Summary

This Masters Thesis describes the state of the art regarding threats to validity in controlled software engineering experiments. Among the 5453 articles published in 13 leading journals and conferences in the decade 1993-2002, 107 articles (2%) reported controlled experiments in which individuals or teams conducted one or more software engineering tasks. This thesis has a special focus on generalization regarding subjects, tasks, and environment (threats to external validity). I mainly look on two different aspects for each generalization type; if replications strengthen the validity and what kinds of arguments are used for generalizing or not generalizing. At the end of the analysis the raw data found for threats to internal validity are also summarized, but not presented in detail. The main result from this research is that most researchers are very vague in their conclusions regarding results of their experiments, and that they are to some degree ambiguous. Another result from this research is that researchers struggle with more or less the same problems regarding generalization of their results. The most important proposition from this thesis will be that researchers should write more precise discussions whether their results can be generalized or not.

Acknowledgement

This thesis would not have been possible without the help of several persons. First of all, I would like to thank my supervisors, Dag I.K. Sjøberg and Amela Karahasanovic. I could not have had any better supervisors than you. You have given me a lot of support and time, and showed great interest in my work. I would also like to thank the other participants of the Context project, Vigdis By, Jo E. Hannay, Ove Hansen, and Anette C. Rekdal, for ideas, feedback, and most of all for being great co-workers. Also a special thank to Barbara Kitchenham for useful suggestions and Marit Solveig Finseth for SAS production. Last, but not least, a special thank to my wonderful wife, Ingvild. Thanks for supporting me the last 1.5 year, even though I have been very little home with you.

Some of the sections in this thesis are a product of the whole project. That is particularly applicable for chapter 3 and 4 that are partly written by other members in the project team.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

At Simula Research Laboratory there has been an ongoing project, *Context*, for the last 2 years, and this master thesis is one of the outcomes of that project. The aim of the project is to describe the state of the art in Software Engineering today. It is also interesting to see what kind of knowledge is needed in the empirical Software Engineering field and to place Simula Research Laboratory in the big picture.

This thesis investigates reflections and discussions done by researchers, around the validity of their experiments. The main focus is external validity and generalization, but also internal validity is briefly summarized.

The external validity of experiments is a matter of sampling. Most controlled software engineering experiments generally do not employ random or probability samples, and it is therefore difficult to generalize the results of these studies.

I look on three types of generalization; subjects, tasks, and environment. Regarding subjects, I also look on the subgroup of experiments with student subjects and how the researchers describe or not describe their sample and target population. What I will try to find out, is whether researchers face the same difficulties for generalizing, whether replications increase the number of generalizations, and if there are any trends that points in one or the other direction for the decade 1993 to 2002. For a discussion of all the topics covered by Context, see [11]. In [12] you find a deeper discussion of subject selection.

The articles analyzed in the project are published in a sample of nine journals and four conference proceedings in the decade 1993-2002. The sample is the journals ACM Transaction on Software Engineering Methodology (TOSEM), Empirical Software Engineering (EMSE), IEEE Computer, IEEE Software, IEEE Transaction on Software Engineering (TSE), Information and Software Technology (IST), Journal of Systems and Software (JSS), Software Maintenance and Evolution (SME), Software: Practice and Experience (SP&E), and the proceedings of International Conference on Software Engineering (ICSE), IEEE International Symposium on Empirical Software Engineering (ISESE), and IEEE International Symposium on Software Metrics (METRICS). The conference Empirical Assessment & Evaluation in Software Engineering (EASE) is included in that selected papers (10) from EASE appear in especially dedicated issues of JSS, EMSE, and IST. We consider the above journals to be leading in software engineering, ICSE is the principle conference in software engineering, and EASE, ISESE, and METRICS are the major venues in empirical software engineering and report a relatively high proportion of controlled software engineering experiments.

1.1 Motivation

My motivation for this research is the importance of external validity. The primary goal in all sciences, including software engineering, is to produce general knowledge. For controlled experiments to produce general knowledge, it is essential for any study, or families of studies [1][5], to be generalizable. This can either be accomplished by having representative subjects, tasks, and environments, or by replicating studies to see if theories hold beyond subjects, tasks, and environments.

1.2 Structure

The remainder of this thesis is organised as follows.

- Chapter 2 describes definitions used in this thesis.
- Chapter 3 presents related work.
- Chapter 4 describes the research method, including how the articles were selected, and how the analysis was done.
- Chapter 5 describes the structure of the database with a short description of the relevant fields for this thesis.
- Chapter 6 contains the complete analysis of the external validity reporting to the experiments in the survey.
- Chapter 7 summarized the results for the internal validity reporting of the experiments in the survey.
- Chapter 8 concludes from the data gathered for this thesis.

Chapter 2

Definitions

This chapter presents definitions of validity and population that are used in this paper.

2.1 Validity

We classify the threats to validity as internal and external as described by [8]. [9] extend this list to include conclusion and construct validity as well, but we will not look on those aspects, because we find internal and external validity the most important to consider in controlled experiment[1] [10]. In this thesis the definitions for internal and external validity described by Wohlin *et al.* [10] will be used. The definitions are summarized in the following sections.

In addition to the threats described by [10], we have added some other, five internal threats and one external threat. The internal threats we have added are described in Table 2.3, and the external threat is described in Table 2.4 together with the other external threats.

2.1.1 Internal validity

> If a relationship is observed between the treatment and the outcome, we must make sure that it is a causal relationship, and that it is not a result of a factor of which we have no control or have not measured. In other words, that the treatment causes the outcome (the effect).

Threats to internal validity concern issues that may indicate a causal relationship, although there is none. Factors that impact on the internal validity are how the subjects are selected and divided into different classes, how the subjects are treated and compensated during the experiment, if special events occur during the experiment etc. All these factors can make the experiment show a behavior that is not due to the treatment but to the disturbing factor.

[1] "In applied research, which is the target area for most of the software engineering, the priorities are different…….The priorities for experiments in applied research are in decreasing order: internal, external, constructs and conclusion."

Table 2.1: Internal validity; single group threats

History	In an experiment, different treatments may be applied to the same object at different times. Then there is a risk that the history affects the experimental results, since the circumstances are not the same on both occasions. For example if one of the experiment occasions is on the first day after a holiday or on a day when a very rare event takes place, and the other occasion is on a normal day.
Maturation	This is the effect of that the subjects react differently as time passes. Examples are when the subjects are affected negatively (tired or bored) during the experiment, or positively (learning) during the course of the experiment.
Testing	If the test is repeated, the subjects may respond differently at different times since they know how the test is conducted. If there is a need for familiarization to the tests, it is important that the results of the test are not fed back to the subject, in order to support unintended learning.
Instrumentation	This is the effect caused by the artifacts used for experiment execution, such as data collection forms, document to be inspected in an inspection experiment etc. If these are badly designed, the experiment is affected negatively.
Statistical regression	This is a threat when the subjects are classified into experimental groups based on a previous experiment or case study, for example top-ten or bottom-ten. In this case there might be an increase or improvement, even if no treatment is applied at all. For example if the bottom-ten in an experiment are selected as subjects based on a previous experiment, all of them will probably not be among the bottom-ten in the new experiment due to pure random variation. The bottom-ten cannot be worse than remain among the bottom-ten, and hence the only possible change is to the better, relatively the larger population from which they are selected.
Selection	This is the effect of natural variation in human performance. Depending on how the subjects are selected from a larger group, the selection effects can vary. Furthermore, the effect of letting volunteers take part in an experience may influence the results. Volunteers are generally more motivated and suited for a new task than the whole population. Hence the selected group is not representative for the whole population.
Mortality	This effect is due to the different kinds of persons who drop out from the experiment. It is important to characterize the dropouts in order to check if they are representative of the total sample. If subjects of a specific category drop out, for example, all the senior reviewers in an inspection experiment, the validity of the experiment is highly affected.
Ambiguity about direction of causal influence	This is the question of whether A causes B, B causes A or even X causes A and B. An example is if a correlation between program complexity and error rate is observed. The question is if high program complexity causes high error rate, or vice versa, or if high complexity of the problem to be solved causes both.

Table 2.2: Internal validity; multiple group threats and social threats

Interactions with selection (multiple group threats)	The interactions with selection are due to different behaviour in different groups. For example, the selection-maturation interaction means that different groups mature at different speed, for example if two groups apply one new method each. If one group learns its new method faster than the other, due to its learning ability, does, the selected groups mature differently. Selection-history means that different groups are affected by history differently, etc.
Diffusion of imitation of treatments	This effect occurs when a control group learns about the treatment from the group in the experiment study or they try to imitate the behaviour of the group in the experiment study. For example, if a control group uses a checklist-based inspection method and the experiment group uses perspective-based methods, the former group may hear about the perspective-based method and perform their inspections influenced by their own perspective. The latter may be the case if the reviewer is an expert in a certain area.
Compensatory equalization of treatments	If a control group is given compensation for being a control group, as a substitute for that they do not get treatments; this may affect the outcome of the experiment. If the control group is taught another new method as a compensation for not being taught the perspective-based method, their performance may be affected by that method.
Compensatory rivalry	A subject receiving less desirable treatments may, as the natural underdog, be motivated to reduce or reverse the expected outcome of the experiment. The group using the traditional method may do their very best to show that the old method is competitive.
Resentful demoralization	This is the opposite of the previous threat. A subject receiving less desirable treatments may give up and not perform as good as it generally does. The group using the traditional method is not motivated to do a good job, while learning something new inspires the group using the new method.

In Table 2.1 single group threats are listed. These threats apply to experiments with single groups. As such experiments have no control group (group with no treatment) there is a problem in determining if the treatment or another factor caused the observed effect.

Most of the threats to internal validity can be addressed through the experiment design. For example, by introducing a control group many of the threats to internal validity can be controlled. On the other hand, multiple group threats are introduced instead.

Multiple group threats are presented in Table 2.2, and only one such threat is considered (Interactions with selection). In a multiple groups experiment, different groups are studied. The threat to such studies is that the control group and the selected experiment groups may be affected differently by the single group threats as defined above. Thus there are interactions with the selection.

Social threats are also presented in Table 2.2. These threats are applicable to both single group and multiple group experiments.

Table 2.3: Internal validity, own definitions

Accuracy of subjects registration	If the subjects are not reporting properly, or not reporting at all, the analysis can be confounded by this. If for example the subjects do not report their time data properly, this can impact the results.
Motivation	The subject motivation for participating in the experiment. If, for example, experiment participation is a mandatory part of a course, this threat can be applicable.
Plagiarism	If the subjects exchange information about the experimental material between sessions, this threat can be applicable. This can occur if, for example, the second experimental run is performed one week after the first.
Replication	This is the effect of not replicating an experiment correctly.
Training	If the order in which two techniques are presented to the subjects can have impact of the understanding or attitude for the techniques, this effect can be applicable.

Table 2.4: External validity

Interaction of subjects[2] and treatment	This is an effect of having a subject population, not representative of the population we want to generalize to, i.e. the wrong people participate in the experiment. An example of this threat is to select only programmers in an inspection experiment when programmers as well as testers and system engineers generally take part in the inspections.
Interaction of environment[3] and treatment	This is the effect of not having the experimental setting or material representative of, for example, industrial practice. An example is using old-fashioned tools in an experiment when up-to-date tools are common in industry. Another example is conducting experiments on toy problems. This means wrong 'place' or environments.
Interaction of history and treatment	This is the effect of that the experiment is conducted on a special time or day which affects the results. If, for example, questionnaire is conducted on safety-critical systems a few days after a big software-related crash, people tend to answer differently than a few days before, or some weeks or months later.
Interaction of task and treatment[4]	This is the effect of not having the experimental task representative of, for example, industrial practice.

2.1.2 External validity

The external validity is concerned with generalization. If there is a causal relationship between the construct of the cause, and the effect, can the result of the study be generalized outside the scope of our study? Is there a relation between the treatment and the outcome?

[2] Changed by us from " Interaction of selection and treatment" to better fit our description.
[3] Changed by us from "Interaction of setting and treatment" to better fit our description.
[4] Added by us to have a complete set of external threats.

Threats to external validity concern the ability to generalize experiment results outside the experiment setting. External validity is affected by the experiment design chosen, but also by the objects in the experiment and the subjects chosen. There are four main risks: having wrong participants as subjects, conducting the experiment in the wrong environment, performing it with a timing that affects the results, and having wrong tasks.

2.2 Sample and target population

To decide whether sample and target populations are defined explicitly or implicitly in the articles, we used the following definitions.

Sample population is denoted explicit if at least type of participants were (e.g. graduate students) described in the article. Note that this does not say anything about how the sample populations are selected (e.g. random sample, convenience sample etc.). If nothing is described about the sample population, but we understood from the text what sample the experimenters have used, the sample population is denoted implicit, otherwise sample population is denoted 'Unknown' which means that no information about the sample population was found in the article.

Target population is denoted explicit if the target group of the study is described in the article or if a generalization from the sample population to another population (or an extension of the sample population to a broader set of the same population) is done. If the experimenters discuss a generalization from the sample population, but do not generalize for some reason, target population is denoted implicit; otherwise target population is denoted 'unknown', which means that no information about the target population was found in the article.

According to [6] a random sample has the following definitions:

> A sample of size *n* selected from a population of *N* distinct objects is said to be a *random sample* if each collection of size n has the same probability 1/(*Nn*) of being selected.
>
> A *random sample* of size *n* from a population *f(x)* is a collection of *n independent* random variables $X_1,…,X_n$, *each having the distribution f(x)*.

Chapter 3

Related work

This chapter describes related work published.

Table 3.1 describes the purpose, scope and extent of sampled papers in four major surveys as well as our survey. Tichy *et al.* [4] compare the amount of experimental work published in a few computer science journals and conference proceedings with the amount of experimental work published in one journal on artificial neural network and one journal on optical engineering. In total, 403 articles are surveyed and classified into the five categories: formal theory, design and modelling, empirical work, hypothesis testing and ‘other’. Zelkowitz and Wallace [13] propose a taxonomy of empirical studies in software engineering and report a survey in which 612 papers are classified within this taxonomy. Glass *et al.* [14] investigate 369 articles with respect to topics, research approaches, research methods, reference disciplines and level of analysis.

The above surveys give a comprehensive picture of research methods used in software engineering. They differ in purpose, criteria for selection of papers and taxonomies of empirical studies. Their results, nevertheless, suggest the same: The major part of published papers in computer science and software engineering provide little or no experimental validation; the proportion of controlled experiments being particularly low. The surveys also propose means to increase the amount of empirical studies and their quality.

The major difference between those surveys and ours is that they describe the extent and some characteristics of all empirical studies, while we provide an in-depth study of controlled experiments, only. The survey by Zendler [15] also focuses on experiments. He reports the results of 31 experiments with the aim of developing a preliminary software engineering theory. Shaw [16] has categorised the research reported in papers submitted and accepted for ICSE 2002.

In addition to the general surveys described above, there are of course many surveys within sub-disciplines of software engineering, for example, object-oriented technology [17], testing techniques [18], and software effort estimation [19].

Table 3.1: Surveys of empirical studies in software engineering

	(Tichy *et al.* 1995)	**(Zelkowitz *et al.* 1997)**	**(Glass *et al.* 2002)**	**(Zendler 2001)**	**Our survey**
Purpose	Comparing the extent of empirical studies in computer science with other fields	Classifying empirical studies in SE and to validate the taxonomy of empirical studies proposed by the authors	Surveying topics, research approaches, research methods, reference disciplines and level of analysis	Developing a preliminary SE theory from the results of various SE experiments	Surveying topics, subjects, tasks, environments, and generalization of controlled experiments in SE
Scope	Comp. Sci., incl. SE	SE	SE	SE	SE
Journals	ACM (random publications), TSE, PLDI Proc., TOCS, TOPLAS	ICSE Proc., IEEE Software, TSE	IEEE Software, IST, JSS, SP&E, TOSEM, TSE	Various journals and conference proceedings	EASE, EMSE, ICSE, IEEE Computer, IEEE Software, ISESE, IST, JSME, JSS, METRICS, SP&E, TOSEM, TSE
Sampling of papers	Partly random 1991-1994; one to four volumes per journal, random selection of work published by ACM in 1993	All papers in 1985, 1990, and 1995	Random in the period 1995-1999	Not reported	All papers in the period 1993-2002
Number of investigated papers	403	612	369	49 papers assessed, 31 papers analyzed in depth	5453 papers scanned, 107 papers analyzed in depth

Chapter 4

Research Method

This chapters describes the kind of experiments being the subject of this survey, the selection of journals and conferences in which the experiments are reported, and the procedure for identifying and analyzing the relevant articles.

4.1 Controlled experiments in software engineering

The common attribute in all experiments is control of treatment, though control can take many different forms. Shadish, Cook, and Campbell [9] provide the following definitions:

> Experiment: A study in which an intervention is deliberately introduced to observe its effects.
>
> Randomized experiment: An experiment in which units are assigned to receive the treatment or an alternative condition by a random process such as the toss of a coin or a table of random numbers.
>
> Quasi-Experiment: An experiment in which units are not assigned to conditions randomly.
>
> Correlation study: Usually synonymous with nonexperimental or observational study; a study that simply observes the size and direction of a relationship among variables.

This survey focuses upon experiments in which individuals or teams (the experimental units) apply a process, method, technique, language or tool (the treatments) to conduct one or more software engineering tasks. (An organisation or company could also be an experimental unit, but we found no such cases in our survey.) The insistence of treatment excludes empirical studies such as pure correlation studies, re-sampling studies and other studies that are solely based on calculations on existing data. Moreover, usability experiments are not included since we regard those as part of another discipline (human computer interaction). Articles that focus on methodological issues but that still describe experiments, and articles that only summarise experiments are also not included; our survey focuses on articles that provide the main reporting of an experiment.

In addition to randomized experiments, we include quasi-experiments. General random assignment of experimental units to treatments may not always be feasible, e.g., for logistic reasons. Laitenberger *et al.* [20] report an experiment in which units are imported into the experiment from intact training groups in a company. Randomised assignment would in this case have disturbed the training process. See also [21].

Since the term ‘experiment’ is inconsistently used in the software engineering community (often used synonymously with empirical study), we use the term ‘controlled experiment’ to emphasize the control of application of treatment.

4.2 Identification of articles reporting controlled experiments

To identify the controlled experiments, one person systematically read the title and abstract of 5453 scientific articles published in the selected journals and conference proceedings for the period 1993-2002. Excluded from the search were editorial columns, prefaces, article summaries, interviews, news, reviews, correspondence, discussions, comments, reader's letters and summaries of tutorials, workshops, panels and poster sessions.

If it was unclear from the title or abstract whether a controlled experiment was described, the whole article was read by both the same person and another person in the project team. In the end, 107 articles were selected. Note that identifying the right articles was not straightforward since the terminology in this area is confusing. For example, several authors claimed they described experiments even though no treatment was applied in the study.

4.3 Analysis of the articles

The survey data was stored in a relational database (MS SQL Server 2000). You find a detailed description of the database in Chapter 5 and Appendix A. In addition to the survey database we created a catalogue of all the articles in searchable PDF-format. About 3/4 of the articles were provided in searchable PDF-format by the journal publishers, the remaining 1/4 was OCR-scanned.

Six researchers or research assistants analysed the articles, focusing on certain aspects. Each aspect, corresponding to a set of attributes of the database, was analysed by at least two people. After the initial analysis and after inserting the resulting data into the database, the results from the two observers were compared and possible conflicts resolved by going through the article in common a third time or giving the article to a third person. The main analysis tool was SAS. All tables created by SAS that is applicable for this thesis, is attached in Appendix F.

