White Paper

OpFlex: An Open Policy Protocol

Data Center Challenges

As data center environments become increasingly dynamic, networks are increasingly asked to provide agility and flexibility without compromising performance, security, scalability, and stability. Many of today's software-defined networking (SDN) network virtualization solutions approached this problem by creating software overlay solutions. Although these tools seemed to offer at least a partial solution, they ignored a number of critical issues that limit their applicability. They started with the same Layer 2 and 3 constructs in use in networking today instead of changing the operating model to simplify the way that networking connectively is defined. In fact, even an extremely popular platform like OpenStack has been slightly held back by this approach. OpenStack introduced a logical network and router construct to model current network behavior. However, they have been not been able to quickly adopt complex policies such as service chaining or advanced automation because they don't fit cleanly into the current abstractions.

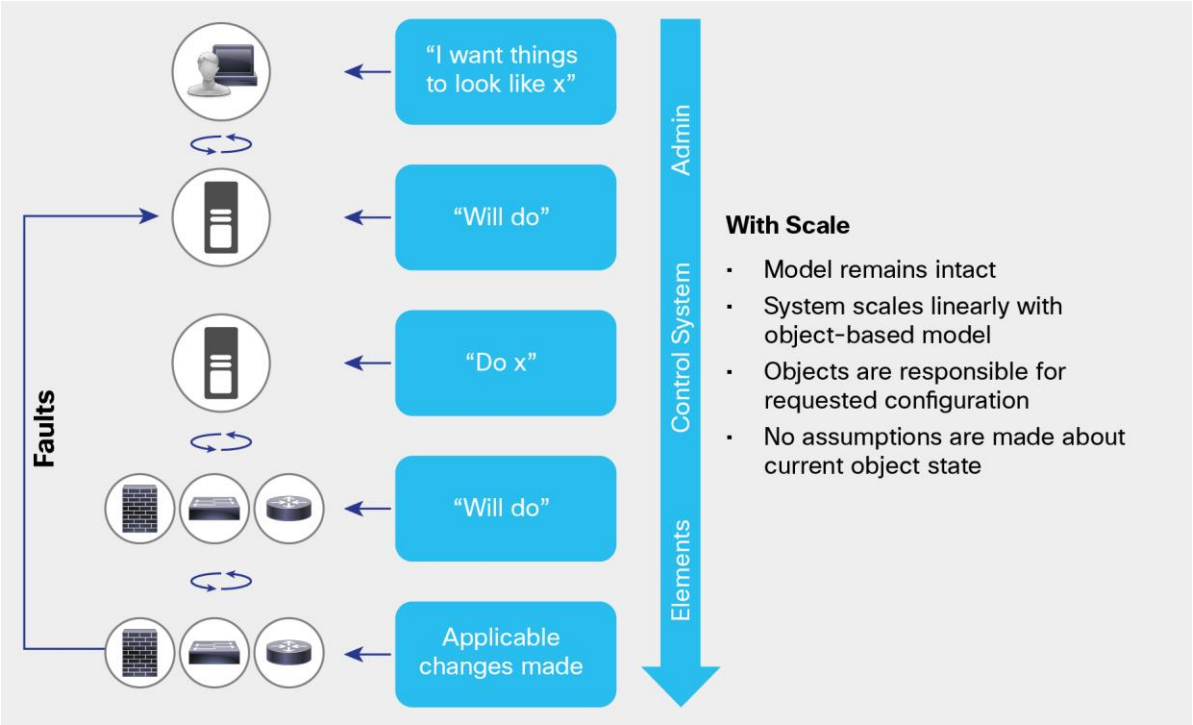
In addition, many of today's SDN solutions increase complexity by separating virtual and physical domains while offering little or no visibility between the two. Many solutions suffer from scalability and performance limitations that result from the use of a centralized, rather than a distributed, control plane. Additionally, they tend to use fixed, rigid schemas for interaction with devices, effectively reducing the network to a lowest common denominator feature set. For example, a protocol such as the Open vSwitch Database (OVSDB) management protocol allows configuration of only basic primitives such as ports and bridges and cannot easily be adapted to expose vendor innovations.

