Effect of Domain Knowledge on Elicitation Effectiveness: An Internally Replicated Controlled Experiment

Alejandrina M. Aranda, Oscar Dieste, *Member, IEEE*, and Natalia Juristo, *Senior Member, IEEE*

Abstract—*Context.* Requirements elicitation is a highly communicative activity in which human interactions play a critical role. A number of analyst characteristics or skills may influence elicitation process effectiveness. *Aim.* Study the influence of analyst problem domain knowledge on elicitation effectiveness. *Method.* We executed a controlled experiment with post-graduate students. The experimental task was to elicit requirements using open interview and consolidate the elicited information immediately afterwards. We used four different problem domains about which students had different levels of knowledge. Two tasks were used in the experiment, whereas the other two were used in an internal replication of the experiment; that is, we repeated the experiment with the same subjects but with different domains. *Results.* Analyst problem domain knowledge has a small but statistically significant effect on the effectiveness of the requirements elicitation activity. The interviewee has a big positive and significant influence, as does general training in requirements activities and interview experience. *Conclusion.* During early contacts with the customer, a key factor is the interviewee; however, training in tasks related to requirements elicitation and knowledge of the problem domain helps requirements analysts to be more effective.

1 Introduction

Requirements elicitation, that is, seeking, capturing and consolidating requirements, is a core activity of any requirements engineering process [1] and has a direct influence on software quality [2]. Requirements elicitation depends on intensive communication between users and analysts in order to gather the right information [3]. Human interactions play an important role in this context. On one hand, customers should be able to interact and communicate their needs to analysts. On the other hand, analysts should be able to draw out and grasp the necessary domain information from customers.

The effectiveness of requirements engineering activities is believed to partially depend on the participating individuals [4]. It has been observed that interview effectiveness can vary significantly depending on interviewer skills, probably because proficiency affects the course of the questioning [5]. As a result, elicitation strongly depends on the individual doing the interviewing [6]. Similar effects have been identified in brain-storming [4] and using other elicitation techniques [7].

Several personal attributes may have a bearing on the effectiveness of any requirements-related task: experience [8], [4], [9], academic education [10], [11], cognitive capabilities [12], etc. One of the aspects that has been proposed to most influence an individual's effectiveness in requirements engineering activities is the knowledge of the problem domain [6], [13], [14], [15].

The software engineering (SE) community tends to take the view that domain knowledge helps professionals to do their job. However, empirical studies examining requirements elicitation have come up with contradictory results: some support the beneficial effects of knowledge [5], others signal possible negative effects [4], [7], [16].

The aim of our research is to experimentally analyse the influence of a requirements analyst's knowledge of the problem domain on the effectiveness of the interview-mediated requirements elicitation activity. Several elicitation techniques have been developed and are in use nowadays. Interviews have traditionally been the most widely but used elicitation technique for requirements acquisition [6]. Despite their importance, scant research has been undertaken on interviews, particularly from an experimental perspective [10], [17].

For research purposes, we divide the elicitation activity into two stages: the elicitation session and the reporting process. The *elicitation session* is the step during which the requirements analyst interacts and talks with customers to gather information about their needs. *Reporting* (referred to in this research as *consolidation* of the elicited information) is the step during which the requirements analyst works to understand and document the information acquired during the elicitation session.

We have conducted two experiments (the second is an internal replication of the first one) whose cause construct is subject domain knowledge. The experimental subjects were MS in Software Engineering students at Madrid Technical

- *The authors are with the Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid, Campus de Montegancedo 28660, Boadilla del Monte, Spain.*
E-mail: am.aranda@alumnos.upm.es, {odieste, natalia}@fi.upm.es.

University's School of Informatics Engineering. The experiment and the internal replication were conducted as assessable exercises of a Requirements Engineering course. The students played the role of analysts and elicited the requirements for two problems (one of which is known and the other unknown to them). The elicitation sessions were conducted using the open interview technique with a time limit of 30 minutes for each problem. The results of the elicitation activity were the recorded interview and report containing the consolidated information of each subject. At the end of the experiment, the students were interviewed individually and completed a post-experimental questionnaire in order to collect information on their personal characteristics: requirements experience, interview experience, qualifications, familiarity with and perceived complexity of the problem domains, etc. In order to check the experiment results and increase their statistical power and external validity, we replicated the experiment with the same subjects and another two problems (again one which was known to subjects and another of which they were ignorant).

Separately neither the experiment nor the internal replication detected a significant influence of domain knowledge on analyst effectiveness. However, the synthesis of the two experiments did manage to identify a small, but significant, effect of knowledge, that is, subjects were slightly more effective in domains which were known to them than in domains of which they were ignorant. Contrariwise, the interviewee and subject's interview experience had a comparable or greater effect than knowledge. Requirements training also has a considerable impact on analyst effectiveness.

The paper is structured as follows. Section 2 describes the background and work related to this research. Section 3 describes the experimental process. Section 4 relates the execution of the experiment. Section 5 reports the results of the experiment, whereas Section 6 gives an account of the execution and results of the internal replication. Section 7 compares the results of the experiment and the replication. Section 8 discusses the results. Section 9 describes the validity threats. Finally, Section 10 outlines the conclusions.

2 Background

It is usually assumed that problem domain knowledge makes requirements analysts more effective [6], [13], [14], [15]. This improvement in effectiveness is believed to be driven by several factors, primarily by better analyst-customer communication. Vitharana et al. [18] draw attention to the fact that analysts with domain knowledge do not have to ask elementary questions, which, apart from taking up elicitation session time, damage the credibility of the analyst in the eyes of customers. Hadar et al. [5] express very similar claims, arguing that analysts with domain knowledge can formulate focused questions in order to capture problem-specific information. This information cannot be elicited using more general questions. Additionally, both papers state that domain knowledge enables analysts to identify and rapidly solve misunderstandings and conflicts.

Some authors have suggested that domain knowledge could have negative effects. In a survey conducted by McAllister [19], customers and users state that analysts with domain knowledge make assumptions about requirements depending on their knowledge and past experience. McAllister conjectures that this may be due to the fact that analysts assume that they know the needs of the customers and, consequently, specify the system that they visualize rather than the system that users really want. In a similar vein, Niknafs and Berry [16], [20] state that although profound domain knowledge eases the understanding of details of the problem, it may also encourage analysts to make assumptions about the requirements. Pitts and Browne [12] are of the same opinion. Browne and Rogich [21] go even further and mention that previous domain knowledge can cause the analyst to overlook implicit requirements.

However, almost all the above claims are based on theoretical standpoints. There are very few empirical studies about the effect of knowledge in the field of requirements engineering. Table 1 summarizes the major empirical studies addressing the influence of knowledge on tasks related to requirements elicitation.

Niknafs and Berry [4], [20] conducted a controlled experiment aimed at studying the impact of domain knowledge and industrial and requirements engineering experience on elicitation effectiveness. The elicitation technique in this case was brainstorming. Twelve subjects with diverse experience and knowledge participated in the experiment; they were divided into four teams. All subjects were 4th-year degree students enrolled in the "Software Requirements and Specification" course at the University of Waterloo. The results showed that domain-ignorant analysts, when added to a team, increased the number of ideas generated by that team (i.e., improved team effectiveness).

Niknafs and Berry [16], [20] conducted another controlled experiment with industry professionals. Eight subjects participated in the study; four of the subjects were employed as developers at a company whose name has been omitted (company C), whereas the other four were from the University of Waterloo. Of the latter four, two were computer science PhD students, while the other two were higher education professionals. The elicitation technique used in the session was brainstorming within a single group. Domain knowledge was operationalized using a problem that was known to the people employed by company C and unknown to the University of Waterloo participants. The observed results are similar between the subjects with and without domain knowledge. However, the subjects that did not have domain knowledge are better at generating new ideas, whereas the ideas generated by the domain-aware people are limited by their acquaintance with the products created in their particular domain.

Niknafs [20] reports two controlled experiments (E1 and E2) with computer science and software engineering students. E1 is the experiment published in [4], and E2 is an internal and exact replication (same procedure, same problem domains, same evaluation process, etc.) of E1. The goal of E2 is to increase the sample size used in E1, as well as to improve the balance between the mix of familiarities (i.e., number of DI and DA subjects that belong to each team). E2

TABLE 1
Empirical Studies of the Effect of Knowledge on Requirements Engineering

Paper	Empirical study type	Subjects	Factor[1]/ Levels	Operationalization of the levels	Elicitation technique	Results
Niknafs and Berry [4], [20]	Controlled Experiment	19 groups of three computer science students each	Domain knowledge: • Domain ignorant (DI) • Domain aware (DA)	Domains were selected based on subject knowledge using a Likert-scale questionnaire.	Brainstorming	A team's effectiveness at generating requirements ideas is affected by the team's mix of domain familiarities. When domain-ignorant analysts joined a team, the number of ideas generated by that team increased (i.e., improved team effectiveness).
Niknafs and Berry [16], [20]	Controlled Experiment	4 domain-aware developers employed by company C and 4 domain-ignorant computer science and higher education professionals	Domain knowledge: • Domain ignorant • Domain aware (DA)	Domains were selected based on subject knowledge using a Likert-scale questionnaire.	Brainstorming	The results show very similar numbers of ideas generated by DA analysts and the DI analysts. However, DI analysts were better at generating new ideas. DAs seemed stuck in the rut of their domain box.
Niknafs [20]	Controlled Experiment	40 groups of 3 people, either computer science and software engineering students, or participants with other bakcgrouds.	Domain knowledge: • Domain ignorant • Domain aware (DA)	Domains were selected based on subject knowledge using a Likert-scale questionnaire.	Brainstorming	The results show that teams with at least one DI were more effective than teams with no DIs.
Hadar et al. [5]	Exploratory empirical study	58 (31 + 27) final-year SE undergraduate students	Domain knowledge: • High-level DK • Low-level DK	The level of domain knowledge for each subject was gathered using a questionnaire.	Interviews	Subjects with domain knowledge formulate more specific questions during elicitation, from which they can gain a better understanding of the domain.
Kristensson et al. [7]	Quasi-experiment	47 subjects including professionals and students: 12 professional developers; 16 computer science students and 19 (non-computer science) students of business administration	Type of user: • Professional Developer • Advanced user • Ordinary user	Previous domain experience and demographic information, as well as personality traits, were gathered using a questionnaire.	Reactive data-gathering technique	Overall, the experiment produced three main results: Ordinary users produced more original new service ideas, indicating a more divergent style of thinking. Ordinary users produced ideas that were assessed as significantly more valuable. Professional developers and advanced users produced the most realizable ideas

1 For quasi-experiments we specify the independent variable instead of the factor.

is analysed in [20] together with E1, as reported below. The results do not reveal that the mix of familiarities has any significant effect on any of the tested response variables (number of raw, relevant, feasible and innovative ideas).

Hadar et al. [5] conducted an exploratory empirical study aimed at evaluating the effect of analyst domain knowledge when performing elicitation using interviews. The study was conducted with final-year SE undergraduate students. The study was composed of two rounds with different groups of participants. Twenty-seven and 31 subjects participated in the first and the second rounds, respectively. Two problem domains were selected for each study round. Subjects were assigned to known or unknown domains according to their knowledge of the experimental problems by means of questionnaires and interviews held by the researchers. Researchers compared the relationship between high/low domain knowledge and the type of questions posed by the analysts. Results showed that participants who lacked domain knowledge mostly phrased general questions, whereas subjects who had domain knowledge were able to put questions more specifically. This could result in their gaining a deeper understanding and eliciting more details about the problem domain.

Kristensson et al. [7] conducted a quasi-experiment in order to study the creativity of user ideas for solving a problem in the mobile technology domain using three types of subjects: 1) advanced users who were computer science students, 2) ordinary users who were business administration or social science students, and 3) professional service developers. Forty-seven subjects participated in the quasi-experiment, of which 12 were professional developers, 16 advanced users and 19 ordinary users. The results indicated that ordinary users create significantly more original and valuable ideas than professional developers and advanced users. On the other hand, professional developers and advanced users created more easily realizable ideas than ordinary users. Kristensson et al. argue that advanced users generate less original ideas than the ordinary user group, possibly due to the restrictive effects of their greater prior knowledge of the domain (mobile phone systems).

With the sole exception of [5], the results of the above studies suggest that previous domain knowledge has more harmful than helpful effects. However, it is at the very least surprising that such simple studies yield results that are so diametrically opposed to the dictates of common sense. Note that most of the empirical studies conducted compare two groups (domain-aware analysts and domain-ignorant analysts) and test rather small sample sizes (from 8 to 31 subjects). Under these conditions, it is possible to repeatedly observe that previous knowledge about the problem domain has negative effects provided that effects are

consistent (i.e., the trend is always the same) and the effect size is large. If this were true, the anecdotal observations which form the groundwork of the theoretical knowledge commonly accepted in RE (i.e., domain knowledge favours the elicitation process) should have revealed that domain knowledge is harmful. In other words, if knowledge is found to have a negative bearing on elicitation in small experiments, this trend should have been apparent under non-experimental conditions. In order to solve this apparent contradiction between theoretical knowledge (based on multiple anecdotal observations) and empirical knowledge (based on a few specially designed investigations), this paper addresses the following research question:

> ***RQ**: Does analyst domain knowledge influence (either positively or negatively) the effectiveness of the requirements elicitation activity?*

We have performed two controlled experiments to answer this research question. The elicitation technique used was an unstructured interview. Note that out of the four related papers, only Hadar et al. [5] uses interviews (the other three use elicitation techniques that are less often used in practice).

In order to operationalize the knowledge construct, we selected two domains, one that was known to the experimental subjects and another of which they were ignorant, in a similar fashion to Niknafs [4], [16], [20]. This contrasts with Hadar et al. [5] and Kristensson et al. [7] experimental studies, where the domain knowledge is operationalized using a questionnaire (i.e., participants' personal opinion about how much they know). We considered that operationalizing the domain knowledge construct by asking subjects what they know about the domain is a rather subjective approach and may pose validity threats to the results [22], [23].

Therefore, our paper experimentally studies the influence of domain knowledge on an elicitation technique that has not been much researched but is very often used (the interview) and operationalizes knowledge objectively instead of subjectively.

3 Research Method

In this section we report the experimental design. The design is reported according to the guidelines for reporting software engineering experiments proposed by Jedlischka and Pfahl [24].

In advance of this experiment, we have run four quasi-experiments (2007, 2009 and two in 2011 which have not yet been published). They are all very similar to the experiment reported here and served as pilot studies for most of the design elements (variables, measurement procedure, validity threats, etc.). The innovations of this experiment were the inclusion of an additional problem domain (domain-aware), as well as the use of a repeated measures design

O. Dieste (Madrid Technical University) and J.W. Castro (Madrid Autonomous University) acted as interviewees A and B (see Section 3.1.3), and A. M. Aranda (Madrid Technical University) acted as invigilator and data collector. A. M. Aranda, O. Dieste and N. Juristo planned and design the experiment. A. M. Aranda was also responsible for measurement. The analysis was conducted jointly by O. Dieste and A. M. Aranda with support from S. Vegas (Madrid Technical University). Note that neither J.W. Castro nor S. Vegas are related to the research (in fact, they are not listed as authors of this article). This helps to avoid experimenter expectancies.