The papers were analyzed according to four aspects: Extent, Topic, Subjects, Task/Environment, and External validity. I have analyzed the articles with focus on External validity. In addition to me, one other person has analyzed the articles with the same focus. Some of the attributes are analyzed by on third person as well.

Chapter 5

Database

As mentioned in Chapter 4, all data was stored in a relation database. The database engine was Microsoft SQL Server 2000. Some information was specific to an article, some was specific to an experiment and some information concerned the combination of article and experiment. Moreover, one article could describe several experiments and one experiment could be described in several articles, then typically with a different focus. Consequently, a data model with the entities Article, Experiment and Article-Experiment were defined with a corresponding set of attributes relevant to our survey. In Figure 5.1 you see a simple diagram of the database. In this thesis, I will only describe the fields relevant for the data presented.

The tables and analyses in this paper are with respect to every article-experiment. The survey consists of 107 articles describing 118 different unique experiments. These 118 experiments are described 125 times in the 107 articles. The number of article-experiment occurrences are therefore 125. Of the experiments, 114 are described once, 2 are described twice, 1 are described thrice, and 1 are described four times in the articles. For the rest of the document, article-experiment occurrence is denoted 'experiment'.

5.1 Fields in the Context database

In Table 5.1 are the most important fields regarding the analysis presented in this thesis briefly described. The complete descriptions of the relevant fields are attached in Appendix A.

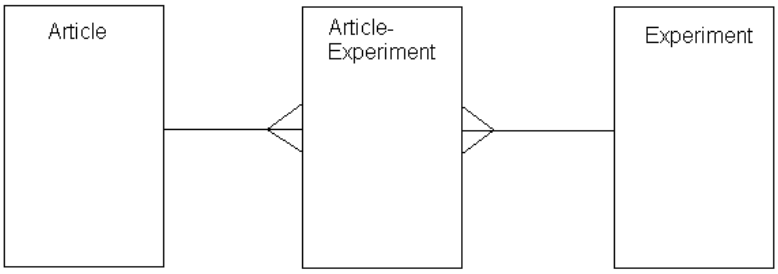

Figure 5.1: Simple outline of the database

Table 5.1: Short description of the relevant fields

Recruitment	Describes how the participants in the experiment where recruited (e.g. as part of a course, volunteers from organisations, recruited by letters etc.).
Paid or rewarded	Describes whether the participants are paid in any way (directly paid, as part of their job, point credits etc.).
Mandatory	Describes whether experiment participation was voluntarily or mandatory.
Selection of participants	Describes who the participants are (e.g. students, professionals etc.), where they come from (e.g. a university, company etc.), and if they are selected from a class, training session or similar.
Sample population	Describes whether the sample population is described explicitly, implicitly or not at all.
Sample type	Describes what kind of sample population that is used in the experiment (e.g. convenience sample, random sample, probability sample etc.).
Sample notation	Describes whether the word *sample* is used for describing the sample population or not.
Target population	Describes whether the target population is described (explicitly described, implicitly described or not described).
Target notation	Describes whether the word *target* is used for describing the target population or not.
Generalization of environment	Describes whether the authors have discussed the validity of the setting and if they are generalizing from it or not.
External and Internal threats	Describes whether the authors have discussed the various threats to internal or external validity.
List of external and internal threats	These fields list the various threats to validity discussed in the article.
Generalization of subjects	Describes whether the authors have discussed the validity of the sample population and if they are generalizing to a broader population or not.
Generalization from students	Describes whether the authors have discussed the validity of the student sample population (if there is students participating), and if they are generalizing the results to professionals.
Generalization type	Describes what kind of generalization from subjects that are made (e.g. students to professional, professionals to professionals).
Reason for generalization	Describes what information or background the authors use for justifying the generalization.
Across or to population	Describes whether the generalization is across populations or to a broader range of the same population. [22]
Generalization from task	Describes whether the authors have discussed the validity of the tasks in the experiment and if they are generalizing to industrial tasks or not.
Replication	If the controlled experiment is a replication this field contains the value 'Yes', if the experiment is not a replication this field contains the value 'No'.

Chapter 6

Analysis of external validity

There are 125 experiments in the survey. Of them 67% address threats to external validity, while the remaining 33% do not address this at all. In Table 6.1, the different threats to external validity (Table 2.4) are listed with their frequencies[5]. Of this we see that 70 of the 125 experiments addresses subject selection as a threat to external validity, 63 addresses the tasks as a threat to external validity, 25 addresses the environment as a threat to external validity, and no one addresses the effects of having the experiment on a special time or day (history). As opposed to the first three threats, the interaction of history and treatment is not necessarily applicable to that many experiments, which may explain no reporting on this category.

The following sections will describe the results for these subcategories of external validity. Section 6.1 discusses generalization from subjects, section 6.2 discusses generalization from tasks, and section 6.3 discusses generalization of the environment. It is possible though, to address generalization without addressing external validity (e.g. one article that have the following statement; *"We can consider our subjects pool a representative sample of the population of professional software developers."*), so the numbers in Table 6.1 may not directly map to generalization of subjects, tasks, and environment. Another thing that we discovered during the analysis process was that it is very hard to draw conclusions from the articles, regarding generalization. Every time a new person analyzed the sample, we got different results. P.t. 4 persons have analyzed the sample on this topic, and all of us have got quite different results. This made it difficult for us to make one united analysis of this topic. The numbers for generalization in this thesis is therefore my personal opinion. The best result we can get from this is that the articles are very vague on this topic. In Appendix B, C, and D, all quotes from the articles regarding generalization are attached.

Table 6.1: Categories of External validity

External validity category	Frequency	Percentage
Interaction of subjects and treatment	70	44.3
Interaction of tasks and treatment	63	39.9
Interaction of environment and treatment	25	15.8
Interaction of history and treatment	0	0.0
Total	**158**	**100.0**

[5] Note that one experiment can have several categories.

6.1 Generalization regarding subjects

6.1.1 Sample and target population

By doing an electronic search through all articles in this survey, we found out that only 9 of the experiments used the word ‘sample’ when describing the sample population. When we did the same thing searching for the word ‘target’ we found only 3 experiments that used this for describing the target population (i.e. the population the results are valid for). So many as 90% did not used any of these terms. None of the articles uses both terms.

Table 6.2 shows the number of experiments that defines the target and sample population for the study. This table does not focus on use of the word ‘sample’ and ‘target’ but on a subjective evaluation done by us. We found that only 2 experiments did not describe their sample population. Rest of the experiments, but one, described it explicitly. The last one described the sample population implicitly. This is not to say that those are random samples. In fact, none of the experiments in this survey have a random sample according to our definition, rather all of them are convenience samples (i.e. the subjects are chosen for convenience [7], [23]), indicating that this is a young science. In [22], J. F. Lucas points on the problem of obtaining random samples in the social sciences. We expect the same problems to be the case here. A complete set of all the different sample populations from the experiments in the survey are attached in Appendix E.

26% of the experiments explicitly describe the target population of the study, in spite of our generous definition. If we also consider those experiments that implicitly define the target population, still 34% of the experiments do not say anything about their target population at all. The reasons for this may be twofold. Firstly, the experimenters did not get the results they wanted or the experiment has low external validity, and therefore they cannot generalize from the sample population. It might then be tempting to not mention this, and let the readers decide for themselves. Secondly, the experimenters may not have considered the target group at all. Although many experimenters do not report on the target population, this is an important issue that should not be left out. All studies have a target population, and it is important for the study, to have any practical use, to define this population. Also that so few experiments as three use the word ‘target’ when describing the target population are making the target population more invisible to the reader.

Table 6.2: Sample (all are convenience samples) and target population

Frequency		Target population			
Percent		Explicit	Implicit	Unknown	Total
	Explicit	32	47	43	122
		25.6	37.6	34.4	97.6
Sample population	Implicit	0	0	1	1
		0.00	0.00	0.8	0.8
	Unknown	0	0	2	2
		0.00	0.00	1.6	1.6
	Total	32	47	46	125
		25.6	37.6	36.8	100.0

Table 6.3: Generalization of subjects

Category	Frequency	Percent
Discussed, generalized	29	23.2
Discussed, inconclusive	6	4.8
Discussed, not generalized	45	36.0
Not discussed	45	36.0
Total	**125**	**100.00**

6.1.2 Generalization of subjects

What are common sample populations? Most researchers in software engineering want their findings to apply for professionals in industry. Despite of this, most researchers use students as their sample population. There might be several reasons for this. Some of the authors of the articles in this survey explained the use of students as lack of resources (mostly financial resources) or trouble getting professionals to take part in experimental studies.

Of the experiments in the survey, 23% generalize from the sample population, to a wider population. If we also include those indicating generalization but that do not conclude[6], there is 28% of the experiments that generalize their findings. 36% do not discuss generalization of subjects at all. It is worth mentioning that there is one experiment that is described in four different articles (with different focus), and that experiment generalizes the sample population in all four articles. From now on I will call this experiment 'the four-article experiment'.

In the subset of experiments with student subjects, 19% generalize their findings to professionals. If we also here include those experiments that are categorized as inconclusive, 24% generalize their findings. This is slightly less than the whole population, so it might seem like there is a tendency to generalize less when students are used as subjects. Anyway, we have not found any evidence that generalization from students to professionals can be done on a general basis. Of the

[6] The experiments we have classified as "Discussed, inconclusive" are not taking a stand of whether to generalize or not, but they go far in saying that the results are generalizable.

Table 6.4: Generalization categorized

Generalization category	Frequency	Percent
Students to professionals	14	48.3
Students to junior professionals	5	17.2
One category of professionals to same category of professionals	5	17.2
Professionals to professionals	3	10.4
One category of students to same category of students	2	6.9
Total	**29**	**100.0**

experiments with student subjects, 33% do not discuss generalization from students to professionals.

There are reasons to believe that experimenters using professional subjects are more concerned with external validity and generalization, but if we look on the subset of experiments with professional subjects, we find evidence for the contrary. So many as 47% did not discussed generalization from the sample population at all.

The reason for this might be that experimenters think that experiments with professional subjects have external validity. This is not always true then, because professionals may behave different within certain groups or settings, so this is always an issue to discuss [3].

As we see in Table 6.4, 19, of the 29 experiments that generalize from the sample population, generalize from students to professionals or junior professionals, 8 generalize from professionals to all professionals in the same category as the experimental subjects or professionals in general, and 2 generalize from students to all students in the same category as the experimental subjects.

According to Lucas [22], generalization can be categorized as either *generalizing across* or *generalizing to* other populations or settings. Regarding generalization of subjects, there are 29 experiments that generalize from the sample population to a broader population. Of these, 76% generalize across (students to professionals, students to junior professionals, professionals to professionals) and 24% generalize to (one category of professionals to same category of professionals and one category of students to same category of students) another population.

Strictly speaking, when you have a convenience sample rather than random samples from a well defined population, it is not possible to generalize the findings of one single study to other than the experimental subjects through statistical hypothesis theory [6]. This means that many of the generalizations made by scientists in the SE field are too optimistic. But still, if one study is not generalizable, a family of similar studies (obtained for example by replications) can be generalizable.

Table 6.5: Argumentation for generalization

Generalization argument	Argument explanation	Frequency	Percent
Related work	Base their generalization on existing work/theory. Refer to other articles or books explaining this.	9	25.0
Discussion	General argumentation for that their generalization holds.	9	25.0
Background	The participants' background justifies the generalization.	6	16.7
Soon professionals	The participants will soon be professionals, and thus the generalization is justified.	5	13.9
Task	The experimental tasks are of a 'professional' nature, and the generalization from student subjects is therefore justified.	3	8.3
No difference between students and professionals	There was no significant difference between the students and professionals in the study or in a similar study.	2	5.6
Statistic	The generalization is based on statistical results.	1	2.8
Conditions	The experimental conditions are under such circumstances, that generalization beyond the sample population is justified.	1	2.8
Total		**36**	**100.0**

6.1.3 Argumentation for generalizing

In table 6.5, the different reasons authors use for generalization of subjects are listed. Of this, 22 experiments have one argumentation category, while 7 experiments have two categories. The most frequent argumentations for generalizing the subjects to another or a wider population of the sample population is related work and general discussions. 'Related work' means that the experimenters base their generalization on others work. 'Discussion' means that the experimenters argument in favor of the generalization. That so few as one experiment generalize the subjects based on statistics, shows that there are only a few experiments that get significant results. Other argumentation like that the students soon will become

professionals, and that there are no significant difference between students and professionals, may be generalizations that do not hold. This is especially true for the subjects that soon become professionals. There is no reason to believe that they will act like professionals, just because they soon finish their education. Most of

Table 6.6: Relation between replicated experiments and generalization of subjects

Total		Replication		
Percent		Yes	No	Total
Generalized from subjects	Yes	6	23	29
		4.8	18.4	23.2
	No	15	81	96
		12.0	64.8	76.8
	Total	22	103	125
		17.6	82.4	100.0

them have probably not hands on experience from industry and will therefore not be representative of professionals.

The 51 experiments that discuss generalization from sample population, but do not generalize, explains the problem with generalizing mainly by use of students or a non-representative sample. A few of them also find it difficult to generalize from one single study only.

6.1.4 Replication

A replication is a study trying to reproduce the results of an earlier study, either strictly speaking or by varying something (e.g. the way the experiment is run), to produce greater validity around the topic [1]. We look upon an experiment as a replication if the author describes the experiment as a replication. We only register if

the experiment is a replication or not. No information about type of replication (i.e. strict, that vary the manner in which the experiment is run etc. [1]) is registered. Of all the experiments in this survey, 22 are replications.

Normally, replications of studies, brings greater validity to the experimental results. By this it would be natural to think that generalization of the results would be more applicable to those studies. Our results also support this view. There is a slight increase of experiments that generalize from the sample population to a broader population between the non-replication group and the replication group, from 22% to 29%. In Table 6.6, you see the division of experiments to 4 groups. There are 6 experiments that both generalize their results and are a replication.

6.1.5 Trends

See Appendix G for the distribution of experiments to years.

Figure 6.1 shows the trend-line for the experiments that generalizes from the sample population in the period 1993-2002. The years from 1993-1995 have so few experiments in total, so it is not easy to draw conclusions based on those years, but if we look on the experiments from 1996-2002 there is a clear trend towards a greater deal of experiments that generalize their subjects to a broader population.

Figure 6.2 shows the trend-line for those experiments that do not discuss generalization from the sample population in the period 1993-2002. Also here the years 1993-1996 have too few experiments to draw any conclusions, but even if we include them, there will not be a clear trend here. Except for the year 2001, the percentages of experiments that do not discuss generalization from subjects are quite high, and stable around 40-50 %.

Figure 6.3 shows the distribution of experiments each year to all the categories presented in Table 6.3.

Figure 6.4 shows the trend-line for the experiments with student subjects that generalize to professionals. If we look on the period 1997-2002, there is a trend towards a higher percentage of experiments generalizing from students to professionals. This may look strange, as one might expect the generalization from students to professionals to decrease as the science mature. The 6 experiments that both generalize from the sample population and are replications cannot explain this, because only two of them are in this group (one from 1999 and one from 2000), and if we remove them, the trend line will be almost the same. So the reasons might come from larger numbers of students being subjects. As shown in figure 6.6, the share of not relevant experiments, which is the experiments that only use professionals as subjects, have decreased over the years, so that is evidence for a higher percentage of experiments with students.

As shown in figure 6.5, there is no clear trend that more or fewer experiments do not discuss generalization from students to professionals. If we do not look on the year 2002, when the percentage of experiments not discussing generalization from students where very low, there is probably a weak raising trend to not discuss this topic, since the first years have so few experiments, but there is still not much difference between the years here.

Figure 6.6 shows the distribution of experiments each year to all categories for generalization from students to professionals.

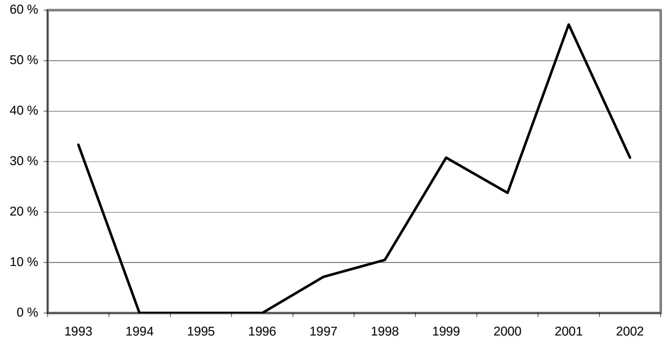

Figure 6.1: Experiments generalizing their sample population

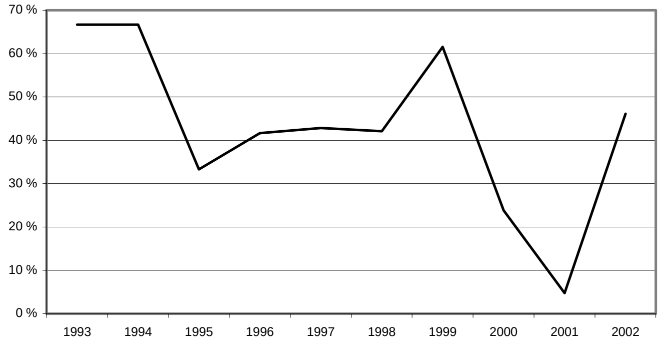

Figure 6.2: Experiments not discussing generalization from sample population

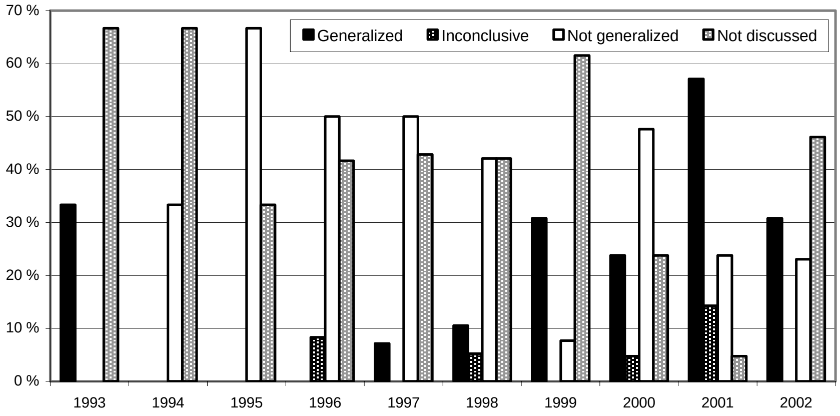

Figure 6.3: Generalization from subjects, all categories

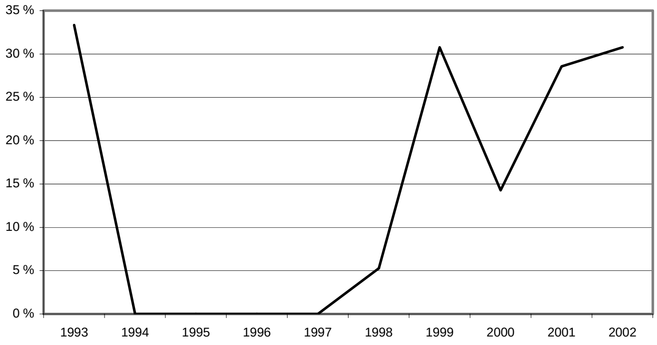

Figure 6.4: Generalizing from student sample to professionals

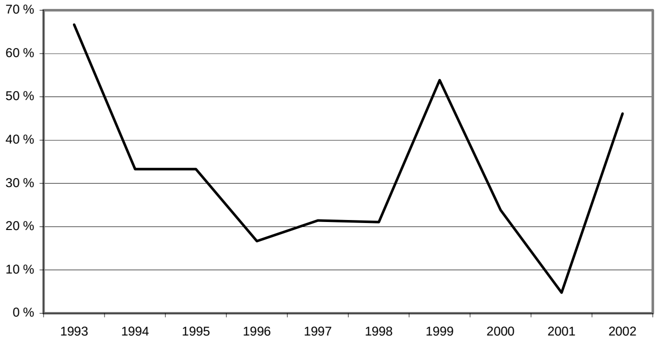

Figure 6.5: Experiments not discussing generalization from students to professionals

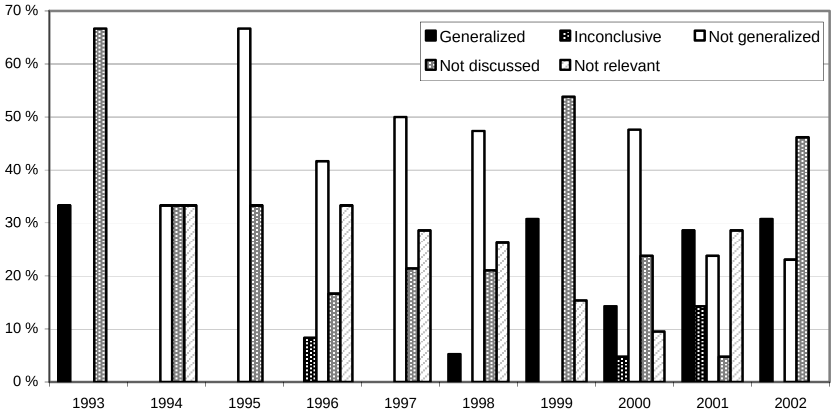

Figure 6.6: Generalization from students to professionals, all categories

Table 6.7: Generalization from tasks

Generalization category	Frequency	Percentage
Discussed, generalized	7	5.6
Discussed, inconclusive	2	1.6
Discussed, not generalized	63	50.4
Not discussed	53	42.4
Total	**126**	**100.0**

Table 6.8: Relation between replicated experiments and generalization from tasks

Frequency Percentage		Replication		
		Yes	**No**	**Total**
	Yes	1	6	7
Generalized from task		0.8	4.80	5.6
	No	20	98	125
		16.0	78.4	94.4
	Total	21	104	125
		16.8	83.2	100.0

6.2 Generalization regarding tasks

By task we mean the experimental tasks performed by the subjects and applications being subject of the task.