Declarative Control: A New Approach to SDN

Cisco® Application Centric Infrastructure (ACI) uses a fundamentally different approach. To understand it, you need to understand a new operating model called declarative control, which is based on a concept called promise theory. Promise theory offers a declarative control model based on scalable control of intelligent objects, originally proposed by Mark Burgess, founder of CFEngine. Declarative control dictates that each object is asked to achieve a desired state and makes a promise to reach this state, without being told precisely how to do so. The more traditional imperative model employs top-down management to specify every element of configuration to reach the desired state. One way to think about this distinction outside of the networking world is look at how things operate at an airport. You could think of the air traffic control system as a good example of a declarative control system. Air traffic controllers tell pilots to take off or land in particular places but they do not describe how to actually reach them. That job, actually flying the plane, adjusting the air speed, flaps, landing gear, etc. falls on the intelligent, capable, and independent pilot.

In a system managed through declarative control, underlying objects handle their own configuration state changes and are responsible only for passing exceptions or faults back to the control system. This approach reduces the burden and complexity of the control system and allows greater scale. This system increases scalability by allowing the methods of underlying objects to request state changes from one another and from lower-level objects (Figure 1).

Figure 1. Declarative Control in Large-Scale Systems



Advantages of Declarative Control

Declarative control systems such as Cisco ACI offer a number of advantages over imperative systems. They inherently separate out application, operation, and infrastructure requirements and allow each to be specified independently. This separation can accelerate application deployment by allowing the system, rather than the administrator, to coalesce these requirements. For example, the Cisco ACI fabric always operates as a routed network, and application policies are dynamically distributed to the leaf switches by the Cisco Application Policy Infrastructure Controller (APIC) on demand, with no human interaction. Additionally, Cisco ACI and its policy model aid application developers, who no longer need to understand the details of the underlying systems; the solution also creates an accurate, auditable record of each developer's requests that can be huge benefit to a cloud administrator. Systems built on declarative control can achieve high performance at scale with strong resiliency by moving complexity to edge devices, which do most of the processing. Finally, declarative systems, which allow policies to be specified in abstract terms, have strong interoperability characteristics. Multiple vendors can consume and honor the same policy without the need to have identical hardware configurations or software versions. In fact, vendors can continue to innovate on their platforms and expose new features as long as they honor the semantics of the abstract policy. This approach thus removes the lowest-common-denominator limitation established by today's software-based overlay solutions.

Introducing OpFlex: A New, Open Policy Framework

To implement declarative control, a new mechanism is required to transfer abstract policy from a network policy controller to a set of smart devices capable of rendering abstract policy. Unfortunately, existing protocols such as OVSDB favor imperative models with rigid schemas, so they are not appropriate for this use case. In fact, devOps tools such as Puppet or CFEngine take a similar approach to OpFlex in using declarative languages to configure server resources.

OpFlex was designed to augment rather than replace these tools by focusing on additional requirements of the network and policies that must span multiple network devices. For example, OpFlex includes a native mechanism for identity resolution used to define declarative policies between two different network endpoints.

Cisco, along with partners including Intel, Microsoft, Red Hat, Citrix, F5, Canonical, and Embrane, developed OpFlex to address this challenge. OpFlex is an open and extensible policy protocol for transferring abstract policy in XML or JavaScript Object Notation (JSON) between a network policy controller such as the Cisco APIC and any device, including hypervisor switches, physical switches, and Layer 4 through 7 network services. Cisco and its partners are working through the IETF and open source community to standardize OpFlex and provide a reference implementation.

