Abstract - This work presents an experimental study of an idea related to the automatic generation of performance and stress testing by reusing functional testing. The idea was implemented in a tool named FERRARE GT. This tool is able to generate both test scripts as well as the data required for their execution. In this study we verified that the use of the method can generate benefits related to cost reduction, from the reduction of test effort and, at the same time, benefits related to test quality, from the improvement of the test relevance for the software development.

Keywords - software testing; data generation; non-functional requirements; experimental study.

I. INTRODUCTION

Testing is a critical element in the software quality control and represents the final review of the analysis, design and implementation. However, testing is usually not performed as it should. A fundamental factor that contributes to this situation is the activity cost, which can get as high as 50% of the total project cost [2].

A test category performed by many organizations is functional testing. It aims at verifying the software behavior [6]. There are various other test objectives, as, for instance, performance and stress testing. They are much less often executed than the functional test and have different purposes. The performance test aims at validating the performance requirements, as, for instance, the response time in a specific context, like the access of 100 users in a local network environment. Stress testing is similar to performance testing, however, the execution context is elevated to levels above the average, verifying the system operation in such cases and certifying that no unusual behavior occurs.

In general many organizations that develop Web systems perform functional tests. However, few execute performance and stress tests before launching the system, even though they are just as important for the Web environment. It is fundamental the development of mechanisms which motivates the execution of such tests so needed by Web systems.

From this scenario, it was noticed that the creation of a mechanism, which aids the development of performance and stress testing, from any common artifact to the software development, could result in the cost reduction. Because they are so well known and have a big portion of the necessary information, the functional test was chosen for the input of such automation. From that it was developed a tool, named FERRARE [8], with such an objective. The initial prototype tool enabled the generation of performance and stress testing scripts from functional testing scripts. However, limitations in the tool, which prevented its use in an industrial environment, were discovered.

This paper describes the extensions performed in FERRARE to allow its use in an industrial environment related to software development, as well as an experimental study performed in order to evaluate the benefits related to incorporate the tool in a software development environment.

There are related works that propose the generation of performance tests based on models describing the system under test [4,5,9]. By the other hand, Bertolini et al [1] propose four black box test techniques in order to crash the system, by using a special kind of functional tests. It is important to emphasize that FERRARE does not use models. It generates performance and stress tests scripts by reusing functional tests scripts. Besides, FERRARE can infer the data related to execute a functional test and generate suitable data to execute several tests instances, isolating each one from the others.

This paper is organized as follows: Section II presents FERRARE GT; Section III presents the experimental study; Section IV presents a discussion about the idea and the study results; Section V concludes the paper and presents directions for future works.

II. FERRARE GT

FERRARE is a tool developed for the generation of performance and stress testing scripts from functional testing scripts. It is divided into two modules: Extractor and Generator.

FERRARE was conceived to work with any functional testing tool and any performance and stress testing tool, as long as the extractor and the generator for the desired tools are created, as it is going to be discussed later on. Nowadays, FERRARE works with the functional testing tools Selenium IDE and Canoo Web Test, besides performance and stress testing tools Apache JMeter and WebLoad. This means that it generates performance and stress testing to be executed on JMeter or the WebLoad, from functional testing created with Selenium IDE or Canoo WebTest (input tools) [8]. A sketch of the functioning of FERRARE can be seen in Figure 1.

The Extractor module is responsible for the extraction of the information inside the functional test script. This included the identification of the actions related to the test (test procedure) and of the input data, expected outputs and other
test conditions (test case). The extraction generates an abstract representation of the functional test, independent of technology.

The Generator module is responsible for the generation of performance testing based on the information supplied by the Extractor. This generation involves the specification of different parameters such as the quantity of concurrent users, time limits and number of machines used for test execution.

**Figure 1.** FERRARE GT overview.

FERRARE generates performance and stress testing from the creation of “copies” from functional testing, taking in consideration the restrictions associated to the inputs used in this test. If a functional test that performs a book register in an application is used, FERRARE can generate 100 “copies” of this test, in the format required by performance and stress testing, respecting the characteristics of the fields, as an obligatoriness, sizes and formats. It is important to emphasize that the tool does not perform a simple “copy”, since several other actions are also executed, in order to allow its concurrent execution [8].

