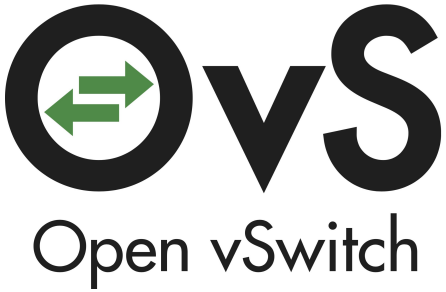


The Open vSwitch and OVN Projects

Netfilter Workshop 2017

Joe Stringer

Highlights from the Year

- The Open vSwitch project moved to the Linux Foundation
- Released the 2.6 and 2.7 series
- Moving to a more regular six month release interval
  - Release 2.8 in August
- First release of OVN

Who Works on the OVS Projects?

- 230 individual contributors
- Contributions from a wide variety of companies
- 16 “committers”
- Diversity of contributors has increased with OVN

OVS Project Releases

- Improved support for OpenFlow in every release
- Version 2.6
  - OVN
  - NAT support (Linux kernels)
  - QoS and policing for DPDK
  - Basic connection tracking on DPDK and Hyper-V
- Version 2.7
  - DPDK now fully supported
  - OVN: traffic shaping, DSCP
- Version 2.8
  - 802.1AD “QinQ”
  - DPDK support for NAT
  - OVN: DNS, RBAC, IPv6
  - L3 tunneling with GRE, VXLAN-GPE