3.1 Variables

3.1.1 Dependent Variable

The dependent variable is the **effectiveness of the elicitation process.** There is not as yet any widely accepted metric for measuring requirements elicitation effectiveness. However, the theoretical and empirical literature reports several alternative measurements: Agarwal and Tanniru [9] measure effectiveness in terms of the total number of identified rules; Pitts and Browne [12] use the number of acquired requirements; Burton et al. [25] take into account the number of elicited rules and clauses; Browne and Rogich [21] make a distinction by categories: total number of requirements, processes and information; Niknafs and Berry [4], [20] account for effectiveness according to the numbers of raw, relevant, feasible, and innovative requirement ideas generated by the RE team.Therefore, effectiveness has been operationalized as the number of items (be they concepts, rules, processes, etc.) acquired by the analyst during elicitation. We have adopted a similar procedure and measure the **effectiveness** of the requirements elicitation process as the *total percentage of problem domain elements identified by experimental* subjects. As specified in Section 3.11, a benchmark list (or gold standard) is required to relationships between actions, functions or states. More modern formulations, like the FRISCO Report [28], add items that were implicit in the earlier literature, such as actors or rules, to the above concepts, actions or states.

We use two main types of elements in order to measure elicitation effectiveness: concepts and processes. Although we originally considered goals, they were later omitted because they were far outnumbered by processes or concepts and they are always identified by all subjects. Consequently, they do not provide relevant information about the effects of domain knowledge on effectiveness. As we have selected simple domains (with few actors and no business rule), these more sophisticated aspects are, like goals, not helpful for measuring effectiveness.

Domain processes and concepts can be described at several detail levels (inputs, outputs, attributes, relationships, etc.). Considering that the experiment involves performing an early interview within a requirements process, the analyst is unlikely to apprehend the details of the domain. On this ground, we have not taken into account the above details (e.g., relationships) and concentrated on coarse-grained elements which provide an incomplete, though probably fairer, picture of the effectiveness of analysts in a preliminary elicitation session addressing a simple problem.

In order to simplify the instrumentation, we have omitted the non-functional requirements from the study and focused exclusively on functional requirements. From the viewpoint of information systems, requirements are primarily concerned with the automation of pre-existing domain processes. Processes and requirements can be regarded as redundant for the purpose of measuring how much

information subjects gather. However, these particularities of information systems do not apply to other system types, like, for example, control systems. Jackson's characterization [29], which considers requirements (as opposed to specifications) as optative statements in a domain, is more appropriate in such settings. From this viewpoint, requirements are a part of the problem domain, that is, separate from processes and concepts, and should be included as such. This is a more plausible option for this experiment, as none of the problem domains used in the experiments could be qualified as an information system.

In short, this research considers the following problem domain elements: *concepts, processes and requirements.* The make this meaturement.

The problem domain is composed of different types of elements. There is agreement within the literature on the key elements, but differences on the fine points. According to Yadav et al. [26], for example, the problems are usually analysed in terms of organizational goals, business processes and tasks to be performed within the processes to achieve the goals. In a similar vein, Davis [27] mentions that, irrespective of the language, notation or technique used, requirements: 1) define an object, a function or a state; 2) limit or control the actions associated with an object, a function or a state; or 3) define measure of effectiveness was calculated according to the following formula:

$$\text{Effectiveness} = \frac{\frac{\#\text{identified}}{\text{concepts}} + \frac{\#\text{identified}}{\text{processes}} + \frac{\#\text{identified}}{\text{requierement}}}{\text{total number of elements}}.$$

We have opted not to calculate a weighted mean, as the fact that there is a greater proportion of a particular element in a domain does not mean that this is intrinsically more important. All the elements play an important role in the construction of the future software system.

3.1.2. Factors and Levels

The *factor* under study is analyst ***Knowledge*** of the problem domain. It has two levels: *aware* and *ignorant,* as defined by Niknafs and Berry [4], [20]. We define an analyst as domain ignorant (DI) if he has not had previous contact with the domain; otherwise the analyst is domain-aware (DA).

Domain knowledge is a characteristic of the analyst with which, in principle, researchers cannot tamper (i.e., they cannot allocate levels to subjects). The strategy that we have applied is to select two problem domains, knowledge of which on the part of the experimental subjects can be predicted. This is a different strategy to the one applied by Niknafs and Berry [4]. As shown in Section 3.4, we have used a repeated-measures design where all subjects are domain ignorant in one case and domain aware in the other. In the case of Niknafs and Berry [4], [20], there is only one domain for which there are domain-ignorant and domain-aware subjects.

This was feasible as we had thorough knowledge of the experimental population (postgraduate students of a subject that we have been teaching for over 10 years) and can hence accurately determine problem domains of which graduate students are aware or ignorant.

Using students rather than professionals has the added advantage that domain knowledge and experience will be fairly disassociated, as students are not generally very experienced. On this ground, the observed effects in the experiment will be able to be definitely attributed to the knowledge factor. All the assumptions on analyst domain knowledge made during the design were checked a posteriori by means of a questionnaire.

3.1.3. Blocking Variable

Variables that are likely to have an effect on the dependent variable but are not being investigated by the experiment are termed blocking variables [30]. This experiment has been blocked based on interviewee.

In order to execute an experiment with similar characteristics to requirements elicitation by means of open interview, it is necessary to have one interviewee per interview. The characteristics of the interviewee are more than likely to influence analyst effectiveness (e.g., an analyst will gather more information from a more forthcoming than a reserved interviewee). Ideally the same interviewee (which would be an experiment parameter) should participate in all interviews. But this is unworkable, as the interviewee would have to assume a workload (around 30 interviews about two different problem domains) that could bias the result of the experiments due to the effects of fatigue or even learning on the part of the interviewee. Additionally, the external validity of the experimental results would be rather low due to dependency on only one interviewee.

Out of convenience (resources, number of hours and available classes, number of students), the number of interviewees was set at two. Therefore, *the subjects were blocked by interviewee.* Although three interviewees would have been better, two appeared to be sufficient to counteract the fatigue and learning effects:

- *Fatigue effect*: Experience from earlier studies conducted in 2007, 2009 and 2011 suggests that interviewees can complete up to five interviews per session without experiencing fatigue. Conducting from five to seven interviews, lasting no more than a total of four hours, is tolerable. As of this point, interviewees start to find the interviews hard going. According to the experimental design planned below and the number of available subjects, interviewees are not required to go over seven interviews per session.
- *Learning effect*: Interviewees repeating the same interview over and again might not always provide consistent information (e.g., the first interviewers could receive more or less information than the last). Our experience from past studies suggests that information supplied to interviewers varies more between days than on the same day. On the same day, the interviewees tend to give mechanical responses (i.e., repeat the same response to identical questions). This is probably due to the concentration required of interviewees during the elicitation session helping them to recall the answers given not long before which they tend to repeat. The planned experimental design only requires one day for each interview session, which we believe removes (or counteracts) the interviewee learning effects.

The use of two interviewees solves another important practical problem. The requirements elicitation is a phase where communication is first and foremost; therefore subjects using their mother tongue are likely to be more effective than subjects using a second language. In order to prevent language from influencing the effectiveness of both experimental subjects and interviewees, we established that native Spanish speakers should use Spanish during the interview, whereas the other subjects (of different nationalities, including German, Romanian, Serbian, Swedish, Danish and United States) should use English. Note that English is a second language for all of these students, except one. The first group interviewed interviewee B, whereas the second group interviewed interviewee A.

Interviewees A and B are native Spanish speakers. Interviewee A has an adequate knowledge of English. As regards this point, note that English is the working language of the degree programme that all the experimental subjects are taking. Interviewee A teaches this master's programme. Interviewee B does not have a comparable knowledge of English; that reason justify using Spanish during his interviews.

The fact that interviewee and language are confounded does not pose a threat to validity in this case. By blocking the subjects by interviewee, we prevent the two variables (interviewee and language) from interacting (i.e., interviewee B cannot speak good enough English). Such interactions increase the variance of the domain knowledge factor and are an obstacle to the identification of significant effects. Of course, the problems of communication in a second language (the case of interviewee A) may detract from the effectiveness of subjects. However, the between-subjects design assures that each subject does the DA and DI elicitation in the same language, thereby cancelling out the negative effects of communication in English.

3.2 Hypothesis

The main aim of the research is to experimentally analyse the influence of analyst problem domain knowledge on the effectiveness of the requirements elicitation activity. In order to achieve the proposed objective we have stated the following experimental hypothesis:

H_0: The effectiveness of domain-aware and problem-ignorant analysts is the same.

H_1: The effectiveness of domain-aware and problem-ignorant analysts differs.

As this is an exploratory study and different trends regarding the effect of knowledge have been observed in the literature, we could not predict the direction of the potential effects of the problem domain knowledge. Therefore, the alternative hypothesis was two-tailed.

3.3 Subject Selection

We have used convenience sampling to select the experimental subjects. The subjects that participated in the experiment were students of Madrid Technical University's School of Informatics Engineering, enrolled in the MS in Software Engineering Requirements Engineering course. While experimenting with students has been associated with a lack of realism [31] and reduced external validity [32], [33], self-selected students are regarded in several disciplines as **an appropriate subject pool for the study of social behaviour** [34]. Additionally, subject motivation is ensured, as the experimental task is equivalent to a graded exercise. This increases experiment validity.

TABLE 2
Two Level, Within-Subjects Design Including Blocking

Interviewee group	Blocking variables	Time sequence	
	Interviewee	Session 1	Session 2
G1	A (English)	Domain aware	Domain ignorant
G2	B (Spanish)		

The information related to the experimental subjects was gathered using a post-experimental questionnaire (in order to avoid any type of experimenter bias). This questionnaire gathered information about such aspects as were likely to have an impact on how effective subjects are at capturing information during requirements elicitation: academic qualifications, specific knowledge of requirements elicitation techniques, interview or requirements experience, familiarity with problems addressed during the experiments. The questionnaire is shown in Appendix A, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TSE.2015.2494588. Section 4.1 discusses the characteristics of our sample.

3.4 Design

Our research is a controlled experiment composed of a *factor* (subject problem domain knowledge) with *two levels* or *treatments* (domain aware and domain ignorant) blocked by interviewee as shown in Table 2.

We opted to apply a within-subjects design or repeated measures design, primarily on the grounds of statistical power. It takes about 34 experimental units (some 18 subjects) in order to achieve a power of 80 percent with respect to the detection of mean effects (Cohen's $d >= 0.5$)[2] using a repeated measures design with a within-subjects factor. A similarly powerful equivalent between-subjects design requires 102 subjects. Within the context of the degree programme as part of which the experiment was conducted, the average number of students per class is from 15 to 20, so the within-subjects design looks like the best option.

The within-subjects design is powerful because it essentially entails a blocking by matching procedure. Each subject is matched with himself or herself, cancelling out any inherent variability and greatly increasing power [35]. In particular, any influence due to subject experience (i.e., skills and knowledge acquired during their professional career) is eliminated. So any observed effects could definitely be ascribed to the problem domain knowledge.

The design shown in Table 2 is a paired design with two experimental objects [30], which Cook and Campbell generically refer to as a pre-post design [36]. With a pre-post design, the problem domains have to be studied in a particular order (in our case, first the domain-aware problem, followed by the domain-ignorant problem. As a result, the

2. The calculations were made using G*Power 3.1.3 default parameters.

domain and order in which the elicitation sessions are conducted (execution order) are confounded. A crossover design [37] would address this validity threat, albeit at the cost of introducing the risk of domain information being filtered between sessions. There is a very high risk of this occurring in an academic environment, which precludes the use of the crossover design. However, internal replication emulates a crossover design, and this is the option that we went for. Section 6 discusses the executed internal replication in which the order of treatments was inverted.

Repeated measures designs are open to other threats, apart from execution order. These threats and the strategies that we applied to remedy them are discussed in the Section 3.5.

3.5 Evaluation of Validity

The within-subjects design is a controlled experiment that is exposed to a number of validity threats: fatigue, learning and carryover effects. We believe that we have managed to address these effects in our experiment by applying the following strategies.

- *Fatigue effect*: The fact that each subject has to analyse two problems means that two sessions are required to run the experiment. If these sessions are performed in close succession, subjects may experience fatigue resulting in a drop in their effectiveness as the experimental sessions progress. To avoid this pernicious effect, the experimental sessions were held two days apart. Therefore fatigue has not affected the second session, which transpired in similar conditions to the first session.
- *Learning effect*: The source of the learning effect is the performance of the same experimental task by the same experimental subject on repeated occasions. In this experiment, each subject have analysed two completely different domains, and therefore the information was unlikely to be reusable from the domain-aware problem (AP1) to domain-ignorant problem (IP1). On the other hand, the elicitation was performed using the open interview, and subject skills were a priori unlikely to improve substantially after a mere 30-minute conversation (actual elicitation sessions were even shorter) and over the two days between sessions.
- Furthermore, the learning effect can be identified through experiment synthesis and internal replication, and its influence can be counterbalanced (as we have done).
- *Carryover effect*: The residual effect that administering one treatment to a subject has on another treatment administered later to the same subject, where the residual effect increases or decreases the effectiveness of the later treatment, is known as carryover [38]. Carryover is an important risk in medical experiments, as drug residues can remain in the body for quite some time and interact with later treatments [39]. However, in this experiment, the experimental subjects always used the same technique (open interview), which rules out the possibility of there being carryover, as any effects would be due to learning as discussed above.

TABLE 3
Problem Domains Used in the Experiment

Problem	Brief description	Levels
AP1	A text messaging application for mobile devices	Domain Aware
IP1	A battery recycling machine control system	Domain Ignorant

Apart from the threats to validity posed by the type of experiment (within-subjects) used, there are other more subtle threats to the validity caused by design decisions:

- *Exclusion of non-functional requirements*: Omitting non-functional requirements from the research may detract from the effectiveness of subjects who tend to focus on such issues. In order to minimize this threat, interviewees intentionally led the conversation when subjects asked about this type of requirements. The conversation was led by truthfully answering that the respective non-functional issue was not relevant.
- *Management-related issues*: As for non-functional issues, the effectiveness of subjects paying special attention to management issues (e.g., deadlines, costs) could be compromised. To minimize this threat, we have applied the same strategy as for non-functional requirements.

