# Survey on Workload Techniques for Load, Performance and Stress Tests

Francisco Nauber Bernardo Gois,Pedro Porfírio Muniz de Farias,André Luís Vasconcelos Coelho, Thiago Monteiro Barbosa[a,b,b,a]

[a]*Servio Federal de Processamento de Dados, Avenida Pontes Vieria ,832, Fortaleza, Ceará 60130-240*
[b]*Universidade de Fortaleza, Avenida Pontes Vieria ,832, Fortaleza, Ceará 60130-240*

**Abstract**

Many software must respond to thousands or millions of concurrent requests. These systems must be properly tested to ensure that they can function correctly under the expected load. Load, Performance and Stress Evolutionary testing aims to find test scenarios which produce execution times violating the timing constraints specified. The purpose of this paper is propose the use of a approach using hybrid metaheuristc in load, performance and stress test models using Genetic Algorithms, Simulated Annealing and Tabu Search Algorithms. A tool named IAdapter , a JMeter Plugin to perform evolutionary load, performance or stress tests, was developed. Two experiments were conducted to validate the proposed approach. The first experiment has been applied in an emulated component and the second experiment has been applied in an installed Moodle application. In both experiments, the use of a hybrid metaheurisct has obtained better fitnesse values.

*Keywords:* Evolutionary Testing, Tabu Search, Hybrid Metaheuristics

## 1. Planning and Conducting of Research

In this section, the steps of planning and review of driving are presented in details. The first part of the study was a systematic review.

A systematic review is a process of assessment of all available research related to a research question or subject of interest. The planning of the systematic review was carried out from the protocol defined by Biolchini [1] [2]. Systematic literature review (SLR) is a approach to conducting survey on a research topic. It aims at producing an ”engineering” approach with a well-defined methodology so that different investigators can produce survey results effectively and reliably [3].

Planning is the starting point for the review, whose main points are the definition of one or more research questions. The activities to conduct the research includes formulate Research Questions, The Paper idenfification process (selection of sources to search, search strategies and the use of keywords).

A key activity in the planning of a typical SLR project is to formulate a set of Research Questions (RQs) before identifying, reading, and analyzing articles of the topic [4].

### *1.1. Research Questions*

The work aims to answer two research questions:

First Research Question : How to define a suitable workload for Load Testing to generate realistic loads models (Realistic Workloads)?

Second Research Question : How to set a suitable workload for Load Testing to fault-inducing loads ?

Third Research Question: In which fora is research on load, performance and stress testing published?

Fourth Research Question: Which topics for load, performance and stress testing have been investigated and to what extent?

Fifth Research Question: What types of research are represented and to what extent?

### *1.2. Search strategy*

In this section,we present the search strategy of our paper. For each question, keywords were chosen and used in a search strategy. The search strategy for the selection of studies was carried out through search in repositories (ACM, Springer, IEEE, Google Scholar, Science Direct, Mendeley ) , language ( Portuguese and English) and the keywords defined. Using the results, new keywords have been included, feeding back the process. The research strategy included these two practices:

1. Identification of other words and synonyms for terms used in the research questions. This prac-

tice is used for minimize the effect of differences in terminologies;
2. The keywords and their possible combinations and synonyms were submitted in the selected repositories search engines;
3. Among the results, were excluded studies not related to load, performance and stress tests.

We used the following search terms:

- Stress Testing: Search-based Testing, Genetic Algorithms, Stress Testing, Test Tools, Test Automation, Empirical Analysis, Denial of Service, Ramp-Up time, Think Timer, Response Time, Bandwidth Throttle, Dynamic Stress Testing, Evolutionary, Heuristic, Search-Based, Metaheuristic. optimization, genetic algorithms, genetic programming.
- Performance Testing: Performance Testing, Web-based Systems, Software Testing, Model-Based Testing, Software Product Line, Regression Testing, Test Failure Prediction, Genetic Metric Selection.
- Load Testing: Markov chain, Automatic Test Case Generation Algorithms, Domain-based reliability measure, Fault detection, Load Test suites, load testing, Reliability, Resource allocation mechanisms, Software testing, System degradation.

¡¡¡¡¡¡¡ HEAD

### *1.3. Study selection procedure*

The selected studies were filtered by one researcher that used the following inclusion or exclusion criteria:

- Include: The researcher is sure that the paper is in scope and that it was properly validated using empirical methods.
- Exclude: The researcher is sure that the paper is out of scope or that the validation was insufficient.
- Uncertain: The researcher is not sure whether the paper fulfills either the inclusion or exclusion criteria above.