As we see in table 6.7, there are only 7 experiments that generalize from their tasks. If we also include those that do not conclude, there are still only 7% of the experiments that generalize the experimental tasks. There are 42% that do not discuss this topic, and that is a little bit higher than for subject generalization. With 50% of the experiments, the main category for task generalization is nevertheless the experiments that discuss generalization, but do not generalize. Also for this generalization type, the four-article experiments generalize in all of the articles.

The reason why so few experiments, compared to subjects, generalize their results may be that the experimenters find it difficult to make realistic tasks, because of for example limited time or resources.

6.2.1 Replication

It is worth noticing that of the 7 experiments that generalize, only 1 is a replication. This indicates that providing representative tasks are difficult, and bigger families of replications are possibly necessary.

6.2.2 Argumentation for generalizing

The 7 experiments that generalize the tasks have 2 arguments for generalizing. One experiment claims that because the application under study is a real application from industry they can generalize from it, and two experiments claims that they can generalize because the method (e.g. inspection method) used is representative of industrial practice. The four-article experiments use both arguments for justifying their generalization.

The experiments that discusses generalization from tasks, but do not generalize describes their problem of generalizing mainly by too small and simple tasks and applications.

6.2.3 Trends

When it comes to generalization from task, it does not make any sense to analyze the trend for the experiments that generalize, because there are as few as 7 experiments. Instead there might be more interesting to see the trend-line for all the experiments that discuss generalization from tasks. Figure 6.7 shows this graph. It is difficult to find a trend here because the share of experiments discussing generalization from tasks are so different from year to year, but there seems like there is a weak trend towards a higher percentage of discussing this.

Figure 6.9 shows the distribution of experiments each year to all categories for generalization from tasks.

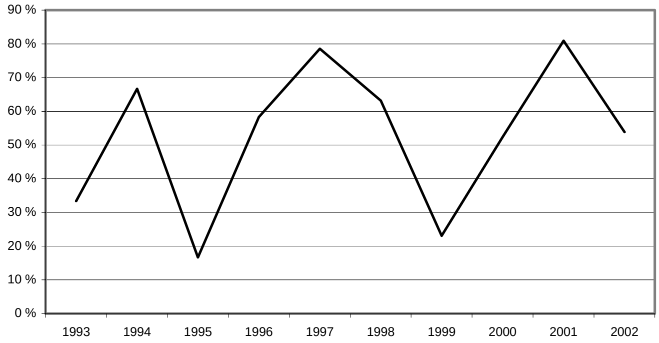

Figure 6.7: Percentage of articles discussing generalization from tasks

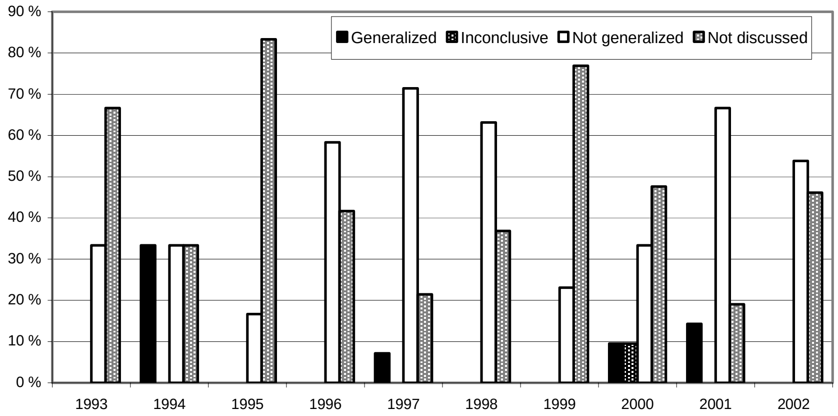

Figure 6.8: Generalization from tasks, all categories

6.3 Generalization regarding environment

By environment we mean, the experimental artifacts (e.g. pen and paper, computer), and the physical environment (e.g. laboratory, industry).

As we can see from table 6.9 there are only 6% of the experiments that generalize from the experimental environment to another environment. If we also here include those that do not conclude, still only 7% of the experiments generalize their findings. 22% discuss this matter, but do not generalize, while so much as 71% do not discuss this at all. Also for this generalization type the four-article experiment generalize in all 4 articles.

From this numbers, it seems like the experimenters find it difficult to provide 'real' settings for their experiments. Only 29% discuss generalization of the environment, and that is far from the reporting of subjects and tasks where about 60% discussed this.

6.3.1 Replication

As shown in Table 6.10, none of the generalized experiments are replications. That none of the experimenters doing a replicated study found their environment to be generalizable, strengthen the assumption that experimenters find it difficult to provide 'real' settings for their experiments.

Table 6.9: Generalization of environment

Generalization category	Frequency	Percentage
Discussed, generalized	7	5.6
Discussed, inconclusive	2	1.6
Discussed, not generalized	27	21.6
Not discussed	89	71.2
Total	**125**	**100.0**

Table 6.10: Relation between replicated experiments and generalization of context

Total **Percent**		**Replication**		
		Yes	No	Total
Generalization of context	**Yes**	0	7	7
		0.0	5.6	5.6
	No	21	97	118
		16.8	77.6	94.4
	Total	21	104	125
		16.8	83.2	100.0

6.3.2 Argumentation for generalizing

The 7 experiments generalizing the environment under study use 4 different arguments for justifying the generalization. The four-article experiment justifies their generalization by claiming they have an environment representative of an industrial development situation. The other three experiments justifies their generalization by one of the following arguments; the working environment gives the experimenters high control over the subjects, the experiment is part of an industrial project, and the working environment is common to many other firms.

The experiments that discuss generalization from the environment, but do not generalize explain why they cannot generalize by experimental materials not representative of industrial practice (e.g. pen and paper), problems generalizing from only one single study, working conditions not representative of industry (e.g. laboratory setup, single person estimating), and only one organization participating in the study.

6.3.3 Trends

Figure 6.9 shows the trend-line for discussing generalization of environment from 1993-2002. If we look on the period 1997-2002, where the numbers of experiments are high, it is no clear trend in any direction if we disregard year 2001, where a very high proportion of the experiments discuss this.

Figure 6.10 shows the distribution of experiments each year to all categories for generalization of environment.

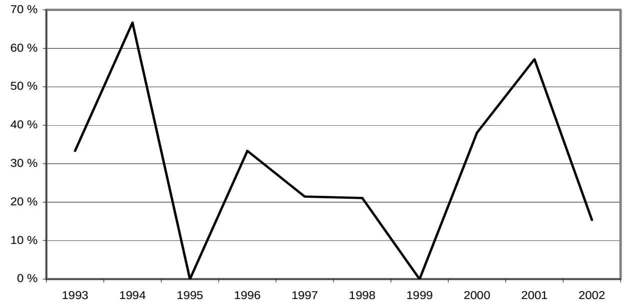

Figure 6.9: Percentage of articles discussing generalization of environment

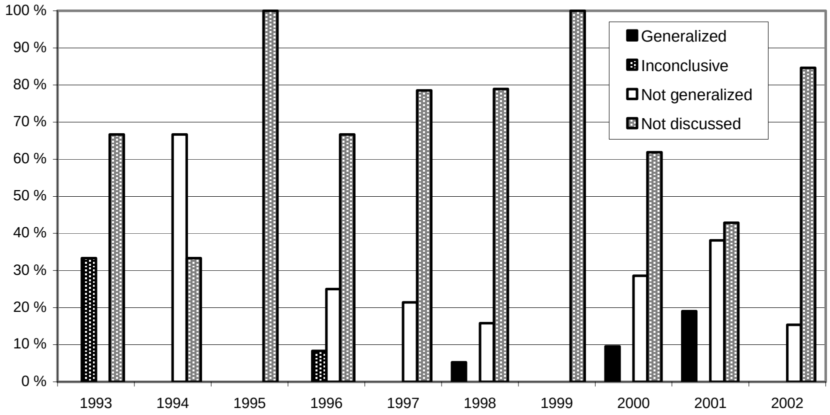

Figure 6.10: Generalization of environment, all categories

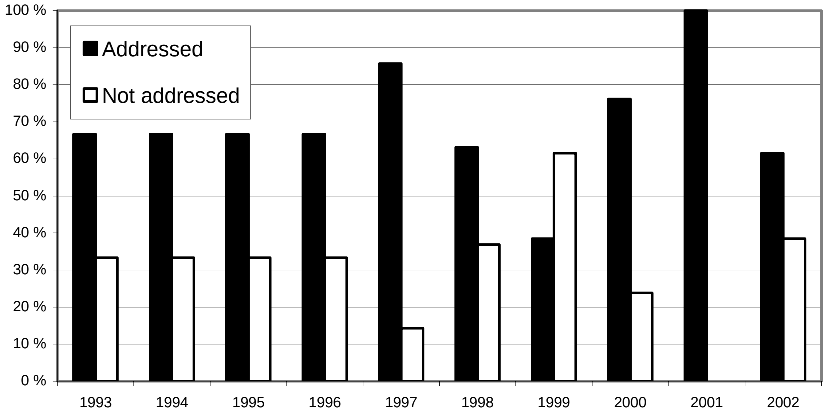

Figure 6.11: External validity 1993-2002

6.4 Trends

Figure 6.11 shows the percentage for addressing and not addressing external validity for the period 1993-2002. The percentages are quite stable with one exception - the year 2001, where all experiments addressed this.

6.5 Discussion

In addition to the four-article experiment, there is only one other experiment that generalizes their results in all three categories. In addition to this there are two other experiments that generalize their results both for the environment and the subjects. This indicates that experimenters find it difficult to provide representative subjects, tasks, and environment at the same time.

Chapter 7

Analysis of internal validity

As we can see from Table 7.1, 67% of the experiments address threats to internal validity, while the remaining 33% do not discuss this at all.

In Table 7.2, the frequencies for the different categories of internal validity threats are listed (see Table 2.1, 2.2, and 2.3). 33% of the 84 experiments addressing internal validity, only addresses one of these categories, while the remaining 67% addresses 2 or more.

Selection, instrumentation, and maturation are the most common validity threats, by 58% of the total threats addressed. There is one experiment that only addresses internal validity in general, without addressing any of the categories explicitly. The 'Other' category contains threats that we do not have defined, like fatigue effects (subjects being tired during experiment execution), persistence effects (the subjects had participated in a similar experiment earlier), demotivation (the subjects being demotivated during the experiment) etc.

Table 7.1: Internal validity

	Frequency	**Percent**
Addressed	84	67.2
Not addressed	41	32.8
Total	**125**	**100.0**

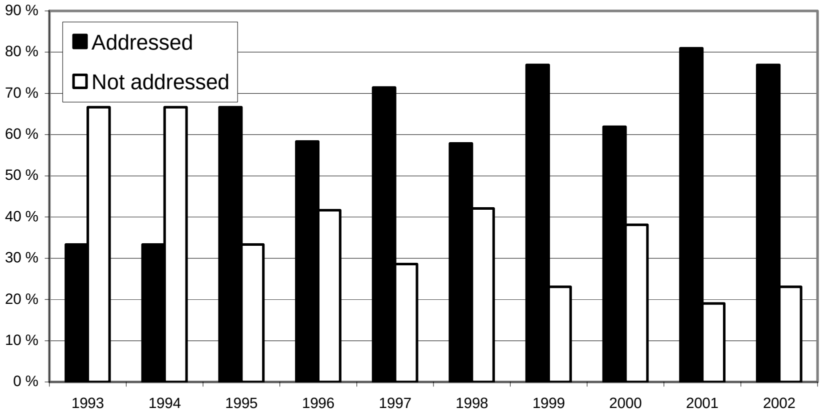

Figure 7.1: Internal validity 1993-2002

Table 7.2: Categories of internal validity

Internal validity category	**Frequency**	**Percent**
Selection	45	20.2
Instrumentation	44	19.7
Maturation	40	17.9
Training	12	5.4
History	10	4.5
Mortality	9	4.0
Interactions with selection	7	3.1
Accuracy of subjects registration	5	2.2
Motivation	5	2.2
Replication	5	2.2
Testing	5	2.2
Plagiarism	4	1.8
Compensatory equalization of treatments	2	0.9
Statistical regression	2	0.9
Ambiguity of the direction of causual influence	1	0.5
Compensatory rivalry	1	0.5
Diffusion of imitation of treatments	1	0.5
Only addresses internal validity	1	0.5
Other	24	10.8
Total	**223**	**100.0**

7.1 Trends

In Figure 7.1 we can see that there is a clear trend towards a higher percentage of experiments addressing threats to internal validity. The trend is quite clear, even without considering the three first years, from about 60% in 1996 to around 80% in 2002.

Chapter 8

Conclusions

The most important conclusion we can get from this thesis, is that researchers are very vague in their argumentation when it comes to generalization. Expressions like *"The subjects were not professional software engineers. However, they were quite experienced programmers and held degrees (many of them advanced) in computer science."*, were difficult to interpret. On the one hand they say that we cannot generalize because the subjects are not professionals, but on the other hand we can generalize because they are very experienced. A suggestion to researchers is therefore to be precise when discussing generalization (this is of course applicable for other topics too, but generalization is this thesis' view).

It is clear that the researchers struggle with more or less the same problems when they want to generalize their results. For subject generalization it is mainly the use of student subjects or a non-representative sample (e.g. professionals from one department in one company), for task generalization the problem is to make tasks that are big and complex enough, and for environment generalization the main problems are non-representative experimental materials and working conditions. The explanation why this is difficult is mainly lack of resources, both money and time.

Do replications increase the number of generalizations? When looking at the results for subject generalization it might seem so, but for task and environment the results are quite different with almost none of the experiments both being a replication and generalizing their results (only one experiment for task generalization).

For most studies one needs replications to generalize the results, still only 18% of the experiments in this survey are replication. So another suggestion will be to carry out more replications. Most researchers provide their research material for free, so replications may be a 'cheap' way to run an experiment, since a lot of the design is already prepared.

Most of the trends I have identified for this thesis are quite stable over the decade 1993 to 2002. There are however some exceptions. Subject (student) generalization has increased over the decade from about 10% (5%) to about 30% (25%) and the share of experiments addressing internal validity have increased from about 60% to about 80%. I have not given the first three years much weight in this analysis because there are so few experiments those years.

More studies with controlled experiments are proposed [2]. Even though the number of studies describing controlled experiment has increased over the decade (see Table G.1) it dropped to 13 in 2002, from 21 in 2001. This was also the case for 1999, and then it increased to 21 again in 2000, so the same might be the case here.

Bibliography

[1] V.R. Basili, F. Shull, & F. Lanubile, “Building Knowledge through Families of Experiments”, IEEE Transactions on Software Engineering, vol. 25, no. 4, pp. 456-473, Jul/Aug 1999.

[2] W.F. Tichy, “Should Computer Scientists Experiment More? 16 Excuses to Avoid Experimentation”, IEEE Computer, vol. 31, no. 5, pp. 32-40, May 1998.

[3] E. Arisholm & D. Sjøberg, “Evaluating the Effect of a Delegated versus Centralized Control Style on the Maintainability of Object-Oriented Software”, IEEE Transaction on Software Engineering, vol. 30, no. 7, 2004.

[4] W.F. Tichy, P. Lukowicz, L. Prechelt, & E.A. Heinz, “Experimental Evaluation in Computer Science: A Quantitative Study”, Journal of Systems and Software, vol. 22, pp. 9-18, 1995.

[5] R.M. Lindsay & A.S.C. Ehrenberg, “The Design of Replicated Studies”, The American Statistican, vol. 47, no. 3, pp. 217-228, August 1993.

[6] G.K. Bhattacharyya & R.A. Johnson, “Statistical Concepts and Methods”, John Wiley & Sons Inc., 1977.

[7] T.R Lunsford & B.R. Lunsford, “The Research Sample, Part I: Sampling.”, Journal of Prosthetics and Orthotics, vo. 7, no. 3, pp. 105-112, 1995.

[8] D.T. Campbell & J.C. Stanley, “Experimental and Quasi-Experimental Designs for Research”, Rand-McNally, 1963.

[9] W.R. Shadish, T.D. Cook, & D.T. Campbell, “Experimental and Quasi-Experimental Designs for Generalized Causal Inference”, Houghton-Mifflin Company, 2002.

[10] C. Wohlin, P. Rundeson, M. Höst, M.C. Ohlsson, B. Regnell & A. Wesslén, “Experimentation in Software Engineering: An Introduction”, Kluwer Academic Publishers, 1999.

[11] D. Sjøberg, J.E. Hannay, O. Hansen, V.By, A. Karahasanovic, N.-K. Liborg, & A.C. Rekdal, ”A Survey of Controlled Experiments in Software Engineering”, Work in progress for IEEE TSE submission, 2004.

[12] O. Hansen, ” Survey of Controlled Software Engineering Experiments with Focus on Subjects”, Master thesis, Work in progress, 2004

[13] M.V. Zelkowitz and D.R. Wallace, “Experimental validation in software engineering”, Journal of Information and Software Technology, vol. 39, pp. 735-743, 1997.

[14] R.L. Glass, I. Vessey, & V. Ramesh, “Research in software engineering: an analysis of the literature”, Journal of Information and Software Technology, vol. 44, no. 8, 2002.

[15] A. Zendler, “A preliminary software engineering theory as investigated by published experiments”, Empirical Software Engineering, vol. 6, no. 2, 2001.

[16] M. Shaw, “Writing good software engineering research paper: Minitutorial”, in Proc. of the 25th International Conference of Software Engineering (ICSE), 2003.