The initial version of FERRARE did not generate the necessary data for the execution of performance and stress testing. Because of that, its automation level was very limited. This originated another project, with the goal of generating data for performance and stress testing, respecting the characteristics of the fields, as an obligatoriness, sizes and formats. It is important to emphasize that the tool does not perform a simple “copy”, since several other actions are also executed, in order to allow its concurrent execution [8].

It is important to emphasize that this approach eliminates the need of having knowledge of all the contraints related to the data model and the application business rules. Because the data are replicated from an instance of the database able to execute a functional test, the replicas should also keep the same feature. This approach is innovative, because it reduces the complexity for the data generation. The innovation is precisely in using a functional test, and the state of the database before its execution, for such replication.

### III. Experimental Study

#### A. Goal Definition

An experimental study with the goal of evaluating the use of FERRARE GT in a scenario of software development was performed. The purpose of the study was to evaluate, regarding effort and quality, from the point of view of the researcher, in the context of students of Computer Science, the feasibility of FERRARE GT in order to automate the data generation and performance and stress tests development for a Web system.

The experiment was executed in a controlled environment (in vitro) with the participation of students from Software Engineering Class from the Computer Science Course from UFPI (Federal University of Piauí).

The subjects had to create performance tests for a Web system on book loaning, named in this work BibSystem. The goal of this system is to allow students to perform book loans available in the library. The system users can authenticate themselves in the system, perform a loan, search for a book and return it. The Login (authentication) function was explored in the observation sections.

#### B. Planning

The experimental study was planned to be executed by students that attended the classes of Software Engineering I and II offered in the first semester of 2010. The Software Engineering I class is offered in the fifth semester of the course and the Software Engineering II class in the sixth, from a total of eight semesters.

The subjects did not have any experience at all in the use of functional testing tools neither in the use of performance and stress testing tools. A verification form was applied to certify this. Therefore, no planning at all was done regarding the type of grouping based on the profile of the subjects [10].

For the execution of the experiment it was planned the use of the tools Selenium (functional testing) and JMeter (performance and stress testing). Both tools were selected because of their big acceptance within the software development industry.

The goal of the study was to evaluate if the reuse of functional testing, for the generation of performance and stress testing, from the support of FERRARE GT is more effective than the development of the same tests in a direct way on JMeter tool. However, generating performance testing without the support of FERRARE GT does not imply only in creating
the test script for the selected tool (JMeter), as well as generating the data required for its execution. This generation is usually done using programs that execute data insertion commands in databases. There are other alternatives, but this was the one used for the study. It is important to emphasize that FERRARE GT does not only support the creation of test scripts, but also generates the data required for its execution.

As mentioned before, the use cases used in the study were Login and Loan. Performance testing for the Login function consisted of executing 100 concurrent authentications, making use of different users and verifying if this happened in up to 5s. Performance testing for Loan consisted of executing 100 different book loans, by different users and verifying if this happened in up to 8s.

Because of that, the main question related with the study was: does the use of FERRARE GT generate a reduction in the required effort for the creation of the performance testing, including the generation of the data necessary for the execution of the tests, when compared with the creation of the same tests using only JMeter? The null hypothesis, related to this question is: there is no difference in terms of effort, measured in minutes, to create tests and to generate data with or without the support of the tool, that is, H0: TestEffort(FERRARE_GT) = TestEffort(JMeter). The alternative hypothesis is that the effort applied in the test, with the support of the method is smaller than the effort applied without the use of the tool, that is, H1: TestEffort(FERRARE_GT) < TestEffort(JMeter).

The experimental sketch used was planned taking in consideration the possible threats to its validity. The Figure 2 summarizes the experimental sketch used. The highlighted parts identify the activities whose time spent by the subjects was registered. The other ones represent the training activities, which respect the times showed in the picture. As it can be visualized in the picture, all the participants had contact with both tools, but in different moments and performing exactly the same tasks. This allowed one group to act as a control of the other. The experiment was divided in two phases. In the first phase all the volunteers generated the data required to execute the 100 book loans simultaneously in the BibSystem. This included the generation of users, books and copies.