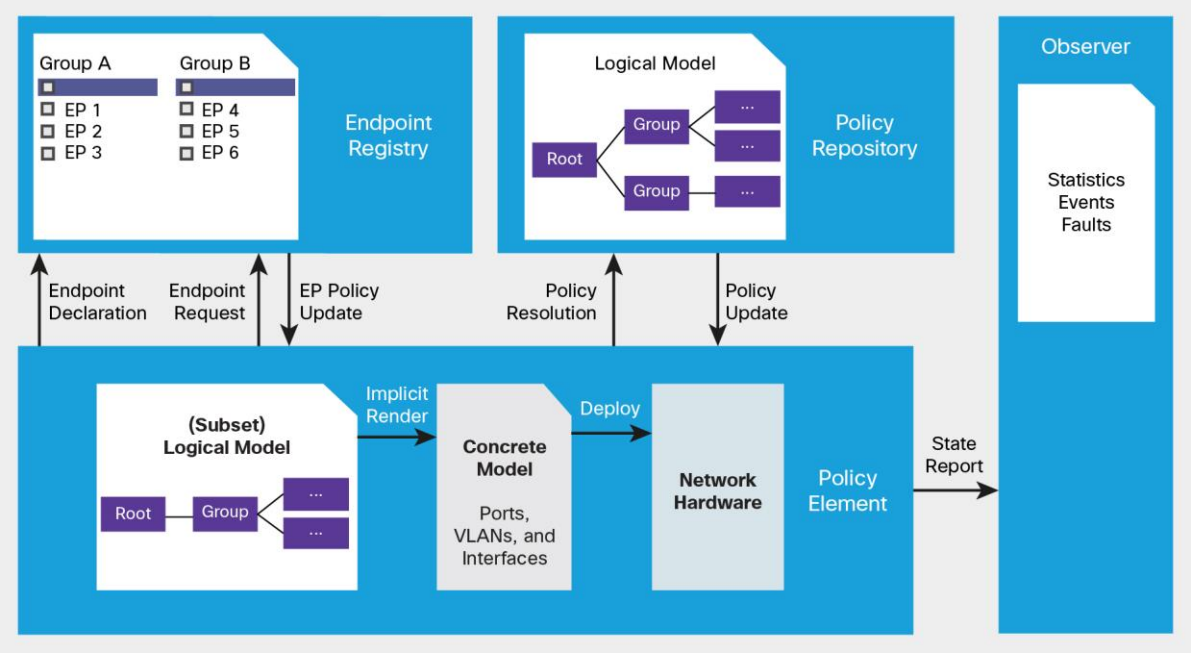
OpFlex Description

OpFlex is designed to allow a data exchange of a set of managed objects that is defined as part of an informational model. OpFlex itself does not dictate the information model and can be used with any tree-based abstract model in which each node in the tree has a universal resource identifier (URI) associated with it.

The protocol is designed to support XML and JSON (as well as the binary encoding used in some scenarios) and to use standard remote procedure call (RPC) mechanisms such as JSON-RPC over TCP. The use of a secure channel through Secure Sockets Layer (SSL) and Transport Layer Security (TLS) is also recommended.

The protocol defines a number of logical constructs required for its operation (Figure 2).

Figure 2. OpFlex Logical Model



Policy Repository

The policy repository (PR) is a logically centralized entity containing the definition of all policies governing the behavior of the system. In Cisco ACI, this function is performed by the Cisco APIC or by the leaf nodes of the network fabric. The policy authority handles policy resolution requests from each policy element.

Policy Element

A policy element (PE) is a logical abstraction for a physical or virtual device that implements and enforces policy. Policy elements are responsible for requesting portions of the policy from the policy authority as new endpoints connect, disconnect, or change. Additionally, policy elements are responsible for rendering that policy from an abstract form into a concrete form that maps to their internal capabilities. This process is a local operation and can function differently on each device as long as the semantics of the policy are honored.

Endpoint Registry

The endpoint registry (ER) stores the current operation state (identity, location, etc.) of each endpoint (EP) in the system. The endpoint registry receives information about each endpoint from the local policy element and then can share it with other policy elements in the system. The endpoint registry may be physically co-located with the policy authority, but it may also be distributed in the network fabric itself. In Cisco's ACI solution, the endpoint registry actually lives in a distributed database within the network itself to provide additional performance and resiliency.

Observer

The observer collects statistics, faults, and events from each policy element in the system. In Cisco ACI, this function is performed by the Cisco APIC, but it could also be separated into a separate system.

OpFlex Protocol Messages

Table 1 summarizes the RPC methods that OpFlex supports. It is not intended to provide a full description but to show how different entities interact through the protocol.