3.6 Assignment of Treatments to Subjects

As Table 2 shows, we have divided the experimental subjects into two equal-sized groups (G1, G2) by language. English-speaking students were assigned to G1, whereas Spanish-speaking students were assigned to G2. These rules out language as a factor having an influence on the effectiveness of the elicitation process for both interviewers and interviewees. English-speaking students were assigned to interviewee A, and Spanish-speaking students to interviewee B.

Notice that this experiment does not require the random allocation of subjects to groups, as every subject performs both treatments and the crossover design has been excluded.

3.7 Experimental Objects

Table 3 illustrates the problem domains used in this experiment. The problem AP1 deals with a mobile application for sending and receiving text messages. AP1 is the instantiation of the domain-aware treatment in this experiment, as mobile devices and instant messaging are often used by students and, generally, by a broad sector of the population. On the other hand, the domain-ignorant problem (IP1) is an uncommon system with which students are unfamiliar. IP1 was selected as an instance of the domain-ignorant treatment. The IP1 problem is related to a battery recycling plant, where a series of very domain-specific machines and processes are used, which students would be unable to infer unless they have specific knowledge of that domain. The IP1 problem is based on a real system, which has been simplified so that it can be addressed by master's students within the time constraints of experimental sessions. Briefly, the key features of the problems are:

TABLE 4
Total Number of Elements That Define the Problems

Problem	Elements that define the problem (#)			
	Requirements	Concepts	Processes	Total
AP1	28	10	16	54
IP1	15	24	12	51

- *Messaging system (AP1)*: an instant messaging system enabling telecommunications operator users to perform basic operations, like plain-text message exchange, user interconnection, chat rooms, etc.
- *Battery recycling control system (IP1)*: an automatic real-time system to control a battery recycling process, from battery sorting to distillation, in order to separate out the poisonous heavy metals contained in the batteries.

Problems AP1 and IP1 have been described in detail, including, apart from requirements, concept and activity models, as shown in Appendix B, available in the online supplemental material. These descriptions are useful for training interviewees in order to provide the right answers to interviewers, and also serve as benchmark for measuring the effectiveness of the experimental subjects.

An important noteworthy aspect is that we tried to assure that the total number of elements used to define the different problems was as close as possible. Otherwise, problem size would be another aspect potentially influencing the results because we would have an undesired Knowledge x Size interaction. Table 4 shows the total number of elements that define and delimit the size of each problem domain.

Although the total number of elements is almost the same in both problems, there are sizable differences when elements are classified as *Requirements* and *Concepts*. The source of these differences (28 versus 15 requirements and 10 versus 24 concepts) is the type of problem domain. Any manipulation (i.e., including or excluding requirements or concepts) could render the problems contrived and/or illogical. Therefore, we decided not to alter the number of requirements and concepts in the original problem statement. However, we are aware that such differences may moderate the effects of knowledge. In order to account for such moderator effects, we will explicitly study the influence of element types (processes, concepts and requirements) on analyst effectiveness during the analysis phase.

Finally, another aspect that we examined was problem complexity. Looking at the activity diagrams (see Appendix B, available in the online supplemental material), we find that problem AP1 *may be* harder to understand than IP1 (e.g., it has a few more processes, the task flow contains cycles, etc.). On top of that, such *complexity effects* are potentially subject dependent. However, it is far from clear how to evaluate the influence of task flow or topology. In this case, we took a conservative approach and introduced an item in the post-experimental questionnaire inquiring about problem complexity as perceived by the subject. Additionally, we counterbalanced the *Complexity* of the experimental objects in the internal replications (see Section 6.1), so any *Knowledge x Complexity* interaction will be cancelled out in the joint analysis (see Section 7).

3.8 Experimental Operation

The experiment was composed of three tasks: elicitation session, report on the gathered information and completion of the post-experimental questionnaire. In the elicitation session, the subjects played the role of requirements analysts (interviewers) and the experimenter acted as customer or user (interviewee). Each subject was competent to perform the experimental task as they were computer science graduate with the technical process knowledge required to elicit requirements.

During the elicitation session, each subject had to identify the key problem domain information. The elicitation session was carried out, as is common practice in the early stages of the requirements process, using the open interview technique (i.e., a conversation with open-ended questions) subject to a 30-minute time limit, with from five to 10 minutes of extra time for subjects to complete the interview as normal. At the end of the elicitation session, the experimental subjects were given 90 minutes to write up all the information that they have acquired during the interview in a report. We believe that the allotted times are realistic (considering domain complexity) and are, in any case, sufficient to acquire and report a sizeable number of the problem domain elements.

Requirements are usually reported using templates. In previous studies, we used both the IEEE 830 template and free-form reporting. Experience has shown that subjects prefer to use free-form reporting when the empirical studies precede training and the template when the empirical study follows upon training (for example, in the first experiment reported in this paper, 10 out of 28 subjects used the template, whereas the template was used by 16 out of 25 subjects in the replication). On this ground, we decided to let subjects use their preferred reporting format rather than asking them to use a particular template which could be source of bias. However, the use of free-form reporting does not necessarily mean that the subjects expressed the domain information narratively. On the contrary, most reports contained a list of items, often divided into sections (e.g., functional requirements, non-functional requirements, etc.). Representative examples of the reports submitted by subjects are shown in Appendix C, available in the online supplemental material.

The subjects finish the experiment by filling in the post-experimental questionnaire, which takes less than five minutes to complete.

3.9 Instrumentation

The instrumentation of the experiment is relatively simple. The experimenter who is the interviewee has been trained in the domains. The analysts did not require specialized training. Finally, we created a post-experimental questionnaire.

As regards interviewee training, the requirements elicitation activity is conducted in an academic environment and was therefore a simulated process. Consequently, the interviewee was not really versed in the problem domain. In order to provide subjects with information, interviewees had to study the problems thoroughly so that they can easily answer the questions posed by experimental subjects as

DAY 1			DAY 2 AND DAY 3	DAY 4		
Office 1	Office 2	Classroom 1		Office 1	Office 2	Classroom 1
Interviewee A / G1	Interviewee B / G2	Invigilator		Interviewee A / G1	Interviewee B / G2	Invigilator
Domain-Aware Elicitation Session		Report	Rest days	Domain-Ignorant Elicitation Session		Report + Post-experimental Questionnaire

Fig. 1. The experimental schedule (planned).

fully and correctly as possible without withholding information or proffering more details than they are asked for. This was possible because problems AP1 and IP1 were clearly and exhaustively described as specified in Section 3.7. The interviewees agreed on how to address any questions not related to the problem domain (which, therefore, were not part of the gold standard) posed by analysts. Such questions typically referred to management issues and non-functional requirements.

As regards the questionnaire, this was implemented in the course-learning environment (Moodle) to which students have access.

3.10 Data Collection Protocol and Procedure

Fig. 1 shows the schedule of the experiment. The week before the execution of the experiment, students were notified by e-mail of the date, time and place and instructions necessary in order to execute the elicitation. In the interests of motivation, the experiment was designed as an assessable practical assignment. As the experiment is presented as assessable practical assignment rather than an experiment, we could avoid validity threats related to the reactive effects of experimental arrangements [36].

According to the experimental design shown in Fig. 1 the subjects performed the requirements elicitation process on two different, non-consecutive days. The elicitation session for domain-aware analysts was held on the first day and the elicitation session for domain-ignorant analysts on the fourth day. On the second and third day, the subjects performed no experiment-related activities in order to avoid fatigue or boredom effects.

Each experimental subject performed the entire experimental schedule. First, they visited the assigned offices (Office 1 or Office 2) to elicit requirements. At the end of the elicitation session, students visit Classroom 1 and created a report about the information acquired in the elicitation session, under the supervision of a researcher (invigilator) who answered any questions that students could have about the exercise. At the end of the reporting process, students handed in a handwritten or digital copy of the report to the invigilator. At the end of the second session of the experiment (Day 4), the subjects answered the post-experimental questionnaire.

3.11 Measurement Procedure

We have used the elements defining the domain (requirements, concepts and processes) as a benchmark list (or gold standard) in order to measure the effectiveness of the requirements elicitation process. Table 5, for example, is an excerpt from this list. The *effectiveness of the elicitation process* was measured based on the reports submitted by subjects at the end of the consolidation process.

The benchmark list was defined beforehand, and no new element was added or removed during the measurement of the subject reports.

Note that the measurement process was not a literal (blind) comparison against the benchmark list. Since there was no specified format for the report submitted by subjects on requirements elicitation, it is performed by carefully reading through the every report.

Any element on the benchmark list appearing in the report was counted just once; repeated occurrences were ignored. In this manner, we were able to generate summary tables like Table 6, which records, for each experimental subject (Ei), the total number and percentage of acquired elements (rows) and the average percentage of subjects that acquired each element (columns). For example, in Table 6, we find that the subject E01 has acquired 50 percent of the concepts defined for the problem.

The measurement was made by one of the researchers (A. Aranda, see Section 4.3). While it is good practice for more than one person to make the measurement, it was not considered necessary in this case due to the simplicity of the domains and the ease of identification of the elements defining the domain. As example, some of the subject reports including the measurement codes are available in Appendix C, available in the online supplemental material.

3.12 Data Analysis

We calculated descriptive statistics and produced box plots and profile graphs in order to check the trends observed in the analyses.

TABLE 5
Excerpt of the Benchmark for the Battery Problem

ELEMENT TYPE	ELEMENT	DESCRIPTION
Requirements	R1	The system will enable manual sorter start-up
	R12	The system will provide an option for entering the recycling batch of the batteries listed on each delivery note.
	...	
Concepts	C1	Metal
	C8	Batteries
	C19	Machines
	...	...
Processes	A1	Enter delivery note
	A2	Sort batteries
	...	...

TABLE 6
Acquired Concepts for Messaging Problem (AP1)

Interview	Concepts										Total (10)	Total (%)
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10		
E01	x		x	x	x					x	5	50%
E02	x	x	x		x	x	x			x	7	70%
E03	x	x	x	x	x	x	x	x	x	x	10	100%
E04			x	x	x						3	30%
E05	x	x	x	x	x					x	6	60%
E06	x	x	x	x	x	x	x	x		x	9	90%
E07	x	x	x	x	x	x	x	x		x	9	90%
E08	x	x	x	x						x	5	50%
E09	x	x	x	x		x		x		x	7	70%
E10	x	x	x	x	x						5	50%
E11	x		x	x	x					x	5	50%
E12	x	x	x	x		x	x	x		x	8	80%
E13		x	x	x	x						5	50%
E14			x	x							2	20%
Averages	86%	71%	100%	93%	71%	43%	36%	36%	7%	71%	5	61%

The repeated measures general linear model (GLM) (also known as repeated measures ANOVA) is the best statistical method for analysing the data of this experiment. If there were no blocking variable different statistical tests, such as the paired-sample t-test or Wilcoxon matched-pairs test could be used, depending on data normality. For the GLM to be reliable, two conditions need to be met: sphericity and normality of residuals.

- *Sphericity*. Sphericity checks that the covariances between each pair of treatments are equal. Mauchly's test is usually employed to test the sphericity condition. In this case, however, there are only two levels of repeated measures, which precludes a sphericity violation [40] and, therefore, the test is unnecessary.
- *Normality of residuals*. The matching procedure should have eliminated the influence of any source of variation on the treatments. This means that the distribution of the residuals must have a zero mean and a random but constant variance. There are assumed to be many independent sources of variation, and therefore the distribution of the residuals should be normal. The normality of the residuals can be tested using the Kolmogorov-Smirnov (KS) and Shapiro-Wilk (SW).

The statistical significance of the results was defined at a level of $\alpha = 0.05$. We used the SPSS V. 21 statistical tool to analyse the results.

4 Experiment Execution

4.1 Sample

Fourteen Madrid Technical University School of Informatics Engineering Master in Software Engineering students participated in the experiment. They were professionals holding a first degree in computing and related areas from different Latin America and European countries. Table 7 summarizes the key characteristics of the sample (note that not all students answered all the questions, so the number of subjects does not total 14). Most subjects had basic knowledge of requirements engineering (e.g., requirements documentation) and a few elicitation-related activities (e.g., elicitation and modelling techniques). The experimental population did not have much experience in tasks related to requirements elicitation. Most students reported less than two years' experience. Only two students reported from three-to-five years' experience. Note again, however, that this study is not concerned with experience whose influence is cancelled out by the within-subjects design.

With regard to familiarity, 11 out of the 14 subjects were familiar with the domain-aware problem and one with the domain-ignorant problem. Our design and the familiarity reported by subjects were generally consistent. To prevent

TABLE 7
Key Characteristics of Experimental Subjects

Experiment (14 subjects)		
Characteristics	Level	#Subjects
Degree	Computer engineering or computer science	11
	Electronic engineering and computer science	1
	Electrical engineering and computer science	1
Knowledge of elicitation-related tasks	Data modelling	3
	Process modelling	3
	Use cases/user stories	10
	Requirements writing	6
	Elicitation techniques	1
Interview experience	0 years	7
	1-2 years	4
	3-4 years	2
	$>= 5$ years	0
Requirements experience	0 years	6
	1-2 years	5
	3-4 years	1
	$>= 5$ years	1
Problem familiarity	AP1 - Familiar	11
	IP1 - Familiar	1

experimenters from influencing results, we opted to implement an intention to treat (ITT) policy [41], whereby the original design was analysed without taking into account the opinion of the participants concerning their familiarity with the problem. The possible effects of familiarity will be studied later and reported in the discussion section.

4.2 Preparation

The first experiment session was executed on the Tuesday and the second session on the Friday of the same week in order to avoid possible fatigue on the part of experimental subjects and interviewees.

One of our goals was to prevent the training given as part of the course biasing the experiment results. On this ground, the experiment was conducted at the start of the course before the subjects received any training in requirements engineering. The experimental subjects relied on the experience and education that they had before taking the master's programme.

4.3 Execution

The experiment was conducted as scheduled. The first session was held on 11 September 2012, whereas the second session was held on 14 September. Each elicitation sessions lasted at most 30 minutes, and none of the subjects in this experiment exceeded the established elicitation time limit. On average, the elicitation sessions lasted 26 minutes, and the minimum and maximum durations were 15 and 30 minutes, respectively. On the other hand, with few exceptions, the subjects used up all the allotted reporting time. At the end of the elicitation session, the subjects switched over to the classroom supervised by the person playing the role of invigilator and started to report all the key information acquired during the interviews. The reports generated by each students was handed into the data collector.

The experimental task was rather tough for the experimenters, as it was quite time consuming and toilsome. Note that for each problem (AP1, IP1), each interviewee (A, B) was questioned by seven experimental subjects for 30 minutes each. For each problem domain, they put in about four hours, including a five-minute break per session. Considering both problems together, the researchers put a total of 8 hours into the elicitation sessions.

