

The Future of Cloud: Infrastructure as a Platform

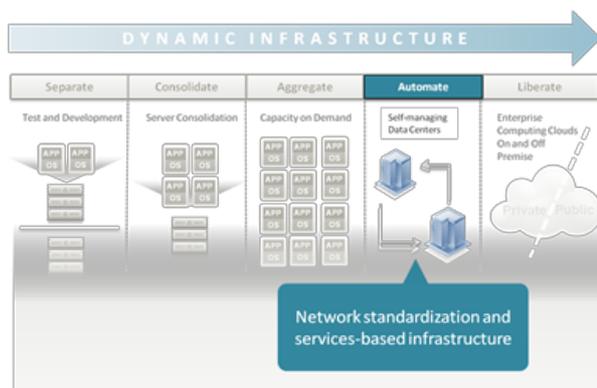


Lori MacVittie, 2011-31-10

Cloud needs to become a platform, and that means its comprising infrastructure must also embrace the platform paradigm.

There's been a spate of articles, blogs, and mentions of OpenFlow in the past few months. [IBM was the latest entry into the OpenFlow game](#), releasing an enabling RackSwitch G8264, an update of a 64-port, 10 Gigabit Ethernet switch IBM put out a year ago. Interest in the specification appears to be growing and not just because it's got the prefix-du-jour as part of its name, implying everything to everyone – free, extensible, interoperable, etc... While all those modifiers are indeed interesting and, to some, a highly important facet of the would-be standard, there's something else about it that is driving its popularity.

That something-else can be summed it with the statement: "infrastructure as a platform."



THE WEB 2.0 LESSON. AGAIN.

The importance of turning infrastructure into a platform can be evidenced by noting commentary on Web 2.0, a.k.a. social networking, applications and their failure/success to garner mind-share. Recently, a high-profile engineer at Google mistakenly posted a length and refreshingly blunt commentary on what he views as Google's failure to recognize the importance of platform to successful offerings in today's demanding marketplace. To Google's credit, once the erroneous posting was discovered, it decided to "let it stand" and thus we are able to glean some insight about the importance of platform to today's successful offerings:

While Yegge doesn't have a lot of good things to say about Amazon and its founder Jeff Bezos, he does note that Bezos – unlike Google – understands that its not just about developing interesting products, but that **it takes a platform to create a great product**.

-- SiliconFilter, "Google Engineer: "Google+ is a Prime Example of Our Complete Failure to Understand Platforms"

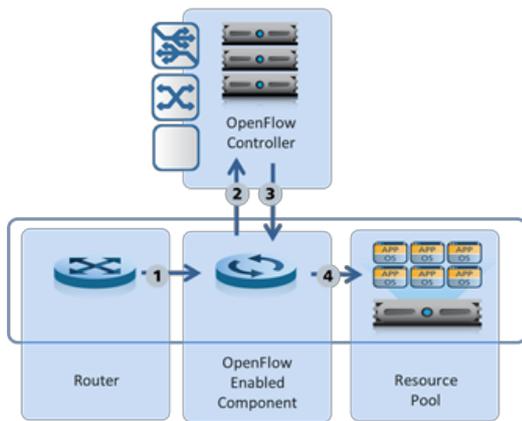
This insight is not restricted to software developers and engineers at all; the rising interest of PaaS (Platform as a Service) and the continued siren's song that it will dominate the cloud landscape in the future is all tied to the same premise: it is the availability of a robust platform that makes or breaks solutions today, not features or functions or price. It is the ability to be successful by building, as Yegge says in his post, "*an entire constellation of products by allowing other people to do the work.*"

Lest you think this concept applicable only to software, let me remind you of Nokia CEO Stephen Elop's [somewhat blunt assessment of his company's failure](#) to recognize this truth:

The battle of devices has now become a war of ecosystems, where **ecosystems include not only the hardware and software** of the device, but **developers, applications**, ecommerce, advertising, search, social applications, location-based services, unified communications and many other things. Our competitors aren't taking our market share with devices; they are taking our market share with an entire ecosystem. This means we're going to have to decide how we either build, catalyse or join an ecosystem.

-- DevCentral F5 Friday, "A War of Ecosystems"

Interestingly, 47% of respondents surveyed by Zenoss/Cloud.com for its Cloud Computing Outlook 2011 indicated use of PaaS in 2011. Like SaaS, PaaS has some wiggle room in its definition, but its general popularity seems to indicate that



yes, indeed, platform is an important factor. OpenFlow essentially provides this capability, turning infrastructure into a platform and enabling extensibility and customization that could not be achieved otherwise.

It basically turns a piece of infrastructure into a giant backplane for new functions, features, and services. It introduces, allegedly, dynamism into what is typically a static network.

It is what IaaS had the promise to be, but as of yet has failed to achieve.

CLOUD as a PLATFORM

The takeaway for cloud and infrastructure providers is that organizations want platforms. Developers want platforms. Operations wants platforms (see Puppet and Chef as examples of operational platforms). It's about enabling an ecosystem that encourages innovation, i.e. new features and functions and services, without requiring the wheel to be reinvented. It's about drag and drop, figuratively speaking, in the realm of infrastructure. Bringing the ability to deploy new services atop a platform that provides the basics.

OpenFlow promises just such capabilities for infrastructure much in the same way Facebook provides these basics for game and application developers. Mobile platforms offer the same for devices and operating systems. It's about enabling an ecosystem in which organizations can focus on not the core infrastructure, but on custom functionality and process automation that delivers efficiency to IT across operations and development alike.

"The beauty of this is it gives more flexibility and control to the network," said Shaughnessy [marketing manager for system networking at IBM], "so you could actually adjust the way the traffic flows go through your network dynamically based on what's going on with your applications."

-- IBM releases OpenFlow-enabled switch

It enables flexibility in the network, the means to deploy more dynamism in traffic policy enforcement and shaping and ties back to cloud with its ability to impart multi-tenant capabilities to infrastructure without completely modifying the internal architecture of components – a major obstacle for many network-focused devices.

OpenFlow is not a panacea, there are myriad reasons why it may not be appropriate as the basis for architecting the cloud platform foundation required to support future initiatives. But it is a prime example of the kind of **platform-focused** capabilities organizations desire to move ahead in their journey to IT as a Service. The cloud on which organizations will be able to build their future data center architecture will be a platform, and that means from the bottom (infrastructure) to the middle (development) to the top (operations).

What cloud and infrastructure providers must do is simulate the Facebook experience at the infrastructure layer. Infrastructure as a platform is the next step in the evolution of cloud computing .

- [IT Services: Creating Commodities out of Complexity](#)
- [IBM releases OpenFlow-enabled switch](#)
- [The Cloud Configuration Management Conundrum](#)
- [IT as a Service: A Stateless Infrastructure Architecture Model](#)
- [If a Network Can't Go Virtual Then Virtual Must Come to the Network](#)
- [You Can't Have IT as a Service Until IT Has Infrastructure as a Service](#)
- [This is Why We Can't Have Nice Things](#)
- [WILS: Automation versus Orchestration](#)
- [The Infrastructure Turk: Lessons in Services](#)
- [Putting the Cloud Before the Horse](#)

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