Table 1. RPC Methods Supported by OpFlex

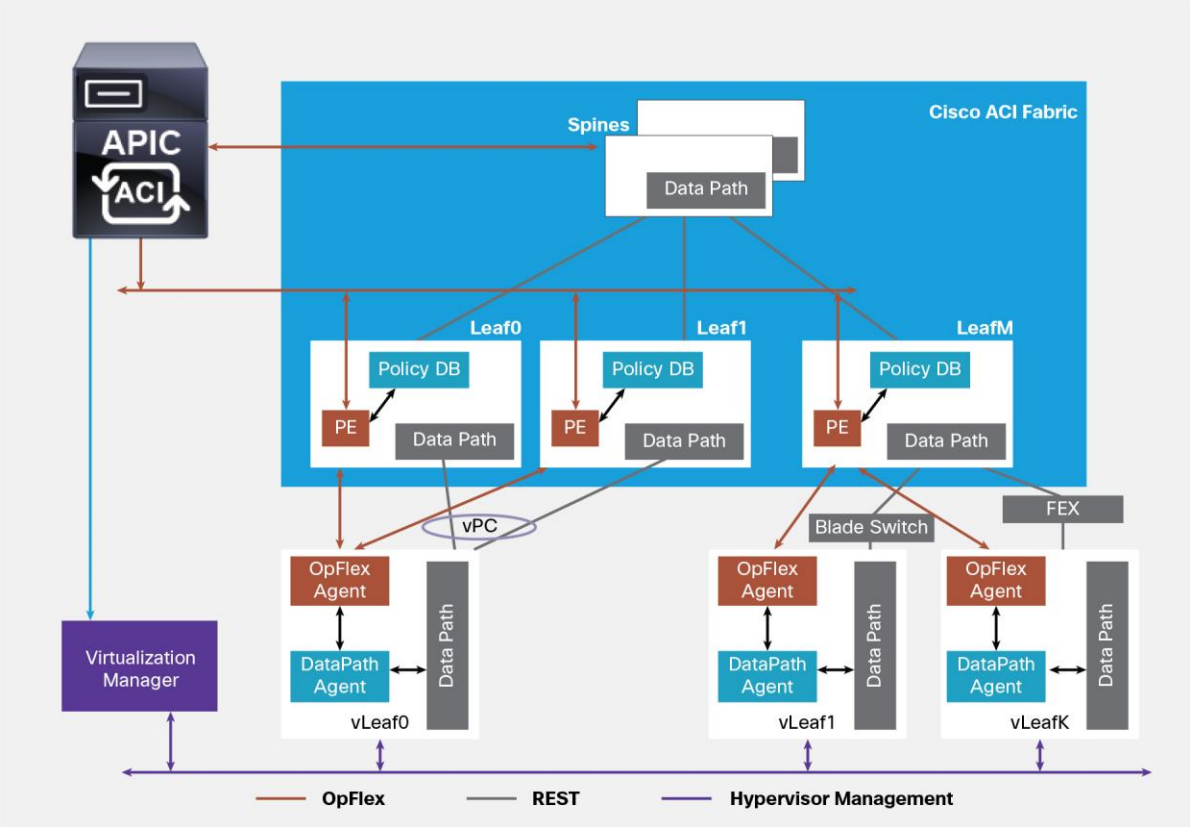
Command	From	To	Description
Identity	Any	Any	Is the first OpFlex message between any entity and its peer (frequently the PE and PR)
Policy Resolution	PE	PR	Retrieves a set of policies for an object
Policy Update	PR	PE	Sent to PEs when the policy definition for policies for which the PE has requested resolution has changed
Policy Trigger	PE	PE	Sends a policy trigger from one PE to a peer PE to trigger a policy resolution on the peer; may be in response to an attachment event that affects an upstream switch
Endpoint Declaration	PE	ER	Indicates the attachment of a new endpoint
Endpoint Request	PE	ER	Queries for an endpoint using a set of identifiers; for example, endpoints may be looked up by MAC address
Endpoint Policy Update	ER	PE	Sent by the ER when a change relating to EP declaration has occurred
State Report	PE	Observer	Sent by each PE to an observer to pass faults, events, and statistics

OpFlex and Cisco ACI: Virtual Leaf Architecture

OpFlex can be used for a number of purposes in Cisco ACI. One common use case involves a policy-enabled virtual switch (vLeaf), a software switch that supports some or all of the Cisco ACI policy model. In this case, the vLeaf effectively plays the role of an edge device and allows policy enforcement directly in the hypervisor. This allows traffic between two virtual machines on the same host to be switched locally with full policy enforcement. From an OpFlex perspective, each vLeaf peers with physical leaf to which it attached to request and exchange policy information. Additionally, OpFlex the Cisco APIC interacts with various hypervisor management systems to configure the virtual switches as needed.

Figure 3 below shows this system in action.