In the second phase, the subjects from the Group 1 (G1) should create tests using JMeter and only later create the same tests using FERRARE GT. The subjects from the Group 2 (G2) should begin creating the tests with FERRARE GT and later create tests with JMeter. The attribution to the groups was planned to be random. The selection of the participants was planned based on convenience; that is why the study is considered a quasi-experiment [10].

It was planned that all the tests created by the subjects would be verified by the paper authors, to certify their quality. This means that the difference between them would be only the way used to create them: either using FERRARE GT or JMeter itself. Each submission of the test generated a verification to certify its quality. If the test was not suitable, the registered errors would be highlighted and the subject should proceed with its correction.

The experimental study was planned to reduce threats related to its Internal Validity and External Validity, which are the most important ones related to the studies in the area of Software Engineering [10].

The internal validity defines if the relationship observed between the treatment and the result is causational and not an influence of other factors which are not controlled. The experimental design used reduces the risk of having a bias, since it was planned that every subject would use both treatments, but in different moments. It was also planned the use of the treatments in an alternated order, to evaluate if the execution order could influence the results.

![Figure 2. Experimental study scheduling.](image)

The chosen experimental design, where all the subjects used both tools allowing one group to act as the control of the other, validates the conclusion obtained, in the same time that it reduces any threat related to the competitive behavior and to the compensatory behavior [10].

The external validity defines the conditions which limit the ability to generalize the results of an experiment for the industrial practice. The subjects, students from the 5th and 6th semesters of the course, find themselves in the final phase of under graduation, having similar skills of a professional with little experience. The performed trainings, along with the fixation exercise contributed to a good formation of the subjects in the used tools. The BibSystem, although small, had the characteristics commonly exists in Web information systems. Thus, it is believed that the conclusions obtained in the study can be extended to other systems, with the usual size and used by professionals with little experience, without losses in the observed results.

C. Operation

Before the beginning of the study activities it was performed a brief presentation related to the activities that would be executed, but the subjects did not have knowledge of the hypotheses that were being tested. It also guaranteed the anonymity of the students explaining even how the data collected would be used.

As mentioned before, the experiment was divided in two phases. In the first phase, all the subjects had training in software testing and in the use of Hibernate¹ and Java to generate data in MySQL database. In the second phase, the subjects had training in the tools that would be used and were asked to create performance tests for the BibSystem.

During the first phase a general vision about software testing was presented to the subjects. The training about tests lasted around 2h and focused on presenting the relevance of the tests, as well as the basic concepts related, the main techniques and existing objectives.
Afterwards, everybody participated in a training about the use of the Java language combined with the Hibernate framework for the generation of data in a MySQL database. The Java language was chosen because it is the one better dominated by the subjects. The Hibernate framework was selected because of the easiness it offers when working with database in the Java language. The MySQL database was used because of its broad use in the academic field and especially in the field where the study took place.

After the training, all the subjects created programs in Java that generated and stored in a MySQL database the required data for the execution of 100 concurrent loans in the BibSystem. It is emphasized that the generated data had to follow the database structure and constraints. The individual time spent by each subject in this activity was registered, since it was executed automatically with the support of FERRARE GT.

During the second phase, the study subjects had to create and execute performance tests for the Loan function. Firstly a training section was performed, lasting about 2 hours, about the tool that would be used. After that the tests were created and executed by the subjects. The Group 1 subjects began using only JMeter in the creation of the tests and used the data generated during the phase one to allow its execution. Afterwards, they used FERRARE GT to create tests for the same function. The Group 2 subjects executed the activities in a reverse path, beginning with FERRARE GT and only later doing the same activity using JMeter.

Twenty-one volunteers participated in the study, from whom 15 concluded all the planned activities. Thus, only the data from these 15 subjects were considered during the analysis and interpretation. This happened because the tests generated by some of them did not reach the quality limit specified. Because of that, their results were not considered concluded.

D. Analysis and Interpretation

Figure 3 presents the data collected from the study. Figure 4 presents the data in the form of a bar chart. Analyzing such data it is perceived that the subjects that used FERRARE GT dedicated an effort considerably smaller for creating performance tests.

