Composing Superclouds on CloudLab

Robbert van Renesse   Hakim Weatherspoon

Department of Computer Science, Cornell University

The cloud paradigm, an elastic cluster of computing and storage resources, is wildly successful. Cloud computing promises to catalyze the technology economy, revolutionize health care, military, government, financial systems, scientific research, and society. Unfortunately, current realizations of public and private cloud have limitations. New applications demand properties that today's cloud platforms either struggle to provide efficiently or lack altogether: robustness and availability despite failure or attack, security of data and integrity of computation, energy efficiency, low latency and high bandwidth to clients worldwide, low cost through efficient use of spot markets, and so on. Many of these disadvantages result from clouds generally being under a single administrative domain.

We are building a Supercloud, that is, a cloud from resources obtained from various cloud providers. A Supercloud will allow companies, organizations, and individuals to move to a cloud computing environment while retaining control over placement and scheduling. In particular, a cloud user could control the location and live migration of their computation, networking, and storage without owning all of the underlying infrastructure—a level of control that is not available today.

The Supercloud is enabled by nested virtualization that paves over heterogeneity such as different virtual machine monitors and image representations as well as small differences in hardware. We have demonstrated live migration of virtual machines from a local cluster to Amazon EC2 to RackSpace to HP Cloud to Cornells public Red Cloud and back to the local cluster. In addition, the Supercloud provides virtual networking and storage that spans the various underlying clouds. Finally, a single cloud manager, based on OpenStack, manages the placement and live migration of VMs.

Using experiment with clients running on PlanetLab nodes, we can demonstrate that the Supercloud provides overall improved availability, as well as significantly decreased and more predictable latency than any of the cloud providers mentioned above. While many cloud providers have multiple availability regions, by combining the various clouds the Supercloud has significantly more availability regions than any other cloud. Moreover, we can migrate VMs live between availability regions, a feature not availability to other cloud providers. We can also take advantage of spot market pricing by live migration to cheap resources and demonstrate significantly cheaper execution of long running computations and services.

While doing experiments on real cloud resources is essential for success, our research will greatly benefit from an experimental resource like CloudLab, and indeed we have already started doing experiments on it. With CloudLab, we will be able to do experiments that would be impossible to perform in the wild, provide parameterization of various variables for better insight into their effect, and perhaps most importantly, provide reproducibility of experimental results. In CloudLab we can instantiate various cloud platforms, which will make it relatively easy to do further development of the Supercloud before deploying it in the wild. We will also be able to do this at a much lower cost than if we had to do all experimentation on real cloud resources. Debugging will be significantly easier, and development faster.

For example, we are currently working on the virtual networking aspect of the Supercloud. In our initial prototype, we have deployed tinc, but while it works well to tie the various clouds together, the overhead is considerable and as a result communication between VMs in the Supercloud is relatively slow. We are considering various options to solve this problem, but ideally have an environment in which we can easily test the various options. The real cloud platforms could be used for this effort, but it would slow down our development significantly and the costs involved might become prohibitive. A similar story is unfolding for Supercloud storage.

As another example, consider the Supercloud manager. Currently we are simply using the existing OpenStack cloud manager, unchanged. However, OpenStack is not aware of the underlying diversity available to it. Ideally it would do intelligent placement and live migration of computing and storage resources. Our initial effort has been on simulation. However, soon we will start implementing some algorithms and integrate them into OpenStack, at which point we will need to do evaluation through running experiments. While we could (and will) run these in the wild, CloudLab will give significantly more control over experimental parameters and provide the important ability to reproduce results.

As a final example of how CloudLab can be useful, consider the educational possibilities. We are currently running various student lab projects, in which we give small groups of students the sources and installation instructions for the Supercloud and ask them to design and implement an extension or application. Currently we essentially run a Supercloud-in-a-box, but this significantly limits the experience that the students get. Using CloudLab, we will be able to instantiate a Supercloud for each group of students.

In conclusion, CloudLab is an ideal resource for development of our Supercloud infrastructure. We expect to make significantly faster progress than we have thus far because of it, make experimentation available to many more of our students, and have results that are fully reproducible.