[17] I.S. Deligiannis, M. Shepperd, S.Webster, and M. Roumeliotis, “A review of experimental investigations into object-oriented technology”, Empirical Software Engineering, vol. 7, no. 3, 2002.

[18] N. Juristo, A.M. Moreno, and S. Vegas, “Reviewing 25 years of testing technique experiments”, Empirical Software Engineering, vol. 9, March 2004.

[19] M. Jørgensen, “A review of studies on expert estimation of software development effort”, Journal of Systems and Software, vol. 70, no. 1_2, pp. 37_60, 2004.

[20] O. Laitenberger, K. El Emam, & T.G. Harbich, “An internally replicated quasi-experimental comparison of checklist and perspective based reading of code documents”, IEEE Transaction on Software Engineering, vol. 27, May 2001.

[21] T.D. Cook & D.T. Campbell, “Quasi-Experimentation. Design & Analysis Issues for Field Settings”, Houghton-Mifflin Company, 1979.

[22] J.W. Lucas, “Theory-Testing, Generalization, and the Problem of External Validity”, Sociological Theory 21, 2003.

[23] R. Ferber, “Editorial: Research By Convenience”, Journal of Consumer Research, vol. 4, 1977.

Appendix A

Table A.1 and Table A.2 contains the detailed description of the context database, for the fields relevant for this thesis.

Table A.1: Description of fields from the article-experiment (focus) table

Fieldname	**Description**	**Who**
Target_population	The target population for the study. The following categories are used. ?? -> no information about target population Implicit -> the target population is implicitly described in the article. Other values -> the target population for the study (explicitly stated in the article).	NKL
Comments_on_target_population	Comment to target_population. Mostly supplementary information and quotes from the article.	NKL
Generalization_of_context	Whether the context (setting) of the experiment are generalized. The following categories are used. Discussed, generalized -> the article discusses generality from the context and do generalize Discussed, inconclusive -> the article discusses generality from the context but do not conclude whether to generalize. Discussed, not generalized -> the article discusses generality from the context, but do not generalize Not discussed -> the article do not discusses generality from the context	NKL
Comments_on_generalization_of_context	Comment to generalization_of_context. Mostly supplementary information and quotes from the article.	NKL
External_threats	Whether external validity is discussed. Discussed -> the article discusses external threats. Not discussed -> the article do not discusses external threats.	NKL
Comments_on_external_threats	Comments to external_threats. Mostly supplementary information and quotes from the article.	NKL
Internal threats	Whether internal validity is discussed. Discussed -> the article discusses internal threats. Not discussed -> the article do not discusses internal threats.	NKL
Comments_on_internal_threats	Comments to internal_threats. Mostly supplementary information and quotes from the article.	NKL
Generalizations_from_students	Whether the students in the samples are generalized to professionals. The following categories are used. Discussed, generalized -> the article discusses generality from students and do generalize Discussed, inconclusive -> the article discusses generality from students but do not conclude whether to generalize. Discussed, not generalized -> the article discusses generality from students, but do not generalize Not discussed -> the article do not discusses generality from students. Not relevant -> the experiment have only professional subjects	NKL/ OH

Comments_on_generalization_from_students	Comments to generalizations_from_students. Mostly supplementary information and quotes from the article.	NKL/ OH
List_of_external_threats	Contains the different threats to external validity the article describes for this experiment. The threats are separated by /.	NKL
List_of_internal_threats	Contains the different threats to internal validity the article describes for this experiment. The threats are separated by /.	NKL
Generalization_of_subjects	Whether the sample population are generalized to a broader population. The following categories are used. Discussed, generalized -> the article discusses generality from subjects and do generalize Discussed, inconclusive -> the article discusses generality from subjects but do not conclude whether to generalize. Discussed, not generalized -> the article discusses generality from subjects, but do not generalize Not discussed -> the article do not discusses generality from subjects	NKL
Sample_notation	Whether the word sample is used for describing the sample population. No -> the article do not use the word sample for describing the sample population for this experiment. Yes -> the article use the word sample for describing the sample population for this experiment.	NKL
Target_notation	Whether the word target is used for describing the target population. No -> the article do not use the word target for describing the target population for this experiment. Yes -> the article use the word target for describing the target population for this experiment.	NKL
Generalization_type	What kind of generalization from subjects this is. The categories are made for this survey. Not relevant -> the experiment do not generalize the subjects Other values -> the type of generalization made	NKL
Reason_for_generalization	What the authors use as background for generalizing the subjects. Not relevant -> the experiment do not generalize the subjects Other values -> the reasons that are used for generalizing	NKL
Across_to_population	What type of generalization according to J.F. Lucas. Across -> the authors generalize across the sample population (from one population to a different population) To -> the authors generalize to a wider sample of the same population	NKL
Sample_population	How the sample population are described. The following categories are used. Explicit -> the sample population are explicitly stated (at least who the subjects are are stated) Implicit -> the sample population are implicitly stated in the article ?? -> no information about the sample population in the article	NKL

Generalisation_from_task	Whether the task are generalized to industrial tasks. The following categories are used. Discussed, generalized -> the article discusses generality from task and do generalize Discussed, inconclusive -> the article discusses generality from task but do not conclude whether to generalize. Discussed, not generalized -> the article discusses generality from task, but do not generalize Not discussed -> the article do not discusses generality from task	NKL
Comment_on_generalisation_from_task	Comment to generalization_from_task. Mostly supplementary information and quotes from the article.	NKL
Other_comments	Other comments to the experiment made by the analyzers during reading. Jo is in charge of all the cryptical abbreviations.	NKL/ ACL/ JIV/ JH/ OH

Table A.2: Description of fields from the experiment table

Fieldname	**Description**	**Who**
Replication	'Yes' if the experiment is a replication, 'No' otherwise.	NKL/ OH
Selection_of_participants	Who the participants are (students, professionals etc.), where they come from (university, company etc.), course/training session etc. (if this is relevant). Information is is categorized with the following labels: INST -> Named institution AINST ->Anonymous institution VINST -> Varied institutions COMP -> Named companies ACOMP -> Anonymous companies VCOMP -> Varied companies CL -> States which class(es) subjects came from CS -> States which course(s) subjects came from VCS -> Varied courses NONE -> No info	NKL/ OH
Comments_on_selection_of_participants	Comments to selection_of_participants. Mostly supplementary information.	NKL/ OH
Recruitment	How the participants were recruited (e.g. as part of a course). Information is categorized with the following labels (some of them may be merged): POC -> Part of course POWS -> Part of workshop V -> Volunteers PI -> Personal invitation POW -> Part of work FC -> From course FCS -> From courses POT -> Part of training POTC -> Part of training course POC+V -> Part of course and volunteers DIV -> Subjects were recruited in different ways POC+PR -> Part of course and professionals NONE -> No info	NKL/ OH

Comments_on_recruitment	Comments to recruitment. Mostly supplementary information.	NKL/ OH
Paid_rewarded	Whether the subjects where paid in som way. The following categories is used. Part of job -> the experiment is done as part of their normal work ?? -> no information about payment Paid -> the subjects are paid Unpaid -> the subjects are not paid Grade -> the subjects receive a grade or factored into final grade for course for participating Credit -> the subjects receive point credit for participating Credit for some -> Some subjects received credit, others not Reward -> the subjects are rewarded in some way (e.g. dinner, trip to an exhibition etc.)	NKL/ OH
Mandatory	Whether experiment participation was mandatory or voluntarily. The following categories are used. ?? -> no information about mandatory/voluntarily No -> experiment participation was voluntarily Not relevant -> experiment is for example done as part of job Yes -> experiment participation is mandatory For some -> experiment participation is mandatory for some, voluntarily for others	NKL/ OH
Sample_type	Type of sample population for the study. For the time every sample is a convenience sample, so this is the only value, but in theory the values can also be random sample, probability sample etc.	NKL

Appendix B

Table B.1 contains the complete set of quotes for all the experiments that addresses generalization of subjects. Remark that all references in the quotes are rewritten to be on the same format (Authors, Year), and to some extent recognizable. The references in the quotes are not present in the reference list of this thesis.

Table B.1: Quotes – Generalization of subjects

Art.	Exp.	My opinion	Quote
11	1	Discussed, not generalized	"First, the original designers and implementors may be the ones who maintain the program. This was not the case in our experiment and our results do not apply to such cases. The maintainers may also have more pattern experience than our participants. The consequences of this difference are unclear; but we do not believe them to be dramatic."
12	2	Discussed, not generalized	"The most frequent concern with experiments using student subjects is that the results cannot be generalized to professionals because the latter are more experienced. In the present case, professional programmers may either have less need for PCL or they may be able to exploit PCL more profitably than our student subjects."
12	3	Discussed, not generalized	"The most frequent concern with experiments using student subjects is that the results cannot be generalized to professionals because the latter are more experienced. In the present case, professional programmers may either have less need for PCL or they may be able to exploit PCL more profitably than our student subjects."
16	5	Discussed, not generalized	"The subjects who participated in this study are unlikely to be representative of software professionals. This is not to say that the results cannot be useful in an industrial context for several reasons. Laboratory settings such as this one allow the investigation of a larger number of hypotheses at a lower cost than field studies. The hypotheses that seem to be supported in the laboratory setting can then be tested further in more realistic industrial settings with a better chance of discovering important and interesting findings. Conversely, laboratory experiments can be used to confirm results obtained in field studies, where control and, therefore, internal validity is usually weaker."
17	6	Discussed, generalized	"We can consider our subject pool a representative sample of the population of professional software developers."
17	193	Discussed, generalized	"We can consider our subject pool a representative sample of the population of professional software developers."
17	194	Discussed, generalized	"We can consider our subject pool a representative sample of the population of professional software developers."
18	7	Discussed, not generalized	"Finally, professional software engineers may have different levels of skill than our participants. A higher skill and experience level may leave less room for improvement, but may also sharpen the eye as to where improvements are most desirable or most easy to achieve with PSP techniques. Conversely, lower skill (which will occur because our students are more skilled than most of the noncomputer scientists that frequently start working as programmers today) may leave more room for improvement but may also impede applying PSP techniques correctly or at all."
19	8	Discussed, generalized	"The major results described in this paper are significant at the .05 level; they can be extended to the underlying normal population. Specifically, There is a significant linear relation between amount of reuse and customer satisfaction. A t-test conducted on the two cases shows that the increase of customer satisfaction after the adoption of a reuse library is significant. The statistical analysis process by itself cannot detail the underlying population other than specifying its statistical parameters."

23	10	Discussed, not generalized	“One possible explanation for the poor performance of the control subjects is that they were college students, not professional programmers. The goal of the experiment, however, was to demonstrate improvement due to the treatment condition. Since the groups were drawn from the same population, performance differences can only be attributed to the treatment. Without replicating the entire experiment using a population of professional programmers, one can only speculate that a similar performance difference would be observed among professionals, except that the base performance levels might have been higher.”
23	11	Discussed, not generalized	“One possible explanation for the poor performance of the control subjects is that they were college students, not professional programmers. The goal of the experiment, however, was to demonstrate improvement due to the treatment condition. Since the groups were drawn from the same population, performance differences can only be attributed to the treatment. Without replicating the entire experiment using a population of professional programmers, one can only speculate that a similar performance difference would be observed among professionals, except that the base performance levels might have been higher.”
29	14	Discussed, inconclusive	“The subjects were not professional software engineers. However, they were quite experienced programmers and held degrees (many of them advanced) in computer science.”
32	16	Discussed, not generalized	“A threat to subject generalizability may exist when the subject population is not drawn from the industrial population. This is not a concern here because our subjects are software professionals. Threats regarding subject and artifact representativeness arise when the subject and artifact population is not representative of the industrial population. This may endanger our study because our subjects are members of a development team, not a random sample of the entire development population and our artifacts are not representative of every type of software professional developers write.”
33	17	Discussed, not generalized	“Threats to external validity are those factors that limit the applicability of the experimental results to industry practice. Such threats include: the student reviewers may not be representative of professional programmers, the software reviewed may not be representative of professional software, and the inspection process may not be representative of industrial practice. These threats are real. Overcoming the first two threats is best accomplished by replication of this study using industrial programmers with real work products. To support this replication, our experimental materials and apparatus are freely available via the Internet (Johnson *et al.*, 1994). To minimize the third threat, the experimental review methods were based on descriptions of industrial practice of software review.”
33	18	Discussed, not generalized	“The graduate student subjects in our experiment may not be representative of software programming professionals. Although more than half of the subjects have 2 or more years of industrial experience, they are graduate students, not software professionals. Furthermore, as students they may have different motivations for participating in the experiment.”
36	195	Discussed, not generalized	“The subjects in our experiment may not be representative of software programming professionals. Although more than half of the subjects have 2 or more years of industrial experience, they are graduate students, not software professionals. Furthermore, as students they may have different motivations for participating in the experiment.”
38	21	Discussed, not generalized	“The subjects used in our experiments, while mature students many of whom had full-time jobs involving software development, might not be representative of the typical programmer. Generally, they had only a couple of years’ experience in commercial software development. Subjects with different backgrounds might perform differently on our experimental tasks; this is a potential avenue for future research.”

38	22	Discussed, not generalized	“The subjects used in our experiments, while mature students many of whom had full-time jobs involving software development, might not be representative of the typical programmer. Generally, they had only a couple of years’ experience in commercial software development. Subjects with different backgrounds might perform differently on our experimental tasks; this is a potential avenue for future research.”
38	23	Discussed, not generalized	“The subjects used in our experiments, while mature students many of whom had full-time jobs involving software development, might not be representative of the typical programmer. Generally, they had only a couple of years’ experience in commercial software development. Subjects with different backgrounds might perform differently on our experimental tasks; this is a potential avenue for future research.”
41	25	Discussed, generalized	“Second, the generality of the current results is tempered by the use of graduate students as managerial surrogates. Surveying over 250 recent studies on the dynamics of small social groups, Bettenhausen (Bettenhausen K.L., 1991) concluded that “... findings using student groups in manipulated settings often generalize to organizational settings as well or better than findings from intact groups that are frequently confounded by unique, unmeasured contextual factors.” In one particularly relevant study, Remus (Remus W.E., 1986) found no significant differences between students and managers in making production scheduling decisions. Although software project management decisions are somewhat different from production scheduling decisions, they are similar enough to apply his findings and assume that software engineering graduate students are acceptable surrogates in this experimental investigation.”
50	28	Discussed, generalized	“We selected a target population to which we wish to generalize the results of this study, viz., analysts and designers with some prior experience in process-oriented modeling (Cook *et al.*, 1979). We then used the principle of randomization to eliminate the possible confounding effects of nuisance variables. The two treatments, i.e., the OO and PO models, were informationally equivalent. Replication helped address the criticism of low generalizability leveled against experiments, and contributed to the external validity of the study. Finally, valid and reliable dependent variables were used to assess the outcomes of the experiment.”
50	29	Discussed, generalized	“We selected a target population to which we wish to generalize the results of this study, viz., analysts and designers with some prior experience in process-oriented modeling (Cook *et al.*, 1979). We then used the principle of randomization to eliminate the possible confounding effects of nuisance variables. The two treatments, i.e., the OO and PO models, were informationally equivalent. Replication helped address the criticism of low generalizability leveled against experiments, and contributed to the external validity of the study. Finally, valid and reliable dependent variables were used to assess the outcomes of the experiment.”
51	203	Discussed, not generalized	“Our conclusions are based on a specific experimental setting, i.e., certain tasks, subjects, and analysis methods. The tasks were moderate in size and complexity, and the subjects were either intermediate or advanced students in information systems engineering. The analysis methods were not trivial. We verified in several ways (as reported in Appendix C) that there were no significant differences between the two test groups in terms of their background and skill level. The efficiency (i.e., the time it takes to complete the task) of specification comprehension and specification generation was not considered as a factor in this experiment. The time allotted for both methods was equal, and the subjects of the experiment knew that their grade depended only on the effectiveness of their solutions, not on their efficiency.”

107	36	Discussed, not generalized	"One important question regarding external validity is whether the subjects are a representative sample of the population. The subjects (mostly undergraduate students, but also some graduate students and professional developers) of the experiment may not be representative of the "general programmer" Furthermore, according to Cockburn, the MF design is typical of the initial designs most students propose Thus, it is possible that the MF design has an unfair advantage when using students as experimental subjects The results may have been quite different if the subjects were OO design experts Thus, we cannot rule out that the subject selection may have biased the results Another threat is whether some subjects actually have read Cockburn's article series or otherwise knew the details of the designs prior to the experiment Because of randomization and the number of subjects involved in the experiment, we believe it is very unlikely that this have affected the results of the experiment."
109	37	Discussed, not generalized	"The largest threat to the external validity is the use of students as subjects. However, this threat is reduced by using fourth-year students which are close to finalise their education and start working in industry."
116	44	Discussed, not generalized	"While evidence has been found in support of the research model, the model needs to be revised to take into account the affects of human-computer interface constraints and the different speeds with which people work."
117	45	Discussed, not generalized	"Secondly, although every attempt was made to control the design, conduct and analysis of the experiments, it is inevitable that the laboratory set up of the experiments threatens the external validity of this research. This is especially so when we found that many defects were undiscovered by the review process in this research. The representativeness of the subjects to professional software engineers is thus questionable. Replication of this experiment in both the laboratory and real setting will help to confirm the findings."
120	48	Discussed, not generalized	"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."
121	49	Discussed, generalized	"The subjects in our initial runs may not be representative of software programming professionals. Although more than half of the subjects have 2 or more years of industrial experience, they are graduate students, not software professionals. Furthermore, as students they may have different motivations for participating in the experiment. This shouldn't be a problem in the replication using professional subjects."
123	51	Discussed, not generalized	"The student subjects involved in the experiment may not be representative of software engineering professionals. This was unavoidable since our choice of subjects was limited by available resources."
124	17	Discussed, not generalized	"Threats to external validity are those factors that limit the applicability of the experimental results to industry practice. Such threats include: the student reviewers may not be representative of professional programmers; the software reviewed may not be representative of professional software; and the inspection process may not be representative of industrial practice."
125	53	Discussed, not generalized	"The subjects in our replication may not be representative of the general software engineering population, e.g. this study used students rather than software professionals. This threat is always a problem, because of the lack of sampling frame, and hence even studies using professionals will be exposed to this threat."

130	59	Discussed, not generalized	"The subjects who participated in this study are unlikely to be representative of software professionals and therefore it is impossible to generalise the results to that population. However, it is argued that student based experiments can provide useful results for several reasons. First, they can be used to focus weak hypotheses on phenomena which appear to be important. These hypotheses can then be tested in more realistic settings with a better chance of important and interesting findings. Second, they can be used as a basis for deciding whether a hypothesis is worth investigating further in, e.g., an industrial case study. And third, they provide confirmatory power for any findings that are replicated in such a case study."
133	62	Discussed, not generalized	"The subjects who participated in the experiments may not be representative of software professionals. Although the participants in the replication and second experiment were a mixture of final year students and new graduate computer scientists and were classed as more experienced programmers, they cannot be categorised as experienced software professionals. For pragmatic considerations, having students as subjects was the only viable option for the laboratory-based experiments."
133	63	Discussed, not generalized	"The subjects who participated in the experiments may not be representative of software professionals. Although the participants in the replication and second experiment were a mixture of final year students and new graduate computer scientists and were classed as more experienced programmers, they cannot be categorised as experienced software professionals. For pragmatic considerations, having students as subjects was the only viable option for the laboratory-based experiments."
133	198	Discussed, not generalized	"The subjects who participated in the experiments may not be representative of software professionals. Although the participants in the replication and second experiment were a mixture of final year students and new graduate computer scientists and were classed as more experienced programmers, they cannot be categorised as experienced software professionals. For pragmatic considerations, having students as subjects was the only viable option for the laboratory-based experiments."
134	65	Discussed, not generalized	"Selection biases may have different effects due to interaction with the treatment. One factor we need to be aware of is that all our subjects were volunteers. This may imply that they are more prone to improvement-oriented efforts than the average developer - or it may indicate that they consider the experiment an opportunity to get away from normal work activities for a couple of days. Thus, the effects can strike in either direction. Also, all subjects had received training in their usual technique, a property that developers from other organizations may not possess."
135	66	Discussed, not generalized	"In this experiment, we did not tell the subjects that we were comparing two inspection techniques. The subjects only knew that they were supposed to use the assigned technique to detect as many usability problems as they could. We asked the subjects not to discuss with other subjects what they had done during the inspection before all subjects had finished participating in the experiment. Our impression was that the subjects were more interested in finding usability problems than using the techniques. The lab environment kept them concentrated on the inspection without distraction or interruption. The awareness that they were observed by others and video recorded may have some impact on their behavior. But since all these apply to both technique groups in the same way, they might not make a significant difference on the relative performance of the two techniques."
212	69	Discussed, not generalized	"The experiment has been conducted with N = 15 subjects, students of business informatics. They cannot be considered as representatives for software design experts found in industry."

