Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Information and Software Technology 46 (2004) 127–147

INFORMATION AND SOFTWARE TECHNOLOGY

www.elsevier.com/locate/infsof

Evaluating the learning effectiveness of using simulations in software project management education: results from a twice replicated experiment

Dietmar Pfahl[a,*], Oliver Laitenberger[b], Günther Ruhe[c], Jörg Dorsch[d], Tatyana Krivobokova[e]

[a]*Fraunhofer IESE, Sauerwiesen 6, 67661 Kaiserslautern, Germany*
[b]*Droege & Comp., Praterinsel 3-4, 80538 München, Germany*
[c]*University of Calgary, 2500 University Drive NW, Calgary, Alberta, Canada*
[d]*Accenture, Campus kronberg 1, 61476 Kronberg, Germany*
[e]*University of Bielefeld, Universitätsstraße 25, 33615 Bielefeld, Germany*

Received 20 December 2002; accepted 20 June 2003

Abstract

The increasing demand for software project managers in industry requires strategies for the development of management-related knowledge and skills of the current and future software workforce. Although several educational approaches help to develop the necessary skills in a university setting, few empirical studies are currently available to characterise and compare their effects.

This paper presents the results of a twice replicated experiment that evaluates the learning effectiveness of using a process simulation model for educating computer science students in software project management. While the experimental group applied a System Dynamics simulation model, the control group used the well-known COCOMO model as a predictive tool for project planning.

The results of each empirical study indicate that students using the simulation model gain a better understanding about typical behaviour patterns of software development projects. The combination of the results from the initial experiment and the two replications with meta-analysis techniques corroborates this finding. Additional analysis shows that the observed effect can mainly be attributed to the use of the simulation model in combination with a web-based role-play scenario. This finding is strongly supported by information gathered from the debriefing questionnaires of subjects in the experimental group. They consistently rated the simulation-based role-play scenario as a very useful approach for learning about issues in software project management.

Keywords: COCOMO; Learning effectiveness; Replicated experiment; Software project management education; System dynamics simulation

1. Introduction

Software development is a dynamic and complex process since many interacting factors impact the costs and schedule of the development project as well as the quality of the developed software product throughout the lifecycle. To monitor and control software development projects, management experience and knowledge on how to balance the various influential factors are required. However, the growing pervasiveness of software and the increasing number of software development projects result in a lack of well-trained and experienced managers.

To address these issues, process simulation techniques have been applied to the domain of software engineering during the last decade, starting with the pioneering work of Kellner et al. [13] and Abdel-Hamid and Madnick [1]. However, experience with process simulation as a means for software project management education and training has rarely been published [6,19,23] although the potential of simulation models for the training of managers has long been recognised [8,20,21].

In fact, only few experimental studies have been conducted with models that simulate typical behaviour of software projects [16,17,27]. The results of these experiments indicate

* Corresponding author. Tel.: +49-6301-707-258; fax: +49-6301-707-203.
E-mail addresses: pfahl@iese.fhg.de (D. Pfahl), oliver_laitenberger@droege.de (O. Laitenberger), ruhe@ucalgary.ca (G. Ruhe), joerg.dorsch@accenture.com (J. Dorsch), tkrivobokova@wiwi.uni-bielefeld.de (T. Krivobokova).

0950-5849/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0950-5849(03)00115-0

that a natural one-way causal thinking could be detrimental to the success of software managers. They must rather adopt multi-causal or systems thinking. Moreover, they must be aware of (unexpected) feedback to their management decisions. These findings highlight the need for new learning and education strategies.

The first strategic step for teaching software project management methods and techniques must already be included in the curriculum of students. University education must teach computer science and software-engineering students not only technology-related skills but also a basic understanding of typical management phenomena occurring in industrial (and academic) software projects. However, practical constraints of a course usually limit the exposure of students to realistic, large-scale industrial software development projects in which they could make their own experiences. This can be partially compensated by making students use software process simulation models that reproduce the behaviour of realistic (i.e. complex) software development projects. The question is whether this approach is viable.

This paper presents the results of a controlled experiment and two external replications that investigate the effectiveness of computer-based training in the field of software project management using a System Dynamics (SD) simulation model. The experiment was originally performed at the University of Kaiserslautern, Germany [23]. Replications took place at the University of Oulu, Finland, and at the University of Calgary, Canada. All results presented in this paper must be viewed as exploratory. The intention is to identify and refine important hypotheses and to investigate them in further detail.

The paper is structured as follows: Section 2 presents the methodological details of the study. Section 3 summarises its results. Section 4 provides a discussion of these results and the associated threats to validity. The paper concludes with improvement suggestions for the experimental design and proposes directions for future research.

1.1. Description of experiment

To investigate the effectiveness of computer-based training in the field of software project management using a SD simulation model, a controlled experiment applying a pre-test–post-test control group design was conducted. The subjects had to undertake two tests, one before the training session (pre-test) and one after the training session (post-test). The effectiveness of the training was then evaluated by comparing within-subject post-test to pre-test scores, and by comparing the scores between subjects in the experimental group, i.e. those who used the SD model, and subjects in the control group, i.e. those who used a conventional project planning model instead of the SD model. In the study, the control group performed their tasks with the well-known COCOMO model [2]. COCOMO was selected since the model is quite comprehensive and can be considered as state-of-the-practice in many industrial software organisations.

The main objective of developing and applying a simulation-based training module was to facilitate effective learning about certain topics of software project management for computer science students. This was done by providing a scenario-driven interactive single-learner environment that can be accessed through the internet by using a standard web-browser. An additional goal was to raise interest in the topic of software project management among computer science students, and to make them aware of some of the difficulties associated with controlling the dynamic complexity of software projects.

The training module used in the study is composed of course material on project planning and control. The core element of the training module is a set of interrelated project management (i.e. planning) models, represented by a simulation model that was created by using the SD simulation modelling method [7,25]. This model simulates typical behaviour of software development projects.

The various possibilities of conducting a training session are depicted in Fig. 1. The first level defines the learning goal, i.e. software project management with focus on project planning and control. The second level defines the type of project planning model used in the training session, i.e. COCOMO model versus SD simulation model. Finally, the third level defines the learning mode as another dimension to characterise the training session, i.e. inclusion or exclusion of a web-based interactive role-play. The combination of the instances in levels two and three yield four different treatments. Our empirical investigations compare the effectiveness of two of them, i.e. SD model-based learning with web-based interactive role-play scenario (experimental group A), and standard COCOMO-based learning without web-based interactive role-play (control group B).

1.2. Experimental hypotheses

Four constructs were used to measure performance of the training session. Each construct is represented by one dependent variable. The experimental hypotheses were stated for the dependent variables, as follows:

1. There is a positive learning effect in both groups (A: experimental group, B: control group), i.e. post-test scores are significantly higher than pre-test scores for each dependent variable.
2. The learning effect in group A is higher than in group B, either with regard to the performance improvement between pre-test and post-test (relative learning effect), or with regard to post-test performance (absolute learning effect). The absolute learning effect is of interest because it may indicate an upper bound

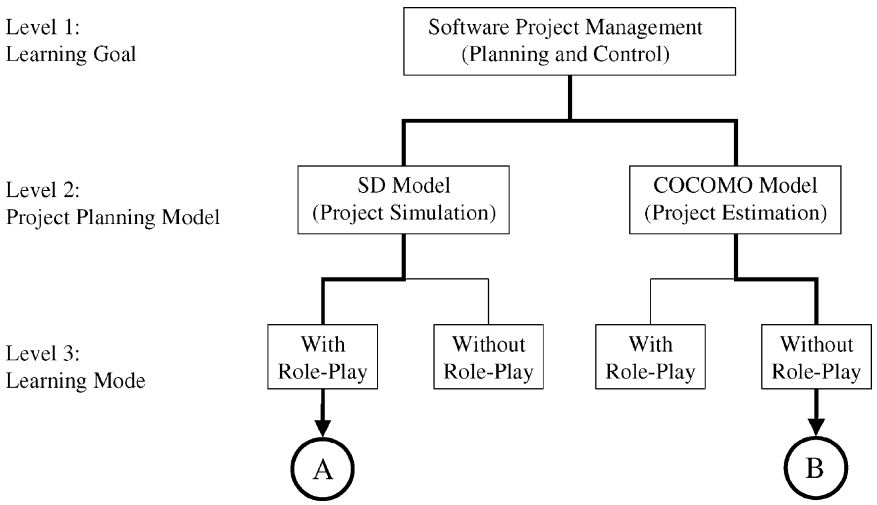

Fig. 1. Training session arrangements.

of the possible correct answers depending on the type of training (A or B).

Note that it is not expected that both relative and absolute learning effects will always occur simultaneously. This reflects on the fact that higher relative learning effects in group A compared to group B are less likely to occur when pre-test scores of group A are significantly higher than those of group B. Similarly, higher absolute learning effects in group A compared to group B are less likely to occur when pre-test scores of group A are significantly lower than those of group B.

2. Method

For evaluating the effectiveness of a training session using SD model simulation, a pre-test–post-test control group design was applied [12]. This design involves random assignment of subjects to an experimental group (A) and a control group (B). The subjects of both groups completed a pre-test and a post-test. The pre-test measured the performance of the two groups before the treatment, and the post-test measured the performance of the two groups after the treatment. The students did neither know that the post-test questions were identical to the pre-test questions, nor were they allowed to keep the pre-test questionnaires. The correct answers were only provided to the students after the end of the experiments.

2.1. Subjects

The initial experiment was conducted with graduate computer science students at the University of Kaiserslautern (KL), Germany, who were enrolled in the advanced software engineering class. Although they had not yet finished their Master degree in computer science or a related field, their skill level was comparable to a Bachelor degree. Twelve students expressed their interest in participation but eventually only nine of them completed all parts of the experiment. The first replication of the initial study was conducted during a summer school with 12 graduate and post-graduate students (one Master degree, one PhD) of the University of Oulu, Finland, having their major in computer science, information technology, information engineering, microelectronics or mathematics. The second replication was performed with 13 senior undergraduate students at the University of Calgary, Canada, major in computer science, electrical engineering and computer engineering.

In all three studies context information about the participants was collected. Students were asked questions about personal characteristics (age, gender), university education (number of terms, major, minor), personal experience with software development, background knowledge on software project management, and preferences and beliefs about learning styles. These questions could be answered on a voluntary basis. The results are presented in Table 1. Note that more than one answer could be given in categories ‘Preferred learning style(s)’ and ‘Opinion about most effective learning style(s)’. In the Calgary experiment most students did not answer the characterisation questions. Therefore, the entry ‘incomplete data’ was put into Table 1. It could not be fully clarified whether the reluctance to provide data is due to higher sensitivity with regards to protection of personal data or simply to lack of clarity in the instructions handed out to the students.

