ExpTool: a Tool to Conduct, Package and Replicate Controlled Experiments in Software Engineering

Joao Pucci Neto, Lilian Passos Scatalon, Rogerio Eduardo Garcia, Ronaldo Celso Messias Correia, Celso Olivete Junior
Department of Mathematics and Computer Science
Faculty of Science and Technology
University State Paulista “Julio de Mesquita Filho”
Street Roberto Simonsen, 305 - CEP 19060-900 Presidente Prudente - SP, Brazil
E-mail: puccineto,lilian.scatalon@gmail.com, rogerio,ronaldo,olivete@fct.unesp.br

Abstract—Running multiples experiments in Software Engineering introduces the need of recording data as well as transferring knowledge across them, especially considering that several researchers are involved on replicating experiments. For that, experimental evaluations generate knowledge that must be registered into a so-called lab package. Researches have reported difficulties on sharing lab packages due to lack of standardization. In this paper we present a tool to support experimenters to conduct controlled experiments, packaging experimental data. In this first version, experimental data are kept into XML file organized based on an ontology proposed to controlled experiment. The tool support aims at organizing lab packages focusing on facilitating creating and sharing lab packages.

Keywords: Controlled Experiment, Lab Package, Experimental Software Engineering.

I. INTRODUCTION

Controlled experiments can be used to test and validate new methods, techniques, languages and tools. Only the results of isolated experiments are not sufficient - they are no enough to be considered trustworthy. The results of experimental studies can only be considered if they are consolidated into a significant body of knowledge for the community that operates in the area. Therefore, it is needed to replicate experimental studies to generate data and compose the body of knowledge [1]. According to [2], the execution of experiment in different contexts allows verifying if conclusions are valid for a wider population. And if a replication present different result, it is possible to analyze the reasons why such difference was obtained [3].

A replication by other experimenters requires reviewing the information from the original experiment in order to understand how it was designed, conducted and analyzed [4]. So, the information about original study must be available for that. All information (process, artifacts, procedures, results and conclusions) are stored into a so-called lab package [5]. Shull cites that researchers have faced some problems on interpreting and understanding lab packages, and consequently, transferring of knowledge between research groups. In addition, Mendonca [6] proposed a framework (FIRE - Framework for Improving the Replication of Experiments) that suggests the sharing of knowledge generated intra and inter groups, but they do not suggest how to organize lab packages. In a previous paper, Garcia [7] proposed an ontology to organized data from controlled experiments. In this context, the aim of this paper is to present a computational tool called ExpTool to assist the researcher in conducting a controlled experiment based on experimental process proposed by Wohlin [2] as well as the creation of a lab package using the ExpOntology. Using the ExpTool the lab package is created: a XML file containing all data from the study is generated according to ExpOntology. The tool maps a sequence of activities in a workflow that allows the experimenter to define their experiment and artifacts used.

In order to present the proposed tool, the paper is organized in the following: in Section 2 is discussed the importance of using controlled experiments, the packaging of data and the importance of knowledge transfer through replication of experiments; in Section 3 is presented the use of ExpTool on creating a lab package (for that, we use a case study for conducting the experiment showing the creation of lab package); in Section 4 we presented related works selected from literature; in Section 5 we presented our final remarks and future works.

II. CONTROLLED EXPERIMENTS

To study a technique (method, language or tool) in a controlled experiment, participants are selected from a population, which apply such technique under controlled conditions, according to what was defined by the responsible researchers in the experimental design. From the analysis, conclusions are drawn about the population from which the participants are considered representative.

There are sources of variation in several parts of this process that influence in the results [5], [8]. The study participants can be from diverse cultural environments or can be under a different set of conditions during execution [9], [6]. Thereby, conclusions keep limited by these factors.

In order to generalize conclusions in a more comprehensive way, these variations should be explored and dealt in replications. So the results can be confirmed or the influence of these variations can be identified and produce more relevant conclusions on the topic. Through replications based in a experiment that investigates some technique, the knowledge about it is consolidated. A body of knowledge in Software Engineering with the goal of support decision making related to software development require the accomplishment of families of experiments in different environments [10].
Undertaking a replication requires the access to information about all the study procedures. Such information are stored into the lab package, which registers the experiment documentation. However, Shull et al. [5] pointed out that the lab packages review present difficulties, since a static lab package can not accommodate relevant aspects of the experiment to the researcher that intends to replicate it.