Figure 3. vLeaf Architecture for OpFlex



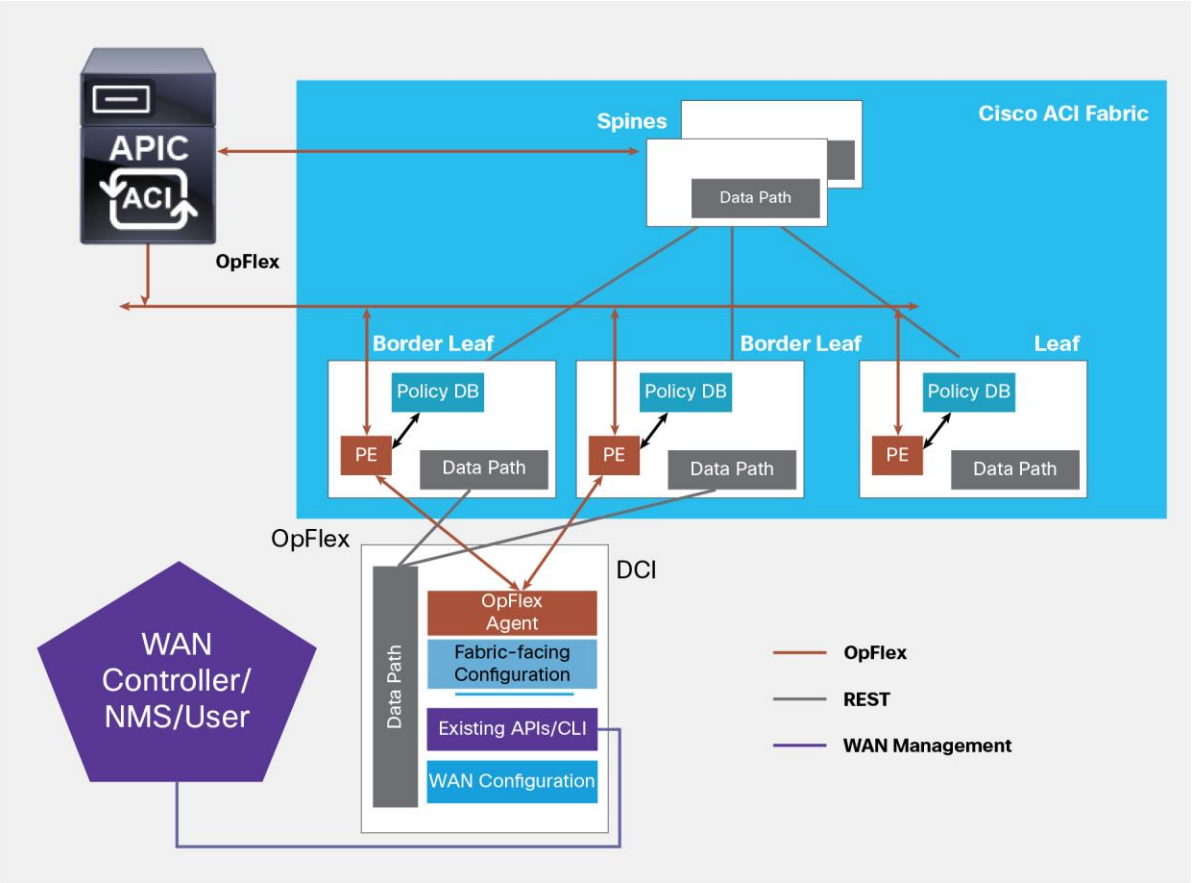
Cisco and ACI: Data Center Interconnect

OpFlex can also be used to communicate policy from a border leaf to a router configured as a datacenter interconnect. In this configuration, the router can be thought of as having two sides: a ACI fabric facing side that peers with a border leaf over OpFlex and a WAN facing side that can be configured manually or through a separate network management system or controller. Cisco's ASR9000 and Nexus 7000 will support OpFlex in this configuration.

As you can see from the diagram below, as in the vLeaf case, the DCI box communicates directly with the border leaf to exchange policy. The goal of this integration is to focus on automating frequently changing per-tenant configurations through OpFlex. Today, the ACI information model does not support a complete end-to-end WAN configuration. For example, parameters that may be exchanged between a border leaf and DCI device include a tenant ID, VRF ID, ASN ID, and DCI-IP.

Over time, as Cisco expands its policy controller framework to cover WAN, users will be able to completely configure a DCI device and gather WAN statistics over OpFlex using abstract policy.

Figure 4. Data Center Interconnect Setup with OpFlex As Part of a Cisco ACI Deployment Handling the Fabric Facing Configuration



Conclusion

OpFlex offers a powerful new tool for managing infrastructure using declarative control. Supported by a robust ecosystem of partners, including Cisco, Microsoft, Red Hat, Citrix, and F5, it offers a standardized, open source mechanism for transferring abstract policies between a controller and devices.

For More Information

http://www.cisco.com/go/aci.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-731302-02 04/15