Elastic Scaling in the Clouds

K. Keahey, J. Bresnahan, T. Freeman, D. LaBissoniere, and P. Marshall

In the recent years, Infrastructure-as-a-Service (IaaS) cloud computing has emerged as an attractive alternative to the acquisition and management of physical resources. The ability to provision resources on-demand allows users to elastically expand and contract the resource base available to them based on an immediate need – a pattern that enables a quick turnaround time when evaluating urgent phenomena, working towards deadlines, or growing an institutional resource base. Furthermore, since access to computational resources no longer requires a significant hardware investment, such resources, even at large scales, become accessible to individual researchers. This leads to a democratization of computing, making more resources available to more scientists. Finally, the adoption of cloud computing may eventually greatly reduce, or even eliminate, the need for institutions to own and maintain large computational resources to support research, allowing them instead to focus their energies and resource on their core competencies.

The ability to elastically expand and contract the resource base as needed is particularly interesting as it allows us to react with a good response time to requests supporting computation for processing events caused by sudden phenomena, such as: hurricanes, ecological disasters, earthquakes, or market disruptions. Such events are typically unpredictable and often result in significant increase in the demand for computational capacity as the requirements for computation propagate through the system. This demand can be satisfied by provisioning resources from private, community and public clouds thus creating a highly available hybrid cloud platform capable of satisfying fluctuating demand.

Our poster presents the architecture and prototype implementation of a system providing such elastic capabilities. The system monitors a queue of requests and, based on its properties (such as length, rate of submitted requests, history files, etc.), it provisions or releases virtual machines (VMs) that are capable of processing requests on the queue. Our system supports plugins to many Infrastrucgure-as-a-Service (IaaS) offerings and thus the VMs may be provisioned on private clouds and community clouds, such as e.g. the FutureGrid clouds or Magellan system configured with Nimbus or a variety of commercial clouds including Amazon or Rackspace. The rate of this provisioning is regulated by a set of policies taking into account queue properties, available resources as well as cost and affordability. Furthermore, the system is designed to be highly available (HA) and work with an operations team to provide those properties.

The poster will describe the system components, including

(1) the information package, including a variety of sensors providing information about the queue, monitoring VMs as well as monitoring IaaS request status as well as a sensor aggregator composing a stream of information from a variety of sources,

(2) a policy engine, which evaluates policies against this information stream and produces directives describing what types of VMs needs to be deployed where,

(3) an elastic processing controller, which carries out the directives of the decision engine, and

(4) provisioning system capable of deploying and contextualizing VMs on a variety of IaaS systems.

Our poster will further include an evaluation of the system, in the context of deployment on Amazon EC2 (www.amazon.com/ec2). The evaluate will illustrate system reaction times in the case of average and worst-case load, showing how the response times vary under different synthetic traces. We will also show another experiment, in which the reliability of the system will be evaluated under different failure conditions induced in a controlled fashion – in this experiment, we will evaluate how the system rebuilds itself.

We will conclude with recommendations for the system's use and an analysis of the impact of these capabilities. We will also provide examples of real-life applications using the system. Our presentation will contain graphs and diagrams of the architecture.