Machine VM + Cloud API - Rewriting the Cloud from Scratch

Thursday, October 21, 2010 at 8:26AM

Todd Hoff in Paper, amazon

Write a little "Hello World" program these days and it runs inside a bewildering Russian Doll of nested environments, each layer adding its own special performance and complexity tax. First, a language executes in its own environment of data structure libraries, memory management, and so on. That, more often than not, will run inside a language VM like the JVM, CLR, or V8. The language VM will in-turn run inside a process that runs inside an OS. An application will run in one or more threads inside a process. And the whole thing will run inside a machine sharing VM layer like Xen. And across all of that are frameworks for monitoring, elasticity, storage, and so on. That's a lot of overhead for a such a little program.

What if we could remove all these taxes and run directly on the new bare metal, which some consider to be a combination of M*achine VM* + *Cloud API*? That's exactly what a system called Mirage, described in the paper Turning down the LAMP: Software Specialisation for the Cloud, sets out to do by treating the *cloud virtual hardware as a compiler target, and converting high-level language source code directly into kernels that run on it.*

From the paper:

> *Frameworks which currently use (for example) fork(2) on a host to spawn processes would benefit from using*

> *cloud management APIs to request resources and eliminate the distinction between cores and hosts...We instead view the cloud as a stable hardware platform, and present a programming framework which permits applications to be constructed to run directly on top of it without intervening software layers. Our prototype (dubbed Mirage) is unashamedly academic; it extends the Objective Caml language with storage extensions and a custom run-time to emit binaries that execute as a guest operating system under Xen. Mirage applications exhibit significant performance speedups for I/O and memory handling versus the same code running under Linux/Xen.*

It's a fascinating idea. Operating systems have historically provided two services: hardware access and sharing of scarce resources. In the cloud, are these really necessary anymore? In the cloud you can't install special hardware so there's really no need to pretend that device drivers are important. Resources are now no longer scarce in the sense that they are acquired elastically via APIs. And with the move to service oriented architectures and the VM already sharing hardware, there's really no need to keep the ghost of a time sharing OS around either.

You are essentially developing inside a language environment using APIs and running that directly on the cloud. This model may have been strange at one time, but with the advent of PaaS products like Google App Engine, Salesforce, and Heroku, the idea of developing inside a language environment, using APIs, and simply deploying on a cloud, has become a far more acceptable way of working. GAE, for example, hasn't dropped all illusion of boundaries yet, each program definitely runs inside an instance, but they aren't very far away from being able to drop that fiction entirely.

Historically, the idea isn't new either, but the cloud as the new computer definitely has put a different spin on it. Daniel Ingalls, author of Design Principles Behind Smalltalk, wrote *An operating system is a collection of things that don't fit into a language. There shouldn't be one.*

The return for switching to this new model should be much better performance as all the layers are collapsed. Mirage reported better performance for database access, but this part of the equation needs a lot more proof, but considered on a $/CPU and $/IO basis, shifting mental models of what programs are and how they run, could pay off handsomely.

Related Articles

- Mirage - an open-source operating system for constructing secure, high-performance, reliable network applications across a variety of cloud computing and mobile platforms.
- A 'Fat-Free' Programming Framework for the Cloud by Chris Kanaracus.
- Good thread on Lambda the Ultimate. Thanks to Z-Bo for the Daniel Ingalls quote.
- Video of a talk on the paper at HotCloud. Unfortunately a password is required. If you get a password we can all use, please let me know.
- Papers related to Mirage
- Mesos - a platform for running multiple diverse cluster computing frameworks, such as Hadoop, MPI, and web services, on commodity clusters.

---

http://highscalability.com/blog/2010/10/21/machine-vm-cloud-api-rewriting-the-cloud-from-scratch.html