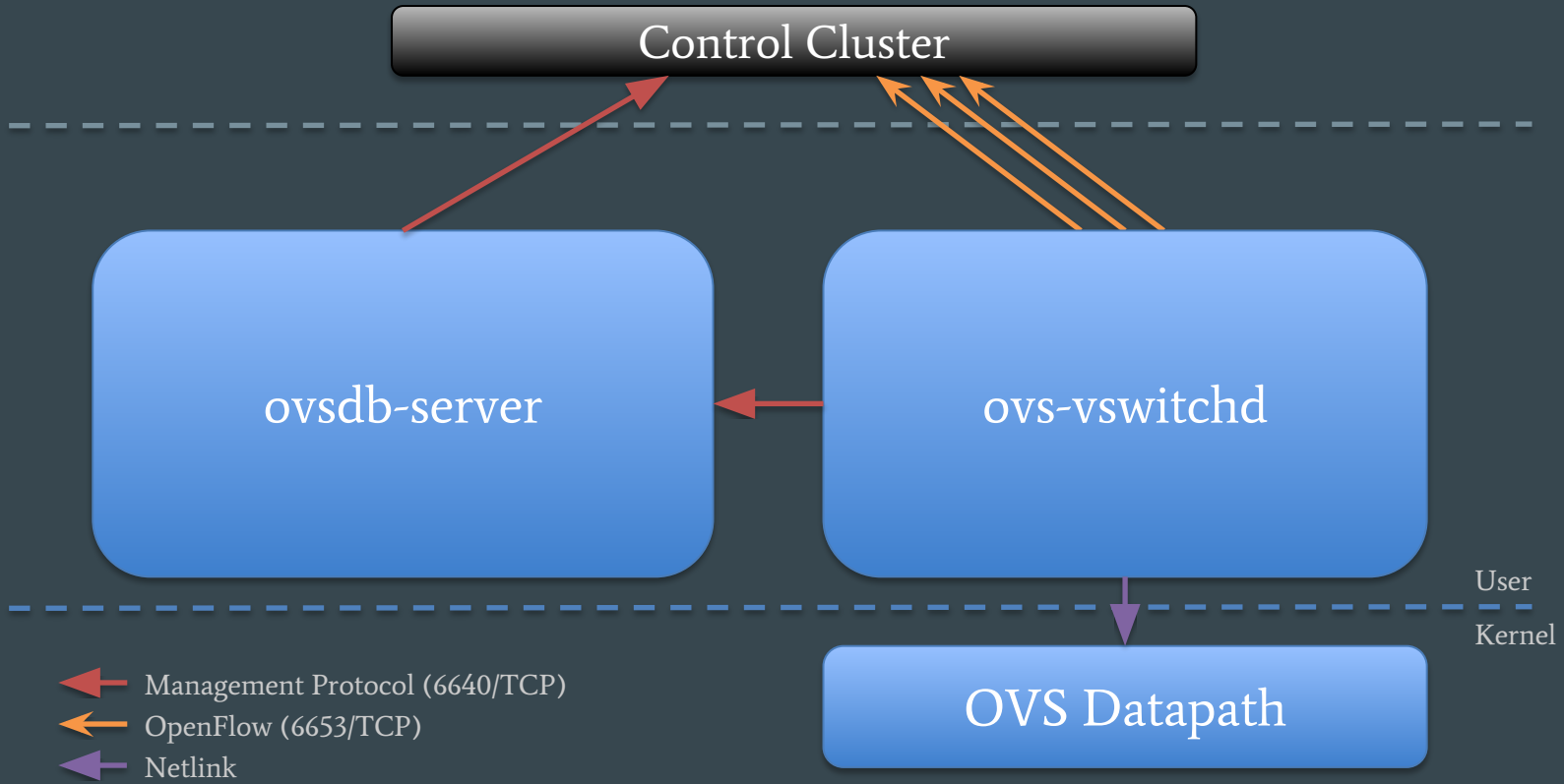
Open vSwitch

Open vSwitch Overview

- OVS is a multi-layer switch
- Visibility (NetFlow, sFlow, SPAN/RSPAN)
- Fine-grained ACLs and QoS policies
- Port bonding, LACP, tunneling
- Centralized control through OpenFlow and OVSDB
- Open source using Apache 2 license*
- Multiple ports to physical switches

* kernel - GPL

OVS Architecture



Platforms

- Linux kernel
- Containers
- DPDK
  - Bypasses the kernel and packets go straight to userspace
    - Potentially very fast if traffic doesn't need kernel
    - Need to recreate services supplied by kernel
- Hyper-V
  - Windows-based hypervisor
  - Different from Windows support, but that's also being worked on
- Non-Linux kernel datapaths sometimes lag on features provided by the kernel

Decoupled Design

- Decoupling Helps
  - A number of different SDN applications have been written without requiring changes to OVS.
  - A number of new OpenFlow protocols have been added without changes to kernel
  - A number of new platforms have been added by implementing just a new datapath
- Flow programming with slow-path/fast-path design often performs better than fixed-pipeline
- NSDI paper on design and implementation:
  - http://openvswitch.org/support/papers/nsdi2015.pdf

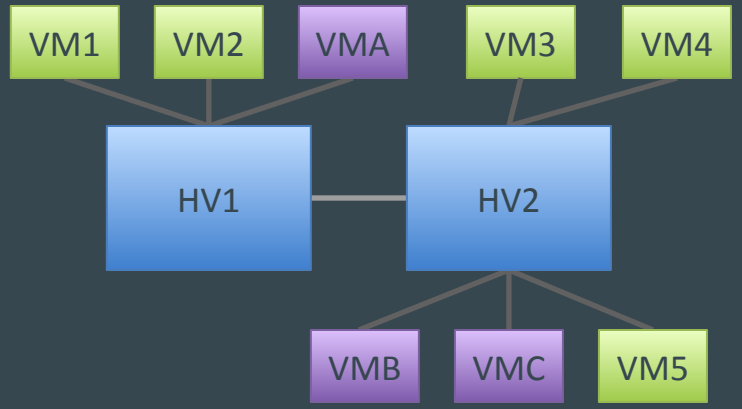
Futures: BPF, P4

- BPF provides a safe, virtual sandbox in the Linux kernel (and other platforms)
  - DPDK-like performance in Linux kernel with XDP
  - Potentially greater portability across kernel versions and platforms
  - Insert new functionality at run-time
- P4 is a domain-specific language for programming packet forwarding planes
  - Run-time addition of new matches and actions
  - Parser can be custom-tuned to important fields for faster flow lookup
  - New matches and actions can be written more compactly than in C
  - A single P4 match-action implementation can be shared across multiple datapaths
- Challenges for OVS
  - Bridging the models of OpenFlow, BPF, P4
  - Complexity allowed by BPF