2.2. Procedure

The initial experiment and its two replications were conducted following the plan presented in Table 2. After a short introduction during which the purpose of the experiment and general organisational issues were explained, data on personal characteristics and background knowledge was collected by means of a questionnaire. Then the pre-test was conducted and data on all dependent

Table 1
Personal characteristics

	KL students	Oulu students	Calgary students
Average age (years)	27.0	31.3	Incomplete data
Share of women	11%	50%	Incomplete data
Share of subjects majoring in			
Computer science, information technology, software engineering, information engineering, information processing science	100%	67%	Incomplete data
Other (non-software related)	0%	33%	
Number of software programs developed			
0	All subjects had written a software program; more details were not asked	17%	Incomplete data
1–3		8%	
4–7		25%	
8–10		33%	
More than 10		17%	
Software project experience			
Experience with teamwork in software projects?	22%	58%	Incomplete data
Active involvement in industrial software projects?	56%	75%	
Responsible for customer contact in commercial software projects?	Not asked	50%	
Number of software project management books read			
0	Not asked	25%	Incomplete data
1–2		58%	
3–5		17%	
More than 5		0%	
Correct understanding of acronym COCOMO	Not asked	33%	Incomplete data
Correct understanding of Brook's Law	Not asked	17%	Incomplete data
Preferred learning style(s)			
Reading (with exercise)	89%	33%	Incomplete data
Web-based training	11%	8%	
In-class lecture (with exercise)	22%	25%	
Working group (with peers)	33%	42%	
Opinion about most effective learning style(s)			
Reading (with exercise)	Not asked	25%	Incomplete data
Web-based training		17%	
In-class lecture (with exercise)		33%	
Working group (with peers)		67%	

variables were collected, again using questionnaires. Following the pre-test, a brief introduction into organisational issues related to the treatments was given. After that, the subjects were randomly assigned to either the experimental or control group. Of the 12 students that agreed to participate in the initial experiment, nine students participated in both pre-test and post-test. Five students were assigned randomly to the experimental group and four students to the control group. Of the 12 students participating in the first replication, six were assigned randomly to the experimental group and six to the control group. Of the 13 students taking part in the second replication, seven were assigned randomly to the experimental group and six to the control group.

Each group underwent its specific treatment. The treatments of each group differed with regards to the specific structure of the respective training scenarios and the type of planning models used within individual scenario blocks. Table 3 summarises the differences between the treatments of the experimental and the control groups, indicating the duration of the scenario blocks applied, and providing information on the nature of the used planning models. The duration of individual scenario blocks is expressed in minutes, first for the initial experiment, then for the replications.

With regard to the training scenarios, the main difference consists in the exclusive application of scenario block 2 to the experimental group. Scenario

Table 2
Schedules of experiment and replications

	Experiment	Replication 1	Replication 2
Introduction to experiment	5 min	5 min	5 min
Personal characteristics and background knowledge	5 min	5 min	5 min
Pre-test			
Interest	3 min	5 min	5 min
Knowledge about empirical patterns	5 min	5 min	5 min
Understanding of simple project dynamics	10 min	10 min	10 min
Understanding of complex project dynamics	12 min	15 min	15 min
Introduction to treatments	5 min	5 min	5 min
Random assignment of subjects to groups	5 min	5 min	5 min
Treatment	45 min	80 min	80 min
Post-test			
Interest	3 min	5 min	5 min
Knowledge about empirical patterns	5 min	5 min	5 min
Understanding of simple project dynamics	10 min	10 min	10 min
Understanding of complex project dynamics	12 min	15 min	15 min
Time need and subjective session evaluation	5 min	10 min	10 min
Total	130 min	180 min	180 min

block 2 involves a role-play with an interactive web-based simulation component. The interactive simulation component invoked a predictive SD simulation model, which was explained in full detail in scenario block 3 to the experimental group and applied in scenario block 4. The SD model can be used to calculate point estimates for planning problems like a statistical black-box model such as COCOMO (Constructive Cost Model [2]). In addition to that, due to the holistic concept underlying SD, it can be used like an explanative (white-box) model that facilitates insights into behavioural aspects of software projects caused by complex information feedback structures (more on SD can be found in Ref. [7]). The treatment of the control group invoked the explanation and application of the planning model COCOMO in scenario blocks 3 and 4. This was considered to be state-of-the-art of university education on project management planning problems.

After completing their treatments, both groups performed the post-test using the same set of questionnaires as during the pre-test, thus providing data on the dependent variables for the second time. In contrast to the pre-test, subjects were allowed to use their respective planning model to answer the post-test questions. Finally, the subjects got the chance to evaluate the training session by filling in another questionnaire, providing data on perceived time pressure and subjective judgement of training quality.

During the whole procedure, the time slots reserved for completing a certain step of the schedule were identical for the experimental and control groups. Note however that a more relaxed schedule was followed during the replications as compared to the initial experiment (cf. columns 'Experiment', 'Replication1' and 'Replication 2' of Table 2).

2.2.1. Treatment details

The scenario blocks used during the treatments can be characterised as follows:

1. *Block 1: PM Introduction.* General introduction into the main tasks of software project managers and the typical problems they have to solve with regard to project planning and control. This includes a brief discussion of problems caused by the so-called 'magic triangle', i.e. the typical presence of unwanted trade-off effects

Table 3
Differences between treatments of experimental and control groups

	Treatment of experimental group	Treatment of control group
Training scenarios	Block 1 (3 min/5 min)	Block 1 (3 min/5 min)
	Block 2 (15 min/30 min): SD model	n/a
	Block 3 (15 min/30 min): SD model	Block 3 (30 min/60 min): COCOMO
	Block 4 (12 min/15 min): SD model	Block 4 (12 min/15 min): COCOMO
PM models	SD model: Behavioural (white box)	COCOMO
	Point estimates (black box)	Point estimates (black box)

between project effort (cost), project duration, and product quality (functionality).
2. *Block 2: PM Role Play.* Illustration of common project planning problems on the basis of an interactive case example in which the trainee takes over the role of a fictitious project manager. This scenario block involved applications of the SD model.
3. *Block 3: PM Planning Models.* Presentation of basic models that help a project manager with planning tasks, namely a process map, and a predictive model for effort, schedule and quality. This scenario block involved detailed explanations of the SD model to the experimental group, and detailed explanations of intermediate COCOMO (for details on the intermediate mode [2]) to the control group.
4. *Block 4: PM Application Examples.* Explanation on how to apply the planning models, i.e. SD model and intermediate COCOMO, respectively, on the basis of examples that are presented in the form of little exercises.

In the following, more details on scenario block 2 are presented. The main purpose of scenario block 2 is to help students better understand the complex implications of a set of empirically derived principles that typically dominate software projects. The set of principles (Table 4) is distilled from the top 10 list of software metric relationships published by Barry Boehm in 1987 [3].

After presentation and explanation of the principles, scenario block 2 involves a role-play based on an interactive project simulation using the SD model. The students take the role of a project manager who is assigned to a new development project. Several constraints are set, i.e. the size of the product and its quality requirements, the number of software developers available, and the project deadline. The first thing to do for the project manager (also in order to familiarise with the simulation model) is to check whether the project deadline is feasible under the resource and quality constraints given. Running a simulation does this check. From the simulation results, the project manager learns that the deadline is much too short. Now, the scenario provides a set of actions that the project manager can take, each action associated with one of the principles and linked to one of the model parameters. Soon the project manager learns that his senior management does not accept all of the proposed actions (e.g. reducing the product size or complexity). Depending on the action the project manager has chosen, additional options can be taken. Eventually, the project manager finds a way to meet the planned deadline, e.g. by introducing code and design inspections (cf. Principle 6 in Table 4).

Table 4
List of principles dominating project performance

No.	Principle
1	'Finding and fixing a software problem after delivery is 100 times more expensive than finding and fixing it during the requirements and early design phases'
2	'You can compress a software development schedule up to 25 percent of nominal, but no more'
3	'Software development and maintenance cost are primarily a function of the number of source lines of code (SLOC) in the product'
4	'Variations between people account for the biggest differences in software productivity'
5	'Software systems and products typically cost 3 times as much per SLOC as individual software programs. Software-system products (i.e. system of systems) cost 9 times as much'
6	'Walkthroughs catch 60% of the errors'

The role-play is arranged in a way that the project manager can only succeed when combining actions that relate to at least two of the principles listed in Table 4. At the end of the role-play, a short discussion of the different possible solutions is provided, explaining the advantages and disadvantages of each.

A detailed description of the SD model used in scenario block 2 can be found in Ref. [23]. Here only the five interrelated sub-models (views) can be briefly characterised:

1. *Production View.* This view represents a typical software development lifecycle consisting of the following chain of transitions: set of requirements (planned functionality) → design documents → code → tested code.
2. *Quality View.* This view models the defect co-flow: defect injection (into design or code) → defect propagation (from design to code) → defect detection (in the code during testing) → defect correction (only in the code). Optionally, additional QA activities will result in defect detection and rework already during design and coding.
3. *Effort View.* In this view, the total effort consumption for design development, code development, code testing, optional QA activities, and defect correction (rework) is calculated.
4. *Initial Calculations View.* In this view, the nominal value of the process parameter 'productivity' is calculated using the basic COCOMO equations for estimating effort and project duration. The nominal productivity varies with assumptions about the product development mode (organic, semi-detached, embedded) and characteristics of the available project resources (e.g. developer skill).
5. *Productivity, Quality and Manpower Adjustment View.* In this view, project-specific process parameters, like (actual) productivity, defect generation, effectiveness of QA activities, etc. are determined based on (a) planned target values for manpower, project duration, product quality, etc. and (b) time pressure induced by unexpected rework or requirement changes.

It should be noted that the complex impact of actions taken during the role-play by the fictitious project manager cannot be predicted using the COCOMO model. This is mainly due to the fact that the COCOMO model does not (yet) fully cover the impact of effort or size changes on product quality.[1] Due to the possibility to model complex cause–effect structures, the SD model used in scenario block 2 has this capability, i.e. constraints on product quality have an impact on project duration and effort consumption, and vice versa.

