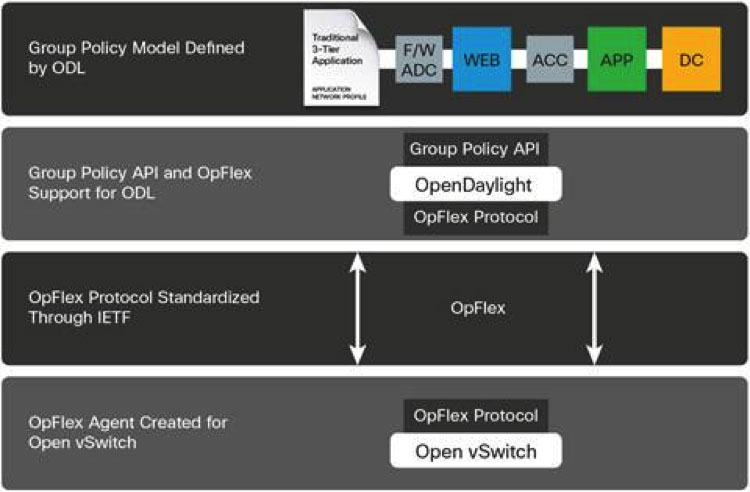
What is Cisco OpFlex?

[Cisco](https://www.sdxcentral.com/cisco/datacenter/) OpFlex is a [southbound protocol](https://www.sdxcentral.com/sdn/definitions/southbound-interface-api/) in a [software-defined network (SDN)](https://www.sdxcentral.com/sdn/) designed to facilitate the communications between the [SDN Controller](https://www.sdxcentral.com/sdn/definitions/sdn-controllers/) and the infrastructure (switches and routers). The goal is to create a standard that enables policies to be applied across physical and virtual switches/routers in a multi-vendor environment.

Overview of Cisco OpFlex

On the surface, [Cisco OpFlex](http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-731302.html) sounds a lot like [OpenFlow](https://www.sdxcentral.com/sdn/definitions/what-is-openflow/), an open standard that enables the SDN Controller to interact with the infrastructure, however, it is quite different in terms of the scope of its capabilities.

While OpenFlow centralizes all the network control functions on the SDN Controller, Cisco OpFlex focuses primarily on the policies. Cisco believes this focus will remove the potential for the controller to become the bottleneck of the network, supporting greater resiliency, availability and scalability, by pushing some of the intelligence out to the devices, using established networking protocols.

[](https://www.sdxcentral.com/wp-content/uploads/2014/07/cisco-opflex.jpg)

Basically, policies are defined within a logical, centralized policy repository in the controller, and the OpFlex protocol is used to communicate and enforce those policies within a set of distributed policy elements on the switches/routers/etc. The protocol allows bidirectional communication of policy, events, statistics and fault information, so potential adjustments can be made to address changes in the environment.

To work, an agent must be embedded in the switches and routers to support the Cisco OpFlex protocol. As a result, [Cisco](https://www.sdxcentral.com/cisco/datacenter/) is working on an [open source](https://www.sdxcentral.com/cloud/open-source/) OpFlex agent that can be used across platforms. [Microsoft](https://www.sdxcentral.com/listings/microsoft/), [IBM](https://www.sdxcentral.com/listings/ibm/), [F5](https://www.sdxcentral.com/listings/f5/), [Citrix](https://www.sdxcentral.com/listings/citrix/), [Red Hat](https://www.sdxcentral.com/listings/red-hat/), [Canonical](http://www.canonical.com/), and [AVI Networks](http://www.avinetworks.com/) have already committed to embedding this agent in their solutions.

The Differences Between Cisco OpenFlow and Cisco OpFlex

Like OpenFlow, OpFlex is designed for communications between a central controller and network devices, but has a different way of distributing the message. While OpenFlow centralizes the network control plane on an SDN Controller and can push commands down to OpenFlow enabled network devices, OpFlex centralizes policy control and relies on traditional and distributed network control protocols to push commands down.

One of the big advantages of OpenFlow is the level of control it offers developers of network control software, which promotes rapid service introduction and customization. Network operators can implement features they want in software they control, rather than having to collaborate with a vendor to put a plan into motion. These benefits are part of OpenFlow’s rapid growth and status as an [SDN](https://www.sdxcentral.com/sdn/definitions/what-the-definition-of-software-defined-networking-sdn/) standard.

In April of 2014, Cisco submitted protocol to the [Internet Engineering Task Force (IETF)](https://www.sdxcentral.com/listings/ietf/) standardization process. Several industry leaders are actively working with Cisco on the standardization process, including Microsoft, IBM, Citrix, and [SunGard Availability Services](http://www.sungardas.com/Pages/default.aspx) to increase adoption and accelerate innovation with the Cisco OpFlex protocol.