Noticing knowledge transfer problems as barriers to the realization of replications, Mendonça et al. [6] proposed the FIRE (Framework for Improving the Replication of Experiments), which is composed by the two activities cycles illustrated in Figure 1. The internal cycle represents the execution of an isolated study. In the external cycle are the activities responsible for integrate the knowledge generated by the study in a common body, in order that its lab package can be effectively reviewed by eventual replicators.

Create/evolve package affect the external activity share knowledge. As in the other point that the circles intercept themselves, understand lab packages is crucial to set experiment goals for the replication.

Thus, the lab package represents the output of the internal cycle carrying the information about a study and also the input of the internal cycle of a possible replication, since researchers must review and understand the lab package of the original experiment. So, to share knowledge and aid a better understanding of lab packages (in the external cycle), the way that its information must be represented is very important, what is confirmed also by the activity standardize packages.

Wohlin et al. [2] have pointed out different sorts of primary studies: Survey, Case Study and Controlled Experiment. The the ExperOntology [7] focuses on Controlled Experiment. Its main concepts are highlighted throughout Experimentation Process, described next.

The Experimentation Process follows a sequence of phases [2]: Definition, Planning, Operation, Analysis and Packaging. In the Definition phase, hypotheses are clearly stated and the experiment goals are established. Based on the definition, in the Planning phase, an execution plan must be detailed, defining the execution environment, the subjects involved and their profile, the dependent and independent variables and their scales. At this stage it is important to discuss the validity of the expected results. These two initial phases are iterative, since it is possible to return to a previous phase or redo the current one.

The Operation phase is divided into three steps: Preparation, Execution and Data Validation. Preparation concerns to preparing the required material to run the experiment, such as data collection forms and training materials. The Execution must ensure that the experiment is conducted as planned. Finally, during Data Validation, replicators try to check the data collected for correctness. These three steps are also iterative. After Operation, the data collected is analyzed (Analysis). The Packaging phase is concerned to documentation, including experimental artifacts, procedures and results into a so-called lab package for future replications. Amaral et al. [12] suggest that such phase should be conducted in parallel throughout the experimentation process. These concepts highlighted in this section are mapped by Garcia et al. [7] into the conceptualization of the ontology in two refinement levels and axioms to formalize it. The first level comprises concepts like controlled experiment, replication, experiment validity and lab package. In the second level, the main aspect is the lab package ontology, which models the concepts that should be considered in the experiment packaging. Scatalon, Garcia and Correia [13] pointed out an approach to package controlled experiments using an evolutionary approach based on ontology based on Garcia et al. [7], but focusing only on creating lab packages – the proposed tool do not support conducting an experiment.

B. The Lab Package

In this section, we present the lab package elements in terms of concepts. The concepts that compose a lab package are presented throughout the experimentation process, highlighted in the following. At first, the initial hypothesis of a controlled experiment is established. It is composed by the object of study, in agreement with a purpose, under a quality focus, and in a specific context.

The Definition phase is the basis for the Planning phase and the initial hypothesis generates the hypotheses formalized. These hypotheses have null hypothesis and the alternative hypothesis, as attributes. From the hypothesis formalized, the experimenter defines the experiment variables – dependent and independent variables. During the planning phase s/he also defines the experiment objects: technologies to be studied (techniques, methods or tools) and artifacts (documents, tools or forms) to be used.
Each subject has his/her profile recorded to characterize his/her background. Capturing the subject background aims at identifying possible influence on results. For instance, previous knowledge about experiment objects or domain application might influence the results obtained. The subjects’ profile must be considered to create the experimental design, which is built combining experiment objects, independent variables and subjects, in agreement with the hypothesis under investigation. In addition, the subjects’ profile must be considered in analysis.

Based on the experimental design, an execution plan must be elaborated in order to describe the entire controlled environment to conduct the experiment. Such plan must consider the training activity, which comprises both theoretical and practical approaches for teaching the involved technology. The plan is obtained by establishing the tasks to be executed, their sequence and their period. During the execution, each task must have its initial and final time recorded, and differences between task planned and task performed must be considered as a threat to validity.