2.2.2. Differences between initial experiment and replications

Since most of the participants of the initial study at the University of Kaiserslautern stated that they did not have enough time available for working through the materials, more time was reserved for the treatment during the replication at the University of Oulu. Another difference refers to the overall arrangements of the experiments. While the initial experiment at the University of Kaiserslautern was conducted on two days with one week of time in between, the replications at the University of Oulu and the University of Calgary were conducted on a single day in each case.

In the initial experiment, on the first day, the steps 'Introduction to experiment', 'Background characteristics', and 'Pre-test' were conducted, consuming a total of 40 min. On the second day, the steps 'Introduction to treatments', 'Random assignment of students to groups', 'Treatment', 'Post-test', and 'Time need and subjective session evaluation' were conducted, consuming a total of 90 min.

The first and second replications were conducted in two parts on one day. The first part, including the steps 'Introduction to experiment', 'Background characteristics', and 'Pre-test', consumed a total of 45 min. The second part, including the steps 'Introduction to treatments', 'Random assignment of students to groups', 'Treatment', 'Post-test', and 'Time need and subjective session evaluation', consumed a total of 135 min. There was a break of 30 min between the first and the second part.

2.3. Data collection

During the experiment, data for two types of variables are collected. Table 5 lists all variables, including four dependent variables (Y.1,…,Y.4) and two variables that capture subjective perceptions about the treatments (Z.1, Z.2).

2.3.1. Dependent variables

Dependent variables Y.1, Y.2, Y.3, and Y.4 are constructs used to capture various aspects of learning induced by the treatments. Each construct was measured through an aggregate of 5–7 questions with answers being provided on a uniform scale. The value of each dependent variable is calculated as an average score. The related questions can be characterised as follows:

Y.1 ('Interest'): Questions about personal interest in learning more about software project management.
Y.2 ('Knowledge'): Questions about typical performance patterns of software projects. These questions are based on some of the empirical findings and lessons learned summarised in Barry Boehm's top 10 list of software metric relations [3].
Y.3 ('Understand simple'): Questions on project planning problems that require simple application of the provided PM models, addressing trade-off effects between no more than two model variables.
Y.4 ('Understand complex'): Questions on project planning problems addressing trade-off effects between more than two variables, and questions on planning problems that may require re-planning due to alterations of project constraints (e.g. reduced manpower availability, shortened schedule, or changed requirements) during project performance.

Table 5
Experimental variables

Dependent variables	
Y.1	Interest in software project management issues ('Interest')
Y.2	Knowledge about typical behaviour patterns of software development projects ('Knowledge')
Y.3	Understanding of 'simple' project dynamics ('Understand simple')
Y.4	Understanding of 'complex' project dynamics ('Understand complex')
Subjective perceptions	
Z.1	Available time budget versus time need ('Time pressure')
Z.2	Session evaluation

2.3.2. Subjective perceptions

The values of variables Z.1, and Z.2 were derived from questionnaires, too. The related questions can be characterised as follows:

Z.1 ('Time pressure'): Questions on actual time consumption per scenario block, and on perceived time need. Note that the actual time consumption per scenario block can differ from the recommendations provided to the subjects via treatment instructions.
Z.2 ('Session evaluation'): Questions on personal judgement of the training session involving an assessment of the perceived degree of usefulness, entertainment, difficulty, and clarity.

2.3.3. Data collection procedure

The raw data for dependent variables Y.1 to Y.4 were collected during pre-test and post-test with the help of questionnaires. The full set of questions used in

[1] This shortcoming might be resolved soon. In Ref. [4], Ray Madachy and Barry Boehm announce the integration of System Dynamics into COCOMO.

the experiments can be found in Ref. [22]. Selected examples are presented in Appendix A. The values for variable Y.1 ('Interest') are average scores derived from five questions on the student's opinion about the importance of software project management issues (i) during university education and (ii) during performance of industrial software development projects, applying a five-point Likert-type scale [15]. Each answer in the questionnaire is mapped to the value range $R = [0, 1]$ assuming equidistant distances between possible answers,[2] i.e. 'fully disagree' is encoded as '0', 'disagree' as '0.25', 'undecided' as '0.5', 'agree' as '0.75', and 'fully agree' as '1'.

The values for variables Y.2 ('Knowledge'), Y.3 ('Understand simple'), and Y.4 ('Understand complex') are average scores derived from five (for Y.2), seven (for Y.3), and six (for Y.4) questions in multiple-choice style. The answers to these questions were evaluated according to their correctness, thus having a binary scale with correct answers encoded as '1', and incorrect answers encoded as '0'. Missing answers were encoded like incorrect answers.

The raw data of the variables related to subjective perception were collected after the post-test (Z.1 and Z.2). Again, the full set of questions can be found in Ref. [22].

The values for variable Z.1 are normalised average scores reflecting the 'time need' for reading and understanding of the scenario blocks 1, 3, and 4, for familiarisation with the supporting tools, and for filling in the post-test questionnaire. For group A, the variable $Z.1_{B2}$ represents the scores related to scenario block 2 only. If a subject wants to express that more than the available time was needed related to a certain task, then 'yes' (encoded as '1') should be marked, otherwise 'no' (encoded as '0'). Adding the scores and dividing them by the number of tasks once again provides a normalised value range $R = [0, 1]$, with '1' indicating time need for all tasks and '0' indicating the absence of time need.

The values for variable Z.2 ('Session evaluation') are based on subjective measures reflecting the quality of the treatment related to scenario blocks 1, 3, and 4. Again, for group A, the variable $Z.2_{B2}$ represents scores related to scenario block 2 only. The subjective perception of the treatment quality was evaluated with regard to four dimensions ('useful' versus 'useless', 'absorbing' versus 'boring', 'easy' versus 'difficult', and 'clear' versus 'confusing') using five-point Likert-type scales, e.g. 'extremely boring', 'boring', 'undecided', 'absorbing', 'extremely absorbing'. Similar to variable Y.1, possible answers were encoded as '0', '0.25', '0.5', '0.75', and '1' depending on whether the subjective judgement was very negative, negative, undecided, positive, or very positive. By taking the average of the values for all four questions the values for disturbing factors could be mapped to range $R = [0, 1]$.

[2] This is a common assumption in experimental software engineering and social science.

2.4. Data analysis

Standard significance testing was used to investigate the effect of the treatments on the dependent variables Y.1 to Y.4. The null hypotheses were stated as follows:

$H_{0,1}$: There is no difference between pre-test scores and post-test scores within experimental group A and control group B.
$H_{0,2a}$: There is no difference in relative learning effectiveness between experimental group A and control group B.
$H_{0,2b}$: There is no difference in absolute learning effectiveness between experimental group A and control group B.

For testing hypothesis $H_{0,1}$, a one-way paired t-test was used, because the data collected for this hypothesis is within-subjects, i.e. post-test scores are compared to pre-test scores of subjects within the same group [26]. For testing hypotheses $H_{0,2a}$ and $H_{0,2b}$, the appropriate test was a one-sided t-test for independent samples [26].

A prerequisite for applying the t-test is the assumption of normal distribution of the variables in the test samples. A test to check this assumption was conducted. While the results of the t-test are often robust against violation of the normality assumption it is strongly influenced by outliers in the data sets. Hence, an analysis to detect the presence of outliers was performed. Checking for the normality assumption showed that no normal distribution of the variables in the test samples could be assumed. On the other hand, the outlier analysis showed that all data points lie within the range of ± 2 standard deviations around the samples' means, and in most cases even within the range of ± 1.5 standard deviations around the samples' means. Although no outliers were detected, additional non-parametric tests were conducted to re-confirm the findings of the t-tests. Since the non-parametric tests, i.e. Wilcoxon matched pair test for hypothesis $H_{0,1}$ and Mann–Whitney U test for hypotheses $H_{0,2a}$ and $H_{0,2b}$, did not yield any difference to the results of the t-tests, the non-parametric test results are not presented in the paper.

Ideally, researchers should perform a power analysis [5] before conducting a study to ensure the experimental design will find a statistically significant effect if one exists. The power of a statistical test is dependent on three different components: significance level α, the size of the effect being investigated, and the number of subjects. Low power will have to be considered when interpreting non-significant results.

Usually, the commonly accepted practice is to set $\alpha = 0.05$. However, controlling a Type I error (α) and Type II error (β) requires either a large effect size or large sample

sizes. This represents a dilemma in a software-engineering context since much research in this area involves relatively modest effects sizes, and in general, small sample sizes. As pointed out in Ref. [18], if neither effect size nor sample size can be increased to maintain a low risk of error, the only remaining strategy—other than abandoning the research altogether—is to permit higher risk of error. Since sample sizes were rather small in the initial experiment and its replications, and no sufficiently stable effect sizes from previous empirical studies were known, it was decided to set $\alpha = 0.1$.

Having the results of the initial experiment and its replications, meta-analysis techniques were applied to produce a single quantitatively synthesized conclusion (cf. Appendix B for details). Meta-analysis consists of a set of statistical procedures performed for the purpose of integrating findings from individual studies. Generalizing and aggregating the results, the meta-analysis also helps to confirm empirical findings from individual studies, each of which can have insufficient statistical power to reliably accept or reject the hull hypothesis.

Statistical methods available for combining results of independent studies, which are known as 'combined tests', range from various counting procedures involving either significance level (p-values), or raw or weighted test statistics as t-test or z-test statistics. Practically speaking, the results of the various combined tests are typically consistent with each other. To strengthen reliability of the results, though, in this work two combined tests were performed: the Fisher and the Stouffer procedures, as they are both suitable for small sample sizes [30].

As statistical tests themselves do not provide any insight into the strength of the relationship or effect of interest, it is also desirable to accompany combined test with indexes of effect sizes. Effect size is expressed as the difference between the means of the two samples divided by the root mean square of the variances of the two samples [26]. For this exploratory study, effects with $\gamma \geq 0.5$ are considered to be of practical significance. This decision was made on the basis of the effect size indices proposed by Cohen [5]. Combining effect sizes can be made in two ways: by calculating the sample mean or by calculating the weighted sample mean. The second estimate has been shown in Ref. [10] to be asymptotically efficient, while the sample mean is slightly biased.

Before combining the results it is suggested to compare studies in order to determine their homogeneity, i.e. to check if they differ among themselves significantly. Heterogeneity provides a warning that it may not be sufficient to combine the studies in one meta-analysis. In this case an additional investigation of reasons that cause variation is necessary. The tests for homogeneity of statistical tests and effect sizes were performed in this work.