214	71	Discussed, generalized	“Nonetheless, this threat to population validity may be reduced to some extent since it is likely that many student subjects will become professional programmers in less than a year from the time they participate in the experiment. As entry level professional programmers, it is also likely that they will be involved in software maintenance during the early years of their professional careers (Swanson *et al.*, 1990). Furthermore, nearly all student subjects would be expected to perform maintenance tasks while developing computer applications for various MIS and computer science courses. Thus it is reasonable to assume that the experimentally accessible population was at least similar to the target population.”
215	72	Discussed, not generalized	“However, students are not professional programmers. One difference between the two is that students generally have little or no experience debugging someone else's code (this may help explain the overall poor performance of the student subjects in this experiment). Certainly there are other differences between students and professional programmers, and therefore, results of this experiment must be considered carefully before making inferences to the professional programming environment.”
217	73	Discussed, inconclusive	“While we believe that students are appropriate subjects for experiments involving human decision-making under uncertainty, it should be noted that student subjects may have different criteria for judging the risk associated with business opportunities than those that would be exhibited by practicing managers. The results should therefore be interpreted with caution until the study is found to be replicable with practicing IS managers as subjects.”
218	74	Discussed, not generalized	“Our subjects may not be representative of the pool of software developers that professionally uses the UML for the analysis and design of object-oriented systems. However, they were all professional developers rather than students.”
219	75	Discussed, generalized	“Selection: This is the effect of performing the study with a population not representing industrial practice. This effect is not considered critical, since the study is performed with professional engineers.”
221	76	Discussed, generalized	“The experimental subjects were students, not professional programmers. However, the conditions in which the experiment took place were intended to mimic those in the real-world as far as possible; the students (who have at least 18 months programming experience) are expected to maintain and modify medium-sized systems, according to changing requirements, as well as be able to develop medium-sized software from specification.”
226	81	Discussed, generalized	“Subjects were 30 students in an undergraduate course in advanced systems analysis at a large US university. Since these subjects lacked other than token amounts of professional systems analysis experience, they are considered “novice” analysts. This is not altogether undesirable because, as Vessey and Conger (Vessey *et al.*, 1994) point out, novice analysts are not biased by experience with other methodologies and have not had time to adopt a personal ”favourite” methodology. All students were Junior or higher in standing and all had taken at least three prior courses in information systems, one of which was “Systems Analysis I”. Thus, all students were well versed in the role of system development methodologies in supporting system design work and all had considerable experience in data modeling with entity relationship diagrams (ERDs) and in process modeling with data flow diagrams (DFDs). Additionally, all students were thoroughly familiar with system development life cycles and with the basic steps involved in building IS applications. Students were given point credit toward their final course grade for participating in the experiment on the grounds that the additional training and system design experience connected with the experiment complemented the learning objectives of the course.”

232	87	Discussed, generalized	"Finally, the issue of using student programmers in experiments introduces the concern that such research does not directly apply to industry programmers. Holt *et al.* demonstrate that advanced students and professional programmers are statistically similar in terms of comparing their mental representation and various performance measures (Holt *et al.*, 1987). This provides support for using students in studies, especially for investigations where an industry validation is to be done, which is true for this research."
234	89	Discussed, not generalized	"Larger studies across the student populations of several institutions would ensure that the variation due to factors such as background, learning experience and environment could be taken into account."
235	91	Discussed, not generalized	"Lastly, the type of subjects is limiting factor. Although advanced students have the same cognitive abilities as their industry counterparts, they certainly lack the experience that comes with practice and working in the industry for several years."
249	174	Discussed, not generalized	"One of the limitations of this study is the use of student subjects. Hence, one should be careful in generalizing the results of this experiment to professional programmers. However, in the case of code comprehension, as opposed to other software tasks (e.g., systems analysis and program construction), empirical results using student subjects may be more extendable to professional programmers. The reason for this is that the task of concise code comprehension involves just that-concise code-regardless of whether or not the code segment is found in a large, industrial program or a smaller student program. At the same time, it should be recognized that students will probably not be as proficient at comprehending concise code as professionals."
402	95	Discussed, generalized	"The biggest threat to the external validity is that students were used during the experiment as subjects. However, students were in the end of their third year of studies in software engineering, close to their start working in the industry. There are more experiments reported in the literature, where students were successfully used as subjects (Höst *et al.*, 2000)(Tichy W.F., 2000)(Travassos *et al.*, 1999)."
403	98	Discussed, generalized	"The largest threat is that students are used as subjects. However, the students are in their third or fourth year of software engineering studies and hence close to start working in industry. The members of one of the subject groups are familiar with the application domain, which is industry-like, as they have developed a requirements specification for a similar, but more extensive system, in a previous course. The comparison between the two subject groups enables blocking with respect to domain knowledge and educational background.
404	99	Discussed, generalized	"In this particular study, the use of students is not critical since the objective is to study the outcome of the PSP for people having different background and experience. In particular, the differences related to ducational background are evaluated. The subjects (students) are, however, not chosen by random. They are chosen based on availability, i.e. the students taken the course. This is often referred to, as being convenience sampling (Robon C., 1993), and the study becomes a quasi-experiment due to the lack of randomisation of subjects."
513	100	Discussed, generalized	"The result of the questionnaire is that the students have similar and good knowledge in all the above mentioned areas. They have especially good domain knowledge in taxi management systems, since they have participated in a previous course in which they developed a requirement specification for taxi management system. Although they are students, the subjects may be compared with software developers, developing similar products in several following projects."

514	103	Discussed, inconclusive	“Students have limited industrial experience and may behave differently from software professionals regarding effort estimation impacts. Arguments in favour of a not too large difference between students and software professionals in our experiments are: - In our interviews with the professional developers and project leaders we found that they had not been taught how to estimate and that they did not get proper estimation accuracy feedback. This means there may not be a very large difference regarding estimation skill between students and software professionals. As reported in (Jørgensen *et al.*, 2000), the estimation skill may not improve very much with increased experience when there is no proper learning environment. - The students in the experiment described in Section 4 should estimate their own work. In this situation the student is a domain expert, i.e. it is a situation similar to the one a programmer faces when estimating a programming task where he/she is a domain (programming) expert, but no expert in effort estimation. - Most of the participants in the experiment described in Section 5 had several years of experience from software organisations. - Results in (Höst *et al.*, 2000) indicate no large differences between students and software professionals regarding assessment of lead-time impact.” “Both experiments, but in particular the second experiment, would have benefited from more subjects. Replications of the experiments are recommended to evaluate the validity of the results. Section 1.1 provides a link to the experiment material necessary for replications.”
514	104	Discussed, inconclusive	“Students have limited industrial experience and may behave differently from software professionals regarding effort estimation impacts. Arguments in favour of a not too large difference between students and software professionals in our experiments are: - In our interviews with the professional developers and project leaders we found that they had not been taught how to estimate and that they did not get proper estimation accuracy feedback. This means there may not be a very large difference regarding estimation skill between students and software professionals. As reported in (Jørgensen *et al.*, 2000), the estimation skill may not improve very much with increased experience when there is no proper learning environment. - The students in the experiment described in Section 4 should estimate their own work. In this situation the student is a domain expert, i.e. it is a situation similar to the one a programmer faces when estimating a programming task where he/she is a domain (programming) expert, but no expert in effort estimation. - Most of the participants in the experiment described in Section 5 had several years of experience from software organisations. - Results in (Höst *et al.*, 2000) indicate no large differences between students and software professionals regarding assessment of lead-time impact.” “Both experiments, but in particular the second experiment, would have benefited from more subjects. Replications of the experiments are recommended to evaluate the validity of the results. Section 1.1 provides a link to the experiment material necessary for replications.”

514	105	Discussed, inconclusive	"Students have limited industrial experience and may behave differently from software professionals regarding effort estimation impacts. Arguments in favour of a not too large difference between students and software professionals in our experiments are: - In our interviews with the professional developers and project leaders we found that they had not been taught how to estimate and that they did not get proper estimation accuracy feedback. This means there may not be a very large difference regarding estimation skill between students and software professionals. As reported in (Jørgensen *et al.*, 2000), the estimation skill may not improve very much with increased experience when there is no proper learning environment. - The students in the experiment described in Section 4 should estimate their own work. In this situation the student is a domain expert, i.e. it is a situation similar to the one a programmer faces when estimating a programming task where he/she is a domain (programming) expert, but no expert in effort estimation. - Most of the participants in the experiment described in Section 5 had several years of experience from software organisations. - Results in (Höst *et al.*, 2000) indicate no large differences between students and software professionals regarding assessment of lead-time impact." "Both experiments, but in particular the second experiment, would have benefited from more subjects. Replications of the experiments are recommended to evaluate the validity of the results. Section 1.1 provides a link to the experiment material necessary for replications."
515	106	Discussed, generalized	"Due to the difficulty of getting professionals to perform the experiments, the original experiment was done by students. In general, more experiments with a larger number of subjects, students and professionals, and with a greater difference between the values of each metric are necessary to obtain more conclusive results regarding the relationship between referential integrity and the analyzability of the relational databases, and, hence, their maintainability." "We tried to increase external validity by performing the replica with professionals, so the results could be more generalized."
520	110	Discussed, not generalized	"Firstly, one needs to be cautious about uncritically accepting the findings of single experiments, especially where small numbers of student subjects are employed. Replication is important since this allows us to have a far greater degree of confidence in the findings."
524	113	Discussed, not generalized	"Threats to external validity are factors which prevent the generalisation of the results to actual software engineering practice. Once again, these threats are the same as for earlier studies subjects, programs, faults, fault densities, or techniques may not be representative of software engineering practice. The first four threats are real and can only be addressed by repeated studies using different subjects, programs, faults and fault densities (hence the importance of replication)."
526	114	Discussed, not generalized	"Clearly the results for the generic documents cannot be generalized to specific application domain documents of the organization. However, the experiment was conducted with professional developers and also with documents from an industrial context which strengthens the ability to generalise. The limited number of data points is a potential threat to external validity but this can ultimately be overcome by further replication."
529	116	Discussed, generalized	"Experiments in a student setting can always be questioned concerning validity in an industrial environment. In this case, this is not regarded as particular critical as one objective of the course is to model an industrial environment. In particular it should be noted that the study is based on comparison of different methods for effort estimation and the evaluation should provide similar results independent of the environment (university or industry)."
702	120	Discussed, not generalized	"The subjects of the experiment (3 rd year computer science students) may not be representative of the general software engineering population."

708	121	Discussed, not generalized	"Some of these included the subjects used (they may not have been representative of the general software engineering population), the Java code (may not be representative in terms of style or complexity – it had eight classes but significant references to the Java API), and learning effect (as an unstructured technique ad-hoc inspection had to be carried out for both groups before systematic inspection - there may still have been a general learning effect)."
709	122	Discussed, generalized	"Of course, the subjects were students participating in a university course. As pointed out in the literature (Curtis B., 1986), students may not be representative of real developers. In our case, this translates to the fact that the participants may not be as effective in their defect detection activity as professional developers, i.e., they find fewer defects. However, this effect impacts all the DCETs in a similar manner. Hence, although our estimates may not be as accurate with students as with professional developers our findings are conservative with respect to the identification of the best models. Hence, our results exhibit a considerable degree of external validity."
710	122	Discussed, generalized	"Of course, the subjects were students participating in a university course. As pointed out in the literature (Curtis B., 1986), students may not be representative of real developers. In our case, this translates to the fact that the participants may not be as effective in their defect detection activity as professional developers, i.e., they find fewer defects. Hence, although our cost-benefit results may not be as good with students as with professional developers, our findings are conservative with respect to the calculation of cost-benefit levels. Hence, our results exhibit a considerable degree of external validity."
715	126	Discussed, not generalized	"For example, Computer Science students may not be representative of any sizable segment of the population of spreadsheet programmers. In particular, they cannot be said to be representative of end-user spreadsheet developers."
721	130	Discussed, not generalized	"The participants are not a sample drawn from industrial practitioners. The participants are Ph.D. students, but some of them have industrial experience. The participants are further discussed in Section 5.2."
727	17	Discussed, not generalized	"Threats to external validity are those factors that limit the applicability of the experimental results to industry practice. Such threats include: the student reviewers may not be representative of professional programmers, the software reviewed may not be representative of professional software, and the inspection process may not be representative of industrial practice. These threats are real. Overcoming the first two threats is best accomplished by replication of this study using industrial programmers with real work products. To support this replication, our experimental materials and apparatus are freely available via the Internet. To minimize the third threat, we based our experimental review methods on descriptions of industrial practice of software review, such as Gilb's Inspection (Gilb *et al.*, 1993)."
734	20	Discussed, not generalized	"the reviewers in the first run of our experiment may not be representative of software programming professionals;"
1007	147	Discussed, generalized	"Some skeptics might feel these results do not transfer to professional programmers. It is plausible that our inexperienced participants had so much more room for improvement than an experienced software engineer that DLDA might be worthless in practice, in spite of our results. However, we found evidence to the contrary in our data. For the test defect density, our data shows a clear trend that the more experienced participants actually obtained a larger improvement than the others. This is true no matter whether we measure experience in years of professional programming experience or by the length of the largest program the participants ever wrote."
1009	122	Discussed, generalized	"We used students as the sample, so the results could reasonably be generalized to people with a comparable background – possibly novice rather than professional developers."

1013	150	Discussed, not generalized	"The majority of participants in our study and in our survey were self-selected pair programmers. Further study is needed to examine the eventual satisfaction of programmers forced to pair program despite their resistance."
1103	153	Discussed, generalized	"Of course, the subjects are students participating in a university course. As pointed out in the literature (Curtis B. 1986), students may not be representative of real developers for software engineering tasks in general. On the other side, a recent report on estimation tasks found no significant differences between students and professional subjects (Höst *et al.*, 2000)."
1104	154	Discussed, generalized	"Our students may not be representative of the population of software professionals. However, a former experiment with NASA developers (Basili *et al.*, 1996) failed to reveal significant relationship between inspection effectiveness and reviewers' experience. Probably, being a software professional does not imply that the experience matches with the skills that are relevant to the object of study. Based on the behavioral theory of group performance, Sauer *et al.* (Sauer *et al.*, 2000) state that task expertise is the dominant determinant of review performance and recommend training to increase to develop reviewers' skills. Since this experiment was part of a software engineering course, we had a chance to train students on both defect detection techniques and inspection process."
1105	122	Discussed, generalized	"Of course, the subjects were students participating in a university course. As pointed out in the literature (Curtis B., 1986), students may not be representative of real developers. In our case, this can have two implications. First, participants may not be as effective in their defect detection activity as professional developers, i.e. they find fewer defects. Second, they find different (types of) defects than professionals. However, these effects impact all the DCETs in a similar manner. Hence, although our estimates may not be as accurate with students as with professional developers our findings are conservative with respect to the identification of the best models. Hence, our results expose some external validity."
1107	68	Discussed, generalized	"The subjects participating in the experiment were all computer science students at an advanced level. It can be expected that the results of the study are to some degree representative for this class of subjects. Any generalisation of the results with regard to education of novice students, or even with regard to training of software professionals should be done with caution."
1111	159	Discussed, not generalized	"Finally, the subjects of the experiment were students. Experiments with students are usually characterized by high internal but low external validity. This limits our possibility to generalize our findings."
1116	162	Discussed, not generalized	"The subjects in our experiment may not be representative of software programming professionals. Although more than half of the subjects have 2 or more years of industrial experience, they are graduate students, not software professionals. Furthermore, as students they may have different motivations for participating in the experiment."
1117	14	Discussed, inconclusive	"The subjects were not professional software engineers. However, they were quite experienced programmers and held degrees (many of them advanced) in computer science."

Appendix C

Table C.1 contains the complete set of quotes for all the experiments that addresses generalization of the environment. Remark that all references in the quotes are rewritten to be on the same format (Authors, Year), and to some extent recognizable. The references in the quotes are not present in the reference list of this thesis.

Table C.1: Quotes – Generalization from environment

Art.	Exp.	My opinion	Quote
11	1	Discussed, not generalized	“First, the original designers and implementors may be the ones who maintain the program. This was not the case in our experiment and our results do not apply to such cases. The maintainers may also have more pattern experience than our participants. The consequences of this difference are unclear; but we do not believe them to be dramatic.” “Third, real maintainers implement and test their solutions (instead of only writing them on paper), that will typically trade some of the incorrectness observed in the experiment against additional time. Furthermore, without an explicit theory of SW maintenance, it is difficult to predict what effect other design patterns (and alternatives) than the five specific ones used in the experiment may have.”
12	2	Discussed, not generalized	“Realistic programs are usually team work. Individual tasks during maintenance may also often be performed by more than one programmer. Such cooperation requires additional communication about the program. In this case, PCL may have further advantages, not visible in the experiments, because one of the major (purported) advantages of design patterns is a common design terminology (Unger *et al.*, 2000).” “Compared to typical industrial size programs, the experiment programs are rather small and simple, neatly designed, and wellcommented. This does not necessarily invalidate the results of the experiments, though. If a positive effect is found, increasing program complexity may magnify the effect because having PCL provides program slicing information. For pattern-relevant tasks, PCL information points out which parts of a program are relevant and enables one to ignore the rest; such information may become more useful as program size increases because more code can be ignored.”
12	3	Discussed, not generalized	“Realistic programs are usually team work. Individual tasks during maintenance may also often be performed by more than one programmer. Such cooperation requires additional communication about the program. In this case, PCL may have further advantages, not visible in the experiments, because one of the major (purported) advantages of design patterns is a common design terminology (Unger *et al.*, 2000).” “Compared to typical industrial size programs, the experiment programs are rather small and simple, neatly designed, and wellcommented. This does not necessarily invalidate the results of the experiments, though. If a positive effect is found, increasing program complexity may magnify the effect because having PCL provides program slicing information. For pattern-relevant tasks, PCL information points out which parts of a program are relevant and enables one to ignore the rest; such information may become more useful as program size increases because more code can be ignored.”
17	6	Discussed, not generalized	“Our study was performed with subjects and code documents from a single organization. While this enjoys greater external validity than doing studies with students in a “laboratory” setting, it is uncertain the extent to which the results can be generalized to other organizations.” “In this study, we assume that defect detection is an individual rather than a group activity. However, other inspection processes in industry may exist that consider defect detection a group activity, such as the one presented in (Fagan M., 1976).”