The main objective of Definition and Planning phases is to establish the experimental design, which must satisfy the requirements to the Analysis phase. Such phases culminate in the experimental design and in the execution plan, defining an environment as controlled as possible to test the hypothesis and minimize the threats to validity. Both of them are the core to guide the operation phase.

The data set gathered during the execution represents the concept results. The analysis of these results is based on hypotheses formalized and on experimental design focusing on dependent variables. Confirmatory analysis aims to test the hypotheses formalized.

From conducting an original experiment a lab package is generated. A replication uses a lab package from previous experiments as the basis for its motivation as well as for generating a new lab package. Both the original experiment and the replication have to be evaluated regarding to their validity. An original experiment is created by a designer, who has his/her profile related to the experiment as a parameter to define a possible threat to validity. In the same sense, a replicator has also his/her profile associated with the replication. It is important to highlight that both designer and replicator profiles might influence, negatively or positively, the conduction of the experiment/replication. The lack of experience, for instance, is a negative influence since it can be difficult to isolate the factors of risk when defining an experiment. Regarding the replication, the lack of experience can also influence the execution fidelity of the original experiment. On the other hand, the high experience is a positive influence since it minimizes the effort for defining the experiment and helps to analyze the lab package both to identify opportunities and combine results of different experimental treatments. So, the designer and replicator profiles must be taken into account during the analysis of the results as an influence on the experiment validity.

The validity evaluation is an issue to be addressed through all phases of the experimentation process. Wohlin et al. [2] pointed out that there are four types of threat to validity: (1) conclusion validity – refers to the relationship between the treatment and outcome; (2) internal validity – refers to the points that assure there is a causal relationship between the factors and the outcome; (3) construct validity – concerns with the relation between theory and observation; and (4) external validity – concerns with generalization. Each type of validity is constrained by threats. A threat to validity constrains the validity of an original experiment or a replication. However, when there are threats, they are identified in the lab package. The influences to any element that integrate a lab package (or the combination of them) cause a threat to validity.

The concepts presented must be organized into the lab package. One might find several papers suggesting guidelines on items to be [14], [15], [2], [16], [17], [18]. Table I summarizes guidelines for packaging, showing at high level what should be packaged – all highlighted items previously presented must be put into the suggested organization of lab package.

There is a large amount of information handled during the experimentation process. The lab package must keep data and decisions during the conception and execution of controlled studies. The packaging phase of a controlled study might be facilitated using a tool. Besides, the tool must be able to support not only the record of data, but also the workflow of controlled experiments. By integrating data collecting with workflow make possible to register data long the process. In the next section we present the proposed tool to support both conducting an experiment and collecting data from its execution.

III. ExpTool: a tool to support controlled experiments

ExpTool was developed using Java programming language as a web application. Servlets and JSP pages (JavaServer Page) were used for treatment and data presentation, as well as traditional Java classes as basic application. Assistive technologies such as jQuery, Ajax and some auxiliary library packages were also essential for the implementation. Its architecture is organized in four layers that exchange data among themselves. They were separated according to their level and function. As a web application, it was necessary to introduce features about security. For that, Sessions between the client (browser) and server are kept while a user is logged.

A. Functionalities to Experimenters

We present the features following the workflow. First, it is necessary to register some experimenter data (name, email and password), what is made by an administrator. The experimenter registered will receive an email to confirm his/her login and password. There is a specific interface to experimenter access their area in ExpTool. The experimenter might register an organization to be the experiment context by registering name, acronym and country. Also, each subject in the experiment must be registered (name, email, phone and link to a previously registered institution). Each subject receives an email (login and password) to access his/her area in ExpTool.

The experimenter can create new experiments and record data for the initial definition (Definition Phase): name, description, theme, technical area, type, domain and language. Also, according to the GQM (Goal, Question and Metric) [2], it
TABLE I: Overview of proposed guidelines for packaging, adapted from Jedlitschka [19]

<table>
<thead>
<tr>
<th>Author</th>
<th>controlled experiment</th>
<th>systematic review</th>
<th>controlled experiment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>type of study</strong></td>
<td>experimental research</td>
<td>experimental research</td>
<td>experimental research</td>
</tr>
<tr>
<td><strong>phases of the study</strong></td>
<td>packaging</td>
<td>all</td>
<td>all</td>
</tr>
<tr>
<td><strong>Structure proposed</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>abstract</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>introduction</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>method</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>procedure</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>results</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>discussion</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>references</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>appendices</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

(* ) indicates that the elements are implicitly required, although not explicitly mentioned.

is necessary to set goals, registering: object of study; purpose; focus; perspective; context factors.