3. Results

In the following sub-sections the descriptive statistics of the dependent variables and disturbing factors are summarised. Then, for each experimental hypothesis the results of the statistical analyses (including meta-analysis) are presented and briefly discussed.

3.1. Descriptive statistics

Table 6 accumulates the descriptive statistics for the initial experiment and both replications. The columns 'Pre-test scores' and 'Post-test scores' show the calculated values for mean, median, and standard deviation of the raw data collected during the pre-test and post-test, respectively, of the initial experiment (E) and the two replications (R1 and R2) for both experimental groups A and control groups B.

The column 'Difference scores' of Table 6 shows the calculated values for mean, median, and standard deviation of the differences between post-test and pre-test scores of the initial experiment (E) and two replications (R1 and R2). The italicised data indicate that the difference between average post-test scores and average pre-test scores is zero or even negative, i.e. based on average data no (or even negative) relative learning effect was observed. This phenomenon occurred twice during the initial experiment (variables Y.4 (group A) and Y.1 (group B)), three times during the first replication (variables Y.4 (groups A and B) and Y.2 (group B)) and three times during the second replication (variables Y.1 (group A), Y.2 (group B) and Y.4 (group A)). Possible reasons for these unexpected outcomes are discussed in Section 3.4.

Table 7 shows the calculated values for mean, median, and standard deviation of the raw data collected on subjective perceptions during the initial experiment (E) and two replications (R1 and R2) for both experimental groups A and control groups B.

In the initial experiment, students in the control groups expressed less need of additional time (variable Z.1) for conducting the treatment and completing the tests than did students in the experimental groups. Probably due to the fact that in both replications more time was available for the treatment, in both groups the request for more time to conduct scenario blocks 1, 3, and 4 decreased below the values of the control group in the initial experiment. This is only partly true when looking at the time need expressed by students of the experimental groups with regards to conducting scenario block 2 (variable $Z.1_{B2}$).

Finally, in all cases—in the initial experiment and both replications, students in the control groups on average perceived their treatment easier, clearer, more absorbing, and more useful (variable Z.2) than the students in the experimental groups. This evaluation, however, relates only to those scenario blocks that are conducted by both groups, i.e. blocks 1, 3 and 4. When looking at the evaluation

Table 6
Scores of dependent variables (E, R1 and R2)

	Pre-test scores				Post-test scores				Difference scores			
	Y.1	Y.2	Y.3	Y.4	Y.1	Y.2	Y.3	Y.4	Y.1	Y.2	Y.3	Y.4
E: initial experiment (KL)												
Group A (5 subj.)												
Mean	0.69	0.56	0.31	0.37	0.79	0.84	0.66	0.43	0.10	0.28	0.34	0.07
Median	0.75	0.60	0.29	0.33	0.85	0.80	0.71	0.33	0.10	0.40	0.43	*0.00*
Stdev.	0.18	0.30	0.26	0.25	0.19	0.17	0.13	0.32	0.09	0.36	0.28	0.19
Group B (4 subj.)												
Mean	0.81	0.50	0.43	0.33	0.79	0.60	0.82	0.46	*− 0.03*	0.10	0.39	0.13
Median	0.78	0.50	0.36	0.25	0.80	0.60	0.86	0.50	*0.00*	0.10	0.50	0.17
Stdev.	0.13	0.26	0.31	0.24	0.19	0.16	0.07	0.37	0.09	0.35	0.38	0.34
R1: first replication (Oulu)												
Group A (6 subj.)												
Mean	0.83	0.57	0.41	0.44	0.85	0.97	0.67	0.44	0.03	0.40	0.26	*0.00*
Median	0.88	0.60	0.43	0.42	0.85	1.00	0.57	0.50	0.03	0.40	0.14	*0.00*
Stdev.	0.14	0.23	0.11	0.23	0.15	0.08	0.27	0.09	0.07	0.28	0.25	0.24
Group B (6 subj.)												
Mean	0.70	0.47	0.33	0.33	0.78	0.43	0.74	0.33	0.08	*− 0.03*	0.41	*0.00*
Median	0.73	0.40	0.36	0.25	0.83	0.40	0.79	0.33	0.08	*0.00*	0.43	0.08
Stdev.	0.18	0.21	0.17	0.30	0.21	0.15	0.17	0.24	0.13	0.32	0.19	0.45
R2: second replication (Calgary)												
Group A (7 subj.)												
Mean	0.84	0.63	0.41	0.62	0.87	0.83	0.61	0.52	0.03	0.20	0.20	*− 0.10*
Median	0.85	0.60	0.43	0.67	0.85	0.90	0.71	0.67	*0.00*	0.20	0.14	*0.00*
Stdev.	0.09	0.15	0.27	0.30	0.10	0.23	0.24	0.20	0.12	0.22	0.35	0.19
Group B (6 subj.)												
Mean	0.82	0.60	0.43	0.44	0.91	0.60	0.55	0.53	0.09	*0.00*	0.12	0.09
Median	0.88	0.70	0.43	0.50	0.95	0.70	0.57	0.50	0.05	*0.00*	0.14	0.07
Stdev.	0.18	0.33	0.13	0.30	0.11	0.42	0.11	0.19	0.11	0.18	0.19	0.21

of scenario block 2 (variable $Z.2_{B2}$), high scores can be observed for the experimental group in the initial experiment, and even higher scores in both replications (cf. Table 8 for detailed evaluation results).

3.2. Inferential statistics

In the following, the results of statistical hypotheses testing are presented for each hypothesis individually.

3.2.1. Hypothesis $H_{0,1}$

Null hypotheses $H_{0,1}$ was stated as follows: There is no difference between pre-test scores and post-test scores within experimental group A and control group B.

Focusing on the experimental groups only, Table 9 shows for each dependent variable separately the results of testing Hypothesis $H_{0,1}$ using a one-tailed *t*-test for dependent samples. Column one specifies the variable

Table 7
Scores of subjective perceptions (E, R1 and R2)

	E: initial experiment (KL)				R1: first replication (Oulu)				R2: second replication (Calgary)			
	Z.1	$Z.1_{B2}$	Z.2	$Z.2_{B2}$	Z.1	$Z.1_{B2}$	Z.2	$Z.2_{B2}$	Z.1	$Z.1_{B2}$	Z.2	$Z.2_{B2}$
Group A												
Mean	0.44	0.20	0.41	0.68	0.17	0.00	0.35	0.82	0.19	0.20	0.37	0.79
Median	0.4	0.00	0.38	0.69	0.20	0.00	0.38	0.82	0.17	0.20	0.44	0.81
Stdev.	0.46	0.45	0.09	0.26	0.15	0.00	0.20	0.17	0.12	0.12	0.19	0.25
Group B												
Mean	0.35		0.66		0.25		0.71		0.21		0.67	
Median	0.3		0.69		0.25		0.75		0.13		0.69	
Stdev.	0.19		0.06		0.19		0.18		0.25		0.14	

Table 8
Subjective evaluation of Scenario Block 2 (for E, R1 and R2)

	$Z.2_{B2}$ (Table 7)	Useful	Absorbing	Easy	Clear
Group A (E)					
Mean	0.68	0.75	0.6	0.65	0.7
Median	0.69	0.75	0.5	0.75	0.75
Stdev.	0.26	0.31	0.29	0.22	0.27
Group A (R1)					
Mean	0.82	0.88	0.85	0.79	0.75
Median	0.82	0.88	1.00	0.75	0.75
Stdev.	0.17	0.14	0.22	0.19	0.27
Group A (R2)					
Mean	0.79	0.93	0.71	0.71	0.82
Median	0.81	1.00	0.75	0.75	0.75
Stdev.	0.25	0.12	0.37	0.30	0.19

(from Y.1 to Y.4) and the related study, i.e. initial experiment (E), first replication (R1), and second replication (R2). Column two represents the size of the effect detected, column three the degrees of freedom, column four the t-value of the study, column five the critical value for $\alpha = 0.10$ (as discussed in Section 2.4) that the t-value has to exceed to be statistically significant, and column six provides the associated p-value.

By examining columns four and five of Table 9, one can see that the experimental groups achieved a statistically and practically significant result for dependent variable Y.1 only in the initial experiment. It should be noted, though, that in both replications the Y.1 values support the direction of the expected positive learning effect. For dependent variables Y.2 and Y.3 statistically and practically significant results were obtained in all three studies, while dependent variable Y.4 did not yield a significant result (neither statistical nor practical) in any of the studies. In both replications, the Y.4 values do not even support the direction of the hypothesis.

Table 9
Results for 'post-test' versus 'pre-test' for group A

Variable/Study	γ	df	t-Value	Crit. $t_{0.90}$	p-Value
Variable Y.1 ('Interest')					
E	1.07	4	2.39	1.53	0.04
R1	0.36	5	0.89	1.48	0.21
R2	0.23	6	0.62	1.44	0.28
Variable Y.2 ('Knowledge')					
E	0.77	4	1.72	1.53	0.08
R1	1.41	5	3.46	1.48	0.01
R2	0.91	6	2.24	1.44	0.04
Variable Y.3 ('Understand simple')					
E	1.23	4	2.75	1.53	0.03
R1	1.06	5	2.61	1.48	0.02
R2	0.59	6	1.55	1.44	0.09
Variable Y.4 ('Understand complex')					
E	0.35	4	0.78	1.53	0.24
R1	0.00	5	0.00	1.48	0.50
R2	−0.50	6	−1.33	1.44	0.88

The next two tables summarise the group A related meta-analysis results for testing Hypothesis $H_{0,1}$.

Table 10 shows the results of comparing and combining the p-values. First, the homogeneity test was performed for each dependent variable. Since no significant result was obtained, the hypothesis that the three studies are homogeneous cannot be rejected, and thus it can be proceeded with combining the p-values. The results of both combined tests (Fisher and Stouffer's Z) were found statistically significant for variables Y.1, Y.2 and Y.3, but no statistical significance can be reported for variable Y.4. The Y.4 data does not even support the direction of the hypothesis.

Table 11 shows the results of comparing and combining the effect sizes. Again, no homogeneity problems were detected. Consistent with the results shown in Table 10, the effect sizes for variables Y.1, Y.2 and Y.3 were found practically significant, and the effect size for variable Y.4 does not even support the direction of the hypothesis.

Focusing on the control groups only, Table 12 shows for each dependent variable separately the results of testing Hypothesis $H_{0,1}$ using a one-tailed t-test for dependent samples. The structure of the table is the same as in Table 9.

