Chapter 114
A Study of Contemporary System Performance Testing Framework

Alex Ng
Federation University, Australia

Shiping Chen
CSIRO Data61, Australia

ABSTRACT

Performance testing is one of the vital activities spanning the whole life cycle of software engineering. As a result, there are a considerable number of performance testing products and open source tools available. It has been observed that most of the existing performance testing products and tools are either too expensive and complicated for small projects or too specific and simple for diverse performance tests. In this chapter, the authors present an overview of existing performance test products/tools, provide a summary of some of the contemporary system performance testing frameworks, and capture the key requirements for a general-purpose performance testing framework. Based on previous works, the authors propose a system performance testing framework that is suitable for both simple and small as well complicated and large-scale performance testing projects. The core of the framework contains an abstraction to facilitate performance testing by separating the application logic from the common performance testing functionality and a general-purpose data model.

INTRODUCTION

System Performance Testing is a vital activity spanning the Software Development Life Cycle (Siva & Sharma, 2006).There are different types of test: Functionality, Graphics, Collision, Performance, Gaming, User Interface, Progression, Correlation, Numerical, Conformance, Interoperability and many more. In this chapter, we find special interest in System Performance Testing. The purpose of system performance testing is to identify bottlenecks by measuring a system's responsiveness and scalability under a certain load (Sarojadevi, 2011). Load testing provides a means for assessing the behavior of a

system under varying traffic burdens. Performance testing is able to provide simulation and modeling help enterprises experiment not just with load but also with specific situations, such as large numbers of users performing the same activity conducting a search or completing an online transaction simultaneously (Menascé, 2002). Performance tests conducted over time provide a good landscape trend which helps companies plan and test for future needs. Such testing can help determine what hardware resources need to be purchased to achieve desired future capabilities (Jain, 1991).

In this chapter, we discuss some of the contemporary system performance testing frameworks and present a general-purpose testing framework for both simple and complicated performance testing. We observed that few of the system performance testing frameworks available have been widely used in industry or the research community (Yigitbasi, Iosup, Epema, & Ostermann, 2009). Our framework proposes an abstraction to facilitate performance testing by separating the application logic from the common performance testing functionalities.

The rest of this chapter is organized as follows. We first provide a definition of system performance and the requirements for system performance testing framework. Afterward, we present a discussion of some of the contemporary system performance testing frameworks. Lastly, we propose a system performance framework with a sample implementation to demonstrate the applicability of this framework.

BACKGROUND

According to (Meier, Farre, Bansode, Barber, & Rea, 2007), Performance Testing is defined as the technical investigation done to determine or validate the speed, scalability, and/or stability characteristics of the product under test. Performance-related activities, such as testing and tuning, are concerned with achieving response times, throughput, and resource-utilization levels that meet the performance objectives for the application under test.

(Meier et al., 2007) classify performance metrics into the following categories:

- **Network-Specific Metrics:** A set of metrics about the overall behavior of the network used to support the system.
- **System-Related Metrics:** A set of metrics helps identify the resource utilization of the system.
- **Platform-Specific Metrics:** A set of metrics related to software that is used to host the application system, such as the Microsoft .NET Framework common language runtime (CLR) and ASP. NET-related metrics.
- **Application-Specific Metrics:** These include custom performance counters inserted into the application code to monitor application health and identify performance issues.
- **Service-Level Metrics:** A set of metrics help measure overall application throughput and latency, or they might be tied to specific business scenarios.
- **Business Metrics:** These metrics are indicators of business-related information, such as the number of orders placed in a given timeframe for a particular department.

There are some common system performance metrics for enterprise systems, such as: *Response Time, Latency,* and *Throughput*. In some contexts it's customary to call these things by different names: *Throughput and Response Time, or Capacity and Delay, or Bandwidth and Latency*. We provide the following definitions to avoid ambiguity:

Related Content

High-Performance Apparel and Wearable Devices for Hot Environments
Radostina A. Angelova (2019). *International Journal of Mobile Devices, Wearable Technology, and Flexible Electronics (pp. 1-14).*
www.irma-international.org/article/high-performance-apparel-and-wearable-devices-for-hot-environments/268888

Perceived Mobile Information Security and Adoption of Mobile Payment Services in China
Fei Gao, Pei-Luen Patrick Rauand Yubo Zhang (2018). *Mobile Commerce: Concepts, Methodologies, Tools, and Applications (pp. 1179-1198).*
www.irma-international.org/chapter/perceived-mobile-information-security-and-adoption-of-mobile-payment-services-in-china/183334

Mobile Application and User Analytics
Venkatraman Ramakrishnaand Kuntal Dey (2018). *Mobile Commerce: Concepts, Methodologies, Tools, and Applications (pp. 691-718).*
www.irma-international.org/chapter/mobile-application-and-user-analytics/183312

On Balancing Energy Consumption, Rendering Speed, and Image Quality on Mobile Devices
Fan Wu, Emmanuel Agu, Clifford Lindsayand Chung-han Chen (2010). *International Journal of Handheld Computing Research (pp. 51-71).*
www.irma-international.org/article/balancing-energy-consumption-rendering-speed/46087

Mobile Ad Hoc Network
Subhankar Dhar (2009). *Mobile Computing: Concepts, Methodologies, Tools, and Applications (pp. 952-960).*
www.irma-international.org/chapter/mobile-hoc-network/26559