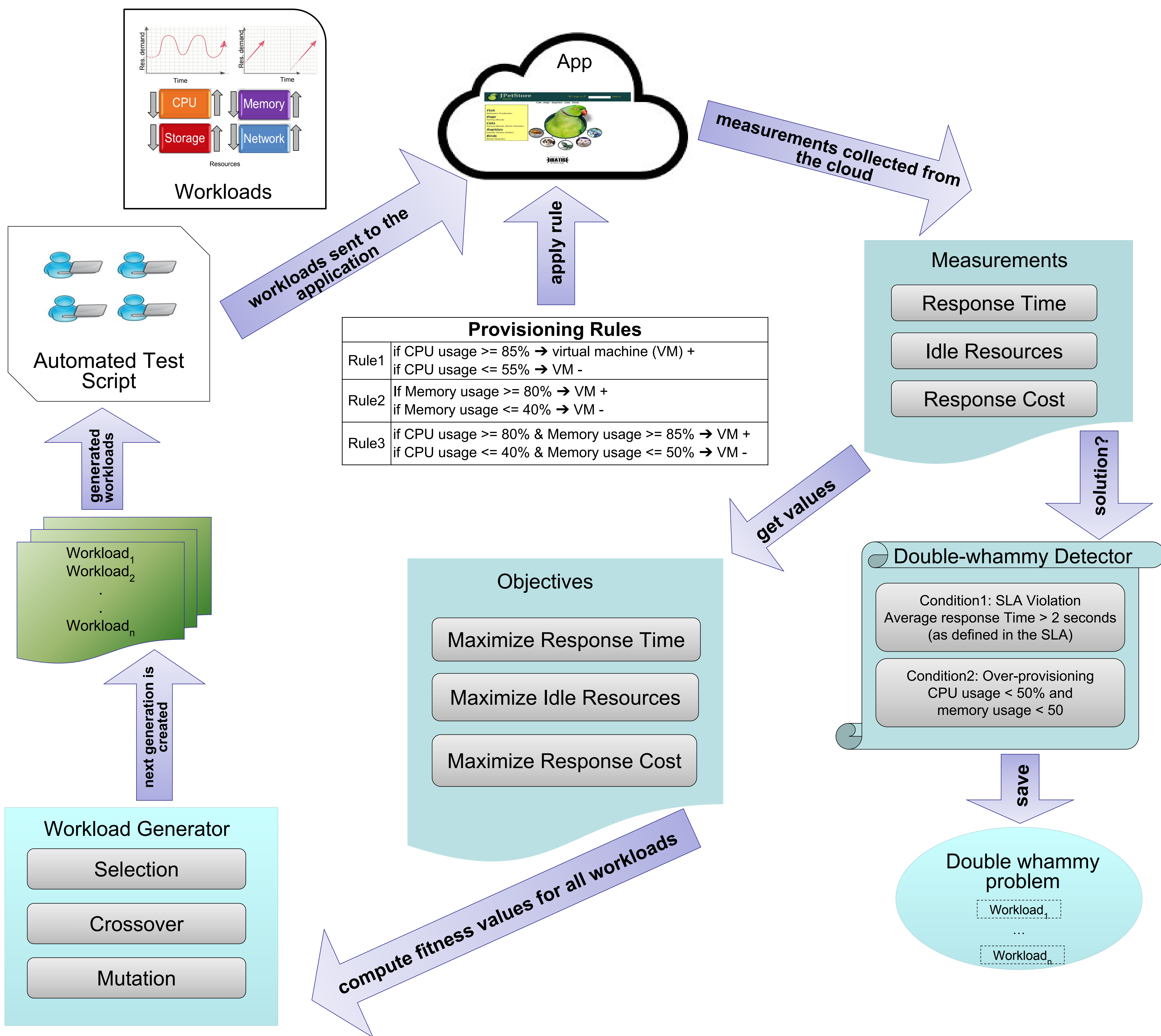


1 Motivation

In a cloud environment, consumers can deploy and run their software applications on a sophisticated infrastructure that is owned and managed by a cloud provider (e.g., Amazon Web Services, Microsoft Azure, and Google Cloud Platform). Cloud users can acquire resources for their applications on demand, and they have to pay only for the consumed resources. This allocation should occur as the load to the application increases, with no degradation of the application's response time.

2 Problem Statement

During allocating of resources to the application, over-provisioning and service level agreement (SLA) violations can occur at the same time, which is referred as the double-whammy problem. Not only does this situation degrade the applications' quality of service (QoS), but also it causes consumers to pay for resources, which are provisioned and do not improve the QoS. Software engineers define various provisioning rules for applications that are deployed in the cloud by studying their performance behaviors during software testing to maintain a particular performance.



4 Experimental Plan

TICLE will be evaluated on two cloud platforms: CloudStack, an open source cloud platform and Microsoft Azure, an enterprise cloud platform using five web-based applications of different sizes and domains such as JPetStore, JForum, LogicalDoc, Kunagi, and PhotoV.

3 Our goal is to improve the cloud elasticity by testing provisioning rules

We developed a framework, coined as TestIng CCloud Elasticity (TICLE), which can automatically generate workloads for the application using GA-based multi objective optimization algorithms during performance testing that will cause both over-provisioning and SLA violations to occur at the same time in the cloud for any given provisioning rules.