The experimenter can create new experiments and record data for the initial definition (Definition Phase): name, description, theme, technical area, type, domain and language. Also, according the GQM (Goal, Question and Metric) [2], it is necessary to set goals, registering: object of study; purpose; focus; perspective; context factors. The Figure 2 shows a screenshot of ExpTool interface during the GQM items definition.

To create the plan, the experimenter has five steps, organized in ExpTool as follow:

1) Planning: consists in five items
   • context: it is defined whether type is in-vitro or in-vivo; whether team is made by students or professionals; whether the problem is real (from industry) or fictitious (classroom); whether domain is specific or general.
   • Hypotheses : the formulation of at least one null hypothesis with a question and the hypothesis itself and the formulation of one or more alternative hypotheses is necessary.
   • Variables: dependent and independent variables are defined according to the hypotheses previously registered.
   • Threats to Validity: set of threats to validity (internal, external and construction threats) are registered.
   • Metric: it is defined metrics and their description shown how data must be collected, registered and analyzed.

2) Instrumentation: consists in three items:
   • Technique: name, description and whether is a method, a technique or a tool under evaluation. Also, it includes description and files of the external artifacts (for example, if a tool is under study, version, files and data about how to install must be kept).
   • Artifact: name, description (questionnaire, document requirement, source code), as well as any included files.
   • Technique versus Artifact: to establish the application of techniques with the artifacts included in the experiment.

3) Participants: there are two items:
   • Selection: select subjects to participate of the study.
   • Characterization: a questionnaire about participants is created to define their profile.

4) Experimental Design: it is necessary to associate each participant to a specific task (apply technique, method or tool) and artifact. For that, it is possible to create groups of participants, and assign or remove participants. So, treatments are associated to groups and the respective artifacts. A schedule about application of treatments by participants must be defined (the experimenter must select the group that will perform a task using the artifact in order to perform the treatment).

5) Planning Conclusion: this step consists in close the planning. After that, it will not be possible to modify the planning.

During the operation phase is not possible to modify any parameter defined during previous phases. At this point, each participant can access his/her area and execute the task designated and record the data obtained during the execution. The features for participants are presented in next section.

During Analysis and Interpretation phase the experimenter evaluates the data obtained from the previous phase, verifying its validity, and makes the relationship between the results and the assumptions, as well as registering the results obtained, with artifacts and conclusions.

Finally, the packaging is done as discussed. The Figure 3 shows the options to generate the lab package: into a XML or a ZIP file. The Figure 4 shows the lab package created (XML
Fig. 2: Screenshot of GQM interface – Definition Phase

Fig. 3: Screenshot of Creation of lab package

Fig. 4: Screenshot with Lab package generated

and related artifacts). The Figure 5 shows partially the content of XML file – it is possible to observe the GQM elements instantiated.

Fig. 5: Screenshot with partial view showing the lab package generated by the ExpTool into XML file

B. Functionalities to Subjects

Each subject receive by e-mail a confirmation of registration and a login and password. S/he can access the system on access to your login and password by selecting the option to access as a participant interface. When the subject open an experiment associated, should answer the questionnaire to characterize the participant before having access to the activities of the study. The questions are presented and, when answered, the access to the list of activities (tasks) is presented.

Regarding the control flow of activities, there are some important points to consider. According to the schedule defined and the subject action, a status is associated to activities. The activity states are:

- Not available: the activity is not approved for implementation, since its start time (defined by experimenter) has not been reached;
- Available and not started: it is possible to start the activity, but the subject did not do it yet;
- Available and Initiated: it was viewed by the subject, but s/he has not sent the artifact resulting from treatment yet (task not completed);
- Started and not completed: the subject initiated the activity and the deadline for its completion was reached; Not initiated and out of time: even the activity was
In order to establish and compute the timing, the clock of the server where the application is hosted is used. The activity can only be opened when it is available or has already started. In Figure 6 is presented a screenshot of the interface presented to a subject during a task associated – it is possible to observe the time still available to the task at bottom region.