As the subjects were given a maximum time limit of 90 minutes to complete the report, taken together every subject put about 4 hours 30 minutes into this activity per problem. Taking both problems together, each subjects put in a total of about 9 hours.

In order to collect the experimental data we spent about two hours per subject on analysing each report. We analysed a total of 28 reports, putting in a total of approximately 56 hours. The data related to population characteristics (e.g., experience) were gathered automatically.

The raw data are available in Appendix D, available in the online supplemental material.

4.4 Deviations

The experiment was carried out according to the planned schedule. Therefore, there were no deviations during experiment execution.

TABLE 8
Descriptive Statistics for Effectiveness in Problems AP1 and IP1

Effectiveness (%)				
	Knowledge of the Problem Domain			
	Aware (AP1)		Ignorant (IP1)	
	A	B	A	B
N	7	7	7	7
Mean	28.04	47.88	24.93	50.98
Maximum	43	69	39	73
Minimum	13	20	14	24
Median	24.07	53.70	23.53	52.94
Variance	105.657	417.728	72.683	246.059
Std. Dev.	10.279	20.438	8.525	15.686

5 Results

5.1 Dataset Reduction

The experimental data set did not have to be reduced as there were no dropouts (all the subjects completed the entire exercise), and we found no outliers that needed to be excluded.

5.2 Descriptive Statistics and Plots

Table 8 lists the total number of subjects who interviewed each interviewee, the mean, maximum and minimum effectiveness achieved by subjects, as well as the median, variance and standard deviation. The domain-aware and domain-ignorant subjects that interviewed interviewee B tend on average to consolidate more elements defining the problem domain than the subjects that interviewed interviewee A.

When comparing domain-dependent effectiveness, the subjects who interviewed interviewee A tend to acquire more information on the domain-aware problem (28 percent) than about the domain-ignorant problem (25 percent). On the other hand, the subjects who interviewed interviewee B acquired on average less information on the domain-aware problem (48 percent) than on the domain-ignorant problem (51 percent).

These differences confirm the foreseeable influence of the interviewee on analyst effectiveness and that we were right to block by interviewee. However, the absolute effectiveness values, that is, the differences in observed effectiveness between the two domains (aware, ignorant) for each interviewee, are minimal at around 3 percent.

The mean effectiveness achieved by the experimental subjects in each problem domain is shown in the profile plot illustrated in Fig. 2. The profile plot represents the problem domains used in the experiment on the x-axis and the mean effectiveness of subjects on the y-axis. Note that effectiveness is separated by interviewee. The profile plot confirms that ***Knowledge* has hardly any effect**, as, irrespective of the ± 3 percent between-problem difference, the lines are more or less horizontal.

The box plots illustrated in Fig. 3 show that the spread is larger for the subjects that interviewed interviewee B, whereas the effectiveness of subjects who interviewed interviewee A is more homogeneous. Note that variability for interviewee B is greater for the domain-aware problem. Again, this difference of spread reflects the effects of the interviewee on analyst effectiveness, which can be isolated thanks to the blocking variable.

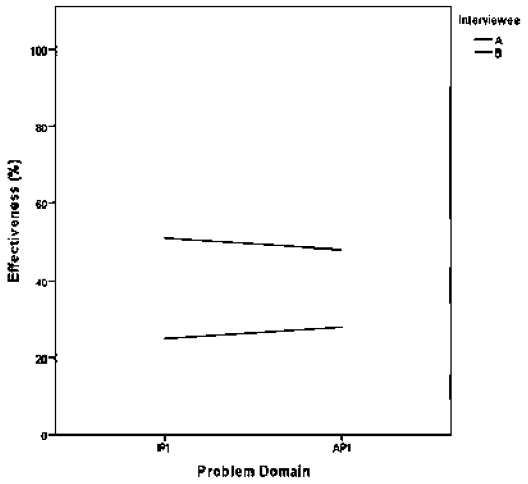

Fig. 2. Mean effectiveness by problem and interviewee.

The above visual perceptions of the effect of knowledge and the influence of the interviewee on the effectiveness of the subjects can be checked statistically by means of the repeated measures GLM analysis.

5.3 Hypothesis Testing

Before going ahead with hypothesis testing, we checked whether or not the experimental data satisfy the condition of normality of residuals required by GLM, as specified in Section 3.12.

We used the Kolmogorov-Smirnov and Shapiro-Wilk tests in order to test the normality of the residuals. The results of the tests show that the data of AP1 (KS: p-value = 0.132 and SW: p-value = 0.129) and IP1 (KS: p-value = 0.200 and SW: p-value = 0.423) come from a normal distribution. The results of the repeated measures GLM are shown in Table 9. First, we find that the ***Knowledge x Interviewee*** **interaction is not significant**. This means that the effects of knowledge for both interviewees can be studied together. The results suggest that ***Knowledge*** (p-value = 0.633 > 0.05) **does not have a significant effect on the effectiveness of the elicitation process**. The null hypothesis ($H_{1.0}$) cannot be rejected, that is, the subjects tend to be similarly effective for both the domain-aware and domain-ignorant problems.

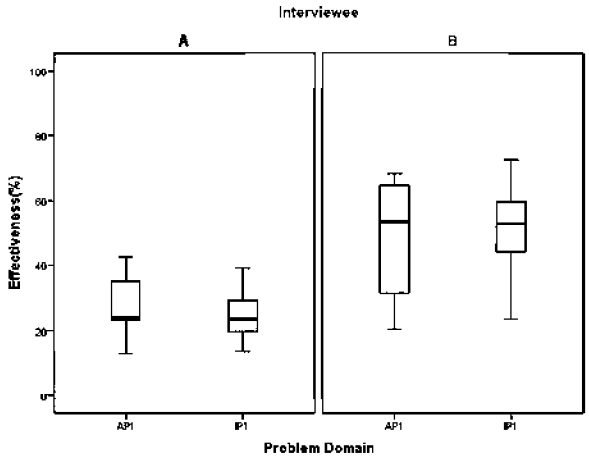

Fig. 3. Mean effectiveness by problem and interviewee (box plot).

TABLE 9
GLM Analysis

Source	Type III sum of squares	df	Mean square	F	Sig.	Eta^2
Knowledge	33.904	1	33.904	.240	.633	.020
Interviewee	3,685.583	1	3,685.583	13.181	.003	.523
Knowledge *x* Interviewee	67.469	1	67.469	.477	.503	.038

As the experiment is powerful enough (see Section 3.4) to detect medium effects, the effect of knowledge, if any, must be very low, a fact that is confirmed by the effect size estimator $Eta^2 = 0.02$. Eta^2 represents the percentage of variance explained by the factor; the greater this is, the more influential the factor is. It is not possible to translate the value of Eta^2 to natural units, which is why the guidelines suggested by Cohen [42] are used for its interpretation. The advantage of using these guidelines is that they are able to compare heterogeneous estimators. In our case, $Eta^2 = 0.02 < 0.2$, which, according to Cohen, corresponds to a very small effect. This very small effect can be equated to a very small effect of the effect size estimator d, that is, there is hardly any effect at all.

Although this was not the aim of this experiment, by blocking by *Interviewee* we discovered **that the *Interviewee* has a positive and statistically significant influence** (p-value = 0.003 < 0.05) **on the effectiveness of the elicitation process**. This shows that the person that provides the information, that is, the customer or user, is a critical component in requirements elicitation. The influence of the *Interviewee* on subject effectiveness is $Eta^2 = 0.523 > 0.5$, which is a very high effect [42].

5.4 Effect of Domain Elements

As specified in Section 3.7, the number of problem domain requirements and concepts differ (although the total number of elements is the same). This could affect our measurement of the effect of domain knowledge. To check whether this threat to validity is realized in this experiment, we ran a multivariate GLM analysis using the dependent variables *Number of Processes, Number of Concepts and Number of Requirements*.

We used absolute values instead of the percentages used for the dependent variable *Effectiveness*. Percentages calculated on the basis of the total number of each type of element (processes, concepts and requirements) and vary in proportion to the differences in the totals (see Table 4). Contrariwise, the number of identified processes, concepts and requirements depends exclusively on the experimental factors (i.e., domain awareness or ignorance). This would appear to be the best way of finding out whether there being more concepts or requirements has an influence on the domain knowledge effect. The dependent variable *Effectiveness* could be expressed in either absolute or relative terms, but the percentage metric appears to make finer distinctions.

The results of the analysis are shown in Appendix E, available in the online supplemental material, and are not included here, as they are secondary to the experimental hypothesis. The results are not significant with respect to

TABLE 10
Complexity (As Perceived by the Subjects)

Problem	Complexity (as perceived by subjects)			
	Low	Medium	High	Total
AP1	10	2	1	13
IP1	0	3	10	13

Knowledge, that is, **domain-aware and domain-ignorant analysts identify approximately the same number of processes, concepts and requirements**. These are similar to the results for the *Effectiveness* response variable and confirm that the **difference in the numbers of individual element types defining the experimental problems does not pose any threat to validity.**

Even if the analysis by element type had pinpointed significant effects, which it did not (it yielded the same results as the joint effectiveness analysis), this would not necessarily have posed a validity threat. Some p-values are in fact fairly low (for more details, see Appendix D, available in the online supplemental material), although they are still far from being statistically significant. Fig. 5 in Appendix D, available in the online supplemental material, illustrates the reason. It is clear that more concepts but fewer requirements are detected for IP1 than for AP1. This is completely consistent with the fact that problem IP1 has more concepts, but fewer requirements than problem AP1. In other words, the subjects detected more (or fewer) elements of a particular type (concepts, requirements) because there are more (or fewer) elements of that type in the domain under study. On the other hand, when they are similar in number, so is the number of detected elements (e.g., processes). This observation suggests that the decision taken in Section 3.1.1 not to make a distinction between processes, concepts and requirements in order to calculate analyst effectiveness was right. Analysts appear to identify the different types of elements depending on their relative number in the domain and not with respect to their type, that is, they are all equally important and should be regarded as such.

The above pattern does not change substantially when the analysis is broken down by interviewee, although there may be a sharper increase or decrease in the number of identified elements. For example, a lot more concepts are elicited from interviewee B than from interviewee A for problem IP1 (see Table 10). This observation supports the fact that the interviewee has a big influence on the elicitation process, as pointed out above.

5.5 Effect of Apparent Complexity on Problems

In Section 3.7, we also pointed out that problem AP1 appears to be more complex than problem IP1. This could subsequently generate a *Knowledge x Complexity* interaction and bias the results of the experiment. In order to study this potential validity threat, we asked subjects about how complex they perceived the problems to be. The results are shown in Table 10. It is evident that the subjects did not regard problem AP1 to be more complex than IP1. The perceived complexity was completely consistent with domain awareness and domain ignorance, which, ultimately, is the dominant factor. As IP1 was considered to be more complex than AP1, any complexity-induced effect would add to the influence of the *domain-ignorant* level and magnify its effects. This did not happen, so complexity does not appear a priori to be a moderator variable threatening the validity of the results. This does not necessarily mean that there is no *Knowledge x Complexity* interaction, as subjects' personal perceptions are not always reliable [23].

We now describe an internal replication of the experiment that counterbalances the *Knowledge x Complexity* levels in order to cancel out any spurious effects and rule out this validity threat once and for all.

6 Internal Replication

The key reason behind the replication was to extend the external validity of the results of the experiment. The results of our experiment suggest that domain knowledge has no influence on analyst effectiveness. This result is somewhat controversial and might be a sign of some sort of validity threat being at work.

The internal replication was conducted as part of the same course by the same experimental subjects and interviewees who participated in the experiment. The replication was run at the end of the course, three months after the baseline experiment. Like the experiment, the replication was conducted as part of a graded practical assignment, which assured that subjects were similarly motivated in both experiments. In the following we adopt the proposal by Carver [43] for reporting replications.

6.1 Changes in the Replication with Respect to the Baseline Experiment

The replication was similar to the experiment in all respects (hypothesis, factor, dependent variable, experimental task, etc.), save that:

- We have modified the problem domains used in the experiment, although one was still *domain-aware* and the other *domain-ignorant*.
- We have modified the order in which the problems were executed, that is, a domain-ignorant problem is used in the first session and a domain-aware problem in the second. This change in the order also counterbalances the possible effects of problem complexity that could be affecting the baseline experiment results.
- The replication was conducted after subjects had received training in requirements engineering and specifically in elicitation.

6.1.1 Change on Experimental Objects

The key difference between the baseline experiment and the internal replication lies in the experimental objects. In a repeated measures experiment, all subjects perform the experimental task on all objects, in our case in all problems. The same problems cannot be used again in the replication because they are already known to all subjects participating in the baseline experiment. Therefore, other problems have to be defined.

TABLE 11
Problem Domains Used in the Experiment

Problem	Brief description	Levels
IP2	Stock trading system	Domain Ignorant
AP2	University information point enrolment management system	Domain Aware

In the internal replication we decided to use a university information point enrolment management system as the domain-aware problem, whereas the domain-ignorant problem addresses a stock portfolio trading system, as shown in Table 11. Problems IP2 and AP2 are equivalent to the instantiation of the domain-ignorant and domain-aware treatment respectively. Like AP1 and IP1, these problems have been defined exhaustively. The full description of the problems is available in Appendix F, available in the online supplemental material, while a summary is provided below:

- *Stock trading system (IP2)*: This is a system for trading stocks on the stock exchange. Users shall be able to buy, sell and query their stock portfolio, as well as receive notifications depending on trading operations.
- *University enrolment control system (AP2)*: This is a university enrolment management system operated via a self-service kiosk, accessed by students using a student card reader. Students shall be able to enrol, pay enrolment fees and query academic records, etc.

6.1.2 Change in Problem Execution Order

In the baseline experiment, the subjects conducted the elicitation first on the domain-aware problem and second on the domain-ignorant problem. This order was chosen to stop students getting frustrated and help them to gain confidence in their interview skills by having them tackle a domain-aware problem first. However, this might have had a negative effect on the effectiveness of subjects tackling the first problem (AP1), as, in order to prevent reactivity effects with the experimental arrangements, no warming-up activity was performed. By changing the order of the problems during the internal replication (first domain-ignorant problem IP2 followed by domain-aware problem AP2), it was be possible by means of a joint analysis of the experiment and the replication (see Section 7) to remove the order effect. Note that this effect does not disappear from the replication, has to be taken into account during its interpretation.

The change in the order of the problems also counterbalances the possible effects of complexity. In the baseline experiment, the AP1 problem appeared to have a somewhat more complex structure than problem IP1. In the replication, we used the structure of the AP1 problem to define problem IP2 (see Appendix F, available in the online supplemental material), whereas the structure of problem IP1 was used to define AP2. Accordingly, the potential effect of the *Knowledge x Complexity* interaction is cancelled out in the joint analysis of the two experiments (baseline and experiment), although the individual replication is still subject to its effect.

