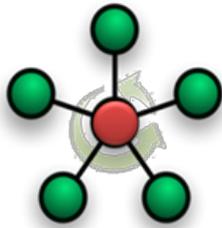


Applying ‘Centralized Control, Decentralized Execution’ to Network Architecture



Lori MacVittie, 2012-27-06



#SDN brings to the fore some critical differences between concepts of control and execution

While most discussions with respect to SDN are focused on a variety of architectural questions (me included) and technical capabilities, there’s another very important concept that needs to be considered: control and execution.

SDN definitions include the notion of centralized control through a single point of control in the network, a controller. It is through the controller all important decisions are made regarding the flow of traffic through the network, i.e. execution.

This is not feasible, at least not in very large (or even just large) networks. Nor is it feasible beyond simple L2/3 routing and forwarding.

HERE COMES the SCIENCE (of WAR)

There is very little more dynamic than combat operations. People, vehicles, supplies – all are distributed across what can be very disparate locations. One of the lessons the military has learned over time (sometimes quite painfully through experience) is the difference between control and execution. This has led to decisions to employ what is called, “Centralized Control, Decentralized Execution.”

Joint Publication (JP) 1-02, *Department of Defense Dictionary of Military and Associated Terms*, defines **centralized control** as follows: “In joint air operations, placing within one commander the responsibility and authority for planning, directing, and coordinating a military operation or group/category of operations.”

JP 1-02 defines **decentralized execution** as “delegation of execution authority to subordinate commanders.”

Decentralized execution is the preferred mode of operation for dynamic combat operations. Commanders who clearly communicate their guidance and intent through broad mission-based or effects-based orders rather than through narrowly defined tasks maximize that type of execution. Mission-based or effects-based guidance allows subordinates the initiative to exploit opportunities in rapidly changing, fluid situations.

-- [Defining Decentralized Execution in Order to Recognize Centralized Execution](#) * Lt Col Woody W. Parramore, USAF, Retired

Applying this to IT network operations means a single point of control is contradictory to the “mission” and actually interferes with the ability of subordinates (strategic points of control) to dynamically adapt to rapidly changing, fluid situations such as those experienced in virtual and [cloud computing](#) environments.

Not only does a single, centralized point of control (which in the SDN scenario implies control over execution through admittedly dynamically configured but rigidly executed) abrogate responsibility for adapting to “rapidly changing, fluid situations” but it also becomes the weakest link. Clausewitz, in the highly read and respected “*On War*”, defines a center of gravity as “the hub of all power and movement, on which everything depends. That is the point against which all our energies should be directed.” Most military scholars and strategists logically imply from the notion of a Clausewitzian center of gravity is the existence of a critical weak link.

If the “controller” in an SDN is the center of gravity, then it follows it is likely a critical, weak link.

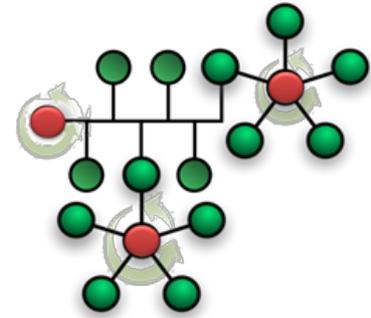
This does not mean the model is broken, or poorly conceived of, or a bad idea. What it means is that this issue needs to be addressed. The modern strategy of “Centralized Control, Decentralized Execution” does just that.

Centralized Control, Decentralized Execution in the Network

The major issue with the notion of a centralized controller is the same one – air combat operations experienced in the latter part of the 20th century: agility, or more appropriately, lack thereof. Imagine a large network adopting fully an SDN as defined today. A single controller is responsible for managing the direction of traffic at L2-3 across the vast expanse of the data center. Imagine a node, behind a [Load balancer](#), deep in the application infrastructure, fails. The controller must respond and instruct both the [load balancing](#) service and the core network how to react, but first it must be notified.

It's simply impossible to recover from a node or link failure in 50 milliseconds (a typical requirement in networks handling voice traffic) when it takes longer to get a reply from the central controller. There's also the "slight" problem of network devices losing connectivity with the central controller if the primary uplink fails.

-- [OpenFlow/SDN Is Not A Silver Bullet For Network Scalability](#), [Ivan Pepelnjak](#) (CCIE#1354 Emeritus) Chief Technology Advisor at [NIL Data Communications](#)



The controller, the center of network gravity, becomes the weak link, slowing down responses and inhibiting the network (and IT) from responding in a rapid manner to evolving situations.

“**Mission-based or effects-based guidance allows subordinates the initiative to exploit opportunities in rapidly changing, fluid situations.**

This does not mean the model is a failure. It means the model must adapt to take into consideration the need to adapt more quickly. This is where decentralized execution comes in, and why predictions that SDN will evolve into an overarching management system rather than an operational one are likely correct.

There exist today, within the network, strategic points of control; locations within the data center architecture at which traffic (data) is aggregated, forcing all data to traverse, from which control over traffic and data is maintained. These locations are where decentralized execution can fulfill the “mission-based guidance” offered through centralized control.

Certainly it is advantageous to both business and operations to centrally define and codify the operating parameters and goals of data center networking components (from L2 through L7), but it is neither efficient nor practical to assume that a single, centralized controller can achieve both managing and executing on the goals. What the military learned in its early attempts at air combat operations was that by relying on a single entity to make operational decisions in real time regarding the state of the mission on the ground, missions failed. Airmen, unable to dynamically adjust their actions based on current conditions, were forced to watch situations deteriorate rapidly while waiting for central command (controller) to receive updates and issue new orders.

Thus, central command (controller) has moved to issuing mission or effects-based objectives and allowing the airmen (strategic points of control) to execute in a way that achieves those objectives, in whatever way (given a set of constraints) they deem necessary based on current conditions.

This model is highly preferable (and much more feasible given today's technology) than the one proffered today by SDN. It may be that such an extended model can easily be implemented by distributing a number of controllers throughout the network and federating them with a policy-driven control system that defines the mission, but leaves execution up to the distributed control points – the strategic control points.

SDN is new, it's exciting, it's got potential to be the “next big thing.” Like all nascent technology and models, it will go through some evolutionary massaging as we dig into it and figure out where and why and how it can be used to its greatest potential and organizations' greatest advantage.

One thing we don't want to do is replicate erroneous strategies of the past. No network model abrogating all control over execution has ever really worked. All successful models have been a distributed, federated model in which control may be centralized, but execution is decentralized. Can we improve upon that? I think SDN does in its recognition that static configuration is holding us back. But its decision to reign in all control while addressing that issue may very well give rise to new issues that will need resolution before SDN can become a widely adopted model of networking.

- [QoS without Context: Good for the Network, Not So Good for the End user](#)
 - [Cyclomatic Complexity of OpenFlow-Based SDN May Drive Market Innovation](#)
 - [SDN, OpenFlow, and Infrastructure 2.0](#)
 - [OpenFlow/SDN Is Not A Silver Bullet For Network Scalability](#)
 - [Prediction: OpenFlow Is Dead by 2014; SDN Reborn in Network Management](#)
 - [OpenFlow and Software Defined Networking: Is It Routing or Switching ?](#)
 - [Cloud Security: It's All About \(Extreme Elastic\) Control](#)
 - [Ecosystems are Always in Flux](#)
 - [The Full-Proxy Data Center Architecture](#)
-

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