. The subjects in the current study were at least as effective in finding B type and C type faults and were only (very) slightly less effective in finding A type faults. Subjects in the current study took, on average, a shorter time to complete the preparation part of the study but a longer time to complete the inspection part of the study. However, it is possible that this was influenced by a time limit for the preparation, of 40 minutes, being imposed on subjects. In other words, the subjects in this experiment took a shorter time to do the experiment because they were instructed to and then, as a consequence, had longer time to do the inspection itself. Recall from Table 7 that five subjects in the current study used the maximum amount of time available for the study (if one includes the break times), and a sixth finished three minutes before that maximum time was reached. In itself, these times do not directly impact the efficiency and effectiveness of the subjects but does provide insights into the design of the study and the motivation of the subjects.

10.2 The hypotheses

The three specific hypotheses for this study were:

H1 There are no particularly high performing or low performing subjects in the current study.

We found some suggestions for high performing and low performing subjects, but we were unable to identify any statistically significant differences between subjects.

H2 There are no particularly easy or difficult faults to find in the current study

We found some suggestions for easier and more difficult faults but, again, we were unable to identify any statistically significant differences. In particular, we found that:

- All A type faults were found by at least one of Thelin et al.'s subjects and by at least one of the current study's subjects. This is consistent with the purpose of UBR.
- Two B type faults (fault 1 and 10) were found by all subjects in the current study, but by much fewer subjects in Thelin et al.'s study. Both Thelin et al.'s subjects and the subjects of the current study were unable to find one B type fault (fault 7). One fault (fault 11) was found by some subjects in the current study but by no subjects in Thelin et al.'s study.
- One C type fault (fault 24) was found by students in the current study, but were not found by any of Thelin et al.'s subjects. Another C type fault (fault 38) was found by Thelin et al.'s CBR group, this study's UBR group, but not Thelin et al.'s UBR group.

H3 There is no difference between the effectiveness of Thelin et al.'s UBR group of students, and the effectiveness of the group of students in the current study

We found that subjects in the current study were more effective at finding B type and C type faults, and very slightly less effective at finding A type faults. We were unable to demonstrate any statistical significance in these differences, however, so we retain the hypothesis.

11 Conclusion

This paper reports on the partial replication of Thelin et al.'s experiment to compare the UBR and CBR reading techniques for finding three types of faults. Practical limitations to the current study meant that it was not possible to conduct a complete replication of Thelin et al.'s study. We were unable to compare the UBR and CBR reading techniques. Instead, we investigated the effectiveness and efficiency of subjects using the UBR reading technique to find the three types of faults.

Broadly speaking, our results are consistent with the findings of Thelin et al.'s study. While the subjects in the current study took longer to complete their inspections, they tended to find more faults during that inspection time. The subjects of the study reported here were almost as effective as Thelin et al.'s subjects in finding A type faults, and were *more* effective than Thelin et al.'s subjects in finding B type and C type faults. The least experienced subject is one of the most poorly performing subjects. But the most experienced subject is *not* one of the best performing subjects.

Examination of the faults that were typically found and rarely found suggests that some faults are more difficult to find. There is also a suggestion that some subjects are better at finding certain kinds of faults. (By 'kinds' I do not mean the types. These 'kinds' might cut across the typology of A, B and C.)

Acknowledgements

We would like to thank the students who participated in the experiment, and Per Runeson and Thomas Thelin, from the Department of Communication Systems at Lund University in Sweden, for providing all of the experimental materials used in the experiment. We would also like to thank Dag and Ragnfrid Sjøberg, from the Simula Center in Oslo, for supplying an appropriate room and equipment at the University of Oslo, where the experiment was actually conducted. This paper analyses data collected by Are Pedersen.

References

[1] T. Thelin, P. Runeson, and C. Wohlin, "An Experimental Comparison of Usage-Based and Checklist-Based Reading," *IEEE Transactions on Software Engineering*, vol. 29, pp. 687-704, 2003.

[2] T. Thelin, "Use Cases for the Taxi Evolution (Version 3.3)," Lund University, Sweden SRS2000001, 23rd March 2004 2004.

[3] T. Thelin, P. Runeson, and B. Regnell, "Usage-Based reading an experiment to guide reviewers with use cases," *Information and Software Technology*, vol. 43, pp. 925-938, 2001.

[4] A. O. Pedersen, "An Empirical Study Of Usage-Based Reading Techniques When Inspecting Design Documents," in *Computer Science*: University of Hertfordshire, 2004.

[5] R. G. Ebenau and S. H. Strauss, *Software inspection process*: McGraw-Hill, 1994.

[6] S. Lausen, *Software requirements. Style and techniques*: Addison-Wesley, 2002.

[7] T. L. Saaty and L. G. Vargas, *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*: Kluwer Academic, 2001.

[8] C. Larman, *Applying UML and Patterns: An introduction to Object-OrientedAnalysis and Design and the Unified Process*, 2nd ed: Prentice Hall PTR, 2002.

[9] ITU-T Z.100, "Specification and description language, SDL. ITU-T Recommendation Z.100," 1993.