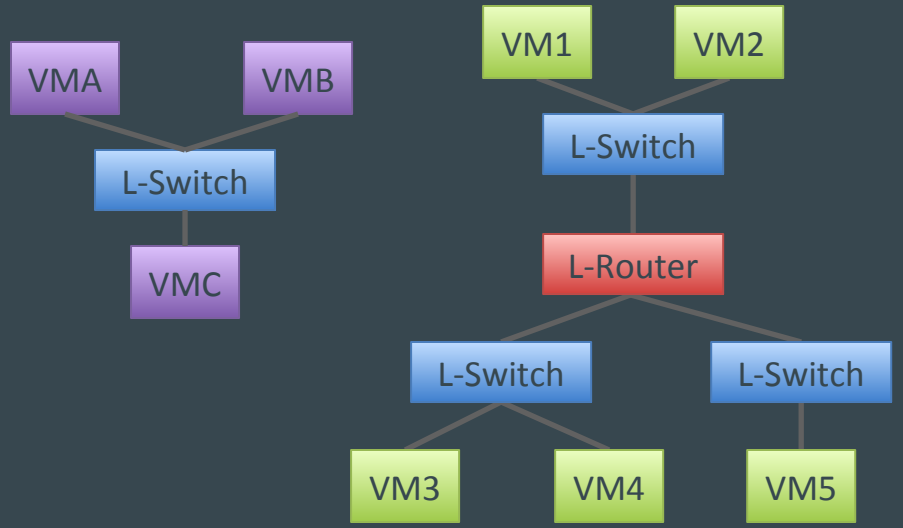
OVN

Virtual Networking Overview

Provides a logical network abstraction on top of a physical network



Physical



Logical

What is OVN?

- Virtual networking for Open vSwitch (OVS)
- Developed within the OVS project
- Linux Foundation Collaborative Project
- License under the Apache 2 license
- First release of OVN came with OVS 2.6
- Bindings available for OpenStack Neutron, Kubernetes, etc.

OVN Feature Overview

- Manages overlays and physical network connectivity
- Flexible security policies (ACLs)
- Distributed L3 routing, IPv4 and IPv6
- Native support for NAT, load-balancing, DHCP
- Works with Linux, DPDK, and Hyper-V
- L2 and L3 gateways
- Designed to be integrated into another system
    - OpenStack, Kubernetes, Docker, Mesos, oVirt

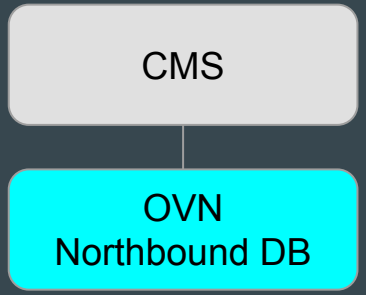
Goals

- Production-quality
- Straightforward design
- Scale to 1000s of hypervisors (each with many VMs/containers)
- Scale to 100s of thousands of ports

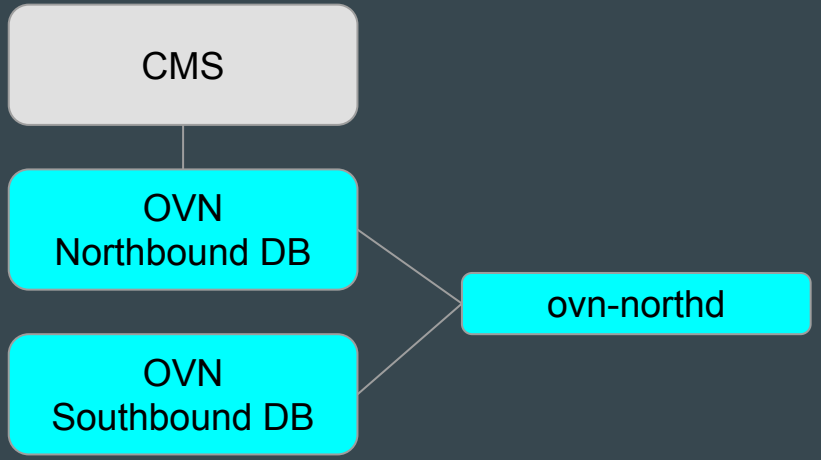
Designed to Scale

- Configuration coordinated through databases
- Local controller converts logical flow state into physical flow state
  - Centrally creating each hypervisor's view is expensive
  - Identical state sent to each hypervisor
- Desired state clearly separated from run-time state
  - Easier to reason about the system
  - Replication story clear
- Grouping techniques reduce Cartesian Product issues
  - High-level grouping constructs in database
  - Use of conjunctive match in switch