TABLE 12
Total Number of Elements that Define the Problems

Problem	Elements that define the problem (#)			
	Requirements	Concepts	Processes	Total
IP2	24	12	14	50
AP2	17	20	17	54

The problems used in the replication are of a similar size to the problems used in the experiment in order to prevent the size of the problem domain from influencing effectiveness. Table 12 shows the total number of elements that define and delimit the size of all the problem domains used in the replication. These values are similar to the problems AP1 and IP1 reported in Table 4. Note that, as for problems AP1 and IP1, the number of requirements, concepts and processes, are not equal in AP2 and IP2. Accordingly, we will study the possible effect of these differences on analyst effectiveness.

6.1.3 Experiment Execution Time

The baseline experiment was executed at the start of the 2012/13 academic year before the students acting as subjects had received any requirements engineering training whatsoever. Considering that subjects reported having very little experience of requirements-related activities, this might very well have an equalizing effect on the effectiveness of subjects, that is, might prevent subjects from excelling in any particular domain.

In order to study this possible effect, the replication was conducted at the end of the requirements engineering course, during which subjects received training (and got practice) in tasks related to requirements elicitation, documentation, as well as analysis, verification and validation. The total course teaching workload was 40 hours, plus independent work and study, adding up to the equivalent of six ECTS credits.[3]

TABLE 13
Total Number of Subjects Familiar with Problems IP2 and AP2

Replication (13 subjects)		
Characteristics	Level	#Subjects
Familiarity with domain	IP2	10
	AP2	12

6.2 Execution of the Replication

6.2.1 Sample

Of the 14 students that participated in the experiment, 13 participated in the replication, as one of the students dropped out of the course. The population data are the same as shown in Table 7 of Section 4.1, except for the familiarity of subjects with the problem domains. According to the information provided in the post-experiemntal questionnaire and specified in Table 13, 10 subjects stated that they were familiar with IP2, whereas 12 subjects are familiar with AP2. Even though several subjects were apparently familiar with IP2, we considered that subjects were ignorant of the IP2 problem. We will discuss whether or not familiarity has an effect on subject effectiveness in Section 8.5.

3. http://ec.europa.eu/education/tools/ects_en.htm

TABLE 14
Descriptive Statistics for Effectiveness in Problems AP2 and IP2 by Interviewee (A, B)

Effectiveness (%)				
	Problem Domain Knowledge			
	Ignorant (IP2)		Aware (AP2)	
	A	B	A	B
N	5	7	6	7
Mean	40.00	62.29	46.30	85.19
Maximum	58	78	67	93
Minimum	26	46	30	76
Median	40.00	60.00	42.59	88.89
Variance	182.000	157.905	224.966	52.583
Std. Dev.	13.491	12.566	14.999	7.251

6.2.2 Preparation

As with the experiment, the subjects were notified by email of the date, time and place where they were to conduct the elicitation session. In order to properly play the role of customers, the interviewees studied and prepared the problems beforehand.

6.2.3 Execution

The internal replication was conducted in January 2013 as another assessable requirements engineering practical assignment for the experimental subjects. The sessions took place on 17 and 19 January, respectively.

The replication was again executed following the experimental procedure described in Section 4.3 adapted to two new problems (IP2, AP2).

Unlike the experiment, the subjects did use up the 30-minute time limit allocated for the elicitation session. On average, elicitation sessions lasted 27 minutes, where the maximum duration was 42 minutes and the shortest interview lasted 15 minutes. Six of the 13 subjects went over the allotted 30 minutes to acquire the information for problem IP2, whereas one overran the time limit for problem AP2. The raw data are available in Appendix G, available in the online supplemental material.

6.2.4 Deviations

No deviations were observed during the execution of the replication, as the replication was carried out according to the preliminary schedule.

6.3 Results of the Replication

6.3.1 Dataset Reduction

We had one incident with respect to the number of experimental subjects: one subject, on personal grounds, did not perform the experimental task for the domain-ignorant problem (IP2). We removed this subject from the analysis.

6.3.2 Descriptive Statistics and Plots

Table 14 shows the total number of subjects that interviewed each interviewee, the mean, maximum and minimum effectiveness achieved by subjects, as well as the median, variance and standard deviation. The trends of the observed results are as expected. On average, the subjects that interviewed interviewee B tend to acquire more information than the subjects that interviewed interviewee A for both the domain-aware and domain-ignorant problem. The subjects that interviewed interviewee B acquire more information on the domain-aware problem (85 percent) than on the domain-ignorant problem (62 percent). With a similar but smaller trend, the subjects that interviewed interviewee A acquire more information on the domain-aware problem (46 percent) than on the domain-ignorant problem (40 percent). These differences again confirm the influence of the interviewee on analyst effectiveness.

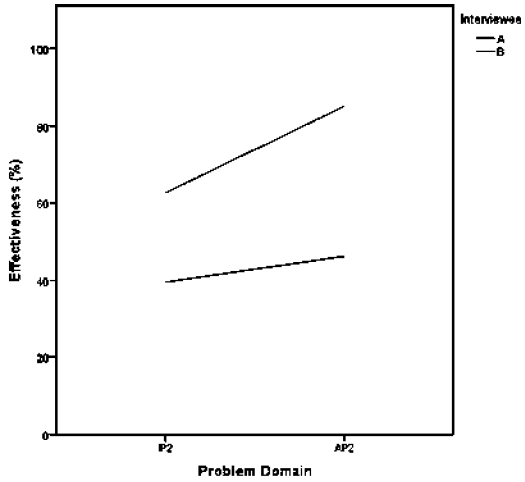

Fig. 4. Average effectiveness by problem and interviewee—internal replication.

The mean effectiveness achieved by experimental subjects in each of the problem domains is shown in the profile plot illustrated in Fig. 4. The profile plot represents the problem domains used in the replication on the x-axis and the mean effectiveness of subjects on the y-axis. Note that effectiveness is broken down by interviewee.

The profile plot illustrates that ***Knowledge* possibly has a positive effect** for the domain-aware problem. The subjects that interviewed interviewee A are 6 percent more effective for AP2, whereas the differences are even more marked with interviewee B at as much as 23 percent. We cannot rule out the possibility of this observed difference in effectiveness between interviewees being due to a possible interaction of the interviewee with knowledge. In any case, this interaction is ordinal, that is, the trend is always the same even though the effects of one level of a factor are not equal for all levels of other factors [44]. Therefore, the analysis model is still valid.

In the box plots shown in Fig. 5, we find, as in the baseline experiment, that the spread of the subjects that interviewed interviewee A is larger than for the subjects that interviewed interviewee B. For interviewee A, variability is greater with respect to the domain-aware problem, whereas for interviewee B, it is larger with respect to the domain-ignorant problem. Again, this difference in spread signals the effects of the *Interviewee* on analyst effectiveness and a possible interaction with the *Knowledge*.

6.3.3 Hypothesis Testing

Before going ahead with the hypothesis testing, we checked whether or not the experimental data satisfied the condition

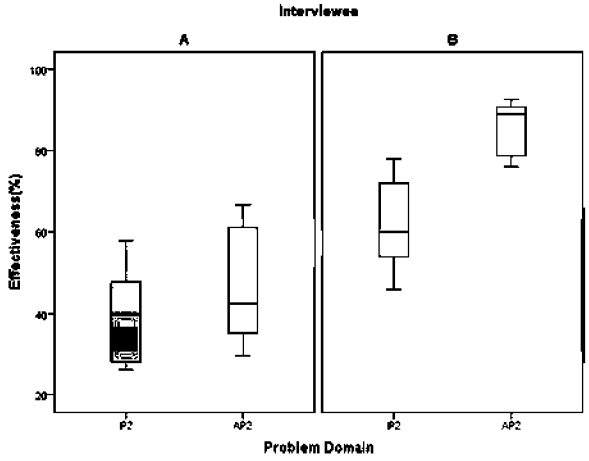

Fig. 5. Box plot – internal replication.

of normality of residuals, as specified in Section 3.11. We used the Kolmogorov-Smirnov and Shapiro-Wilk tests in order to test the normality of the residuals. The results of the tests show that the data of I P2 come from a normal distribution (KS: p-value = 0.200 and SW: p-value = 0.692). However, the data of AP2 have a small deviation: they are normal for KS (p-value = 0.183) but not normal for SW (p-value = 0.046). This deviation is, however, slight (the values for asymmetry and kurtosis are within the usual ranges of ± 1). We therefore believe that the GLM result is reliable. To be sure, however, we also applied non-parametric tests.

The results of the repeated measures GLM are shown in Table 15. The results suggest that ***Knowledge* does not have a significant effect on the effectiveness of the elicitation process** (p-value = 0.301 > 0.05). The null hypothesis ($H_{1.0}$) cannot be rejected, that is, the subjects tend to be similarly effective for both the domain-aware problem and the domain-ignorant problem. This result is wholly consistent with knowledge having a medium-low effect on effectiveness ($Eta^2 = 0.106$).

The ***Interviewee*, like the *Interviewee* x *Knowledge* interaction between interviewee and knowledge, does have a significant effect** (p-value = 0.001 < 0.05 and p-value = 0.038 < 0.05, respectively). This goes to confirm the influence of the *Interviewee* on the effectiveness of the elicitation process, whereas the possible *Knowledge x Interviewee* interaction suggests that elicitation effectiveness is equally or more dependent on the interviewee than on the analyst.

To check these findings, we analysed the effect of knowledge by interviewee. Using this approach, the sample size is very small, and it is virtually impossible to determine whether or not the sample is normal. On this ground, we applied the non-parametric Wilcoxon test for paired samples. Notice that this implies a double check of the earlier GLM analysis, which, as already mentioned, might have been compromised by the non-normality of AP2.

TABLE 16
Wilcoxon Test

Interviewee	Sig.	Z	H_0 (AP2 = IP2)
A	0.138	−1.483	Confirmed
B	0.028	−2.197	Rejected

The results of the test are reported in Table 16. We find that knowledge has no effect for interviewee A (z = −1.483, p-value = 0.138 > 0.05), whereas, on the other hand, knowledge has a significant effect for interviewee B (z = −2.197, p-value = 0.028 < 0.05).

Looking exclusively at the tests, the conclusion is that knowledge has a positive effect, which is, however, strongly moderated by the interviewee. In the case of interviewee A, the problem type (domain-ignorant vs. domain-aware) does not have a big enough effect to make the test significant. In the case of interviewee B, the effect of knowledge is more marked and significant. Note, however, that the p-value in the case of interviewee A is very low (p = 0.138), which, considering the small number of tested subjects (n = 5, taking into account the dropout) indicates that the results are borderline. In other words, this replication clearly signifies, but cannot confirm, that knowledge has a separate effect from the interviewee participating in the elicitation process.

6.4 Effect of Domain Elements

The results of the analysis are shown in Appendix H, available in the online supplemental material, Table 8. Technically speaking, the results are not significant in all cases (concepts, processes and requirements). This confirms the results of the analysis already conducted for *Effectiveness*. However, the p-values are very low for concepts (p-value = 0.051 ≈ 0.05) and processes (p-value = 0.078 ≈ 0.05). For all practical purposes, we can consider these differences to be significant.

As already specified in Section 5.4, it is the type of differences rather than whether or not the differences are significant that is relevant. As Fig. 11 in Appendix H, available in the online supplemental material, shows, the experimental subjects identified more processes and concepts in AP2, which is consistent with the fact that AP2 contains proportionally more concepts and requirements than IP2. Note also that subjects identified more concepts and processes, as was also the case with *Effectiveness*. The results are fully consistent. Therefore, the fact that AP2 and IP2 have different proportions of different types of elements does not appear to pose a threat to the experiment validity.

According to the analysis by *Interviewee* (see Table 8 and Fig. 12 in Appendix H, available in the online supplemental material), the interviewee has the same bearing as already noted.

TABLE 15
Knowledge Effect (GLM) – Replication

Source	Type III sum of squares	df	Mean square	F	Sig.	Eta^2
Knowledge	87.791	1	87.791	1.191	.301	.106
Interviewee	5,523.851	1	5,523.851	23.265	.001	.699
Knowledge* Interviewee	420.148	1	420.148	5.698	.038	.363

TABLE 17
Comparison between the Experiment and the Replication

Problem	Complexity (as perceived by subjects)			
	Low	Medium	High	Total
AP2	4	8	1	13
IP2	3	9	0	12

6.5 Effect of Apparent Problem Complexity

Table 17 shows the complexity of problems AP2 and IP2 as perceived by the experimental subjects. Clearly, the subjects considered both problems to be equally complex and did not take the view that IP1 appears to have a more complex structure. Complexity does not appear to cause any validity threat whatsoever, although this can only be truly determined by means of a joint analysis of the baseline experiment and the replication, as detailed in Section 7.

7 Comparison between the Experiment and the Replication

In this section, we report the similarities and differences between the experiment and the internal replication. As we have the raw data from both runs, we also conducted a joint statistical analysis. Through this analysis, we will be able to:

- Study the effects of the order in which the experimental problems are addressed
- Study the effect of training
- Cancel out the potential effects of experimental problem complexity
- Increase the statistical power of the individual studies (experiment and internal replication).

7.1 Consistent Results

Looking at statistical significance, the results of the experiment and the internal replication can be said to be completely consistent, as shown in Table 18. *Knowledge* has either no or, at most, a small effect (note that the statistical power of the experiment is capable of detecting medium but not small effects). The blocking variable *Interviewee* has a significant effect on analyst effectiveness. Although there is probably a *Knowledge*Interviewee* interaction, the applied statistical model is valid as the interaction is ordinal.

7.2 Inconsistent Results

The results of comparing the mean effectiveness achieved by the subjects in both the experiment and the replication tend to differ slightly, as shown in Table 18.

In the replication the effectiveness of the subjects for the IP2 (ignorant) problem was less than for the AP2 (aware) problem both without blocking and blocked by *Interviewee*. In the particular case of *Interviewee* B, the difference between IP2 and AP2 is even statistically significant. However, in the experiment the effectiveness for problems AP1 and IP1 was more or less equal (a difference of only $\pm$ 3 percent).

Since the main change between the replication and the experiment is the training, the most reasonable explanation is that subject training had a marked influence on their effectiveness. Between the baseline experiment and the internal replication, the subjects received master-level training on requirements engineering and specifically elicitation, as well as other software engineering issues. A training-related improvement in effectiveness is not an extraordinary occurrence; it is a fundamental assumption in academia and had also been observed in controlled situations [45]. This would imply that domain knowledge does have an effect, although it does not operate separately from requirements engineering knowledge (in this case acquired through training). Proper training was required for the effect of domain knowledge to be large enough to be detected with relatively small sample sizes.