17	193	Discussed, not generalized	“Our study was performed with subjects and code documents from a single organization. While this enjoys greater external validity than doing studies with students in a “laboratory” setting, it is uncertain the extent to which the results can be generalized to other organizations.” “In this study, we assume that defect detection is an individual rather than a group activity. However, other inspection processes in industry may exist that consider defect detection a group activity, such as the one presented in (Fagan M., 1976).”
17	194	Discussed, not generalized	“Our study was performed with subjects and code documents from a single organization. While this enjoys greater external validity than doing studies with students in a “laboratory” setting, it is uncertain the extent to which the results can be generalized to other organizations.” “In this study, we assume that defect detection is an individual rather than a group activity. However, other inspection processes in industry may exist that consider defect detection a group activity, such as the one presented in (Fagan M., 1976).”
18	7	Discussed, not generalized	“First, and most importantly, different work conditions than found in the experiment may positively or negatively influence the effectiveness of the PSP training. This is discussed in Section 4. Second, the PSP education of our subjects was only a short time ago. Long-term effects would be more interesting to see.”
19	8	Discussed, generalized	“We think that several significant features of the working environment are common to many other software firms: Business oriented application, RPG-based environment, Commitment to improvement, Project tracking mechanism (i.e., measurement program).”
32	16	Discussed, not generalized	“Experimental scale is a threat when the experimental setting or the materials are not representative of industrial practice. We avoided this threat by conducting the experiment on a live software project.”
39	24	Discussed, not generalized	“The most important future work is to replicate this study with other subjects, different or larger databases, different user interfaces, different indexers, different domains and domain experience, and other representation methods. It would also be valuable to rerun the experiment with subjects who had more experience using the methods. Such replications are necessary, as they are with all scientific experiments, to strengthen the validity of the findings reported here.”
41	25	Discussed, inconclusive	“First, the experimental setting was a computer-based laboratory experiment. Laboratory research entails giving up the richness of context to obtain control. For example, to control for information feedback to the experimental subjects, feedback information was limited to status reports. In reality project managers may rely on more than status reports in determining project status (including conversations with programmers, interviews, etc.). The question is, what price did we pay for such a simplification?”
107	36	Discussed, not generalized	“The changes were coded with pen and paper. This represents another important threat to the external validity. Using a computer one has access to advanced editors, multiple windows, class browsers, etc. Some subjects preferred an exploratory approach to changing the program, which may be difficult to do with pen and paper compared with using a computer. For this experiment, the designs and the change tasks were small. Furthermore, there was a quite even distribution of subjects characterizing their solution approach as exploratory for the MF and RD designs (Section 4 6). Finally, the students are accustomed to working with pen and paper programs on their written exams. This means that the advantage of using a computer is probably not that great. Using a computer would have introduced many new problems regarding training, learning effects and biases towards certain solution approaches depending on the available tool functionality. In this particular experiment, it was in our opinion a better approach to use pen and paper rather than a computer. Still, the only way to eliminate the resulting threats is to replicate the experiment using computers instead of pen and paper.”

116	44	Discussed, not generalized	“While evidence has been found in support of the research model, the model needs to be revised to take into account the affects of human-computer interface constraints and the different speeds with which people work.”
117	45	Discussed, not generalized	“Firstly, it only deals with procedural roles. Other role types (we call expertise roles) are also worth investigation. We also make no assumption about role selection in this research. It is possible that certain reviewer’s characteristics are more suited for certain roles. Secondly, although every attempt was made to control the design, conduct and analysis of the experiments, it is inevitable that the laboratory set up of the experiments threatens the external validity of this research. This is especially so when we found that many defects were undiscovered by the review process in this research.”
127	55	Discussed, not generalized	“The primary threat to the external validity of this experiment is that subjects’ response times were measured in a laboratory environment rather than in a typical work environment. This is a typical problem necessitated by the need to control other aspects of the environment.”
127	56	Discussed, not generalized	“The primary threat to the external validity of this experiment is that subjects’ response times were measured in a laboratory environment rather than in a typical work environment. This is a typical problem necessitated by the need to control other aspects of the environment. The general applicability to visual programming languages is limited by the fact that this experiment used decision statements from a data flow programming languages. The implications of this research to functional or more typical imperative visual languages and to other control structure, e.g. loops, is limited by this.”
133	62	Discussed, not generalized	“Furthermore, it may be that to control and isolate the effect of inheritance on the maintainability of object-oriented software, small systems are required otherwise the effect may become too difficult to detect. As noted by Tiller, more control exerted over an experiment is gained only at the expense of its realism (Tiller D., 1991), an attempt to achieve as fine a balance as possible was made. Although maintainability of software is best evaluated with respect to the entire maintenance process, laboratory-based experimentation on such a scale is not practical; this study has concentrated on the implementation phase of the maintenance process.”
133	63	Discussed, not generalized	“Furthermore, it may be that to control and isolate the effect of inheritance on the maintainability of object-oriented software, small systems are required otherwise the effect may become too difficult to detect. As noted by Tiller, more control exerted over an experiment is gained only at the expense of its realism (Tiller D., 1991), an attempt to achieve as fine a balance as possible was made. Although maintainability of software is best evaluated with respect to the entire maintenance process, laboratory-based experimentation on such a scale is not practical; this study has concentrated on the implementation phase of the maintenance process.”
133	198	Discussed, not generalized	“Furthermore, it may be that to control and isolate the effect of inheritance on the maintainability of object-oriented software, small systems are required otherwise the effect may become too difficult to detect. As noted by Tiller, more control exerted over an experiment is gained only at the expense of its realism (Tiller D., 1991), an attempt to achieve as fine a balance as possible was made. Although maintainability of software is best evaluated with respect to the entire maintenance process, laboratory-based experimentation on such a scale is not practical; this study has concentrated on the implementation phase of the maintenance process.”

134	65	Discussed, inconclusive	“A pretest may affect the subject’s sensitivity of the experimental variable. Both of our groups receive similar pretests and treatments, so this effect may be of concern to us. We cannot avoid the fact that this is an experimental environment, and all subjects knew that. This, by itself, may affect the results and is a limitation of almost any experimental design.” “These effects are due to the experimental environment. In 1994, the pilot study was carried out in the subjects’ own environment, and thus would be valid also in a real setting. We cannot assume the same for the 1995 results since this run was done in a classroom situation. However, the change of experimental environment between the experiment runs has made it easier to concentrate on the techniques and tests to be done, thus separating the techniques better.”
214	71	Discussed, not generalized	“There are several limitations to this study. First, the investigations comprising this research are essentially restricted to laboratory experimentation thereby limiting the external validity (generalization) of the results. Even though conclusions drawn from experimental research (performed in laboratory settings) are empirically stronger than those drawn from non-experimental research conducted in the field, field studies are preferred for their generality and testing of real world phenomena. (Chapanis A., 1983) points out that laboratory experiments are, at best, rough and approximate models of real-life situations and can select only a few independent variables for testing. Second, while psychometric properties of the instrument used to group subjects based on their semantic knowledge have been validated, a systematic evaluation of the psychometric properties of the semantic knowledge instrument through several replications is required before the validity of this instrument can be unequivocally established. The factor analytic method used in deriving and validating the semantic knowledge instrument is not as comprehensive or acceptable as other methods such as Multi-Trait Multi Method (MTMM) for validating instruments.” “The next limitation of this study is the limited manipulation of the independent variables. All independent variables in this study are classified dichotomously as either high or low. Such dichotomous measurement allows only for relative analysis and thus the applicability of the results of the study to real-world situations. For example, the results of the study indicate that presence of time pressure 6 reduces software maintenance effort for programs with poor documentation characteristics. However, no conclusions can be drawn on how much or what level of time pressure is required to induce such reduction in maintenance effort.”
217	73	Discussed, not generalized	“It should also be noted that there are many factors in an organizational setting that may influence decision-making but which were not modeled here. Examples include organizational politics, the presence or absence of competing projects, and so forth. In addition, those variables that were manipulated may have been assigned values that are not characteristic of ’average’ or typical IS projects. For example, projects that threaten a company’s survival (which was part of our operationalization of ``high" magnitude of potential loss) may occur rather infrequently. The manipulations chosen for this experiment were designed to maximize our ’signal to noise’ ratio in testing the relationships among the constructs specified in our model. We make no claim here that they generalize to typical project decisions. Finally, it should be noted that our probability of failure manipulation was numeric while our magnitude of potential loss manipulation was non-numeric. While it is possible that this may have created certain biases in how the subjects responded to the treatments, we do not believe that it poses a significant threat to the design of the experiment or the interpretation of the results. As noted earlier, our manipulation checks clearly show that the range of perceptions recorded on the two variables of interest was comparable, suggesting that the perceptions of what constituted low vs. high were similar for the two manipulated variables despite differences in how they were manipulated.”

219	75	Discussed, generalized	“Setting: This is the effect of performing the study in a setting not representing industrial practice. Since the study is performed in an industrial project and the reviewed code will be part of delivered products, this effect is not considered critical.”
234	89	Discussed, not generalized	“Larger studies across the student populations of several institutions would ensure that the variation due to factors such as background, learning experience and environment could be taken into account.”
252	177	Discussed, not generalized	“Although these results seem positive, the reader should remember that these experiments are small in size. It is not possible to draw dogmatic conclusions from experiments with small sample sizes, such as these. However these results do provide one initial data point for understanding the problems associated with certain verification techniques and safety-critical faults. More experiments are necessary before we can draw firmer conclusions about these results. For example, some possible experiments could look at different verification methods, different locations, different safety-critical systems, specific life cycle phases, etc.”
514	103	Discussed, not generalized	“Although we tried to make the experimental situations realistic, we cannot be sure that the experimental situation had no unwanted impacts on the estimation or development process. For example, the students may have ignored the instruction that the preplanning effort estimate was not based on historical data or expert knowledge because it came from their lecturer, not because pre-planning effort estimate functioned as an anchor value. The development tasks in the second experiment were much smaller than most real-life tasks. Therefore, the results from that experiment are mainly valid for small programming tasks. Large-scale experiments in more realistic environments are needed.”
514	104	Discussed, not generalized	“Although we tried to make the experimental situations realistic, we cannot be sure that the experimental situation had no unwanted impacts on the estimation or development process. For example, the students may have ignored the instruction that the preplanning effort estimate was not based on historical data or expert knowledge because it came from their lecturer, not because pre-planning effort estimate functioned as an anchor value. The development tasks in the second experiment were much smaller than most real-life tasks. Therefore, the results from that experiment are mainly valid for small programming tasks. Large-scale experiments in more realistic environments are needed.”
514	105	Discussed, not generalized	“Although we tried to make the experimental situations realistic, we cannot be sure that the experimental situation had no unwanted impacts on the estimation or development process. For example, the students may have ignored the instruction that the preplanning effort estimate was not based on historical data or expert knowledge because it came from their lecturer, not because pre-planning effort estimate functioned as an anchor value. The development tasks in the second experiment were much smaller than most real-life tasks. Therefore, the results from that experiment are mainly valid for small programming tasks. Large-scale experiments in more realistic environments are needed.”
529	116	Discussed, generalized	“No particular instrumentation concerning data collection was carried out in Step 1. However, we believed that, based our insight into the projects during the execution, that we are able to make rather good interpretations.” “Reliability of the data (the students may not be reporting properly) The progress reporting weekly is believed to make these risks rather small.” “Division into project groups and lack of actual instrumentation. The division into groups is made by the students and it may lead to some groups knowing each other better than others.”
601	16	Discussed, not generalized	“All our results were obtained from one project, in one application domain, using one language and environment, within one software organization. Therefore, we cannot claim that our conclusions have general applicability, until our work has been replicated.”

709	122	Discussed, generalized	“With respect to external validity, we took a specification from a real application context to deal with an inspection object that was representative of an industrial development situation. Moreover, we used inspection activities that had been implemented in a number of professional development environments (Laitenberger *et al.*, 2000).”
710	122	Discussed, generalized	“With respect to external validity, we took a specification from a real application context to deal with an inspection object that was representative of an industrial development situation. Moreover, we used inspection activities that had been implemented in a number of professional development environments (Laitenberger *et al.*, 2000).”
1009	122	Discussed, generalized	“We took a specification from a real application context to deal with an inspection object that was representative of real development specifications. We used a classroom setting in order to control the experiment environment.”
1013	150	Discussed, not generalized	”We would like to run another university study to analyze the effect of pair programming on larger groups. Finally, we would like to see the same experiments applied in an industrial setting— perhaps with part of a larger development team.”
1105	122	Discussed, generalized	“With respect to external validity, we took a specification from a real application context to deal with an inspection object that was representative of an industrial development situation. Moreover, we used inspection activities that had been installed in a number of professional development environments (Laitenberger *et al.*, 2000).”
1113	160	Discussed, not generalized	“Single person estimating. In practice, an estimate is reviewed and quality assured by at least one other experienced person. Thus, it is a group effort, not a single person effort. The worst cases of human performance would therefore not occur in practice. However, we had to make a trade-off between the realism on one side and on the other side getting a large enough sample to permit statistical analysis.”

Appendix D

Table D.1 contains the complete set of quotes for all the experiments that addresses generalization of tasks. Remark that all references in the quotes are rewritten to be on the same format (Authors, Year), and to some extent recognizable. The references in the quotes are not present in the reference list of this thesis.

Table D.1: Quotes – Generalization of tasks

Art.	Exp.	My opinion	Quote
11	1	Discussed, not generalized	"Second, real programs will often be less well documented than the experiment programs, real programs are typically larger, and change tasks rarely revolve closely around a design pattern. The effects of such differences probably differ from one case to the next." "Third, real maintainers implement and test their solutions (instead of only writing them on paper), that will typically trade some of the incorrectness observed in the experiment against additional time. Furthermore, without an explicit theory of SW maintenance, it is difficult to predict what effect other design patterns (and alternatives) than the five specific ones used in the experiment may have."
12	2	Discussed, not generalized	"It is unknown whether the programs and tasks used in our experiments are (or are not) representative of realistic maintenance situations. We have but one indication that our programs are at least not totally different from other programs constructed using design patterns: The ratio of the total number of classes in the program to the number of design pattern instances found in our programs ranges between 3.0 and 5.5. These values are comparable to those found for Java AWT (3.8) and NextStep (3.1) (Gramberg O., 1997). Our article does not claim anything about maintenance tasks that are not pattern-relevant. See the conclusion section for more discussion of pattern-relevant tasks in realistic programs." "Realistic maintenance situations will often be rather different from those found by our subjects. In particular, much larger and more complex programs and tasks may require making changes based on a much lower degree of overall program understanding than could be obtained for the small programs in the experiments. It is hard to say whether or when this will make PCL more useful or less useful than in the experiments. Furthermore, if programmers have to master a large design pattern repertoire, their understanding of individual patterns may be reduced and PCL may become less helpful."
12	3	Discussed, not generalized	"It is unknown whether the programs and tasks used in our experiments are (or are not) representative of realistic maintenance situations. We have but one indication that our programs are at least not totally different from other programs constructed using design patterns: The ratio of the total number of classes in the program to the number of design pattern instances found in our programs ranges between 3.0 and 5.5. These values are comparable to those found for Java AWT (3.8) and NextStep (3.1) (Gramberg O., 1997). Our article does not claim anything about maintenance tasks that are not pattern-relevant. See the conclusion section for more discussion of pattern-relevant tasks in realistic programs." "Realistic maintenance situations will often be rather different from those found by our subjects. In particular, much larger and more complex programs and tasks may require making changes based on a much lower degree of overall program understanding than could be obtained for the small programs in the experiments. It is hard to say whether or when this will make PCL more useful or less useful than in the experiments. Furthermore, if programmers have to master a large design pattern repertoire, their understanding of individual patterns may be reduced and PCL may become less helpful."

16	5	Discussed, not generalized	"The materials used in this study, i.e., the software designs and tasks subjects were asked to complete, may not be representative in terms of their size and complexity."
17	6	Discussed, not generalized	"Our study was performed with subjects and code documents from a single organization. While this enjoys greater external validity than doing studies with students in a "laboratory" setting, it is uncertain the extent to which the results can be generalized to other organizations." "The code documents used in this study can be claimed to be representative of industrial code documents. However, we cannot generalize our findings for other type of documents, such as design or requirements documents."
17	193	Discussed, not generalized	"Our study was performed with subjects and code documents from a single organization. While this enjoys greater external validity than doing studies with students in a "laboratory" setting, it is uncertain the extent to which the results can be generalized to other organizations." "The code documents used in this study can be claimed to be representative of industrial code documents. However, we cannot generalize our findings for other type of documents, such as design or requirements documents."
17	194	Discussed, not generalized	"Our study was performed with subjects and code documents from a single organization. While this enjoys greater external validity than doing studies with students in a "laboratory" setting, it is uncertain the extent to which the results can be generalized to other organizations." "The code documents used in this study can be claimed to be representative of industrial code documents. However, we cannot generalize our findings for other type of documents, such as design or requirements documents."
18	7	Discussed, not generalized	"Third, our task was unusual in several respects (small size, precise requirements, acceptance test indicates expected outputs). It is unknown how these properties might influence the comparison."
29	14	Discussed, not generalized	"The tasks, although quite small, were not at all trivial. The subjects had to understand several important concepts of Motif programming (such as widget, resource, and callback function). Furthermore, they had to learn to use them from a reference manual only, without example programs; we used no examples as we felt that these would have made the programming tasks too simple. Typically, the subjects took between one and two hours for their first task and about half that time for their second." "One must be careful generalizing the results of this study to other situations. For instance, the experiment is unsuitable for determining the proportion of interface defects in an overall mix of defects, because it was designed to prevent errors other than interface errors. Hence it is unclear how large the differences will be if defect classes such as declaration defects, initialization defects, algorithmic defects, or control-flow defects are included." "The results may be domain dependent. This objection cannot be ruled out. This experiment should therefore be repeated in domains other than graphical user interfaces. The results may or may not apply to situations in which the subjects are very familiar with the interfaces used. This question might also be worth a separate experiment."
32	16	Discussed, not generalized	"Threats regarding subject and artifact representativeness arise when the subject and artifact population is not representative of the industrial population. This may endanger our study because our subjects are members of a development team, not a random sample of the entire development population and our artifacts are not representative of every type of software professional developers write." "Experimental scale is a threat when the experimental setting or the materials are not representative of industrial practice. We avoided this threat by conducting the experiment on a live software project."

33	17	Discussed, not generalized	“Threats to external validity are those factors that limit the applicability of the experimental results to industry practice. Such threats include: the student reviewers may not be representative of professional programmers, the software reviewed may not be representative of professional software, and the inspection process may not be representative of industrial practice. These threats are real. Overcoming the first two threats is best accomplished by replication of this study using industrial programmers with real work products. To support this replication, our experimental materials and apparatus are freely available via the Internet (Johson *et al.*, 1994). To minimize the third threat, the experimental review methods were based on descriptions of industrial practice of software review.”
33	18	Discussed, not generalized	“The specification documents may not be representative of real programming problems. The experimental specifications are atypical of industrial SRS in two ways. First, most of the experimental specification is written in a formal requirements notation (see Section 4.3.6). Although some industrial groups are experimenting with formal notations (Ardis M.A., 1994), (Gerhart *et al.*, 1994), it is not the industry’s standard practice. Second, the specifications used are considerably shorter than typical industrial specifications.” “Finally, the review process in our experimental design may not be representative of software development practice. We have modeled our experiment’s review process after the ones used in many development organizations, although each organization may adapt the process to fit its specific needs. Another difference is that the SRS authors are not present at our reviews, although in practice they normally would be. Finally, industrial reviewers may bring more domain knowledge to a review than our student subjects did.”
36	195	Discussed, not generalized	“The inspection process in our experimental design may not be representative of software development practice. We have modeled our experiment's inspection process after the one used in several development organizations within AT&T (Eick *et al.*, 1992). Although this process is similar to a Fagan-style inspection, there are some differences. One difference is that reviewers use the fault detection activity to find faults, not just to prepare for the inspection meeting. Another difference is that during the collection meeting reviewers are given specific technical roles such as test expert or end-user only if the author feels there is a special need for them.” “The specification documents may not be representative of real programming problems. Our experimental specifications are atypical of industrial SRS in two ways. First, most of the experimental specification is written in a formal requirements notation. (See Section 1I.B.) Although several groups at AT&T and elsewhere are experimenting with formal notations (Ardis M.A., 1994), (Gerhart *et al.*, 1994), it is not the industry's standard practice. Secondly, the specifications are considerably smaller than industrial ones.”
39	24	Discussed, not generalized	“The most important future work is to replicate this study with other subjects, different or larger databases, different user interfaces, different indexers, different domains and domain experience, and other representation methods. It would also be valuable to rerun the experiment with subjects who had more experience using the methods. Such replications are necessary, as they are with all scientific experiments, to strengthen the validity of the findings reported here.”
42	26	Discussed, not generalized	“Another limitation comes from the fact that the DBMS used in this research was designed primarily for instructional and research purposes. As such it offered some facilities usually not directly available in a commercial environment. These special features allowed a cross-translation between various query languages and made the results of every phase of the query transformation during its execution available to the user. It is possible that this feedback may have confused some users and reduced their overall performance. However, the fact that the subjects received hands-on training and practice sessions should have reduced this problem to a minimum.”