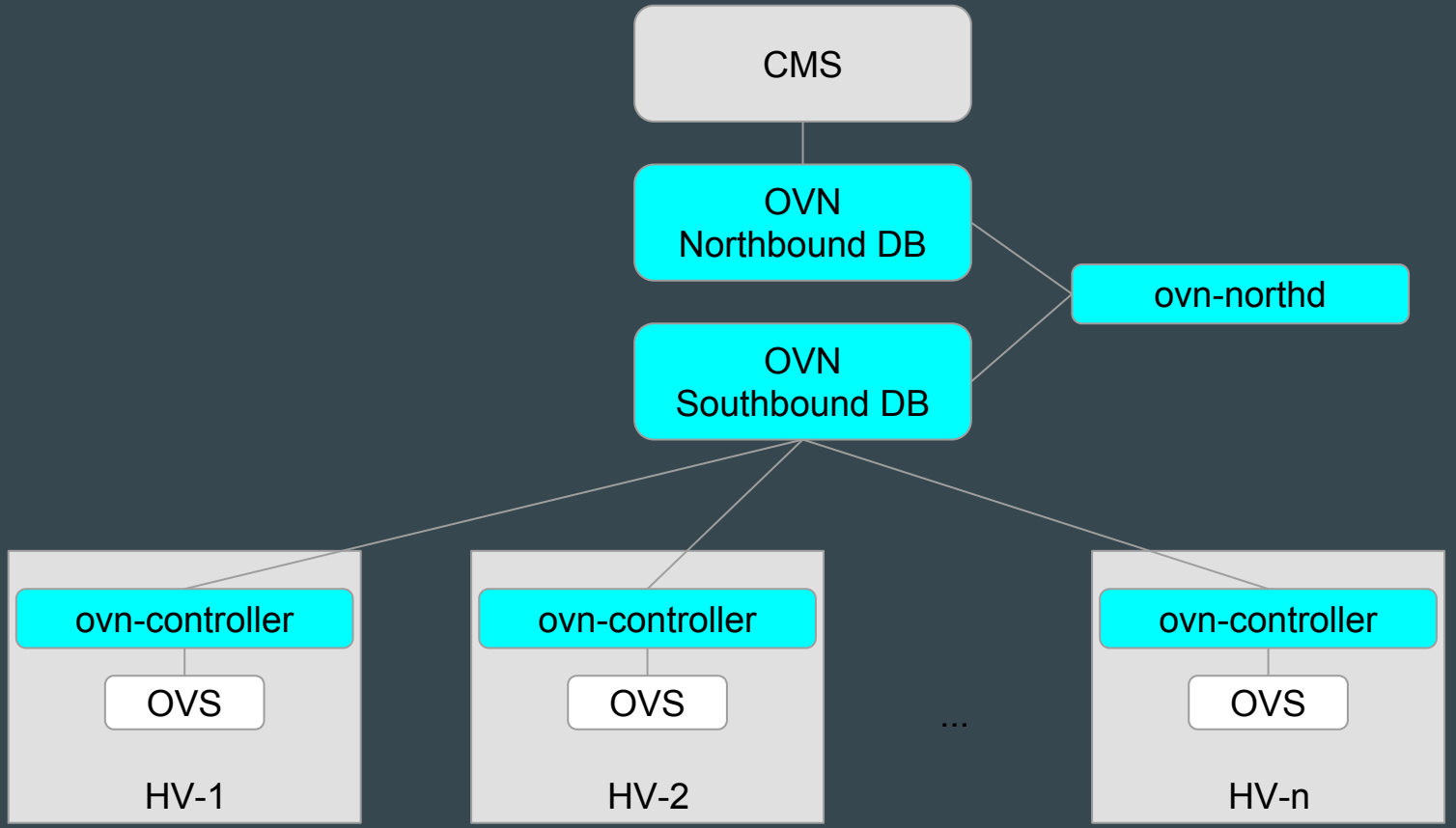
1. Logical configuration in Northbound DB



2. ovn-northd populates Southbound logical flows



3. Hypervisors generate physical flows



OVN Future work

- Database clustering
- Scaling improvements
- Service function chaining
- Encrypted tunnels
- Native DNS support
- ACL Logging

Other Resources

- Website
  - http://www.openvswitch.org/
- OVS/OVN Repository
  - https://github.com/openvswitch/ovs
- OpenStack OVN Integration
  - https://docs.openstack.org/developer/networking-ovn/
- Kubernetes OVN Plugin
  - https://github.com/openvswitch/ovn-kubernetes
- OVS Orbit Podcast
  - https://ovsorbit.org/

Thank you for attending!

...

Joe Stringer
Slides courtesy Ben Pfaff, Justin Pettit