Chapter 91
Cloud Scalability Measurement and Testing

Xiaoying Bai
Tsinghua University, China

Jerry Gao
San Jose State University, USA

Wei-Tek Tsai
Tsinghua University, China & Arizona State University, USA

ABSTRACT

Cloud computing introduces a new paradigm for software deployment, hosting, and service renting. Based on the XaaS architecture, a large number of users may share computing resources, platform services, and application software in a multi-tenancy approach. To ensure service availability, the system needs to support an advanced level of massive scalability so that it can provide necessary resources on demand following the pay-per-use pricing model. This chapter analyzes the unique requirements of cloud performance and scalability, compared with traditional distributed systems. Measurements are proposed with performance indicators, meters, and metrics identified from different perspectives. To support scalability testing in a dynamic environment, an agent-based testing framework is proposed to facilitate adaptive load generation and simulation using a two-layer control architecture.

1. INTRODUCTION

Cloud computing proposes a new architecture for multi-layered resource sharing, such as infrastructure-as-a-service (IaaS), storage-as-a-service (SaaS), data-as-as-service (DaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS) (Rimal, et al., 2009). Resources can be dynamically allocated based on usage demands following negotiated Service Level Agreement (SLA) in a pay-per-use business model to achieve cost-effective performance and resource utilization. Cloud-based infrastructure has a big impact on software engineering methods. It shifts the focus of software development from product-oriented programming to service-oriented reuse, composition, and online renting. It has a great potential to enhance the scalability, portability, reusability,

DOI: 10.4018/978-1-4666-6539-2.ch091

flexibility, and fault-tolerance capabilities of software systems, taking the advantage of the new infrastructure architecture.

However, cloud-based infrastructure also introduces new risks to software systems. Software is remote deployed in a virtualized runtime environment, using rented hardware/software resources, and hosted in a third-party infrastructure. The quality and performance of the software highly depend on its runtime environment. For example, Amazon provides a huge cloud infrastructure EC2 (Elastic Compute Cloud) and web-hosting system AWS (Amazon Web Services) based on EC2 (Amazon). It promises to keep customers' sites up and running 99.95% of the year, which only allows for 4.4 hours of downtime. Unfortunately, an unexpected crash happens in April, 2011 due to operation mistakes during network reconfiguration (News, 2011). More than 70 organizations are affected including FourSquare, the New York Times, and Reddit, which pay to use AWS to run their websites on EC2. Due to the accident, the performance of these websites are greatly decreased, and some sites were even down for dozens of hours.

Scalability is one of the important quality concerns of cloud performance (Bondi, 2000; Chen&Sun, 2006; Duboc, et al., 2006; Gao, et al., 2011). Resources in the cloud allocated elastically to support application executions following a usage-based approach. Built upon the conventional concepts of distributed resource management, cloud computing presents new scalability features. Stakeholders in a cloud-based system have different performance concerns from their individual perspectives including infrastructure providers, software service providers, and end users. From the infrastructure providers' perspective, resource utilization is important. That is, it can timely release sources so that the system can re-allocate to other applications and customers. From the service providers' perspective, it needs to balance between system performance and cost of resource reservation. If resources are reserved more than needed, they have to pay for wasteful resources. If resources are reserved less than needed, they cannot guarantee service availability and response time.

To address the issue, the chapter proposes new analytic techniques for cloud scalability measurement using Radar Chart Model. Performance indicators and meters are analyzed from three perspectives including resource allocation and utilization, system load and system performance. Scalability are measured by taking multiple variables into consideration.

Due to the uncertainties and dynamic nature of cloud infrastructure, continuous testing is necessary to gather data and evaluate system performance (Bai, et al., 2007; Chen, 2006; Gao, et al., 2011; Li, et al., 2010; Liu, 2009; Molyneaux, 2009; Steen, et al., 1998). In counter to the challenges of software testing under uncertainties, new testing capabilities are necessary including:

- **Adaptive Testing:** The ability to sense changes in target software systems and environment, and to adjust test accordingly.
- **Dynamic Testing:** The ability to re-configure and re-compose tests, and to produce, on-demand, new test data, test cases, test plans and test deployment.
- **Collaborative Testing:** The ability to coordinate test executions that are distributed dispersed.

The research proposes an agent-based testing framework to facilitate performance testing of software system built upon the cloud platforms (Bai, et al., 2006; Bai, et al., 2011; Tsai, et al., 2003; Tsai, et al., 2004). Agents are designed with necessary test knowledge, test goal and action plan. Performance testing is defined as a control problem to select the workload and test cases in order to achieve the goal of performance anomaly detection. Test agents are classified into two categories: test coordinator and test runners. Test runners are distributed located on host computers to exercise