### *1.4. Classification scheme*

The classification used by the study is a structure of empirical studies on load,. The scheme consists of six facets, namely quantification approach, abstraction, context, evaluation, research method, and measurement purpose (see Fig. 1). Classification schemes/-taxonomies are rated based on a set of quality attributes. A good taxonomy/classification is: 1. Orthogonality: There are clear boundaries between categories, which makes it easy to classify. 2. Defined based on existing literature: The taxonomy/classification is created based on an exhaustive analysis of existing literature in the field. 3. Based on the terminology used in literature: The taxonomy uses terms that are used in existing literature. 4. Complete: No categories are missing, so that existing articles can be classified. 5. Accepted: The community accepts and knows the classification/ taxonomy.

In order to classify each paper selected in accordance with the characteristics of the survey,

### *1.5. Data Extraction and Mapping of Studies*

The relevant articles are sorted into classification scheme, i.e., the actual data extraction takes place. As shown in Figure the classification scheme evolves while doing the data extraction, like adding new categories or merging and splitting existing categories.

## 2. Load, Performance and Stress Workload: A Brief Introducion

The term Workload represents the size of the demand that will be imposed on the application under test in an execution. The metric unit used for define a Workload is dependent on the application domain, such as the length of the video in a transcoding application of multimedia files or the size of the input files to a file compression application [5] [6] [7].

Workload is also defined by the distribution of load between the identified transactions at a given time. Workload helps us study the system behavior identified in several load model. Workload model can be designed for verify predictability, repeatability and scalability of a system [5] [6].

Workload modeling is the try to create a simple and general model, which can then be used to generate synthetic workloads. The goal is typically to be able to create workloads that can be used in performance evaluation studies. Sometimes, the synthetic workload is supposed to be similar to those that occur in practice on real systems [5] [6].

The workload model is intrinsically linked with the kind of test applied. There are three main tests where these models are usually used:

### *2.1. Performance Testing*

The Performance Test aims at verifying a specified system performance. This kind of test is executed by

simulating hundreds or more simultaneous users over a defined time interval [8]. The purpose of this test is to demonstrate that the system reaches its performance objectives [9]. Other objectives of the performance tests are: Evaluate the adequacy of current capacity

### 2.2. Load Testing

Load Tests are a kind of test where the system is evaluated in pre-defined load levels [8]. The aim of this test is to reach the performance targets for availability, concurrency, throughput and response time of the system. Load Test is the closest test to real application use [6].

### 2.3. Stress Testing

Stress test is a kind of test that verifies the system behaviour against heavy workloads [9]. The Stress Testing is executed to evaluate a system beyond its limits. It's used to validate system response in activity peaks and verify if the system is able from recover from these conditions. Stress Tests differs from other kinds of testing because the system is executed on or beyond its breakpoints. The stress test causes the application or the supporting infrastructure to fail [8] [6].

There are two kinds of Workload models: descriptive and generative. The difference is that descriptive models just try to mimic the phenomena observed in the workload, whereas generative models try to emulate the process that generated the workload in the first place [8].

### 2.4. Descriptive Model

On descriptive models, one finds different levels of abstraction on one hand, and different levels of faithfulness to the original data on the other hand. The most strictly faithful models try to mimic the data directly using statistical distribution of data. The most common strategy used in descriptive modeling is to create a statistical model of an observed workload (Fig. 1). This model is applied to all the workload attributes, e.g. computation, memory usage, I/O behavior, communication, etc [8]. The Fig. 1 shows a simplified workflow of a descriptive model. The workflow has six phases. In first phase, the user uses the system in the production environment. In second phase, the tester collects user's data, like logs, clicks and preferences, in the system . The third phase consists in developing a model to emulate the user's behaviour. The fourth phase is made up of the execution of the test, emulation of the user's behaviour and log's gathering.

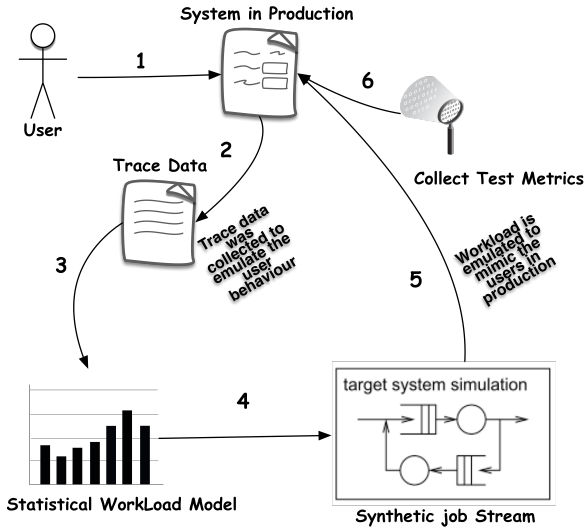


Figure 1: Workload modeling based on statistical data [8]

### 2.5. Generative Model

Generative models are indirect, in the sense that they do not model the statistical distributions. Instead, they describes how users will behave and when they generate the workload. An important benefit of the generative approach is that it facilitates manipulations of the workload. It is often desirable to be able to change the workload conditions as part of the evaluation. Descriptive models do not offer any option regarding how to do so. But with generative models, we can modify the workload-generation process to fit the desired conditions [8]. The difference between the workflows of descriptive and generative models is that user behavior is not collected from logs, but simulated from a model that can receive feedback from the test execution (Fig. 2).