51	203	Discussed, inconclusive	"Our conclusions are based on a specific experimental setting, i.e., certain tasks, subjects, and analysis methods. The tasks were moderate in size and complexity, and the subjects were either intermediate or advanced students in information systems engineering. The analysis methods were not trivial. We verified in several ways (as reported in Appendix C) that there were no significant differences between the two test groups in terms of their background and skill level. The efficiency (i.e., the time it takes to complete the task) of specification comprehension and specification generation was not considered as a factor in this experiment. The time allotted for both methods was equal, and the subjects of the experiment knew that their grade depended only on the effectiveness of their solutions, not on their efficiency."
105	33	Discussed, not generalized	"First, it should be noted that phase was confounded with task orientation versus task performance. Thus, it is not possible to determine whether the changes observed in phase 2 were the result of performance of the task or of additional time to study the program." "Second, participants worked with a single program which implemented a database. To generalize the results it is necessary to repeat the study with other programs in other problem domains. Third, while the program was larger than often used in this kind of study, it was still a small program by industrial standards. Thus, we do not know whether the mental representation of a much larger program would conform precisely to what we found here." "Fourth, in our study participants worked with the program for approximately 2 h, and most did not have time to finish the reuse or documentation task they were given. We might have observed further evolution of the mental representation if they had worked with the program over a longer time."
107	36	Discussed, not generalized	"For this experiment, the designs and the change tasks were small. Furthermore, there was a quite even distribution of subjects characterizing their solution approach as exploratory for the MF and RD designs (Section 4 6)."
109	37	Discussed, not generalized	"The setting is intended to resemble a real inspection situation, but the process that the subjects participate in is not part of a real software development project. The assignments are also intended to be realistic, but the documents are rather short, and real software requirements documents may include many more pages. The threats to external validity regarding the settings and assignments are, however, considered limited, as both the inspection process and the documents resemble real cases to a reasonable extent."
113	41	Discussed, not generalized	"Nature of problem: The problem is a non-trivial problem that the participants worked with for about half an hour. However, it involves a lot of knowledge about software engineering and it is not trivial, and as it was argued in Section 2.1, it is an important area in software engineering."
120	48	Discussed, not generalized	"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." "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."

121	49	Discussed, not generalized	“The inspection process in our experimental design may not be representative of software development practice. We have modeled our experiment’s inspection process after the one used in several development organizations within AT&T (Eick *et al.*, 1992). Although this process is similar to a Fagan-style inspection, there are some differences. One difference is that reviewers use the fault detection activity to to find faults, not just to prepare for the inspection meeting. Another difference is that during the collection meeting reviewers are given specific technical roles such as test expert or end-user only if the author feels there is a special need for them.“ “The specification documents may not be representative of real programming problems. Our experimental specifications are atypical of industrial SRS in two ways. First, most of the experimental specification is written in a formal requirements notation. (See Section 2.2.) Although several groups at AT&T and elsewhere are experimenting with formal notations (Ardis M.A., 1994; Gerhart *et al.*, 1994), it is not the industry’s standard practice. Secondly, the specifications are considerably smaller than industrial ones.”
123	51	Discussed, not generalized	“The inspection process used may not correspond to that used in industry, in terms of process steps and number of participants. For example, the process used did not involve the author presenting an overview of the product, and a rework phase was not used. However, the detection/collection approach used in our experiment is a standard process (Gilb *et al.*, 1993).” “The programs used may not be representative of the length and complexity of those found in an industrial setting. The programs used were chosen for their length, allowing them to be inspected within the time available. However, the amount of time given to inspect each program was representative of industrial practice quoted in popular inspection literature.”
124	17	Discussed, not generalized	“Threats to external validity are those factors that limit the applicability of the experimental results to industry practice. Such threats include: the student reviewers may not be representative of professional programmers; the software reviewed may not be representative of professional software; and the inspection process may not be representative of industrial practice.”
125	53	Discussed, not generalized	“The inspection process may not be representative of industrial software development practice. Despite using a well known and widely used inspection technique (Gilb *et al.*, 1993), many other inspection processes exist in industry which pose a threat to the ability to generalise from this experiment.” “The specification documents may not be representative of industrial problems. The documents used in this study are smaller and less complex than industrial specifications.”

126	54	Discussed, not generalized	“As concerns external validity, analyzing the effect of critics in other types of languages and tasks would be of great interest to furthering the acceptability of critics so they can complement the non-textual, after error capabilities of traditional debugger routines. The most interesting issue may be the impact of visual vs. non-visual programming language. COPE is a visual programming language in the tradition of Visual Basic, Macromedia Director, Icon Author, Asymmetrix Toolbook, HyperCard, JavaScript, and many others. These languages expect programmers to make significant use of built in metaphors, reusable widgets, object default property lists, toggle settings, and script libraries. Programmers in these languages don’t have to think very deeply about code, loops, scripts, etc. So when a non-textual debugger pointing to a line of defective script pops up during execution, theymay not be very attuned to what it is trying to point out. Bycontrast, non-visual languages (e.g., C, C++, FORTRAN, PASCAL, etc.) expect programmers to focus on abstract data structures, structural design, and fine grained programming constructs. Nontextual debuggers that cue the point of error may be enough detail for these programmers. The point is that traditional (non-textual, after error) debuggers arose in the non-visual languages where programmers are mentally attuned to the context. These debuggers have simply been transported without change to the new, visual paradigm. It seems reasonable that a new paradigm with new types of programmers warrants a new set of critic/debugger designs such as found in this study. An interesting question for future research is whether traditional debuggers are best suited to traditional languages, and critics are best suited to visual programming languages. Such a study would also clarify whether the results found in the current paper apply to non-visual languages, as well as visual ones.”
127	55	Discussed, not generalized	“Another threat to the external validity of these experiments is that the tasks involved are very simple. Comprehension of decision statements is only a small portion of the complex process of software development.” ”The general applicability to visual programming languages is limited by the fact that this experiment used decision statements from a data flow programming languages. The implications of this research to functional or more typical imperative visual languages and to other control structure, e.g. loops, is limited by this.”
127	56	Discussed, not generalized	“Another threat to the external validity of these experiments is that the tasks involved are very simple. Comprehension of decision statements is only a small portion of the complex process of software development.” ”The general applicability to visual programming languages is limited by the fact that this experiment used decision statements from a data flow programming languages. The implications of this research to functional or more typical imperative visual languages and to other control structure, e.g. loops, is limited by this.”
129	58	Discussed, not generalized	“Threats to external validity include the short duration of the exercise, and both the uniqueness and prototype nature of the process guidance system.”
130	59	Discussed, not generalized	“The materials used in this study, i.e., the software systems and tasks subjects were asked to complete, may not be representative in terms of their size and complexity.”
133	62	Discussed, not generalized	“The software systems used for the experiments were not large and may not be representative of real software systems. The inheritance depth used in these software systems is representative of real inheritance hierarchies, however-see the characteristics of object-oriented class hierarchies presented in (Chidamber S. *et al.*, 1994). Furthermore, it may be that to control and isolate the effect of inheritance on the maintainability of object-oriented software, small systems are required otherwise the effect may become too difficult to detect. As noted by Tiller, more control exerted over an experiment is gained only at the expense of its realism (Tiller D., 1991), an attempt to achieve as fine a balance as possible was made. Although maintainability of software is best evaluated with respect to the entire maintenance process, laboratory-based experimentation on such a scale is not practical; this study has concentrated on the implementation phase of the maintenance process.”

133	63	Discussed, not generalized	“The software systems used for the experiments were not large and may not be representative of real software systems. The inheritance depth used in these software systems is representative of real inheritance hierarchies, however-see the characteristics of object-oriented class hierarchies presented in (Chidamber S. *et al.*, 1994). Furthermore, it may be that to control and isolate the effect of inheritance on the maintainability of object-oriented software, small systems are required otherwise the effect may become too difficult to detect. As noted by Tiller, more control exerted over an experiment is gained only at the expense of its realism (Tiller D., 1991), an attempt to achieve as fine a balance as possible was made. Although maintainability of software is best evaluated with respect to the entire maintenance process, laboratory-based experimentation on such a scale is not practical; this study has concentrated on the implementation phase of the maintenance process.”
133	198	Discussed, not generalized	“The software systems used for the experiments were not large and may not be representative of real software systems. The inheritance depth used in these software systems is representative of real inheritance hierarchies, however-see the characteristics of object-oriented class hierarchies presented in (Chidamber S. *et al.*, 1994). Furthermore, it may be that to control and isolate the effect of inheritance on the maintainability of object-oriented software, small systems are required otherwise the effect may become too difficult to detect. As noted by Tiller, more control exerted over an experiment is gained only at the expense of its realism (Tiller D., 1991), an attempt to achieve as fine a balance as possible was made. Although maintainability of software is best evaluated with respect to the entire maintenance process, laboratory-based experimentation on such a scale is not practical; this study has concentrated on the implementation phase of the maintenance process.”
212	69	Discussed, not generalized	“Moreover, the application systems to be modelled were too small; thus a generalisation of the results to very large applications seems not to be justified. Nevertheless, the results can be taken as a first indicator that the coarse-grained object-oriented concepts of OML and TOS are more appropriate for structuring database-oriented application systems than those of UML.”
214	71	Discussed, not generalized	“Finally, the fifth limitation involves the nature of the tasks used in this study. The size of the tasks was dictated by the limited availability of the subjects for the experiments. Tasks were designed so that all subjects would be able to complete them within a reasonable amount of time (i.e., 2 h). This led to using C programs whose size ranged from 16 to 97 lines of code. Though these programs were larger than those used in earlier studies on software maintenance, they may not be representative of the size of programs in industry and thus may restrict the external validity of this study.”
215	72	Discussed, not generalized	“At the least, the results of this experiment suggest that in small code segments involving linked lists, programmers may be able to find a bug more easily in the recursive code than the equivalent iterative code.”
218	74	Discussed, not generalized	“The design documents are not necessarily representative of the ones used in industry. The limitation primarily derives from the size of the created design documents. Systems in industry are usually much larger in size than the ones we used in this experiment. However, we regard the amount of material that our subjects were required to inspect in a single inspection as appropriate. The design documents were developed according to the Fusion development process. Although this process is used at Hewlett-Packard (Coleman *et al.*, 1994), other companies may follow another development process, e.g., the RUP (Jacobson *et al.*, 1998). Since all the Fusion models apart from operation schemata can be found in other UML-based development processes as well, we believe that this represents a rather limited threat to validity.”
219	75	Discussed, generalized	“Setting: This is the effect of performing the study in a setting not representing industrial practice. Since the study is performed in an industrial project and the reviewed code will be part of delivered products, this effect is not considered critical.”
221	76	Discussed, inconclusive	“The systems used in this experiment were not large. However, the levels of inheritance in the systems investigated are typical of those found in larger systems.”

232	87	Discussed, not generalized	“It should be noted that this was a limited controlled study. It certainly implies useful information, but we did not study extended-time performance, such as on a typical project that take several months or years of development. Results may be different for long term projects simply because human interactions over an extended period tend to be different than for short term projects.”
235	91	Discussed, not generalized	“Furthermore, the size of the experimental programming problems is a factor. Small programming problems may not accurately represent larger systems. In fact. it is most likely the case that using larger programs would significantly change the results of this study.”
243	168	Discussed, not generalized	“Obviously, the findings of an experiment such as this can be subject to many threats to validity. In this particular case, threats could include the artificiality of the experimental task, possible deficiencies in the operationalization of the two paradigms, and defects in the experimental materials.”
247	172	Discussed, not generalized	“Finally, this research should be extended to databases of increasing size and complexity.”
402	95	Discussed, not generalized	“The design documents were similar to those which are used in practice, but the size of systems in industry is usually larger. However we think, that the amount of documents which subject were required to inspect was appropriate.”
403	98	Discussed, not generalized	“The inspected document is the same in this experiment as in the two former, which is a threat to the external validity. On the other hand, it strengthens the internal validity.”
514	103	Discussed, not generalized	“Although we tried to make the experimental situations realistic, we cannot be sure that the experimental situation had no unwanted impacts on the estimation or development process. For example, the students may have ignored the instruction that the preplanning effort estimate was not based on historical data or expert knowledge because it came from their lecturer, not because pre-planning effort estimate functioned as an anchor value. The development tasks in the second experiment were much smaller than most real-life tasks. Therefore, the results from that experiment are mainly valid for small programming tasks. Large-scale experiments in more realistic environments are needed.”
514	104	Discussed, not generalized	“Although we tried to make the experimental situations realistic, we cannot be sure that the experimental situation had no unwanted impacts on the estimation or development process. For example, the students may have ignored the instruction that the preplanning effort estimate was not based on historical data or expert knowledge because it came from their lecturer, not because pre-planning effort estimate functioned as an anchor value. The development tasks in the second experiment were much smaller than most real-life tasks. Therefore, the results from that experiment are mainly valid for small programming tasks. Large-scale experiments in more realistic environments are needed.”
514	105	Discussed, not generalized	“Although we tried to make the experimental situations realistic, we cannot be sure that the experimental situation had no unwanted impacts on the estimation or development process. For example, the students may have ignored the instruction that the preplanning effort estimate was not based on historical data or expert knowledge because it came from their lecturer, not because pre-planning effort estimate functioned as an anchor value. The development tasks in the second experiment were much smaller than most real-life tasks. Therefore, the results from that experiment are mainly valid for small programming tasks. Large-scale experiments in more realistic environments are needed.”
515	106	Discussed, not generalized	“We tried to use schemas and operations representative of real cases in the experiments although more experiments with larger and more complex schemas are necessary.”

520	110	Discussed, not generalized	“In this case our results differ from those at Strathclyde, certainly inasmuch as they found that the problems caused by inheritance did not materialise until a deeper level (5) of inheritance was used. Clearly, further research would be useful in shedding additional light upon the impact of inheritance. Another problem with experiments relates to the scale and plausibility of the materials. Obviously this is not addressed by faithful replication. Second, at least in the opinion of the author, there is a mounting body of evidence to suggest that inheritance in OO systems is not unequivocally a ‘good thing’. This is borne out by the review in this paper of other empirical studies such as the investigation of a much larger (133 KLOC) industrial C11 system where it was reported that the classes in inheritance structures have significantly greater defect densities than the other classes (7). However, this is not to argue causality. One possible explanation is that class inheritance is employed to deal with the more complex aspects of a problem. Nonetheless, it is disturbing that even in the case of a highly, contrived problem (as used by the Strathclyde and Bournemouth experiments), which was essentially designed to be dealt with by a specialization solution, subjects still seemed to find the flat version easier to work with. Thus, there is a pressing need for further empirical research utilising more subjects and dealing with industrial scale tasks.”
524	113	Discussed, generalized	“The techniques are intentionally chosen to be representative of those used in industry. There is some question about the level of industrial usage of the code reading technique employed. Inspection techniques in industry tend to be less formal, but consequently less easily taught, and their successful application requires a significant amount of experience. For this reason it was felt that the subjects would perform better with a technique that is more methodical to apply and hence the code reading technique was kept.”
526	114	Discussed, not generalized	“Clearly the results for the generic documents cannot be generalized to specific application domain documents of the organization. However, the experiment was conducted with professional developers and also with documents from an industrial context which strengthens the ability to generalise. The limited number of data points is a potential threat to external validity but this can ultimately be overcome by further replication.”
601	16	Discussed, not generalized	“All our results were obtained from one project, in one application domain, using one language and environment, within one software organization. Therefore, we cannot claim that our conclusions have general applicability, until our work has been replicated.”
702	120	Discussed, not generalized	“The inspection process used during the experiment may not have been representative of industrial software practice. This experiment did not have some of the phases commonly associated with the full inspection process, e.g. presentational overview by the author or rework phase. The group phase that was carried out was partially defect collation and partially defect detection. The main focus of the experiment was the individual inspection phase (sometimes referred to as the preparation phase of the full inspection process).” “Java code may not be representative (in complexity or stylistically) of industrial software. In this case though, the code inspected was part of a substantially larger software system, diminishing some of the complexity concerns.” “The defects seeded in the code may not be representative of the problems currently experienced in industry. As mentioned earlier, this has hopefully been overcome by basing defects on information from various sources (literature, industrial survey, and previous investigations).”
708	121	Discussed, not generalized	“Some of these included the subjects used (they may not have been representative of the general software engineering population), the Java code (may not be representative in terms of style or complexity – it had eight classes but significant references to the Java API), and learning effect (as an unstructured technique ad-hoc inspection had to be carried out for both groups before systematic inspection - there may still have been a general learning effect).”

