

Where networks and application architecture converge lies Devops



Lori MacVittie, 2014-07-05

#SDN #devops On this side is a variant of SDN: network service virtualization (NSV). On the other side is an emerging application architecture: microservices. Where they meet lies devops.

One of the most fascinating things to watch in the technological shifts occurring today is to see them all converging on a singular point: applications. Whether it's securing or delivering, deploying or access, all key areas of IT are all converged on a singular focus: applications.

Technology Shifts Center on Applications



From a developer-turned-network-geek perspective, that's doubly interesting. That's because one impacts the other, and vice-versa. One of the trends in application architecture today is a shift toward microservices. I'll oversimplify for a moment and explain that as SOA without all the baggage. A recent post on [High Scalability](#) explains the architecture - and the impact on infrastructure requirements:

Where a monolithic application might have been deployed to a small application server cluster, you now have tens of separate services to build, test, deploy and run, potentially in polyglot languages and environments.

All of these services potentially need clustering for failover and resilience, turning your single monolithic system into, say, 20 services consisting of 40-60 processes after we've added resilience.

Throw in load balancers and messaging layers for plumbing between the services and the estate starts to become pretty large when compared to that single monolithic application that delivered the equivalent business functionality!

[Microservices - Not A Free Lunch!](#)

Now let's shift gears and peek at what's going on over in network land. You might recall we recently discussed [network service virtualization](#). If not, here's a quick summary from [Nick Lippis](#):

NSV seeks to virtualize enterprise appliances, such as firewalls, load balancers, application accelerators, application delivery controllers, Intrusion Protection Systems, WAN optimizers, call managers, etc., instantiated for each application. Each instance of each NSV is created for a specific application. That is, if there are 10 applications that require network services, then each application will be configured with its own instantiation of that service. That is, 10 applications, then 10 NSV firewalls.

In short, NSV seeks to virtualize network services by creating an instance of the network service for each application versus virtualizing a network service once for all applications. NSV hopes to present significant capex and opex relief from hardware appliances, as well as an efficient way of applying network services to applications without chaining or tagging packets and rapid automated, on-demand application deployment.

[Lippis Report 217: It's Network Service Virtualization in the Enterprise rather than Network Function Virtualization](#)

Reading both, one might assume some level of collusion between the two but that's unlikely to be the case. The divide between application architects and networky groups is well established; they really don't play well together. And yet both these trends recognize the need to meet in the middle, in the L4-7 service layer, to provide for scalability and other "plumbing" services.

From a scalability perspective, this is very much a verticle [partitioning-based scalability pattern](#), where load is spread across distinct functional boundaries of a problem space, each handled by different processing units. Those functional boundaries in today's architectures are embodied by microservice definitions. One service is responsible for a discrete function, as the point of microservices is, to a large extent, to decompose monolithic applications into individual, domain (functional) specific services.

Overall, this means services can be scaled individually on-demand, which is far more efficient than scaling a monolithic application. But it does introduce complexity, as there are necessarily more moving parts, and it does tend to complicate monitoring and force the need for more [application-centric monitoring](#).

A Symbiotic Relationship

The application architect recognizes the need and, to some extent, laments the complexity it will introduce. Network service virtualization, on the other side, offers to fulfill the need and recognizes the need for efficiency and, ultimately, simplification in providing them in a "rapid automated, on-demand" fashion.

These issues - the plumbing and the monitoring - fall squarely into the realm of issues that can be resolved by applying devops to operations. Automated provisioning, treating infrastructure as code, and enabling a more holistic view of "applications" are all enabling capabilities of what devops aims to achieve.

For one of the first times I can remember, the operational burden imposed by technological shifts in application architecture is nearly simultaneously being addressed by the technological shifts in the network. In fact, one could argue that the shifts occurring in the network toward network service virtualization are actually **enabling** the shift in application architecture. Being able to rapidly provision, manage and monitor the L4-7 services necessary to deliver microservices increases the ability to take advantage of the architecture.

Like the question of the chicken and the egg, it really doesn't matter which came first. What matters is that they're complementary and both driving toward the same goal: accelerated application deployment and delivery of an exceptional end user experience.

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

©2016 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at f5.com. Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5. CS04-00015 0113