## 3. Load, Performance and Stress Cloud Testing

Cloud computing alters the way of obtaining computing resources, managing and delivering software, technologies and solutions. Meanwhile, cloud computing also brings new issues, challenges, and needs in cloud-based application testing and evaluation. In the past decades, there were numerous published technical papers focusing on scalability analysis and performance evaluation.

The presented research work found one systematic review on cloud software testing:

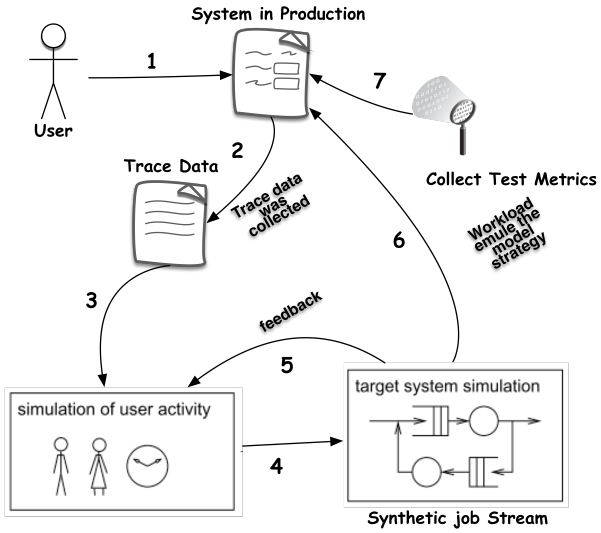


Figure 2: Workload modeling based on Generative Model [8]

- 5W+1H pattern: A perspective of systematic mapping studies and a case study on cloud software testing [10].

The Figure 3 shows a comparison between the researches found in the presented research work. The x axis represents the kind of cloud used ( SaaS, PaaS or IaaS ) and the y axis presents the type of workload strategy used by each research (Generativive or Descriptive Workload). The Figure also divides the researches by the type technique used in each paper.

### *3.1. Papers found in the presented research work*

This subsection presents details about the papers found in the presented research work.

Gao et. al. proposes a new formal graphic models and metrics to test SaaS performance and analyze system scalability in clouds. This paper presents a analytic models in a radar chart graphic format to evaluation for system performance and scalability of SaaS applications in a cloud. The paper consider three types of system loads[11].

- The communication traffic load: The amount of incoming and outgoing communication messages and transactions in a given time unit during system performance and evaluation.
- The system user access load during:The number of concurrent users who access the system in a given time unit.

| | Descriptible WorkLoad | Generative WorkLoad | |
|---|---|---|---|
| Diagram Model | | Gao. et. al. 2011 | SaaS |
| Record and Playback User Actions Model | | Snellman et. al. 2011 | |
| Statistic Model | Tsai et. al. 2011 Jackson et. al. 2010 | | |
| Parameter File | | Patil et. al. 2011 | PaaS |
| Test Automation Tool or Framework | | Vasar et. al. 2012 Jayasinghe et. al. 2012 | IaaS |
| Test Automation Tool or Framework | | Yan et. al. 2012 Tómasson et. al. 2013 Chen et. al. 2015 | SaaS, IaaS and PaaS |
| Multi-Criterion Approach | | Pinheiro et. al. 2015 | |

Figure 3: Distribution of the researches over kind of cloud

- The system data load: the underlying system data store access load, such as the number of data store access, and data storage sizing.

Snellman et. al. propose the ASTORIA framework to identify problems using Record and Playback User Actions. The ASTORIA is a framework for automatic performance and scalability testing of Rich Internet Ap-

plications.[**?** ].

Tsai et. al. propose scalability metrics that can be used to test the scalability of SaaS applications.

Jayasinghe presents Expertus, a tool to automate large scale distributed experiment studies in IaaS clouds. The Expertus have the goal of addressing three challenges discussed in Section II.

## 4. Evolutionary Test in Load, Performance and Stress Tests

The search for the longest execution time is regarded as a discontinuous, nonlinear, optimization problem, with the input domain of the system under test as search space [12]. The main objective of evolutionary testing in performance,stress and load tests is to find test scenarios which produce execution times violating the timing constraints specified. If a temporal error is found, the test was successful [12]. The application of evolutionary algorithms to load, performance and stress tests involves finding the best and worst case execution times (BCET, WCET) to determine if timing constraints are fulfilled [13].