TABLE 18
Comparison of the Results of the Experiment with the Replication

Independent variable	Significance		Comparison of means	
	Exp.	Repl.	Exp.	Repl.
Knowledge	No	No	AP1 $\approx$ IP	IP2 < AP2
Interviewee	Yes	Yes	B > A	B > A
Knowledge*	No	Yes	Int. A AP1 $\approx$ IP1	Int.A IP2 < AP2
Interviewee			Int. B AP1 $\approx$ IP1	Int.BIP2 < AP2^{+}

$^{+}$*Statistically significant comparison.*

A second possibility, which does not necessarily rule out the first, is that it could be due to the effects of order. As mentioned earlier, IP2 was always presented to subjects before AP2. The AP2-IP2 sequence was never tested. Subjects who have become practised at interviewing (i.e., experienced a learning effect between IP2-AP2) might be more effective with respect to the problem that they tackled in second place. The fact that this improvement in effectiveness is not observed between AP1 and IP1 is not contradictory. Assuming that *Knowledge* does have an effect, tackling IP1 (which is a domain-ignorant problem) in second place should lead to an increase in effectiveness, which is precisely the observed result.

Each alternative can be evaluated by statistical analysis, thanks to the changes made to the design of the internal replication. The joint analysis discussed in Section 7.3 has the additional advantage of cancelling out any effect of complexity on knowledge by counterbalancing the baseline experiment and the replication and thus strengthening the conclusions of this paper.

7.3 Joint Analysis

The baseline experiment and internal replication can be considered on the whole as a single experiment with two factors and two blocking variables.

The factors are *Knowledge*, as in both experiments, and the *Order of Execution* of the elicitation sessions. The *Order of Execution* reflects the order in which the subjects tackled the problems. In the baseline experiment, the order was aware-ignorant, whereas the order was reversed ignorant-aware in the replication. As the two possible orders have been tested, the *Order of Execution* can be interpreted as a factor with two levels: *Before (B)* and *After (A)*.

The blocking variables are the *Interviewee*, as in both experiments, and the *Training* at the time when the experiments were run. The levels of the *Training* blocking variable are *Before Training* (*BT*) and *After Training (AT)*, which coincide

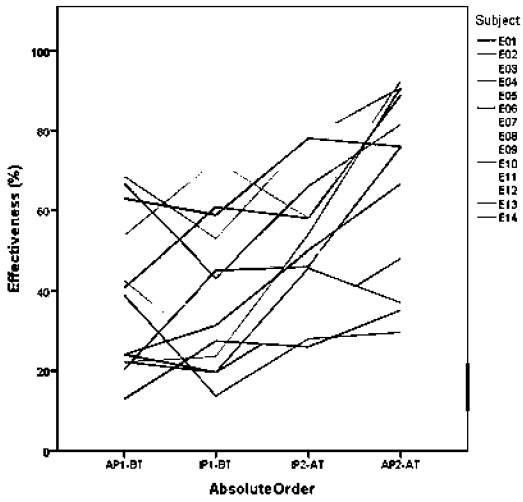

Fig. 6 Effectiveness by subject.[4]

with the execution of the experiment and the replication, respectively. The *Training* blocking variable will reduce data variability by explicitly taking into account subject training.

The complexity of the analysis rules out the use of the repeated measures GLM. Note that effectiveness was calculated four times per subject but not under the same conditions. There are two variables (*Order of Execution* and *Training*) that determine the correlations between repeated measures for each subject, which are potentially different in each case.

Fig. 6 shows how the repeated measures are distributed by subject. The x-axis shows the absolute order in which the measures were taken, irrespective of *Training*. The plot has not been corrected for knowledge, but the trends are clear enough. Accounting for the behaviour of each subject with respect to the problems will help to reduce factor variability and thus increase the power of the statistical tests.

Instead of the repeated measures GLM, we used a mixed model, whereby we could take into account both fixed effects (the separate effects of the factors and blocking variables) and random factors (the increases or decreases in subject effectiveness depending on the *Order of Execution* and *Training*). More specifically, the design of the fixed effects model is as follows:

$$Effectiveness = Knowledge + Training + Order\,of\,Execution + Interviewee + \varepsilon_1.$$

The design of the random effects model is as follows:

$$Effectiveness\,(Subject) = Training + Order\,of\,Execution + \varepsilon_2.$$

We have used an AR(1) covariance matrix, as the nature of the experiment suggests that close measures will be more strongly correlated than measures that are far apart for each subject. The results of the mixed model did not vary substantially if we use an unstructured covariance matrix (which is the most general structure possible). An additional advantage of AR(1) is that it was favourable to the convergence of the REML (restricted maximum likelihood) procedure.

Table 19 reports the results of the statistical analysis. The table differs somewhat from the typical ANOVA table, but is easily interpreted and, more importantly, directly states the effect sizes in natural units. We cannot provide Hedges' d or similar statistics, as it is unclear how they should be calculated for models with multiple error terms. The results (rounded, except for the p-value, to 1 decimal place) are as follows:

- The blocking variable ***Interviewee*** **has a marked effect on the elicitation effectiveness** of subjects. In natural units, one *Interviewee* (B) provides 26.7 percent more information than the other (A).
- The blocking variable ***Training*** **has a marked effect on elicitation the process effectiveness** of subjects. Subjects were 20.9 percent more effective on average *After Training*.
- The *Order of Execution* of the elicitation sessions does have an effect, as we had feared when it was classed as a threat to the validity of the baseline experiment. However, it is smaller than visual inspection suggested. In percentage terms, **the subjects were 7.3** percent **more effective for the problem that they tackled in second place.** Although the result is not significant ($p = 0.061$), the results are very close to the significance level α.
- As in both the experiment and the replication, **domain *Knowledge* was significant ($p = 0.005$) but had a rather small effect**. Subjects were only 7.3 percent more effective for the domain-aware problem than for the domain-ignorant problem.

Such a small effect of Knowledge is rather surprising considering the prevalent belief in SE literature that domain knowledge makes requirements analysis more effective [6], [13], [14], [15]. Additionally, it makes sense of the results of the related empirical studies that were described in Section 2, as discussed in the next section.

8 Discussion

8.1 Comparison with Related Empirical Research

Our experimental results show that domain knowledge has a small, but significant effect on the effectiveness of analysts during the requirements elicitation process. This result is consistent with the more widespread opinion in SE on the effect of knowledge, that is, subjects with problem domain knowledge are more effective than subjects that have no such knowledge.

The results of our experiment are contrary to experimental findings by Niknafs and Berry [4], [16]. In both papers, Niknafs and Berry conclude that the inclusion of an ignoramus improves the performance of brainstorming groups (e.g., generates more innovative ideas). However, as the authors gathered more evidence, the results became inconclusive. In [20], Niknafs analysed 40 groups (more than double the number used in [4]) and found that the inclusion of an ignoramus did not have any statistically significant effects. However, the raw data do suggest there are differences between domain-ignorant and domain-aware people,

4. AP1-BT: domain-aware problem before training; IP1-BT: domain-ignorant problem before training; AP2-BT: domain-ignorant problem 2 after training; IP2-BT: domain-ignorant problem 2 after training.

TABLE 19
Mixed Model Results (AR)

Parameter[5]	Estimate	Std. error	df	t	Sig.	95% Confidence interval	
						Lower bound	Upper bound
Intercept	52.9	4.7	26.9	11.3	0.000	43.3	62.5
[Knowledge = IP]	−7.3	2.2	12.2	-3.4	0.005	−12.0	−2.6
[Knowledge = AP]	0[b]	0.0	.	.	.	.	.
[Training = BT]	−20.9	3.1	14.0	−6.8	0.000	−27.5	−14.3
[Training = AT]	0[b]	0.0	.	.	.	.	.
Interviewee	26.7	5.3	14.4	5.0	0.000	15.3	38.2
[Order = B]	−7.3	3.6	13.7	−2.0	0.061	−15.1	0.4
[Order = A]	0[b]	0.0	.	.	.	.	.

a. Dependent variable: consolidated elements (percent).
b. This parameter is set to zero because it is redundant.
5. Acronyms: IP: domain-ignorant problem; AP: domain-aware problem; BT: before training; AT: after training; B: before; A: after.

and the differences are not always in favour of the latter group. For example, 3I groups (with three domain-ignorant members) are the one that generate a greater number of feasible ideas (see [20, p. 48], Table 4).

There are many possible explanations for the inconsistencies. As regards the inconclusiveness of Niknafs and Berry's results [20] compared with the significant effects reported in this study, we believe that it is due to the fact that the sample size of the combined experiments E1+E2 reported in [20] is not large enough to detect small effects. Note that Niknafs and Berry used a between-subjects design, which requires over 100 experimental units to detect medium effects. The two experiments reported in this paper use within-subjects designs, which require a smaller size, albeit at the cost of posing some threats to validity.

Irrespective of whether or not the results are significant, it remains to be explained why some groups composed exclusively of domain-ignorant people are more effective than other groups. This could be interpreted as domain knowledge having a negative effect. One possible cause of the difference in the results is the elicitation technique used (interviews versus brainstorming). Brainstorming is essentially a creative process. As domain-ignorant subjects are unconditioned by previous experiences, there are fewer constraints on the ideas that they propose. An interview, however, requires communication between the interviewee and interviewer, and previous knowledge could be helpful for making the communication more effective. In our opinion, Niknafs and Berry might be reporting an Einstellung phenomenon [46] in the field of requirements elicitation rather than a difference in effectiveness that is traceable to domain knowledge.

However, these are no more than preliminary conclusions. For example, it is curious that the groups reported as being the most effective in [20] are 0I (all domain-aware members) and 3I (all domain-ignorant members). Teams 1I and 2I almost always generate fewer ideas. It is possible that many aspects (domain knowledge, task, ease of communication, etc.) are interacting in a complex manner during the elicitation task, causing the observed instability in the effect of domain knowledge. In turn, such instability is an explanation for the fact that there is no clear position (each practitioner speaks from the blinkered viewpoint of his or her experience) on the effect of domain knowledge. For example, after conducting a survey of 40 practitioners with varying levels of experience, Mehrotra [46] found that domain awareness and ignorance are both positive for requirements elicitation.

Finally, Niknafs and Berry [4], [16], [20] use a mix of professionals and students. We have no objection to this procedure, but it does entail two risks. In the first place, the effects of knowledge are confounded with the effects of experience. Second, it is hard to get experienced professionals to understand the need to behave effectively [47]. If professionals are insufficiently motivated, knowledge may appear to exercise a negative effect due to poor performance by professionals with experience.

Kristensson et al.'s study [7], using a larger sample size (47 subjects, mix of professionals and students), can be interpreted likewise as *Einstellung* and reaches similar conclusions to Niknafs and Berry.

The results of our experiments are compatible with findings by Hadar, Soffer and Kenzi [5]. In Hadar et al.'s study, the subjects with domain knowledge asked more specific questions than domain-ignorant subjects. Hadar et al. suggest that phrasing questions more specifically could be related to a better effectiveness during the elicitation process. In the case of students, the effect size associated with the specificity of the questions is small (Hedges' $g = 0.36$), which is aligned with our results. In order to observe this effect, Hadar et al. used 56 experimental subjects. This is more or less the same number of experimental units as we used in our experiments.

Our experiments are unable to determine the reason why analysts with domain knowledge are almost equally effective as analysts without knowledge, although they can suggest hypotheses. The most widespread opinion, as inferred from the literature (see Section 2), attributes this lower than expected effectiveness to the fact that domain-aware analysts infer rather than capture customer/user needs. If this were true, it would imply that domain-aware analysts finish the interviews earlier than domain-ignorant analysts. As shown in Fig. 7 the data that we have collected suggest this is the case. On average, the time taken in interviews conducted by domain-aware analysts is slightly shorter than the time taken by domain-ignorant analysts $t_{ignorant} = 30 : 20$, $t_{aware} = 24 : 27$).

The differences are smaller than one might expect (just over 5 minutes, 16 percent of total time), although this is probably merely a reflection of the fact that, in academic

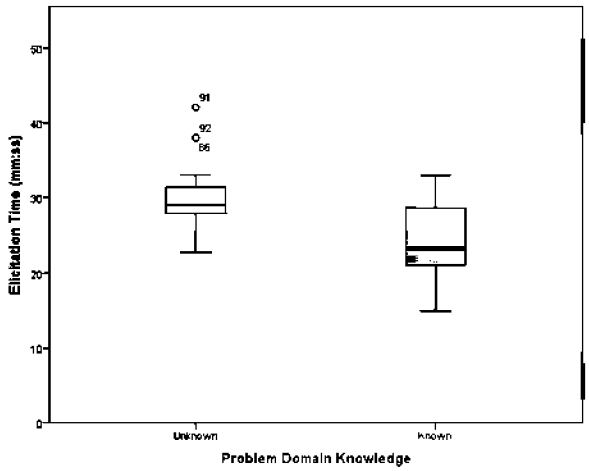

Fig. 7. Average length (minutes) of the elicitation sessions for domain-ignorant and domain-aware analysts (problems AP1, AP2, IP1 and IP2).

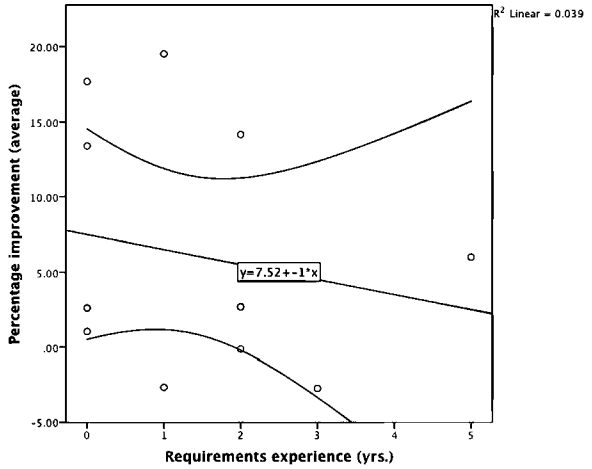

Fig. 8. Correlation between requirements experience and individual effectiveness.The curved lines represent the confidence intervals at 5 percent.

contexts, subjects tend to use up all the time that they are allowed (take an examination, for example, most students hand in their exam paper towards the end of the allotted time). In real-world environments, the differences in interview times might be greater. Future studies should investigate this issue further.

8.2 Influence of Experience on the Effectiveness of the Requirements Elicitation Process

Again we have to make clear that the results cannot be extrapolated to analyst *Experience*, that is, we are not saying that *experienced* analysts are less effective than *inexperienced* analysts. The potential effect of experience was eliminated by means of the *within-subjects* design, as each subject acted as his or her own control, thereby cancelling out any effect that experience might have.

However, the *within-subjects* design does not assure that the effect of experience is cancelled out if there is an *Experience* x *Knowledge* interaction, that is, if experienced and inexperienced domain-aware and domain-ignorant analysts react differently, then *Experience* may have a mediating effect on *Knowledge*.