By examining columns four and five of Table 12, one can see that the control groups achieved statistically and practically significant results for dependent variable Y.1 only in the replications. In the initial experiment, the Y.1 data did not even support the direction of the hypothesis. For dependent variables Y.2 and Y.4 no significant results could be found. The data of the initial experiment and the second replication at least support the direction of expected positive learning effect. The results for dependent variable Y.3 were found statistically and practically significant in all studies.

The next two tables summarise the group B related meta-analysis results for testing hypothesis $H_{0,1}$.

Table 13 shows the results of comparing and combining the p-values. As in the case of the experimental groups, no homogeneity problem could be found and thus the p-values for the control groups could be combined. It can be seen from the last two columns of Table 13 that variables Y.1 and Y.3 showed statistical significance. The combined data for variables Y.2 and Y.4 at least support the direction of expected positive learning effect.

Table 14 shows the results of comparing and combining the effect sizes. Again, as for p-values, no significant results in the homogeneity tests were obtained. The combined effect sizes could be considered as practically significant only for variable Y.3, while the combined data for variables Y.1, Y.2, and Y.4 at least support the direction of the hypothesis, i.e. the expected positive learning effect.

3.2.2. Hypothesis $H_{0,2a}$

Null hypothesis $H_{0,2a}$ was stated as follows: There is no difference in relative learning effectiveness between experimental group A and control group B, i.e. the difference

Table 10
Comparing and combining the p-values for 'post-test' versus 'pre-test' for group A

Variable	Homogeneity test			Combined p-values			
	Q	Crit. $Q_{0.90}$	p-Value	P	Crit. $P_{0.90}$	Fisher	Stouffer's Z
Y.1	0.80	4.61	0.67	12.26	10.64	0.06	0.03
Y.2	0.47	4.61	0.79	21.03	10.64	0.0018	0.00068
Y.3	0.24	4.61	0.89	19.70	10.64	0.0031	0.00112
Y.4	1.86	4.61	0.39	4.50	10.64	0.61	0.61

between post-test and pre-test scores of group A is not significantly larger than the one of group B.

Table 15 shows for each dependent variable separately the results of testing hypothesis $H_{0,2a}$ using a one-tailed t-test for independent samples. It turns out that for variable Y.1 hypothesis $H_{0,2a}$ can be rejected only for the initial experiment, whereas for both replications the values do not even support the direction of the expected relative learning effect. For dependent variable Y.2 statistically significant level was achieved in both replications and the result of the initial experiment supports the direction of the expected relative learning effect with practical significance, i.e. showing a medium effect size. In none of the studies the values for variables Y.3 and Y.4 can be considered significant and only in the second replication the result for variable Y.3 does not contradict the direction of hypothesis, demonstrating though small effect size, i.e. not having practical significance.

The next two tables show the meta-analysis results for testing hypothesis $H_{0,2a}$.

Table 16 shows the results of comparing and combining the p-values. As can be seen, the homogeneity test was found significant for variable Y.1 (indicated in italics). Examining the p-values for variable Y.1, as presented in Table 15, indeed shows a statistically significant result with a p-value of 0.04 in the initial experiment which is complemented by relatively large p-values in the replications (both p-values are 0.82). The test for homogeneity rejects the hypothesis that this variation is due to a sampling error. It should be noted, though, that there is some debate in the literature concerning what to do with the results of independent studies when their results were found significantly non-homogeneous. Hedges [10] and Hunter et al. [11] suggest that it is inappropriate to combine these results in a meta-analysis. Harris and Rosenthal [9] argue that heterogeneity is analogous to individual differences among subjects within single studies and is common whenever studies by different investigators using different methods are examined. Possible reasons for the variation between the studies reported in this paper are further discussed in Section 3.4. For the remaining variables, no heterogeneity effect was found.

Table 11
Comparing and combining the effect sizes for 'post-test' versus 'pre-test' for group A

Variable	Homogeneity test			Combined effect sizes	
	Q	Crit. $Q_{0.90}$	p-Value	Average	Weighted average
Y.1	0.50	4.61	0.78	0.56	0.49
Y.2	0.28	4.61	0.87	1.03	1.02
Y.3	0.32	4.61	0.85	0.96	0.91
Y.4	0.54	4.61	0.76	−0.05	−0.10

Combining the p-values yields a statistically significant result for variable Y.2, while the data for variable Y.1, Y.3, and Y.4 not even support the direction of the expected relative learning effect.

Table 17 shows the results of comparing and combining the effect sizes. The meta-analysis results for effect sizes are consistent with those for p-values: the data for variable Y.1 can neither be considered homogeneous nor do they show any significant results; the data for variable Y.2 are practically significant regarding hypothesis $H_{0,2a}$, while the data for variables Y.3 and Y.4 do not even support the direction of the hypothesis.

3.2.3. Hypothesis $H_{0,2b}$

Null hypothesis $H_{0,2b}$ was stated as follows: There is no difference in absolute learning effectiveness between

Table 12
Results for 'post-test' versus 'pre-test' for group B

Variable/study	γ	df	t-Value	Crit. $t_{0.90}$	p-Value
Variable Y.1 ('Interest')					
E	−0.29	3	−0.58	1.64	0.70
R1	0.65	5	1.58	1.48	0.09
R2	0.82	5	2.02	1.48	0.05
Variable Y.2 ('Knowledge')					
E	0.29	3	0.58	1.64	0.30
R1	−0.10	5	−0.25	1.48	0.60
R2	0.00	5	0.00	1.48	0.50
Variable Y.3 ('Understand simple')					
E	1.05	3	2.09	1.64	0.06
R1	2.13	5	5.22	1.48	0.0017
R2	0.63	5	1.54	1.48	0.09
Variable Y.4 ('Understand complex')					
E	0.37	3	0.73	1.64	0.26
R1	0.00	5	0.00	1.48	0.50
R2	0.41	5	1.01	1.48	0.18

Table 13
Comparing and combining the p-values for 'post-test' versus 'pre-test' for group B

Variable	Homogeneity test			Combined p-values			
	Q	Crit. $Q_{0.90}$	p-Value	P	Crit. $P_{0.90}$	Fisher	Stouffer's Z
Y.1	2.76	4.61	0.25	11.59	10.64	0.07	0.08
Y.2	0.30	4.61	0.86	4.82	10.64	0.57	0.44
Y.3	1.53	4.61	0.47	23.01	10.64	0.00079	0.00043
Y.4	0.45	4.61	0.80	7.53	10.64	0.27	0.18

experimental group A and control group B, i.e. the post-test scores of group A are not significantly larger than those of group B.

Table 18 shows for each dependent variable separately the results of testing hypothesis $H_{0,2b}$ using a one-tailed t-test for independent samples. It turns out that variable Y.2 shows statistically significant results in the initial experiment and its first replication, and practically significant results for the second replication. Thus, hypothesis $H_{0,2a}$ can be rejected for Y.2. Apart from the first replication, where a practically significant result for variable Y.4 was achieved, for variables Y.1, Y.3 and Y.4 no significant results were obtained in any of the studies. The data for variables Y.3 and Y.4 in the initial experiment, for Y.3 in the first replication, and for Y.1 and Y.4 in the second replication do not even support the direction of the hypothesis.

The next two tables show the meta-analysis results for testing hypothesis $H_{0,2b}$.

Table 19 shows the results of comparing and combining the p-values. As can be seen, the homogeneity test was found statistically significant only for variable Y.2. Looking at the corresponding p-values for this variable (Table 18), one can see that the p-value obtained in the first replication is extremely small (0.000009) comparing to the rest ones (0.03 in the initial experiment and 0.13 in the second replication). So, in contrast to the case of hypothesis $H_{0,2a}$ (cf. discussion related to Table 17), in this case, the individual studies showed statistically or, at least, practically significant results, but to a strongly varying degree. Again, possible reasons for the detected variation will be further discussed in Section 3.4. For the remaining variables no heterogeneity effect was found.

Apart from the statistically significant result for variable Y.2, combining the p-values does not yield any significant results for the remaining variables Y.1, Y.3, and Y.4. The data for variables Y.1 and Y.3 do not even support the direction of the hypothesis.

Table 20 shows the results of comparing and combining the effect sizes. Again, the meta-analysis results for effect sizes are fully consistent with those for p-values: the data for variable Y.2 cannot be considered homogeneous but does show practically significance; the data for variable for variables Y.1 and Y.3 do not even support the direction of the hypothesis, while the data for variable Y.4 at least support the direction of the hypothesis.

Table 14
Comparing and combining the effect sizes for 'post-test' versus 'pre-test' for group B

Variable	Homogeneity test			Combined effect sizes	
	Q	Crit. $Q_{0.90}$	p-Value	Average	Weighted average
Y.1	0.79	4.61	0.68	0.39	0.47
Y.2	0.09	4.61	0.95	0.06	0.03
Y.3	1.32	4.61	0.52	1.27	1.18
Y.4	0.15	4.61	0.93	0.26	0.24

3.3. Other data

In addition to filling in the pre-test and post-test questionnaires and the questionnaires about personal characteristics and subjective perceptions, participants in the experimental studies had the chance to make comments or improvement suggestions, and could raise issues or problems that they encountered during the treatments. Apart from some improvement suggestions related to technical aspects of the role-play and tool usage, comments and problem statements mainly supported the findings of

Table 15
Results for 'performance improvement'

Group A versus B					
Variable/study	γ	df	t-Value	Crit. $t_{0.90}$	p-Value
Variable Y.1 ('Interest')					
E	1.38	7	2.06	1.42	0.04
R1	−0.56	10	−0.98	1.37	0.82
R2	−0.54	11	−0.97	1.36	0.82
Variable Y.2 ('Knowledge')					
E	0.51	7	0.75	1.42	0.24
R1	1.43	10	2.48	1.37	0.02
R2	1.00	10	1.73	1.37	0.06
Variable Y.3 ('Understand simple')					
E	−0.16	7	−0.23	1.42	0.59
R1	−0.65	10	−1.13	1.37	0.86
R2	0.30	11	0.53	1.36	0.30
Variable Y.4 ('Understand complex')					
E	−0.23	7	−0.33	1.42	0.62
R1	0.00	10	0.00	1.37	0.50
R2	−0.92	11	−0.65	1.36	0.94

Table 16
Comparing and combining the p-values for 'performance improvement'

Group A versus B							
Variable	Homogeneity test			Combined p-values			
	Q	Crit. $Q_{0.90}$	p-Value	P	Crit. $P_{0.90}$	Fisher	Stouffer's Z
Y.1	4.81	4.61	*0.09*	7.24	10.64	*0.30*	*0.52*
Y.2	1.03	4.61	0.60	16.85	10.64	0.01	0.01
Y.3	1.25	4.61	0.53	3.76	10.64	0.71	0.67
Y.4	1.29	4.61	0.52	2.46	10.64	0.87	0.85

the quantitative analyses. Positive comments mainly correlated with the high scores for scenario block 2 in the subjective evaluation of the experimental group. In addition, positive statements were made about the clarity of the presentation of the COCOMO model and its usefulness. Negative comments or problem statements mainly addressed the difficulty of understanding the structure of the SD model, and the lack of time for getting acquainted with the tools, for working through the treatments, and for answering the questions in the pre-test and post-test questionnaires. The time issue, however, was less prominent during the replications.