The experimenter set a deadline to each task and whether the treatment (in hours). The time to begins at the point when participant view the activity for the first time. A regressive counter to the participant execute a task is presented with the remaining time. When deadline is reached, the tool prevents the transmission of results to the server. If for some reason the subject performed the activity, but was unable to load the files on time, s/he should contact the experimenter responsible in charge and the experimenter must assess the situation and decide whether use or not the data generated by subject.

IV. RELATED WORKS

Some applications have been developed in order to support the controlled experiments. We presented two similar works, pointing out their features and contrainst that motivated our proposed tool. The first one is the eSEE (Experimental Software Engineering Enviroment): a computational infrastructure based on web services that aims to analyze data repository about experimental studies and recover decisions about their design. Lopes and Travassos pointed out that eSEE have been used to evolve a glossary of terms concerned Experimental Software Engineering [21].

Another tool to suppor experimental studies is PontoLab [13]. This tools focuses on instantiate concepts of ExperOntology [7]. The interface provides tabs to edit the information related to the experimental process [2], according to the ontology concepts. In the tab corresponding to the definition phase, are inserted the general directions of the experiment, represented by the concepts object of study, quality focus, purpose and context. Next, this directions are formalized in hypotheses, that express the cause-effect relationship to be analyzed through the experiment execution. From the hypotheses formulation, in the tab of planning, are defined the dependent and independent variables, with the respective treatment that can be assumed by each. Also, are selected the subjects of the experiment. Once established the independent variables and the participants, it is possible to define the study schedule, that is, the experimental project: assign to each subject a set of treatments, each of a independent variable. These assignments, represented in the ontology by the predicate Design, generate tasks, inserted in the execution plan in the next tab, of the operation phase, in which is also possible to register about the results obtained with the execution. Finally, in the last tab, of analysis phase, it is registered about statistical tests of confirmatory analysis of the hypotheses and observations that emerged from a exploratory analysis, leading to unforeseen relations in the experimental project. It is important to note that the PontoLab tool focuses on packaging an experiment using the ontology, but do not support the execution of controlled experiments.
V. Final Remarks

Experimental Software Engineering attempts to evaluate and measure the performance of models and techniques in practical contexts, in order to establish a body of knowledge base to support decision-making. Also, results from multiples experiments can be used in order to support new ideas and theories. To build a body of knowledge, it is necessary to conduct controlled experiments and their replications. Also, it helps to generalize the results packaged and stored into lab packages. To replicate studies. It is necessary review the lab package from original study. Difficulties rise from the lack of standard to organize data stored into lab packages and, consequently, make difficult to understand its contents.

In this context, we proposed a tool to support the conduct of controlled experiments, presented in this paper. ExpTool provides support to the experimenter: each task defined in workflow to execute an experiment is supported, including the packaging. During each task, data collected are registered. It is possible to keep record (and links) to external files used during the experiment, such as tool and artifacts. After the execution, data analysis and results are registered using external files, if necessary – it is allowed to keep spreadsheet and graphics in separate files. Threats to validity, conclusions and further works can be registered in text format. As a last step, the lab package is instantiated into a XML file, organized as suggested by Jedlitschka [19].

In addition, the ExpTool allows experimenter to use an XML file previously created as input to another experiment. In this case, the XML file represents a lab package and the experimenter would use it to replicate the experiment. The ExpTool allows both replication (repetition of original experiment) and replication with variation – the experimenter is allowed to modify the original plan (different parameters or artifacts, for example) to execute the experiment in order to create a family of studies. As main contribution, we point out: 1) the organization of lab package using the workflow steps and related concepts; 2) facilitate exchanging the lab package among experimenters (since adopting ExpTool, experimenters are allowed to exchange their lab package); 3) an lab package can be used as input to an experiment, facilitating the integration among experiments (and experimenters).

As further work, we are developing a new feature: to instantiate lab packages using an ontology. For that, the concepts will be kept according to ExpOntology [7]. Also, we are aware that exchanging lab package is difficult to validate. So, we intend to create a repository of controlled experiments and to make them available to other experimenters, as well as the ExpTool.

References