The *Experience* x *Knowledge* interaction can be studied using the experience data that we have gathered from the experimental subjects (see Section 4.1). Years of experience (either total or confined to requirements-related activities) can be correlated to individual effectiveness (i.e., the difference in effectiveness between the problem-aware and problem-ignorant subjects). If there were no correlation (i.e., the regression line is horizontal), then we could conclude that there is no *Experience* x *Knowledge* interaction. An upward line would mean that there is a direct interaction (experience makes domain-aware analysts more effective). Strictly speaking, a direct interaction could also mean that analysts with domain knowledge are less effective in domains of which they are ignorant. However, our analysis cannot single out which of these two possibilities is true. If the line is downward, the interaction would be inverse (the interpretation would be just the opposite).

Fig. 8 shows the scatter plot and the regression line for Professional Experience measured in years, whereas Fig. 9 represents requirements experience, also measured in years. Strictly speaking, we find that *Requirements Experience* appears to interact inversely with *Knowledge*, but in practice the slope is virtually zero (−0.49 percent per year, $r = -0.1$). In turn, *Professional Experience* (years of experience in related requirements activities) interacts directly. The *Professional Experience* x *Knowledge* interaction intercept is quite low (3.67 percent), which means that predictably an inexperienced subject is not influenced by this interaction. Experienced subjects improve at a rate of 1.21 percent per year, that is, for every year of experience, analysts gather 1.21 percent more information about the domain of which they are aware than about the domain of which they are ignorant. In terms of correlations, the relationship is also substantial ($r = 0.35$). Our experiment cannot explain this phenomenon, although it is probably related to the process of expert knowledge consolidation, which is well documented in the literature [48].

There are two reasons why it is important that we identified a *Professional Experience* x *Knowledge* interaction. First, it can have a sizeable influence when experimental subjects have lengthy experience (note that a subject with 10 years of experience may be 12.1 percent more effective than an

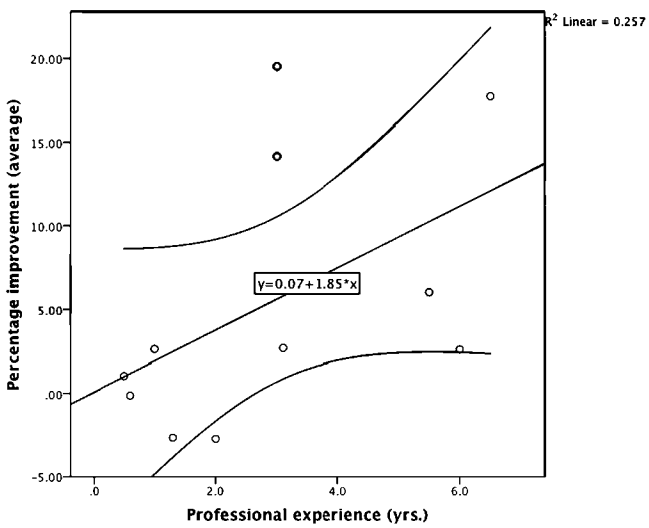

Fig. 9. Correlation between professional experience and individual effectiveness. The curved lines represent the confidence intervals at 95 percent.

TABLE 20
Significance of Factors and Covariables[a]

Source	Numerator df	denominator df	F	Sig.
Intercept	1	10.0	42.4	.000
Knowledge	1	9.9	8.2	.017
Training	1	11.5	57.8	.000
Order	1	11.0	6.2	.030
Interviewee	1	10.6	14.8	.003
ProfExp	1	10.4	.5	.518

a. Dependent variable: effectiveness (percent).

inexperienced subject). In such cases, the effect of the *Professional Experience* x *Knowledge* interaction cannot be overlooked and should be included in the analysis.

Second, our experimental subjects did not have a wide range of experience, although they did have considerable general software experience (0-6 years), and so the knowledge effect might have been overestimated. There are two ways for testing this:

- By averaging the improvement in subject effectiveness predicted by the regression line, we can estimate (with reservations, as it is a post-hoc analysis) that the improvement attributable to the *Professional Experience* x *Knowledge* interaction is 6.9 percent. This effect is comparable to the knowledge effect, which is estimated at 7.3 percent. However, the difference is still positive (7.3% – 6.9% = 0.4%). This means that even discounting the effect of experience, analysts with domain knowledge were still slightly more effective than domain-ignorant analysts.
- Adding *Professional Experience* as a covariable to the mixed model. The statistical significance of the results is shown in Table 20 whereas the effects are shown in Table 21. There are few changes with respect to the analysis reported in Section 7.1. The effect of knowledge is still statistically significant, although the effect is estimated to be slightly less (from 7.3 to 6.9 percent, confirming the influence of the Professional Experience x Knowledge interaction).

In both cases, **domain knowledge is confirmed to have a positive** effect. We also have to highlight that we have few dropouts and results should be with taken due caution.

8.3 Other Influential Aspects

The experiment revealed another two variables that have a strong impact on analyst effectiveness: the interviewee and requirements training (particularly, interview training).

It is widely recognized that the customer/user plays an extremely important role during the requirements process [49], [50]. The experimental data that we have gathered suggest that, at least as regards the early elicitation of requirements, interviewing one or other interviewee accounts for a difference of 25 percent on average in analyst effectiveness. Note that *Interviewee* and *Language* are confounded in our experiment. The interviews of interviewee A were conducted in English, whereas the interviews of interviewee B were held in Spanish. This difference may be partly explained by the fact that the interview was conducted in a first (B - J.W. Castro) or second (A - O. Dieste) language. The difference may be partly explained by the fact that the interview was conducted in a first or second language. In any case, the observed difference in effectiveness (i.e., 25 percent) suggests that the incidental aspects of the elicitation process may play a role that is equally or more important than the essential aspects (e.g., analyst knowledge).

The second notable issue is the enormous improvement in analyst effectiveness through specialized training and coaching.

The baseline experiment was conducted *before* the subjects *received training* in elicitation, which unquestionably explains the resulting low average scores for effectiveness. However, subject effectiveness grew tremendously (around 20 percent) during the internal replication, irrespective of the interviewee and domain knowledge (aware or ignorant) addressed. The major difference between the execution of the experiment and internal replication was the *training* that subjects received during the requirements course as part of which this experiment was conducted. Consequently, it is only logical to conclude that **requirements training increases analyst effectiveness considerably**, at least for the convenience sample used (MS students with out requirements experience).

A possible objection to the learning effect (from a methodological and scientific viewpoint rather than out of real conviction, as few will question the beneficial effects of training) would be the risk of the course lecturer (O. Dieste) biasing the training process (not necessarily deliberately) in such a

TABLE 21
Estimation of Significance of Factor Effects

Parameter	Estimate	Std. error	df	t	Sig.	95% Confidence interval	
						Lower bound	Upper bound
Intercept	59.5	6.8	13.9	8.8	0.000	45.0	74.0
[Knowledge = IP1]	−6.9	2.4	10.0	−2.9	0.017	−12.2	−1.5
[Knowledge = AP1]	0b	0.0	.	.	.	.	.
[Time = BT]	−22.7	3.0	11.5	−7.6	0.000	−29.3	−16.2
[Time = AT]	0b	0.0	.	.	.	.	.
[Order = BS]	-9.3	3.7	11.0	−2.5	0.030	−17.5	−1.1
[Order = AS]	0b	0.0	.	.	.	.	.
Interviewee	25.5	6.6	10.6	3.8	0.003	10.8	40.2
ProfExp	−1.1	1.6	10.4	−0.7	0.518	−4.6	2.4

a. Dependent variable: consolidated elements (percent).
b. This parameter is set to zero because it is redundant.

TABLE 22
Mixed Model Substituting Familiarity for Knowledge

Estimates of Fixed Effects[a]

Parameter	Estimate	Std. Error	df	t	Sig.	95% Confidence Interval	
						Lower Bound	Upper Bound
Intercept	37.8	6.8	33.2	5.6	0.000	24.0	51.7
[Time = BT]	−18.8	3.2	29.5	−5.8	0.000	−25.4	−12.2
[Time = AT]	0b	0.0	.	.	.	.	.
Interviewee	29.5	6.4	11.0	4.6	0.001	15.4	43.6
[Order = BS]	−10.4	3.5	14.7	−3.0	0.009	−17.9	−3.0
[Order = AS]	0b	0.0	.	.	.	.	.
Familiarity	5.4	1.9	27.1	2.9	0.007	1.6	9.2

a. Dependent variable: consolidated elements (percent).
b. This parameter is set to zero because it is redundant.

way that students performed better during the replication than they would do in other elicitation activities (e.g., an experiment executed by independent researchers or a real interview). This cannot be ruled out. However, thanks to the repeated measures of the two experimental replications, the improvement of subjects can be calculated not only before and after requirements training but also between elicitation sessions (using the session variable listed in Tables 20 and 21.

There is no between-session training, and therefore any difference would necessarily be attributable to increased proficiency gained by the analyst through merely conducting interviews (or, alternatively, by on-the-job training). The between-session improvement of analysts is over 9 percent on average, which is a very high value, comparable to the effect of knowledge itself (which is also around 9 percent). This means that **practice at least (and more than likely training too) has a very beneficial and almost immediate effect on analyst effectiveness.**

8.4 Familiarity with the Domain versus Knowledge of the Domain

The levels of the *Domain Knowledge* factor were established by the researchers based on their knowledge of the population on which the experiment was to be executed (postgraduate software engineering students). These students tend to have a very similar profile, and therefore we were able to predict quite confidently that they would be ignorant of the *Battery recycling* and *Stock trading* domains and aware of the *Mobile messaging* and *University enrolment* domains.

In order to confirm that our assumptions were correct, we asked students about their Familiarity with the respective domains. Generally (see Tables 7 and 13) over 85 percent of students rated the *Domain-aware* and *Domain-ignorant problems* as *Familiar* or *Unfamiliar*, respectively. In the case of the *Stock trading* domain, however, the figures were reversed: 77 percent of subjects (10 out of 13) stated that they were familiar with the problem, which had been rated by researchers as a *Domain-ignorant problem*.

The results reported in previous sections were calculated by means of a strict ITT[6] policy [51]. However, because of the high familiarity of subjects with one of the *domain-ignorant* problems, we need to check whether the results hold if we replace the *Knowledge* factor by the independent variable Familiarity (i.e., the subjects' opinion of their domain knowledge). Table 22 reports the results of the respective mixed model. The effect of Familiarity is small (5.4 percent), comparable to the effect of *Knowledge* (7.3 percent) reported in Section 7.3, and equally significant (p-value = 0.007). This is not surprising, as *Knowledge* and *Familiarity* are strongly correlated ($r = 0.51$, p-value < 0.001). Indeed, they are different operationalizations (own opinion, external opinion) of the same variable (domain knowledge). Therefore, we have to conclude that domain knowledge (irrespective of whether it is operationalized as *Knowledge* or *Familiarity*) has positive but small effects on analyst effectiveness.

The question then is, why the subjective perception of subjects is so contrary to the ITT analysis? We think that subjects were mistaken in their perception. Fig. 10 shows a box plot illustrating the effectiveness of subjects depending on their familiarity with the Stock trading domain. Note that the subjects that claimed to be more familiar with the domain were clearly also the least effective.

In order to confirm that *Familiarity* is not a good operationalization of domain *Awareness* or *Ignorance* for *Stock trading*, we held a one-to-one post-experimental interview (after the internal replication in January 2013) with the experimental subjects. In all cases, the subjects confirmed that they

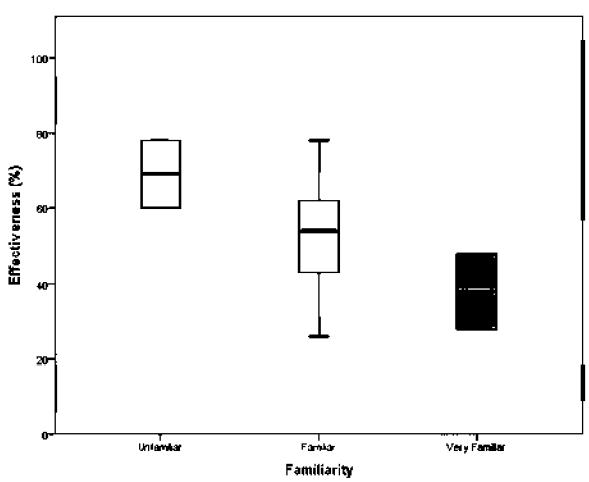

Fig. 10. Familiarity of subjects with IP2.

6. Intention to Treat.

were not familiar with the actual domain (*Stock trading*). The *Familiarity* that subjects reported appeared to be due more to the implementation technology (it was a mobile application), to its similarity with other commercial systems, or to causal knowledge gained, e.g.: watching TV news, than to stock trading itself. Niknafs [20] made the same observation about poor self-assessment of domain familiarity.

We cannot explain why computer engineers in our case made such unfounded judgements about their ability. However, computer engineers have been reported in the literature (e.g.: [52]) and observed [23] to be overoptimistic, and thus it is more of an inconvenience (as it is a potential validity threat) than a surprise.

8.5 Methodological Remarks on Sample Size

Finally, and at the risk of being repetitive, we believe that it is very important to stress the implications of using an unsuitable sample size to gather empirical knowledge. In very small samples (note that a small sample can be as large as 150 subjects [53], although 50 is probably a more accurate figure [54], [55], large effect sizes in either direction, provoked entirely by error, are very often observed [54], leading to the mistaken conclusion that effects are statistically significant. During experimental design it is imperative to analyse statistical power. Statistical power reports the possibilities of detecting a particular effect size (e.g., $d = 0.5$ as in our case). If we have few subjects, rather than extracting conclusions from individual experiments separately, it will be necessary to replicate the experiment a fair number of times and synthesize the results to achieve reasonable type II error rates. Note that synthesis must be formal, using either meta-analysis or blocked ANOVA or any equivalent procedure. Classical vote counting (that is, counting how many experiments have generated statistically significant results) would yield erroneous results [56], e.g., neither of the two replications reported in this paper have generated statistically significant results.

9 Validity Threats

9.1 Threats to Statistical Conclusion Validity

The small sample size of the original experiment might have caused the results not to be statistically significant. To combat this threat, we replicated the experiment internally, and we also analysed the data (of both the experiment and the replication) jointly as if we were dealing with a single experiment. In this manner, we have increased the statistical power and, consequently, the reliability of the results.

9.2 Threats to Internal Validity

The baseline experiment was conducted without subjects receiving any specific training or warming-up activity regarding interviewing. This may have meant that subjects were less effective in the experiment, especially for the domain-aware problem, which they tackled in the first place. We have dealt with this threat to validity, albeit indirectly, during the internal replication, as subjects received requirements engineering and specifically elicitation training. Additionally, we inverted the order in which the tasks were executed so as to offset the missing warming-up activity. As a result, the effects reported in the joint analysis are, we believe, reliable.