3.4. Summary

This section first summarises the results of the initial experiment and its two replications with regards to null hypothesis $H_{0,1}$ (Table 21), and null hypotheses $H_{0,2a}$ and $H_{0,2b}$ (Table 22) for each dependent variable separately. Then the respective results of the related meta-analyses are presented (Table 23). The following abbreviations are used for table entries.

Statistical significance (stat. sig.): null hypothesis could be rejected at significance level $\alpha = 0.1$.

Practical significance (pract. sig.): null hypothesis could not be rejected but effect size $\gamma \geq 0.5$. If statistical significance is achieved, practical significance is not mentioned.

Positive effect (+): no practical significance could be observed but effect size $\gamma > 0$. The number in parentheses indicates how many subjects would have been needed to achieve statistical significance with the given effect size (only in Table 22).

No effect or negative effect (−): t-value ≤ 0.

Table 21 shows that null hypothesis $H_{0,1}$ could only be rejected in all experiments for variable Y.3 (experimental and control groups). In addition, for the experimental groups, $H_{0,1}$ could be rejected in all cases for Y.2 and in one case for Y.1. For the control groups, $H_{0,1}$ could be rejected in two cases for Y.1, too.

Table 22 shows that null hypothesis $H_{0,2a}$ could only be rejected in all cases for variable Y.2. In addition, a significant result was achieved in one case for variable Y.1. Regarding null hypothesis $H_{0,2b}$, statistical testing yielded statistically and practically significant results for variable Y.2, too. Practically, significant results were achieved in one case for variable Y.4.

Table 23 summarises the meta-analysis results related to null hypotheses $H_{0,1}$, $H_{0,2a}$, and $H_{0,2b}$. The italicised data indicate the cases, where the homogeneity test failed.

In general terms, the meta-analysis results fully confirm—but simplify and focus—the interpretation of the assembled individual studies' results.

Table 17
Comparing and combining the effect sizes for 'performance improvement'

Group A versus B					
Variable	Homogeneity test			Combined effect sizes	
	Q	Crit. $Q_{0.90}$	p-Value	Average	Weighted average
Y.1	5.20	4.61	*0.07*	*0.09*	*− 0.10*
Y.2	0.98	4.61	0.61	0.98	0.99
Y.3	1.35	4.61	0.51	−0.17	−0.15
Y.4	1.33	4.61	0.52	−0.38	−0.39

Table 18
Results for 'post-test improvement'

Group A versus B					
Variable/Study	γ	df	t-Value	Crit. $t_{0.90}$	p-Value
Variable Y.1 ('Interest')					
E	0.01	7	0.02	1.42	0.49
R1	0.36	10	0.63	1.37	0.27
R2	−0.36	11	−0.64	1.36	0.73
Variable Y.2 ('Knowledge')					
E	1.45	7	2.16	1.42	0.03
R1	4.40	10	7.63	1.37	0.000009
R2	0.69	10	1.19	1.37	0.13
Variable Y.3 ('Understand simple')					
E	−1.53	7	−2.28	1.42	0.97
R1	−0.32	10	−0.56	1.37	0.71
R2	0.33	11	0.60	1.36	0.28
Variable Y.4 ('Understand complex')					
E	−0.07	7	−0.11	1.42	0.54
R1	0.63	10	1.08	1.37	0.15
R2	−0.02	11	−0.04	1.36	0.51

Table 19
Comparing and combining the p-values for 'post-test improvement'

Group A versus B							
Variable	Homogeneity test			Combined p-values			
	Q	Crit. $Q_{0.90}$	p-Value	P	Crit. $P_{0.90}$	Fisher	Stouffer's Z
Y.1	0.76	4.61	0.68	4.64	10.64	0.59	0.50
Y.2	5.51	4.61	*0.06*	34.08	10.64	*0.000006*	*0.000015*
Y.3	3.11	4.61	0.21	3.30	10.64	0.77	0.87
Y.4	0.81	4.61	0.67	6.33	10.64	0.40	0.31

Regarding variable Y.1 (interest in software project management) the expected positive learning effect could be observed for both experimental and control groups (hypothesis $H_{0,1}$). However, the expected positive impact of involving the SD model and using a role-play in the treatment was neither found for hypothesis $H_{0,2a}$ nor for hypothesis $H_{0,2b}$. Even worse, not even the directions of the hypotheses are supported by meta-analysis results. While the interpretation of the result for variable Y.1 is clear for the case of testing for absolute learning effectiveness (hypothesis $H_{0,2b}$), the failed test for homogeneity in the case of hypothesis $H_{0,2a}$ leaves some room for further interpretation. Clearly, the lack for homogeneity was caused by the fact that in the initial experiment the result was statistical significant, while in the following two replications the data not even supported the direction of the underlying hypothesis. In order to explain this variation, one can assume the impact of other factors, e.g. difference in personal characteristics or background knowledge of subjects (Table 1), difference in the perception of time pressure by subjects (variable Z.1), or difference in the evaluation of the treatments by subjects (variable Z.2). The application of an ANCOVA [29], however, did not yield any improvement of the situation[3] or—in the case of personal characteristics and background knowledge—could not be conducted due to lack of data. Having no further knowledge of other plausible explanations at hand, as another source of variation the differences in the schedules of the initial experiment as compared to both replications could be assumed. This assumption would support the conclusion that choosing the schedule of the initial experiment yields a statistically significant increase of interest in software project management when applying a treatment involving role-play and using SD models, while choosing the schedule of the replications does not.

Regarding variable Y.2 (knowledge about empirical patterns in software projects) the expected positive learning effect was significant only for the experimental group (hypothesis $H_{0,1}$). On the other hand, meta-analysis results clearly support the expectation that subjects in the experimental group perform significantly better than subjects in the control group for both relative and absolute scores (hypotheses $H_{0,2a}$ and $H_{0,2b}$). Here, though, for hypothesis $H_{0,2b}$, homogeneity of the data cannot be assumed. Thus, the question of validity of the meta-analysis results arises again. In this case, however, differently to the situation of variable Y.1, revisiting the data used for the meta-analysis (Table 19, last column for variable Y.2) suggests that the discovered heterogeneity effect is simply caused by the fact that the achieved level of significance in one study (first replication) is too large as compared to the achieved significance level of the others. Since all three studies either show statistical or practical significance in testing hypothesis $H_{0,2b}$, the combination of the data in the meta-analysis, again yielding a statistically significant result, should not become subject of doubt just because the level of significance between the individual studies is varying too much.

Regarding variable Y.3 (understanding of simple project dynamics) again the expected positive learning effect could be observed for both experimental and control groups (hypothesis $H_{0,1}$). However, when comparing the relative and absolute learning effects of the experimental groups with those of the control groups (hypotheses $H_{0,2a}$ and $H_{0,2b}$), the expected positive impact of involving the SD model and using a role-play in the treatment was not achieved. Even worse, as in the case of variable Y.1 not even the directions of the hypotheses $H_{0,2a}$ and $H_{0,2b}$ are supported by meta-analysis results.

Finally, regarding variable Y.4 (understanding of complex project dynamics), meta-analysis results do not

Table 20
Comparing and combining the effect sizes for 'post-test improvement'

Group A versus B					
Variable	Homogeneity test			Combined effect sizes	
	Q	Crit. $Q_{0.90}$	p-Value	Average	Weighted average
Y.1	0.80	4.61	0.67	0.01	−0.01
Y.2	9.42	4.61	*0.01*	*2.18*	*1.49*
Y.3	3.92	4.61	0.14	−0.51	−0.32
Y.4	0.84	4.61	0.66	0.18	0.19

[3] Results of the ANCOVA for the initial experiment can be found in Ref. [24].

Table 21
Summary of individual results for $H_{0,1}$

Variable	Group A			Group B		
	Experiment	Replication 1	Replication 2	Experiment	Replication 1	Replication 2
Y.1	Stat. sig.	+	+	−	Stat. sig.	Stat. sig.
Y.2	Stat. sig.	Stat. sig.	Stat. sig.	+	−	−
Y.3	Stat. sig.	Stat. sig.	Stat. sig.	Stat. sig.	Stat. sig.	Stat. sig.
Y.4	+	−	−	+	−	+

Table 22
Summary of individual results for $H_{0,2}$

Variable	Group A versus B					
	$H_{0,2a}$			$H_{0,2b}$		
	Experiment	Replication 1	Replication 2	Experiment	Replication 1	Replication 2
Y.1	Stat. sig.	−	−	+ (1000)	+ (65)	−
Y.2	Pract. sig.	Stat. sig.	Stat. sig.	Stat. sig.	Stat. sig.	Pract. sig.
Y.3	−	−	+ (110)	−	−	+ (90)
Y.4	−	−	−	−	Pract. sig.	−

show any statistically or practically significant effect, neither for hypothesis $H_{0,1}$ nor for hypotheses $H_{0,2a}$ and $H_{0,2b}$.

4. Discussion

Starting out from the results presented in the previous section, interpretations and possible explanations of the outcomes of the experiments will be given below, followed by a discussion of the validity of the results.

4.1. Interpretation of results

Testing the positive learning effect within experimental groups confirmed a statistically significant positive impact on the change of scores from pre-test to post-test for dependent variables Y.1 to Y.3. This provides evidence for the assumption that the training session involving the SD model instead of COCOMO plus performing a role-play significantly increases interest in the topic of project management, knowledge about empirical patterns in software projects, and understanding of simple project dynamics of students participating in the training. On the other hand, no positive effect could be found for variable Y.4 (understanding of complex project dynamics) even though using the SD model was assumed to be an excellent tool to explain and analyse the complex interdependencies between project duration, effort consumption, and quality of the project outcome, i.e. the software product.