Evolutionary tests uses a cost (fitnesse) function to select the best individuals. There has two measurement units normally associated with the fitnesse function in load, performance or stress test: Processor Cycles and Execution Time. The Processor Cycles approach describes a fitness function in terms of processor cycles. The Execution Time approach involves executing the application under test measuring the execution time [2] [14]. The Figure 4 shows a comparison between the presented research work and the load, performance and stress test researches presented by Afzal et. al. [2]. Afzal's work was added with some of the latest research in the area ([15] [16]). The x axis represents the type of tool used ( Prototype or Functional Tool ) and the y axis presents the metaheuristic used by each research (Genetic Algorithm, Tabu Search, Simulated Annnealing or a Customized Algorithm). The Figure also divides the researches by the type of function fitnesse (Execution Time or Processor Cycles). Most research is limited to making prototypes on genetic algorithms. The presented research work is distinguished from others by having a functional tool using a hybrid approach.

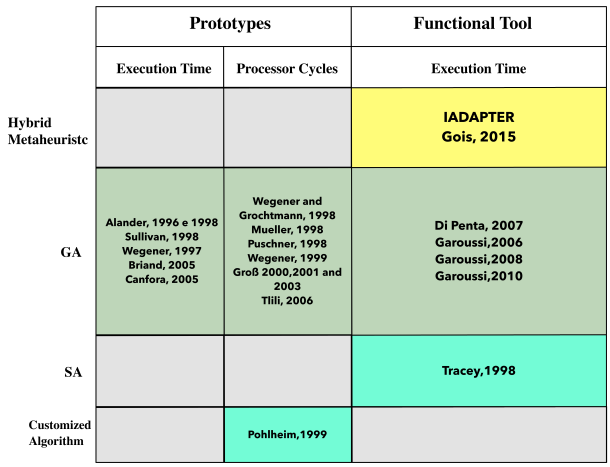


Figure 4: Distribution of the researches over range of applied metaheuristics

## Reference


[1] J. Biolchini, P. G. Mian, A. Candida, C. Natali, Systematic Review in Software Engineering, System Engineering and Computer Science Department COPPE/UFRJ, Technical Report ES 679 (2005) 165–176.

[2] A systematic review of search-based testing for non-functional system properties, Information and Software Technology 51 (2009) 957–976.

[3] B. Kitchenham, S. Charters, Guidelines for performing Systematic Literature Reviews in Software Engineering, Engineering 2 (2007) 1051.

[4] P. Jamshidi, A. Ahmad, C. Pahl, Cloud Migration Research: A Systematic Review, IEEE Transactions on Cloud Computing 1 (2013) 142–157.

[5] D. G. Feitelson, Workload Modeling for Computer Systems Performance Evaluation, Cambridge University Press, 2013.

[6] I. Molyneaux, The Art of Application Performance Testing, "O'Reilly Media, Inc.", 2009.

[7] M. C. Gonçalves, Um Processo de Inferência de Desempenho para Apoiar o Planejamento da Capacidade de Aplicações na Nuvem Um Processo de Inferência de Desempenho para Apoiar o Planejamento da Capacidade de Aplicações na Nuvem (2014).

[8] G. a. Di Lucca, A. R. Fasolino, Testing Web-based applications: The state of the art and future trends, Information and Software Technology 48 (2006) 1172–1186.

[9] C. Sandler, T. Badgett, T. Thomas, The Art of Software Testing (2004) 200.

[10] C. Jia, Y. Cai, Y. T. Yu, T. Tse, 5W+1H pattern: A perspective of systematic mapping studies and a case study on cloud software testing, Journal of Systems and Software 000 (2015) 1–14.

[11] J. Gao, P. Pattabhiraman, X. Bai, W. T. Tsai, SaaS performance and scalability evaluation in clouds, Proceedings - 6th IEEE International Symposium on Service-Oriented System Engineering, SOSE 2011 (2011) 61–71.

[12] M. O. Sullivan, S. Vössner, J. Wegener, D.-b. Ag, Testing Temporal Correctness of Real-Time Systems — A New Approach Using Genetic Algorithms and Cluster Analysis — (????) 1–20.

[13] W. Afzal, R. Torkar, R. Feldt, A systematic review of search-based testing for non-functional system properties, Information and Software Technology 51 (2009) 957–976.

[14] N. J. Tracey, A search-based automated test-data generation framework for safety-critical software, Ph.D. thesis, Citeseer, 2000.

[15] V. Garousi, Traffic-aware Stress Testing of Distributed Real-Time Systems based on UML Models using Genetic Algo-

rithms, August, 2006.
[16] V. Garousi, A Genetic Algorithm-Based Stress Test Requirements Generator Tool and Its Empirical Evaluation, IEEE Transactions on Software Engineering 36 (2010) 778–797.