The results certify that the idea proposed in FERRARE GT, that reusing functional testing for automation of performance testing is a good alternative. The effort for creating performance testing, as well as the preparation of the environment for its execution, mainly related to the generation of data, can be considerably reduced. This was noted analyzing all the subjects (that is, independent of group) as well as the groups individually. In both cases the gain was expressive and confirmed through student t test.

Group 1 subjects began the study using JMeter directly, while Group 2 began through the use of FERRARE GT. It is possible to notice that the time to generate a test with the use of FERRARE GT by Group 2 was expressively bigger than the time registered by Group 1 for the same task. This indicates that the learning obtained by utilizing JMeter beforehand favoured the use of FERRARE GT. This was expected, once the tools are similar and have similar goals. The knowledge gained with the use of one can influence the use of the other.

![Figure 3. Study data summary.](image)

![Figure 4. Bar chart showing the study results.](image)

Analyzing the threats to the study, it can be noticed that it does not seem to have been any underlying factor that have interfered in the study. The results reflect the use of treatments and not uncontrolled factors.

Since the subjects had to automate tests for the same part of the system, using both FERRARE GT and JMeter, the problem of instrumentation was not noticed, since there is no difference in the problem used in the study.

Nothing related to the study indicated that there were any threats related to history, that is, it was not identified any external effects that could influence directly in the results differently from what was mentioned above. Relating to maturing, it is believed that the subjects improved with the experience, since they are students and are submitted to new contents, but nothing that could influence directly the result of the developed study.

IV. DISCUSSION

In this article, an experimental study to evaluate the impact of the reuse of functional testing for the generation of performance testing was presented. This was enabled by the tool FERRARE GT, which implements such an idea. It was
possible to verify that this approach is very promising. The effort to generate performance testing with support of the tool FERRARE GT is smaller than the effort to generate the same test without the support of the aforementioned tool.

Performance testing requires data for its execution. The required effort for such preparation can be an inhibiting factor in its use in the industrial environment. The example used in this work illustrates this well: to test the performance of the Loan function of the BibSystem it is necessary to have 100 users and 100 book copies available. Note that this is a fairly simple example and even so it demands a considerable effort. Any tool that helps reducing such effort is contributing to a systematization of performance testing, which is still seldom used by organizations.

The idea explored in this work was based on the reuse of functional testing. Since functional testing indicates the expected inputs and outputs to evaluate a behavior, we can execute such instance various times concurrently to evaluate the performance of this function. However, it is necessary to create copies of the required data, so that each test uses its own data and does not interfere in the execution of the other. The study certified the viability of the proposal, having obtained surprising results to the continuity of the work.

Reusing functional testing to generate performance and stress testing can still bring indirect gains not measured in the study: bigger diffusion of the use of functional testing, because now they have a more important role in the development process, and a bigger quality of the generated tests, since the saved effort can be used in nobler activities.

V. CONCLUSION AND FUTURE WORKS

In this work an experiment performed in order to validate the applicability of a tool for the generation of data and performance and stress testing by reusing functional testing, named FERRARE GT, was described. The basic idea is to reuse the existing information from functional tests, in such a way that both the performance and stress testing and the required data for its execution are automatically generated.

Based on the study performed, it was noticed that the use of FERRARE GT in an organization that already performs functional testing can bring a reduction in the effort required to create performance tests and to generate the required data for its execution. This facilitates the systematic use of performance and stress testing by organizations, since the required effort for its use is reduced. Also, the relevance of the functional test increases, what is benefic to the organization, which will be able to give more importance to something that is already considered fundamental by most industries.

As a future work it is intended to increase the experimental study. Also, it is intended to use the tool developed in an industrial environment. FERRARE GT also has many possibilities of improvement, which will be able to allow its use in a simpler and more effective way, raising the associated gains.

VI. ACKNOWLEDGEMENTS

This work was supported by grants from UFPI and CNPq (560128/2010-0). Infoway Technology provided some software products for testing and IT professionals to discuss the directions.

REFERENCES