Testing the performance of relative and absolute learning effectiveness between experimental and control groups (hypotheses $H_{0,2a}$ and $H_{0,2b}$) showed that training involving SD model and role-play yields significantly better scores for variable Y.2 (knowledge about empirical patterns in software projects) than using COCOMO without role-play. On the other hand, no consistent significant difference between experimental and control groups could be observed regarding variable Y.1, Y.3 and Y.4 (interest in the topic of project management, understanding of simple project dynamics, understanding of complex project dynamics).

The strong effect observed for variable Y.2 when comparing the performance of experimental to control groups can probably be attributed to the inclusion of

Table 23
Summary of meta-analysis results $H_{0,1}$ and $H_{0,2}$

Variable	$H_{0,1}$		$H_{0,2a}$, Group A versus B	$H_{0,2b}$, Group A versus B
	Group A	Group B		
Y.1	Stat. sig.	Stat. sig.	−	−
Y.2	Stat. sig.	+	Stat. sig.	*Stat. sig.*
Y.3	Stat. sig.	Stat. sig.	−	−
Y.4	−	+	−	+

the role-play (scenario block 2) in the treatments of the experimental groups. As mentioned earlier (Section 2.2.1), scenario block 2 provided information on typical patterns of software project behaviour to subjects in the experimental groups, which was not given in such an explicit form to subjects in the control groups. It should be mentioned that the role-play relied on a planning model that covers complex interdependence between factors affecting project duration, effort consumption, and quality of the project outcome. The SD model offered the required functionality while COCOMO is restricted to model interdependencies between project duration and effort consumption only (simple project dynamics).

Inclusion of the role-play, on the other hand, imposed additional time pressure on the subjects in the experimental groups, which might have resulted in low scores for questions related to dependent variables Y.3 and Y.4. Although more time was allowed during the replications, and the values for time need were lower than in the initial experiment, qualitative results from the replication still support this subjective feeling of time pressure. Another reason for difficulties of the experimental group with variable Y.3 and-particularly variable Y.4 might be that the presentation of the SD model in scenario block 3 was too hard to grasp, due to the high complexity of the model structure. Some subjects mentioned this issue in the debriefing questionnaires.[4]

Moreover, it seems that the treatment of the experimental groups, as it is now, does not yet fully exploit all potentially available features that learning through SD model usage and SD model building could offer. SD models not only make causal relationships explicit and allow for variation of the strength of the relationships, but also offer means to change the structure of these relationships and make the effects of such changes on project performance (i.e. duration, effort, quality) visible through simulation. During the treatments, however, subjects of the experimental groups were only allowed to use the provided SD simulation model as-is. That is, students were confronted with a pre-defined model including very complex causal relationships and feedback structures without being allowed to really investigate individual relationships between model variables and their effects on project behaviour. Without having the chance to actively alter model structures it seems to be difficult to understand the SD model in its full complexity as compared to the mature and well structured, though simpler, COCOMO. Combining limitations, i.e. lack of time for model understanding and lack of active involvement in model building, it seems that chances are low to achieve a positive learning effect with regards to variable Y.4 in the experimental groups. This, however, would be a prerequisite for achieving significantly better learning effects in comparison to the control groups.

Finally, the fact that the performance of the experimental group with regards to variable Y.1 was only significantly better in the initial experiment but not in the two replications might be attributed to differences in the subjects' personal characteristics and/or background knowledge on software project management. At least in the case of the Oulu experiment, it can be observed that subjects generally were more mature and experienced as compared to the Kaiserslautern experiment. Since there was obviously still some positive learning effect in both experimental and control groups, one might conclude that the involvement of a role-play and simulations is no longer a distinguishing feature. This interpretation certainly needs more investigation in future replications.

4.2. Threats to validity

In the following, this section discusses various threats to validity of the study.

4.2.1. Construct validity

Construct validity is the degree to which the variables used in the study accurately measure the concepts they purport to measure. The following issues associated with construct validity have been identified:

1. The mere application of a SD model might not adequately capture the specific advantages of SD models over conventional planning models, since it has often been claimed that model building—and not the application of an existing model—is the main benefit of SD simulation modelling [14].
2. Interest in a topic and evaluation of a training session are difficult concepts that have to be captured with subjective measurement instruments. To counteract this threat to validity in the studies, the instruments for measuring variables Y.1 and Z.2 were derived from measurement instruments that had been successfully applied in a similar kind of study [28].
3. There are indications that the distinction between 'simple dynamics' and 'complex dynamics', as it was made for measuring variables Y.3 and Y.4 (Section 2.3.1), was too simplistic.
4. It is difficult to avoid 'unfair' comparison between usage of SD models and COCOMO, because SD models offer features that per definition are not available for COCOMO (e.g. simulation of parameter changes over time/on-the-fly modification of model assumptions, etc.). In addition, only scenario block 2 explicitly provides information about typical behaviour patterns of software projects, because this is an important prerequisite for conducting the role-play. Since exclusively subjects of

[4] It should be noted however, that a recent replication of the experiment at the University of Reading conducted by Daniel Rodriguez achieved significant results for variable Y.4 with regards to hypotheses $H_{0,1}$ and $H_{0,2a}$. The causes for the strong differences between the results reported here and those of the Reading experiment could not yet been clarified.

the experimental group perform scenario block 2, subjects of the control group might be disadvantaged.

4.2.2. Internal validity

Internal validity is the degree to which conclusions can be drawn about the causal effect of the independent variable on the dependent variables. Potential threats include selection effects, non-random subject loss, instrumentation effect, and maturation effect.

1. A selection effect was avoided by random assignment of subjects. In addition, existing differences in ability between groups were captured by collecting pre-test scores and by measuring the level of experience of subjects.
2. Non-random drop-out of subjects has been avoided by the experimental design, i.e. assignment of groups only directly before the treatment, and not before the pre-test at the beginning of the experiment.
3. The fact that the treatments of the experimental and control groups were different in the number of scenario blocks involved and, as a consequence, in the time available to perform each scenario block, may have induced an instrumentation effect. The post-mortem evaluation of the experiment with regard to time requirements indicated in the initial experiment that most subjects of the experimental group did not have enough time to execute both scenario blocks 2 and 3. Due to more relaxed schedules, this effect could be reduced during the replications of the initial experiment. In addition, even though this was thrived to be avoided by careful design, the planning models used in both treatments might slightly differ in scope and handling.
4. A maturation effect could have been caused if subjects had been informed before or during pre-test that at the end of the experiment they will complete a post-test with exactly the same questions. Since this information was not given to the subjects, and all materials were re-collected after the pre-test, it can be assumed that a maturation effect did not occur.

4.2.3. External validity

External validity is the degree to which the results of the research can be generalised to the population under study and other research settings. Two possible threats have been identified: subject representativeness and materials.

The subjects participating in the experiment were all students in computer science or related fields 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 (e.g. undergraduates at freshman level), or even with regard to training of software professionals should be done with caution.

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.

In any case, the point should be emphasised that the presented research at its current stage is exploratory of nature and just the first step of a series of experiments, which—after modification of the treatments and stepwise inclusion of subjects with different backgrounds—might yield more generalisable results in the future.

5. Summary and future work

The empirical studies presented in this paper investigated the effect of using a SD simulation model to assist software project management education of computer science students. The treatment focused on problems of project planning and control. The performance of the students was analysed with regard to four aspects, i.e. interest in the topic of software project management (Y.1), knowledge about typical project behaviour patterns (Y.2), understanding of simple project dynamics (Y.3), and understanding of complex project dynamics (Y.4). This was done by comparing the test results of students who completed a training session using the SD model (with web-based role-play) to the test results of students who performed a training session using COCOMO (without web-based role-play).

Many, but not all, findings of the initial experiment were corroborated by the two replications: SD models and role-play significantly stronger improve the students' knowledge of typical software project behaviour patterns. Qualitative data indicates that the inclusion of role-plays with SD models in project management education is perceived as a highly useful exercise. This is supported by the subjective evaluation of the role-play scenario involving simulation with the SD model, which received very high scores with regards to usefulness, entertainment, difficulty, and clarity.

Although the results of the three studies are promising, two negative results were found. First, the expectation that students receiving training with an SD model achieve a better understanding of complex project dynamics than students that work with COCOMO was not supported by any of the studies presented here. Second, the positive impact of working with an SD model on raising the interest in the field of software project management which was found in the initial experiment, was not confirmed by the replications.

A plausible explanation for the first negative finding could be that because of its high maturity and its relatively simple structure, COCOMO is much easier to grasp (and apply) by the students within the given time frame than the rather complex SD model with its multiple feedback loops. As a consequence, in future experiments, the presentation of the SD model should be improved. Maybe, due to its high complexity, it is even necessary that the students be involved in model building in an early phase of scenario block two of the treatment. This more active involvement

would yield a more constructive approach which could be realised in the form of an interactive session with tutoring or in a moderated group work setting. Whether such a revised treatment has a better learning effect regarding the understanding of complex project dynamics must be tested in future experiments.

As mentioned before, in order to develop an improved treatment involving simulation, further experiments must be conducted. These experiments should systematically alter individual elements of the treatments of the experimental and control groups in order to gain better understanding of the factors that increase the learning effect, i.e. drop scenario block 2 for the experimental group or include a role-play (using COCOMO) for the control group. The following questions should be addressed by future experiments: What is the main reason why the initial experiment yielded significant results for variable Y.1 (increase of interest in software project management) while the two replications did not? Is this due to the change of schedule, high pre-test scores, or other (yet) unknown factors? Why were the pre-test scores of variable Y.1 in the Oulu and Calgary replications much higher than in the initial experiment conducted in Kaiserslautern? What is the reason why the experimental group did not have a significant learning effect with regard to variable Y.4 (understanding of complex project dynamics)? Is it simply too difficulty (or boring) to understand a predefined SD model, i.e. should students be involved in the development of the model, or is the lacking effect just caused by inadequate presentation of the SD model and an inadequate time schedule? Even though, the role-play involving the SD model was considered as useful, absorbing, clear and easy by most of the students in the experimental group, could it be further improved so as to have a (more) positive impact on the post-test results of variables Y.3 and Y.4? In general, it would be important to identify specific causal relationships between certain elements of the treatment, i.e. scenario blocks, and the post-test performance related to variables Y.1 to Y.4. In order to reconfirm new findings from future experiments, again replication will be required to rule out specific threats to validity that are always exhibited when running empirical studies.