709	122	Discussed, generalized	"With respect to external validity, we took a specification from a real application context to deal with an inspection object that was representative of an industrial development situation. Moreover, we used inspection activities that had been implemented in a number of professional development environments (Laitenberger *et al.*, 2000)."
710	122	Discussed, generalized	"With respect to external validity, we took a specification from a real application context to deal with an inspection object that was representative of an industrial development situation. Moreover, we used inspection activities that had been implemented in a number of professional development environments (Laitenberger *et al.*, 2000)."
715	126	Discussed, not generalized	"Similarly, the spreadsheets used in the experiment may not be representative of the population of spreadsheets. However, although the spreadsheets may seem rather simple, given the limited testing time of the experiment, few subjects achieved 100% du-adequacy (Clock: 21.7%;Grades: 1.4%). To determine whether the results of this study generalize to a larger segment of the spreadsheet programming population and to other spreadsheets, we are planning to conduct additional studies."
718	200	Discussed, not generalized	"The variability in the skills of participants and the modest number of participants limits the generalizability of our results. We chose to make this trade-off because, in these exploratory studies, we were interested. in how the participants worked with the approach; the quantitative data supported the analysis of the qualitative data." "The external validity of the experiments is also affected by the problems we chose as a basis for the experiment and the limited training provided to the participants. The faults seeded into the system for the debugging experiment, for instance, were all synchronization problems that could be solved by altering Cool code. We informed the participants that synchronization faults had been seeded into the program; AspectJ participants may thus have been pointed towards the Cool code. The performance of all of the participants may also have been affected by being asked to work with either new languages (the AspectJ participants), or with particular constructs (Java and Emerald) introduced to provide a basis of similarity between the languages."
718	204	Discussed, not generalized	"The variability in the skills of participants and the modest number of participants limits the generalizability of our results. We chose to make this trade-off because, in these exploratory studies, we were interested. in how the participants worked with the approach; the quantitative data supported the analysis of the qualitative data." "The external validity of the experiments is also affected by the problems we chose as a basis for the experiment and the limited training provided to the participants. The faults seeded into the system for the debugging experiment, for instance, were all synchronization problems that could be solved by altering Cool code. We informed the participants that synchronization faults had been seeded into the program; AspectJ participants may thus have been pointed towards the Cool code. The performance of all of the participants may also have been affected by being asked to work with either new languages (the AspectJ participants), or with particular constructs (Java and Emerald) introduced to provide a basis of similarity between the languages."
721	130	Discussed, not generalized	"The assignments for which the effort is estimated is small and performed by only one person. The result of the experiment is only valid for applications such as the ones used in the experiment. This is further discussed in Section 5.2."
727	17	Discussed, not generalized	"Threats to external validity are those factors that limit the applicability of the experimental results to industry practice. Such threats include: the student reviewers may not be representative of professional programmers, the software reviewed may not be representative of professional software, and the inspection process may not be representative of industrial practice. These threats are real. Overcoming the first two threats is best accomplished by replication of this study using industrial programmers with real work products. To support this replication, our experimental materials and apparatus are freely available via the Internet. To minimize the third threat, we based our experimental review methods on descriptions of industrial practice of software review, such as Gilb's Inspection (Gilb *et al.*, 1993)"

734	20	Discussed, generalized	”the inspection process in our experimental design may not be representative of software development practice.” “the specification documents may not be representative of real programming problems;” “We avoided the third threat by modeling the experiment’s inspection process after the design inspection process described in Eick, *et al.* (Eick *et al.*, 1992), which is used by several development organizations at AT&T; therefore, we know that at least one professional software development organization practices inspections in this manner.”
1009	122	Discussed, generalized	”We applied inspection activities that had been used in a professional development environment (Porter *et al.*, 1994) to work with an inspection process that was representative of software development practice.” “We took a specification from a real application context to deal with an inspection object that was representative of real development specifications. We used a classroom setting in order to control the experiment environment.”
1103	153	Discussed, not generalized	“With respect to external validity, we take a specification whose size, structure and notation is in our opinion realistic for administrative information systems. However, generalizing our results for other types of projects like real-time applications is certainly limited.”
1104	154	Discussed, not generalized	“The inspection process in this experiment may not be representative of industrial practice. Although there are many variants of the inspection process in the literature and industry, we conducted inspections on the basis of a widely spread inspection process (Wheeler *et al.*, 1996). However, our inspections differ from industrial practice of inspections because inspection meetings occurred simultaneously in big rooms, and did not include the document’s author.” “The requirements documents inspected in this experiment may not be representative of industrial requirements documents. Our documents are smaller and simpler than industrial ones although in the industrial practice long and complex artifacts are inspected in separate pieces. Furthermore, we cannot exclude that meeting losses and meeting gains would occur with the same frequency also for other software artifacts, such as design documents and code.”
1105	122	Discussed, generalized	“With respect to external validity, we took a specification from a real application context to deal with an inspection object that was representative of an industrial development situation. Moreover, we used inspection activities that had been installed in a number of professional development environments (Laitenberger *et al.*, 2000).”
1107	68	Discussed, not generalized	“Even when the training sessions are applied to students, adequate size and complexity of the applied materials might vary depending on previous knowledge about SD modelling and COCOMO.”
1110	158	Discussed, not generalized	“There was some debate as to whether the abstraction process can produce real errors from seeded faults. Since the faults were seeded individually, not starting from seeded errors, it was felt that the results of error abstraction for these documents were necessarily arbitrary. (And if so, mightn’t different results be obtained for a document with real faults and errors?) This debate really hinges upon the question of how representative the seeded faults were in the experiment, and we had to admit we really didn’t know.”

1111	159	Discussed, not generalized	”First, our findings are tied to the chosen inspection process. In the context of our experiment, participants individually looked for defects and then performed a classical face-to-face meeting. Some authors suggest variations of this process such as replacing the classical face-to-face meeting with a distributed, asynchronous discussion step (Todd Dennis *et al.*, 1992) (Cohen *et al.*, 1983). Since this inspection approach is only feasible with adequate tool support, the usefulness rating might be different there. Second, the usefulness and ease of use measures are based on self-reported questionnaire items as opposed to objectively measured ones. However, our results show that the questionnaire items are reliable and valid. Furthermore, there are no objective measures to capture usefulness and ease of use. Hence, the only possibility is to investigate the mechanisms driving user behavior with the help of subjective measures. We strongly believe that the “people factor” reflected in user behavior is an important one to consider while developing any particular tool or suggesting any new software engineering technique. This opinion is shared by many experts as reported by the National Research Council (Web: http://www.nap.edu/readingroom/books/statsoft/). However, this factor has often been neglected in software engineering research and practice. One reason might be the lack of valid and reliable measurement instruments. This research provides one step to overcome this obstacle.”
1116	162	Discussed, not generalized	”The inspection process in our experimental design may not be representative of software development practice. We have modeled our experiment’s inspection process after the ones used in many development organizations, although each organization may adapt the process to fit its specific needs. Another difference is that the SRS authors are not present at our inspections, although in practice they normally would be. Finally, industrial reviewers may bring more domain knowledge to an inspection than our student subjects did.” “The specification documents we used may not be representative of real programming problems. Our experimental specifications are atypical of industrial SRS in two ways. First, most of the experimental specification is written in a formal requirements notation (see Section 2.2). Although several groups at AT&T and elsewhere are experimenting with formal notations (Ardis M.A., 1994, Gerhart *et al.*, 1994), it is not the industry’s standard practice. Second, the specifications used are considerably shorter than industrial specifications.”
1117	14	Discussed, not generalized	”The results may be domain-dependent. This objection cannot be ruled out. This experiment should therefore be repeated in domains other than graphical user interfaces. The results may not apply to situations were the subjects are very familiar with the interfaces used.”

Appendix E

Table E.1 presents summarizes for all sample populations in the experiments. For two of the experiment, we could not find any information about the sample population, and these two experiments are marked with ‘??’ in the table.

Table E.1: Summary of sample populations in all experiments

Exp.	**Sample summary**
1	Professional software engineers from one firm doing an experiment as part of their job.
2	Students in computer science from one intensive course.
3	Students in computer science from one standard course.
4	Mostly undergraduate students from one standard course.
5	Volunteer students with varying levels of degrees from one course.
6	Professional programmers from a particular business unit at one firm doing an experiment as part of their job.
7	Computer science master students from different graduate courses. Some were paid, and some had to participate (mandatory).
8	Professional developers from one firm doing an experiment as part of their job.
10	Students at all levels from two universities and from several different courses. Some had to participate (mandatory).
11	Students at all levels from two universities and from several different courses. Some had to participate (mandatory).
14	A mixture of post docs, phd, and graduate students doing an experiment voluntarily with no payment.
16	Professional developers from one firm doing an experiment as part of their job.
17	Undergraduate students from two different courses at one university.
18	A mixture of graduate students in computer science from one university and professional software developers (don’t know if they are from one or several companies).
20	1st and 2nd year graduate students from one course at one university held by one of the authors.
21	Graduate and upper division undergraduate students majoring in computer science from one course at one university. Several of the students were full time employees in the computing field.
22	Graduate and upper division undergraduate students majoring in computer science from one course at one university. Several of the students were full time employees in the computing field.
23	Graduate and upper division undergraduate students majoring in computer science from one course at one university. Several of the students were full time employees in the computing field.
24	Primarily professional software engineers and analysts from one consortium who have several member companies.
25	5th and 6th quarter masters students in computer systems management from one course at one university.
26	Junior and senior students from one course at one university.
28	Undergraduate students majoring in information systems from one course at one university. Subjects received credit based on their perfomance, which was factored into the final grade for the course.
29	Undergraduate students majoring in information systems from one course at one university. Subjects received credit based on their perfomance, which was factored into the final grade for the course.
33	A mixture of professionals from different software development enterprises (both United States and France, all but one were male, recruited by electronic advertisements or by nomination from colleagues), and advanced undergraduate computer science students from one course at one university (enrolled in the course now, or had been enrolled earlier, all but 4 were male, recruited through announcement at the university).
34	Last year students in informatics from several graduate courses.
36	Mainly undergraduate students in computer science, but also some graduate students. All students from one university, enrolled in several undergraduate courses. The subjects were paid.
37	Fourth year master students in computer science and engineering and electrical engineering from one course at one university. The experiment was a mandatory part of the course and they got a grading based on serious participation in the study (not on their performance).

41	A mixture of fourth year master students in computer science and engineering and electrical engineering from one course at one university and professionals from one company.
43	Full time masters students in software engineering from one university.
44	Staff and students from one university. No incentives for participating were given.
45	Third year undergraduate students from one course. All students in the course participated, but they were nor aware that they were experimental subjects. The assessment was graded.
48	Bachelor of science students from one course. Experiment was a mandatory part of the course, but the grading was only concerned about the subjects' precense. They got rewarded with a free trip to an exhibition with free lunch. Only 3 female subjects.
49	Professional developers from one firm enrolled in a professional training course.
51	Third year undergraduate software engineering students from one course at one university. The experiment was a mandatory part of the course, and the practical aspects of the experiment was an assessment which contributed to their overall degree in the class.
53	3rd year undergraduate students from one course at one university.
54	Information systems graduate students from one course at one university.
55	Domain experts. Nothing else stated.
56	Programmers and technical non-programmers.
58	Staff from one research group in Software Engineering at one university (quarter time appointments masters degree candidates and full time PhD candidates) doing an experiment as part of their regular working hours. The subjects were in part self-selected (volunteers).
59	Undergraduate computer science students (varying degrees, but most of the subjects had their vordiplom) from one course at one university. The experiment subjects were volunteers.
60	Volunteers. Nothing else stated.
61	Third and fourth year undergraduates from one advanced course.
62	Students and recent graduates from an intensive taught postgraduate conversion course. Experiment constituted 60 % of the grading in the course and was mandatory.
63	A mixture of bachelor computer science students going into final (fourth) year and new graduates from one course. Experiment participation was voluntarily.
65	Professional software developers from the Software Engineering laboratory at one firm doing an experiment as part of their job. The experiment was voluntarily, and everyone who volunteered was accepted.
66	Professional from one government organization doing an experiment as part of their job. The participation was voluntarily.
67	Junior and senior students from one department at one university.
68	Graduate computer science students from one course at one university. Participating was voluntarily.
69	Students of business informatics that were at least in the fifth term from one course at two universities.
71	A mixture of students from one course at two departments at one university and professionals from several local software development organizations and a senior-level course at the same university. All subjects received extra credit for merely participating, the top 50% within their respective group was eligible for a cash incentive (lottery tickets).
72	Students (computer and information science and information science majors) from 16 different classes.
73	Undergraduate business students from one course (different sections of the same course) at one university. Participating was voluntarily, and students from some of the sections received extra credit for participating. (Approximately equal amount of males and females.)
74	Practitioners with various backgrounds from one course.
75	Professional engineers from one firm doing an experiment as part of their job.
76	Second year bachelor computer science students from one software engineering unit at one university. Participating was voluntarily.
81	Advanced undergraduate systems analysis students from one course at one university. Subjects received extra point credit toward their final course grade for participating in the experiment.
82	Undergraduate college students from one course.
83	Students from several courses at one university. Received course credit for their participation.
84	Third and fourth year undergraduate students from one course (held by the author) at one university.
85	Third and fourth year undergraduate students from one course (held by the author) at one university.
86	Professional maintainers from one organization.

87	Computer science senior college students from one course (typically include the top of the class and are highly motivated).
88	Undergraduate business majors.
89	Undergraduate and postgraduate students from six different courses at one university. Experiment participation was voluntarily.
91	Computer science seniors.
95	Third year bachelor students from one course at one university. The experiment was a mandatory part of the course and the grading if the course depended on their performance in the experiment.
98	Third year software engineering bachelor students and fourth year software engineering master students from two courses at two universities. Experiment participation was a mandatory part of the courses and the grading of the course depended only on their participation in the experiment.
99	Mostly fourth year students from one course at one university.
100	Third year software engineering bachelor students from one course at one university. Experiment participation was a mandatory part of the course.
103	Undergraduate computer science students from one course (given by the authors) at one university.
104	Students from one course at one university.
105	Graduate computer science students from one course at one university.
106	Professionals from one company.
110	Software engineering bachelors students in their final year from one university.
113	Honours students from one course at one university. Experiment participation was a mandatory part of the course.
114	Professional software developers from one company doing an experiment as part of their job.
115	First and second year computer science or business information technology students from one course at one university.
116	Students from one course at one university.
120	Third year computer science honours students from one course at one university. The subjects received course credit.
121	Third year computer science honours students from one course at one university.
122	Undergraduate students from a two semester university software development workshop.
125	Second year computer science students.
126	Advanced computer science students from two upper division undergraduate courses and one graduate course.
127	Software engineering students from one course at one university.
130	PhD students from one course at one university.
131	Professional developers from one company doing an experiment as part of their job.
139	Fifth year undergraduate students from one course at two universities.
140	Fourth year computer science students from one university. Experiment participation was voluntarily and the subjects received a point credit to the final examination.
141	Second year computer science students from one university. Experiment participation was voluntarily and the subjects received a point credit to the final examination.
147	Graduate and senior undergraduate students in computer science, electrical engineering, and computer engineering from one university. All subjects were male.
150	Senior software engineering students from one university.
151	Students from one university. Experiment participation was voluntarily and the participants were paid.
153	Undergraduate informatics students from a software engineering workshop at one university.
154	Undergraduate students from one course at one university. Experiment was mandatory (run as a midterm exam) and was graded.
158	Students from to classes. Experiment participation was mandatory and was graded.
159	Computer science graduate students from one course at one university.
160	Partners and managers from one company doing an experiment as part of their job. Participation was voluntarily.
161	A mixture of staff and students from several academic institutions and computing professionals from industrial software companies. Experiment participation was not paid, and was voluntarily.
162	Graduate software engineering students from one course.
166	??

168	Middle and senior business managers from a part-time course at one university. The managers came from several different organizations.
170	Developers. Nothing else stated.
172	Freshmen to graduate college students with varying backgrounds. Experiment participation was voluntarily.
174	Computer science majors from one course.
175	Experienced professional programmers from one organization doing an experiment as part of their job. Experiment participation was voluntarily.
176	Graduate and undergraduate computer science students.
177	Graduate software engineering students from two universities. Experiment participation was voluntarily.
181	Senior and graduate students.
188	Graduate and undergraduate computer science students.
192	Developers from one company doing an experiment as part of their job.
193	Professional programmers from a particular business unit at one company doing an experiment as part of their job.
194	Professional programmers from a particular business unit at one company doing an experiment as part of their job.
195	First and second year graduate computer science students from one course (held by one of the authors).
198	Computer science bachelor students going into final (fourth) year and new graduates. Experiment participation was voluntarily.
200	Graduate students and professors in computer science and an undergraduate in computer engineering.
201	??
203	Undergraduate and graduate information systems engineering students from one university. Experiment participation was mandatory and was graded.
204	Graduate and undergraduate computer science and computer engineering students from one university and one participant from the industry doing an experiment as part of a summer job.

Appendix F

This appendix contains all the SAS tables created for this thesis.

TABLE N11: External validity

	Frequency	Percent
Discussed	84	67.20
Not discussed	41	32.80
Total	125	100.00

TABLE: List_of_external_threats

	Frequency	Percent
Interaction of subjects and treatment	70	44.30
Interaction of tasks and treatment	63	39.90
Interaction of environment and treatment	25	15.80
Total	147	100.00

TABLE N2: Sample and target population
Table of sampop by targpop

Frequency Percent	Explicit	Implicit	Unknown	Total
Explicit	32 25.60	47 37.60	43 34.40	122 97.60
Implicit	0 0.00	0 0.00	1 0.80	1 0.80
Unknown	0 0.00	0 0.00	2 1.60	2 1.60
Total	32 25.60	47 37.60	46 36.80	125 100.00

TABLE N3: Generalization of subjects

	Frequency	Percent
Generalized	29	23.20
Inconclusive	6	4.80
Not discussed	45	36.00
Not generalized	45	36.00
Total	125	100.00

TABLE N7: Generalization categorized

	Frequency	Percent
Professionals to professionals	3	10.34
Sample to same categories of professionals	5	17.24
Sample to same categories of students	2	6.90
Students to junior professionals	5	17.24
Students to professionals	14	48.28
Total	29	100.00

TABLE N8: Number of reasons for generalizing

	Frequency	Percent
One reason	22	75.86
Multiple reasons	7	24.14
Total	29	100.00

TABLE N9: Reason for generalization (One exp. can have more than one reason, see table N8.)

	Frequency	Percent
Argumentation	9	25.71
Background	5	14.29
Conditions	1	2.86
No difference between students and professionals	2	5.71
Soon professionals	5	14.29
Statistic	1	2.86
Task	3	8.57
Theory	9	25.71
Total	35	100.00

TABLE N10: Relation between replicated experiments and generalization of subjects
Table of Generalizations_of_subjects by Replication

Frequency Percent	No	Yes	Total
Generalized	23 18.40	6 4.80	29 23.20
Inconclusive	6 4.80	0 0.00	6 4.80
Not Discussed	39 31.20	6 4.80	45 36.00
Not generalized	36 28.80	9 7.20	45 36.00
Total	103 82.40	22 17.60	125 100.00

TABLE N20: Generalization from task

	Frequency	Percent
Generalized	7	5.60
Inconclusive	2	1.60
Not discussed	53	42.40
Not generalized	63	50.40
Total	125	100.00

TABLE N21: Relation between replicated experiments and Generalization from task
Table of Generalisation_from_task by Replication

Frequency Percent	No	Yes	Total
Generalized	6 4.80	1 0.80	7 5.60
Inconclusive	1 0.80	1 0.80	2 1.60
Not Generalized	50 40.00	13 10.40	63 50.40
Not Discussed	47 37.60	6 4.80	53 42.40
Total	104 83.20	21 16.80	125 100.00

TABLE N18: Generalization of environment

	Frequency	Percent
Discussed, generalized	7	5.60
Discussed, inconclusive	2	1.60
Discussed, not generalized	27	21.60
Not discussed	89	71.20
Total	125	100.00

TABLE N17: Relation between replicated experiments and generalization of context
Table of Generalization_of_context by Replication

```
Frequency|
Percent  |No       |Yes      |  Total
---------+---------+---------+
Yes      |        7|        0|      7
         |     5.60|     0.00|   5.60
---------+---------+---------+
No       |       97|       21|    118
         |    77.60|    16.80|  94.40
---------+---------+---------+
Total          104        21      125
             83.20     16.80   100.00
```

TABLE N14: Internal validity

	Frequency	Percent
Discussed	84	67.20
Not discussed	41	32.80
Total	125	100.00

TABLE N16: Categories of Internal validity

	Frequency	Percent
Accuracy of subjects registration	5	2.24
Ambiguity of the direction of causual influence	1	0.45
Compensatory rivalry	1	0.45
Compensatory equalization of treatments	2	0.90
Diffusion of imitation of treatments	1	0.45
History	10	4.48
Instrumentation	44	19.73
Interactions with selection	7	3.14
Maturation	40	17.94
Mortality	9	4.04
Motivation	5	2.24
No categories	1	0.45
Other	24	10.76
Plagiarism	4	1.79
Replication	5	2.24
Selection	45	20.18
Statistical regression	2	0.90
Testing	5	2.24
Training	12	5.38
Total	223	100.00

Appendix G

Table G.1: Distribution of experiments to years

	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002
Number of exp.	3	3	6	12	14	19	13	21	21	13
Percentage	2.4	2.4	4.8	9.6	11.2	15.2	10.4	16.8	16.8	10.4