The source of another threat to internal validity is the non-foreseeability of the questions that interviewers are likely to ask. For instance, we excluded non-functional requirements from the benchmark list (gold standard) defining both the domain-aware and domain-ignorant problem domain elements. Therefore, interviewers who focused on this type of requirements could be less effective. To counteract this validity threat, the interviewees gave to-the-point responses and tried to steer the conversation to other avenues (see Section 3.5). However, this procedure may not have worked in all cases. The same applies to other types of information like, for example, management issues (which, although they are not part of the domain, often crop up in customer-analyst conversations).

9.3 Threats to Construct Validity

Analyst *Familiarity* with the problem domain was initially measured using subjective measures rated using Likert scales. This measurement is not reliable, as subjects' opinions may be biased and affect the findings. In order to combat this threat, we controlled *Knowledge* by having subjects tackle two problem types (domain-aware problem and domain-ignorant problem) in both the baseline experiment and the replication.

9.4 Threats to External Validity

The fact that experimental subjects come from a convenience rather than a random sample (i.e., the subjects have not been recruited from a larger population but are students enrolled in a particular course) is a threat to the external validity of the experiment. Therefore, due caution must be exercised when generalizing our results to professional analysts. However, the fact that the students were taking a professional master's course and most also had professional experience in computer-related jobs, mitigated this threat, as their results can be considered to be representative of their kind. Therefore, we believe our results can be generalized to junior developer novices in elicitation techniques.

The experimental setting was quite different from what professional analysts may be used to: simulated customer and limited time. We believe that this threat has a marginal effect at most. The students were highly motivated and performed professionally. Time was not an obstacle to information elicitation. Most of the students finished the elicitation session before running out of time.

In order to increase the external validity of our experiments, we used two different interviewees and four different problems. Therefore, our results are not restricted to one problem and one interviewee, but further experimentation needed in order to generalize our results to more problems and respondents.

10 Conclusions

The aim of the research reported in this paper is to study the effects of problem domain *Knowledge* on the effectiveness of requirements elicitation. We executed a controlled experiment with Madrid Technical University students as part of a Requirements Engineering course. This experiment was

internally replicated in order to check the observed effects in the experiment and increase statistical power.

The results suggest that *Knowledge* has a small, albeit statistically significant, effect on the effectiveness of the elicitation process. As a by-product of our research, we have found that the interviewee is a key factor during the requirements elicitation process, exercising much more influence on the final result of the elicitation process than analyst domain knowledge itself. Additionally, proper analyst training in aspects related to requirements engineering is of utmost importance. Training has an effect comparable to the interviewee and therefore greater than the domain knowledge effect.

Note that the above conclusions should not be overgeneralized. First, the experimental setting simulated the early stages of the elicitation process when analysts have very little information about the problem domain. Domain knowledge might have larger than detected effects later on. Additionally, experimental subjects were master's students with little or no requirements engineering experience. More experienced subjects might be more effective at tackling domain-aware problems, as already suggested in the discussion.

Let us stress that experimenters must pay special attention to sample size/statistical power when designing an experiment. If we had not decided to conduct an internal replication in order to increase the statistical power of the experiment, our results would have turned out to be completely opposite.

A very interesting (although co-lateral) finding is that subjects' personal opinions have to be used with due care as operationalization of knowledge in SE experiments since they may be biased on any number of grounds (e.g., overoptimism on the part of computer engineers), and this bias may spread to the findings of the experiment.

Acknowledgments

This work was supported in part by the project **TIN2011-23216** funded by the Spanish Ministry of Ministry of Economy and Competitiveness. Alejandrina Aranda holds a PhD grant from Itaipú Binacional, Paraguay.

References

[1] I. Sommerville, *Requirements Engineering: A Good Practice Guide*. New Delhi, India: Wiley, 2009.

[2] A. Aurum and C. Wohlin, *Engineering and Managing Software Requirements*. New York, NY, USA: Springer, 2006.

[3] F. Anwar, R. Razali, and K. Ahmad, "Achieving effective communication during requirements elicitation - A conceptual framework," in *Proc. 2nd Int. Conf. Softw. Eng. Comput. Syst.*, 2011, pp. 600–610.

[4] A. Niknafs and D. M. Berry, "The impact of domain knowledge on the effectiveness of requirements idea generation during requirements elicitation," in *Proc. IEEE 20th Int. Requirements Eng. Conf.*, 2012, pp. 181–190.

[5] I. Hadar, P. Soffer, and K. Kenzi, "The role of domain knowledge in requirements elicitation via interviews: An exploratory study," *Requirements Eng.*, vol. 19, pp. 143–159, 2014.

[6] D. Zowghi and C. Coulin, "Requirements elicitation: A survey of techniques, approaches, and tools," in *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin, Eds. Berlin, Heidelberg: Springer, 2005, pp. 19–46.

[7] P. Kristensson, A. Gustafsson, and T. Archer, "Harnessing the creative potential among users*," *J. Prod. Innovation Manage.*, vol. 21, pp. 4–14, 2004.

[8] G. M. Marakas and J. J. Elam, "Semantic structuring in analyst and representation of facts in requirements analysis," *Inf. Syst. Res.*, vol. 9, pp. 37–63, 1998.

[9] R. Agarwal and M. R. Tanniru, "Knowledge acquisition using structured interviewing: An empirical investigation," *J. Manage. Inf. Syst.*, vol. 7, pp. 123–141, 1990.

[10] D. Carrizo, O. Dieste, and N. Juristo, "Systematizing requirements elicitation technique selection," *Inf. Softw. Technol.*, vol. 56, pp. 644–669, Jun. 2014.

[11] A. Albayrak and J. Carver, "Investigation of individual factors impacting the effectiveness of requirements inspections: A replicated experiment," *Empirical Softw. Eng.*, vol. 19, pp. 241–266, Feb. 01, 2014.

[12] M. G. Pitts and G. J. Browne, "Stopping behavior of systems analysts during information requirements elicitation," *J. Manage. Inf. Syst.*, vol. 21, pp. 203–226, 2004.

[13] P. Loucopoulos and V. Karakostas, *Systems Requirements Engineering*. London, U.K.: McGraw-Hill, 1995.

[14] M. G. Christel and K. C. Kang. (1992). Issues in requirements elicitation [Online]. Available: Http://www. Sei. Cmu. edu/publications/documents/92. reports/92. Tr. 012. Html, vol. CMU/SEI-92-TR-012

[15] R. R. Young, "Recommended requirements gathering practices," *Crosstalk*, vol. 15, pp. 9–12, Apr. 2002.

[16] A. Niknafs and D. M. Berry, "An industrial case study of the impact of domain ignorance on the effectiveness of requirements idea generation during requirements elicitation," in *Proc. 21st IEEE Int. Requirements Eng. Conf.*, 2013, pp. 279–283.

[17] O. Dieste and N. Juristo, "Systematic review and aggregation of empirical studies on elicitation techniques," *IEEE Trans. Softw. Eng.*, vol. 37, no. 2, pp. 283–304, Mar./Apr. 2011.

[18] P. Vitharana, H. Jain, and F. M. Zahedi, "A knowledge based component/service repository to enhance analysts' domain knowledge for requirements analysis," *Inf. Manage.*, vol. 49, pp. 24–35, 1, 2012.

[19] C. A. McAllister, *Requirements Determination of Information Systems: User and Developer Perceptions of Factors Contributing to Misunderstandings*. Ann Arbor, MI, USA: ProQuest, 2006.

[20] A. Niknafs, "The impact of domain knowledge on the effectiveness of requirements engineering activities," Ph.D. dissertation, Univ. Waterloo, Waterloo, ON, Canada, 2014.

[21] G. J. Browne and M. B. Rogich, "An empirical investigation of user requirements elicitation: Comparing the effectiveness of prompting techniques," *J. Manage. Inf. Syst.*, vol. 17, pp. 223–249, 2001.

[22] B. C. Choi and A. W. Pak, "A catalog of biases in questionnaires," *Prev. Chronic Dis.*, vol. 2, p. A13, Jan., 2005.

[23] A. Aranda, O. Dieste, and N. Juristo, "Evidence of the presence of bias in subjective metrics: Analysis within a family of experiments." in *Proc. 18th Int. Conf. Eval. Assessment Softw. Eng.*, London, U.K., 2014, pp. 24–27.

[24] A. Jedlitschka, M. Ciolkowski, and D. Pfahl, "Reporting experiments in software engineering," in F. Shull, J. Singer and D. K. Sjøberg, Eds. Springer London, pp. 201–228, 2008.

[25] A. M. Burton, N. R. Shadbolt, G. Rugg, and A. P. Hedgecock, "Knowledge elicitation techniques in classification domains," in *Proc. 8th Eur. Conf. AI*, 1988, pp. 85–90.

[26] S. B. Yadav, R. R. Bravoco, A. T. Chatfield, and T. M. Rajkumar, "Comparison of analysis techniques for information requirements determination," *Commun. ACM*, vol. 31, pp. 1090–1097, 1988.

[27] A. Davis, *Software Requirements: Objects, Functions and States*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.

[28] E. D. Falkenberg, W. Hesse, P. Lindgreen, B. E. Nilsson, J. L. H. Oei, C. Rolland, R. K. Stamper, F. J. M. Van Assche, A. A. Verrijn-Stuart, and K. Voss, "FRISCO: A framework of information system concepts," The IFIP WG 8.1 Task Group FRISCO, 1996.

[29] C. A. Gunter, E. L. Gunter, M. Jackson, and P. Zave, "A reference model for requirements and specifications," *IEEE Softw.*, vol. 17, no. 3, pp. 37–43, May/June 2000.

[30] N. Juristo and A. M. Moreno, *Basics of Software Engineering Experimentation*. Norwell, MA, USA: Kluwer, 2001.

[31] D. I. K. Sjoberg, B. Anda, E. Arisholm, T. Dyba, M. Jorgensen, A. Karahasanovic, E. F. Koren, and M. Vokac, "Conducting realistic experiments in software engineering," in *Proc. Int. Symp. Empirical Softw. Eng.*, 2002, pp. 17–26.

[32] P. Runeson, "Using students as experiment subjects–an analysis on graduate and freshmen student data," in *Proc. 7th Int. Conf. Empirical Assessment Softw. Eng.*, 2003, pp. 95–102.

[33] M. Höst, B. Regnell, and C. Wohlin, "Using students as subjects—a comparative study of students and professionals in lead-time impact assessment," *Empirical Softw. Eng.*, vol. 5, pp. 201–214, Nov. 01, 2000.
[34] F. Exadaktylos, A. M. Espín, and P. Brañas-Garza, "Experimental subjects are not different," *Sci. Rep.*, vol. 3, Feb. 14, 2013, Doi:10.1038/srep01213
[35] R. B. Bausell and Y. F. Li, *Power Analysis for Experimental Research: A Practical Guide for the Biological, Medical and Social Sciences*. Cambridge, U.K.: Cambridge Univ. Press, 2002.
[36] T. D. Cook and D. T. Campbell, *Quasi-Experimentation: Design & Analysis Issues for Field Settings*. Skokie, IL, USA: Rand McNally, 1979.
[37] R. O. Kuehl, *Design of Experiments: Statistical Principles of Research Design and Analysis*. Belmont, CA, USA: Duxbury/Thomson Learning, 2000.
[38] B. W. J. Brown, "The crossover experiment for clinical trials," *Biometrics*, vol. 36, pp. 69–79, 1980.
[39] S. Senn, *Cross-Over Trials in Clinical Research*. New York, NY, USA: Wiley, 2002.
[40] L. S. Meyers, G. Gamst, and A. J. Guarino, *Applied Multivariate Research: Design and Interpretation*. Newbury Park, CA, USA: SAGE, 2012.
[41] S. K. Gupta, "Intention-to-treat concept: A review," *Perspect. Clin. Res.*, vol. 2, pp. 109–112, Jul., 2011.
[42] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. Hillsdale, NJ, USA: Lawrence Erlbaum Assoc., 1988.
[43] J. C. Carver, "Towards reporting guidelines for experimental replications: A proposal," in *Proc. 1st Int. Workshop Replication Empirical Softw. Eng. Res. SIGSOFT Softw. Eng. Notes*, May, 2010.
[44] M. Lewis-Beck, A. E. Bryman, and T. F. Liao, *The SAGE Encyclopedia of Social Science Research Methods*. Newbury Park, CA, USA: SAGE, 2003.
[45] A. H. Schoenfeld and D. J. Herrmann, "Problem perception and knowledge structure in expert and novice mathematical problem solvers," *J. Experimental Psychol.: Learning, Memory, Cognition*, vol. 5, pp. 484–494, 1982.
[46] G. Mehrotra, "Role of domain ignorance in software development," M.S. thesis, Univ. Waterloo, Waterloo, ON, Canada, 2011.
[47] D. Friedman and S. Sunder, *Experimental Methods: A Primer for Economists*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
[48] K. A. Ericsson, M. J. Prietula, and E. T. Cokely, "The making of an expert," *Harv. Bus. Rev.*, vol. 85, p. 114, 2007.
[49] C. Pacheco and I. Garcia, "A systematic literature review of stakeholder identification methods in requirements elicitation," *J. Syst. Softw.*, vol. 85, pp. 2171–2181, Sep. 2012.
[50] R. Razali and F. Anwar, "Selecting the right stakeholders for requirements elicitation: A systematic approach," *J. Theoretical Appl. Inf. Technol.*, vol. 33, pp. 250–257, 2011.
[51] V. M. Montori and G. H. Guyatt, "Intention-to-treat principle," *Can. Med. Assoc. J.*, vol. 165, pp. 1339–1341, Nov. 13, 2001.
[52] M. Jørgensen, B. Faugli, and T. Gruschke, "Characteristics of software engineers with optimistic predictions," *J. Syst. Softw.*, vol. 80, pp. 1472–1482, Sep. 2007.
[53] R. H. Hoyle, "Preface," in *Statistical Strategies for Small Sample Research*, R. H. Hoyle, Ed. Newbury Park, CA, USA: Sage, 1999, pp. v-vii.
[54] F. Richy, O. Ethgen, O. Bruyere, F. Deceulaer, and J. Reginster, "From sample size to effect-size: Small study effect investigation (SSEi)," *Internet J. Epidemiol.*, vol. 1, 2004, http://ispub.com/IJE/1/2/10397
[55] J. W. Graham and J. L. Schafer, "On the performance of multiple imputation for multivariate data with small sample size," in *Statistical Strategies for Small Sample Research*, R. H. Hoyle, Ed. Newbury Park, CA, USA: Sage, 1999, pp. v–29.
[56] L. V. Hedges and I. Olkin, *Statistical Methods for Meta-Analysis*. New York, NY, USA: Academic, 1985.