

# Software Performance Testing in Complex Environments

*Proposta de Mestrado*

Orientador: Simão Melo de Sousa  
Co-Orientador: João Paulo Fernandes

2016-2017

## 1 Context

We propose the study of dynamic deployment, control and gathering of performance metrics in context of software development. The motivation for this study is:

1. In today's software market, time to market is a key factor to success but one cannot disregard quality. Performance is a key factor since software should not only do what is it's supposed to (which is expressed in terms of functional requirements) but also do it on time (to be included in one kind of non-functional requirements).
2. Performance testing introduces an added complexity due to the necessity of actively collect metrics on the System Under Test (SUT) and afterwards relate (in time) the collected values with the action taken in the test execution.
3. Complexity increases when the SUT is composed of multiple technologies divided in multiple hosts, each with its own specific metrics and collection tools.
4. Platform as a Service for software deployments is a reality for both production solutions and testing environments as this reduces the costs of infrastructure but also introduces a dynamic environment deployment and the necessity of automation tools.

With our work we want to address three aspects:

1. Automate the metric collection on a given SUT;
2. Automatetheinjectorsspawnbasedonloadpatternsandinjectioncontrol.
3. Relate the metric values collected with the injection applied.

Our goal is to understand the possibility of having a framework for orchestrating the deployment of performance testing tools given an inventory of a SUT and its technologies, for metric collection, and the necessities of injection to measure the performance under different load patterns.

## 2 Research Questions

The following research questions have to be answered in the project:

- How different are the conditions for performance testing in a static environment and dynamically allocated one?
- What challenges rise from dynamic orchestration from both injection and collection?
- When a set of functional automated tests already exist, can one make use of those as an input for performance testing?

## 3 Work Program

The central work areas for the proposed research break down into the phases described below:

- Trimester 1: Study the literature, the tools and the frameworks that are available for performance testing as well as for software deployment/installation. From the identified ones, realize those that are the most fitted for the collection of the metrics of the target technologies. Identify the different types of performance testing and their respective goals. From the analysis, realize how the load injection must be controlled.
- Trimester 2: Develop a framework for the automated deployment of the tools for both metric collection and load injection. The framework to develop should be general enough to consist of a re-usable tool that other developers may build their work upon.
- Trimester 3: Perform a proof of concept of performance testing using the developed framework.
- Trimester 4: Write a thesis, and an overview article that introduces the conducted research.

## 4 Expected Outputs

- A computational prototype.
- Involvement in the dissemination of the research achievements.
- An MsC dissertation.

## 5 Relevance to the beneficiaries

The beneficiaries of this research are software developers because we provide a framework to improve and facilitate the implementation and deployment of performance testing tools. The software artifacts we produce in the process will be of use to a wide audience of software developers.

## 6 Contacts

Simão Melo de Sousa (desousa@di.ubi.pt)

Company Overseer: Eng. João Casteleiro. (joao.casteleiro@readinessit.com,  
Readiness IT)

