Applications. Islands No More.

Don MacVittie, 2011-10-08

There was a time when application developers worried only about the hardware they were developing the application for. Those days passed a good long while ago, and then AppDev's big concern was the OS the application was being developed for. But the burgeoning growth of the World Wide Web combined with the growth of Java and Linux to drive development out of the OS and into the JVM. Then, the developer focused on the JVM in question, or in many cases on the interpreted language interfaces – but not the OS or hardware. For our purposes I have lumped .NET in with JVMs as a simple head-not to where it came from. If you find that offensive, pretend I called them "AVMs" for "Application Virtual Machines". In the end, interpreted bytecodes, no matter how you slice it.

An interesting and increasing trend in this process was the growing dependence of both application and developer on the network. When I was a university lecturer, I made no bones about the fact that application developers would be required to know about the network going forward, and if the students didn't like that news, they should find an alternative career. I stopped teaching eight years ago to dedicate more time to some of my other passions (like embedded dev, gaming, and my family), so that was a long time ago, and here we are. Developers know more about the network than they ever dreamed they would. Even DBAs have to know a decent amount about general networking to do their jobs. But they don't know enough, because the network is one of the primary sources of application performance woes.

And that makes the network a developers' problem. Not maintaining it or managing it, but knowing how your applications are reliant upon it, both directly and indirectly, to deliver content to customers. And by knowing what your application needs, influencing changes in the architecture. In the end, your application has to have the appearance of performance to end users. In that sense it is much like the movies. It can look totally ridiculous while they are filming a moving – with people hanging in air and no backdrop or props around them – as long as the final product looks realistic, it impacts the viewers not one bit. But if computer animation or green screening causes the work of actors to look bad, people will be turned off. So it is with applications. If your network causes your application performance to suffer, the vast majority of the population is going to blame the application. The network is a nebulous thing, the application is the thing they were accessing.

Technology is catching up to this concept – F5 has its new application based ADC functionality from iApps to application level failover – and that will be a huge boon to developers, but you are still going to have to care – about security, about network performance, about ADCs, about load balancing algorithms. The more you know about the environment your application runs in, the better a developer you will be. It is far better to be able to say "that change would increase traffic from the database to our application and from the application to the end user – meaning out our Internet connection, and unless I'm wrong, we don't have the bandwidth for that…" than it is to make the change and *then* discover this simple truth. If you're lucky enough to have an F5 product in your network, ask about iAPP, it's well worth looking into, as it brings an application-oriented view to network infrastructure management. And when the decision is made to move to the cloud, you can take your iAPP templates with you. So they aren't simply a single-architecture tool, running in BIG-IP LTM-VE, they will apply in the cloud also.

While you're learning, you'll discover that some network appliances can help you do your job. Our Application Protocol Manager and Web Application Manager spring to mind, one helping with common security tasks, the other helping with delivery performance through caching, compression, and TCP optimization of application-specific data. You'll also get ideas for optimizing the way your application communicates on the network, because you'll understand the underlying technology better.

It's more to learn, but you have been freed from the nitty-gritty of the hardware and the OS, and in many cases from the difficult parts of algorithmic development. Which means adding high-level networking knowledge is a wash at worst. And it is one more way to separate yourself from the pack of developers, a way that will make you better at what you do.

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com