Finally, it should be mentioned that recently another replication following the Oulu/Calgary schedule was conducted at the University of Reading with 11 computer science students (six subjects in the experimental and five subjects in the control group). With regards to variables Y.1 to Y.3 the results of this replication reconfirm the results of the meta-analyses presented in Section 3. A major difference, however, was observed with regards to variable Y.4. Within both experimental and control groups a positive learning effect could be observed (even statistically significant for the experimental group). Moreover, with regards to relative learning effectiveness, the experimental group performed significantly better than the control group. Further analysis is required to determine the reasons for this result. Since content and schedule of the treatment were identical to those of the Oulu and Calgary replications, in addition to treatment development, future effort should be allocated to analysing the context of individual studies in order to better understand under which circumstances treatments show expected effects.

Acknowledgments

The authors thank all students that participated in the studies and all members of the technical staff at the Universities of Calgary, Kaiserslautern, and Oulu for helping with setting-up the technical environments of the experiments. Comments on an earlier version of this manuscript received from Patrick Waterson and the anonymous reviewers were highly appreciated. Part of this work was funded by the European Union under grant IST-1999-11634.

Appendix A. Example questions (pre-test, post-test, subjective perceptions)

Example question related to dependent variable Y.1 ('Interest'):

071	I consider it very important for computer science students to know as much as possible about software project management (1 = fully agree/5 = fully disagree)						
		1	2	3	4	5	
	Agree	O	O	O	O	O	Disagree

Example question related to dependent variable Y.2 ('Knowledge'):

081	For a typical software project, finding and fixing a software problem (defect) after delivery is about —3 times —5 times —10 times —100 times More expensive than finding and fixing it during the requirements and early design phases

Example question related to dependent variable Y.3 ('Understand simple'):

091	You have to estimate schedule and effort for a project of size 60,000 SLOC (source lines of code). Assume that you do not have any additional information about project specifics so that you can take the standard (or nominal) project performance as a baseline. Which cost-optimal schedule is most probable for the phases Design (high-level and detailed)—Coding (incl. unit test)—Test? —10 months—14 months—18 months—22 months

Example question related to dependent variable Y.4 ('Understand complex'):

101 Assume you are responsible for a software project of size 1000 tasks. Assume that you don't have any additional information about project specifics so that you can take the standard (or nominal) project performance as a baseline. The standard process implies that 50% of the design and code documents are inspected. Using the standard process, Y person-months is the cost-optimal effort consumption for conducting the phases Design—Coding (incl. unit test)—Test. Due to new customer requirements the quality level of the software has to be 'very high', i.e. 0.2 defects per task (instead of 'nominal', i.e. 1.7 defects per task). Without changing the standard process, which development phase(s) will be intensified most (by adding effort and extending the schedule) in order to achieve the increased reliability level?
- — Design
- — Implementation (coding)
- — Test
- — All phases are intensified equally

Example question related to variable Z.1 ('Time pressure'):

051 I did not have enough time to
- — read the materials on project management (during training session), particularly:
  - — Block 1
  - — Block 2
  - — Block 3
  - — Block 4
- — Familiarize with the tool(s) (during training session)
- — Complete the post-test

Example question related to variable Z.2 ('Session evaluation'):

061a I consider the explanations/information provided in Block 2 (Role Play) in general

	1	2	3	4	5	
Useful	O	O	O	O	O	Useless
Boring	O	O	O	O	O	Absorbing
Difficult	O	O	O	O	O	Easy
Clear	O	O	O	O	O	Confusing

061b I consider the explanations/information provided in Block 1, Block 3, and Block 4 in general

	1	2	3	4	5	
Useful	O	O	O	O	O	Useless
Boring	O	O	O	O	O	Absorbing
Difficult	O	O	O	O	O	Easy
Clear	O	O	O	O	O	Confusing

Appendix B. Meta-analysis procedures

According to the standard procedure, first comparing the p-values was executed via testing the statistic

$$Q = \sum_{i=1}^{K} (Z_i - \bar{Z}),$$

which under H_0 of Homogeneity is χ^2 -distributed with $K - 1$ degrees of freedom. K is the number of studies and Z_i are standard normal deviates corresponding to the one-tailed p-values.

Combining the p-values for the case with very small sample sizes can be performed either with the Fisher procedure, i.e. testing the statistic

$$P = -2\sum_{i=1}^{K} \ln(p_i),$$

which under H_0, corresponding to p_i, is χ^2 -distributed with $2K$ degrees of freedom, or via calculating the p-value out of Stouffer's Z statistic:

$$Z = \frac{1}{\sqrt{K}} \sum_{i=1}^{K} Z_i.$$

Further, comparing the effect sizes was performed through testing the statistic:

$$Q = \sum_{i=1}^{K} \gamma_i^2/\sigma^2(\gamma_i) - \frac{\left(\sum_{i=1}^{K} \gamma_i/\sigma^2(\gamma_i)\right)^2}{\sum_{i=1}^{K} 1/\sigma^2(\gamma_i)},$$

which under H_0 of homogeneity is χ^2-distributed with $K - 1$ degrees of freedom. Here, $\sigma^2(\gamma_i) = (8 + \gamma_i^2)/2N$ is the estimated variance of γ_i and N is the total number of subjects.

Calculating of either sample mean or weighted average makes combining the effect sizes. The weighted average is given by:

$$D = \frac{\sum_{i=1}^{K} \gamma_i/\sigma^2(\gamma_i)}{\sum_{i=1}^{K} 1/\sigma^2(\gamma_i)}.$$

References

[1] T.K. Abdel-Hamid, S.E. Madnick, Software Project Dynamics—An Integrated Approach, Prentice Hall, Englewood Cliffs, NJ, 1991.
[2] B.W. Boehm, Software Engineering Economics, Prentice Hall, Englewood Cliffs, NJ, 1981.
[3] B.W. Boehm, Industrial software metrics top 10 list, IEEE Software (1987) 84–85.
[4] B.W. Boehm, C. Abts, W.A. Brown, S. Chulani, B.K. Clark, E. Horowitz, R. Madachy, D.J. Reifer, B. Steece, Software Cost Estimation with COCOMO II, Prentice Hall, Upper Saddle River, 2000.
[5] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, Academic Press, New York, 1988.
[6] A. Drappa, J. Ludewig, Quantitative modeling for the interactive simulation of software projects, Journal of Systems and Software 46 (1999) 113–122.

[7] J.W. Forrester, Principles of Systems, Productivity Press, Cambridge, 1971.
[8] A.K. Graham, J.D.W. Morecroft, P.M. Senge, J.D. Sterman, Model-supported case studies for management education, European Journal of Operational Research 59 (1992) 151–166.
[9] M.J. Harris, R. Rosenthal, Mediation of interpersonal expectancy effects: 31 meta-analysis, Psychological Bulletin 97 (1985) 363–386.
[10] L. Hedges, Estimation of effect size from a series of independent experiments, Psychological Bulletin 92 (1982) 490–499.
[11] J.E. Hunter, F.L. Schmidt, G.B. Jackson, Meta-Analysis: Cumulating Research Findings Across Studies, Sage, Beverly Hills, CA, 1982.
[12] C.M. Judd, E.R. Smith, L.H. Kidder, Research Methods in Social Relations, International Edition, sixth ed., Harcourt Brace College Publishers, Fort Worth, 1991.
[13] M.I. Kellner, G.A. Hansen, Software process modeling: a case study. Proceedings of the 22nd Annual Hawaii International Conference on System Sciences, 1989.
[14] D.C. Lane, On a resurgence of management simulation games, Journal of the Operational Research Society 46 (1995) 604–625.
[15] R. Likert, A technique for the measurement of attitude, Archives of Psychology 22 (140) (1932).
[16] C.Y. Lin, Walking on battlefields: tools for strategic software management, American Programmer (1993) 33–40.
[17] C.Y. Lin, T. Abdel-Hamid, J.S. Sherif, Software-engineering process simulation model (SEPS), Journal of Systems and Software 38 (1997) 263–277.
[18] M. Lipsey, Design Sensitivity, Sage, Beverly Hills, CA, 1990.
[19] R. Madachy, D. Tarbet, Case studies in software process modeling with system dynamics. Proceedings of the Second Software Process Simulation Modeling Workshop (ProSim'99), Silver Falls, Oregon, 1999.
[20] P. Milling, Managementsimulation im Prozeß des Organisationalen Lernens [Organisational Learning and its Support by Management Simulators]. Zeitschrift für Betriebswirtschaft, Ergänzungsheft (supplement) 3/95: Lernende Unternehmen, 1995, pp. 93–112. (Also available at URL http://iswww.bwl.uni-mannheim.de).
[21] J.D.W. Morecroft, System dynamics and microworlds for policy-makers, European Journal of Operational Research 35 (1988) 301–320.
[22] D. Pfahl, An Integrated Approach to Simulation-Based Learning in Support of Strategic and Project Management in Software Organisations, vol. 8, Experimental Software Engineering, Fraunhofer IRB Verlag, Stuttgart, PhD Thesis, 2001.
[23] D. Pfahl, M. Klemm, G. Ruhe, A CBT module with integrated simulation component for software project management education and training, Journal of Software and Systems 59 (2001) 283–298.
[24] D. Pfahl, N. Koval, G. Ruhe, An experiment for evaluating the effectiveness of using a system dynamics simulation model in software project management education, Proceedings of the Seventh International Software Metrics Symposium, London, United Kingdom, 2001, pp. 97–109.
[25] G.P. Richardson, A.L. Pugh, Introduction to System Dynamics Modeling with DYNAMO, Productivity Press, Cambridge, 1981.
[26] D.J. Sheskin, Handbook of Parametric and Nonparametric Statistical Procedures, CRC Press, Boca Raton, 1997.
[27] B.J. Smith, N. Nguyen, R.F. Vidale, Death of a software manager: how to avoid career suicide through dynamic software process modeling, American Programmer (1993) 10–17.
[28] J.A.M. Vennix, Mental Models and Computer Models—Design and Evaluation of a Computer-based Learning Environment for Policy-making, PhD Thesis, University of Nijmegen, 1990.
[29] A.R. Wildt, O.T. Ahtola, Analysis of Covariance, Sage University Paper Series on Quantitative Applications in the Social Sciences, Series No. 07-012, Sage, Newbury Park, 1978.
[30] F.M. Wolf, Meta Analysis. Quantitative Methods for Research Synthesis, Sage University Paper Series on Quantitative Applications in the Social Sciences, Series No. 07-059, Sage, Newbury Park, 1978.