Software Defined Networking

Dr. Nick Feamster
Professor

In this course, you will learn about software defined networking and how it is changing the way communications networks are managed, maintained, and secured.

Module: Network Virtualization

- Lessons
  - What is network virtualization and how is it implemented?
  - Examples of network virtualization and applications.
  - Virtual networking in Mininet
  - Virtualizing control: FlowVisor
  - Use case: Multi-tenant data centers (Nicira NVP)

This Lesson: Multi-Tenant Datacenters

- What is a multi-tenant datacenter?
  - Components: Network, compute, storage
- Virtualizing the network (case study: NVP)
- Challenges: Forwarding speed and scaling
- The role of SDN in network virtualization

Koponen, Teemu, et al. "Network Virtualization in Multi-tenant Datacenters." *NSDI*. April 2014.

Multi-Tenant Datacenter

- Single physical datacenter shared by many "tenant" users
  - Customers (Amazon, Rackspace)
  - Applications/services (Mail, Search, …)
  - Developers
- Challenges
  - Workloads require **different topologies, services**
  - Address space **overlaps with physical network**

Multi-Tenant Datacenter Architecture

- Each host in the datacenter has multiple VMs
  - Each host has a hypervisor with an internal switch
  - Switch forwards to local VM or another hypervisor

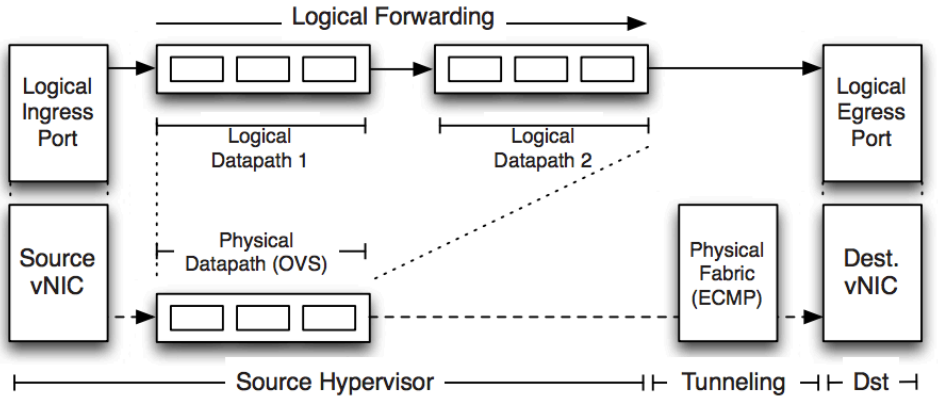
- **Need:** **Network hypervisor** to build right network abstractions for tenants

Network Hypervisor Abstractions

- **Control abstraction:** Tenants define a set of logical network data-plane elements that they can control.

- **Packet abstraction:** Packets sent by endpoints should see the same service as in a “native” network.

Implementing the Abstractions

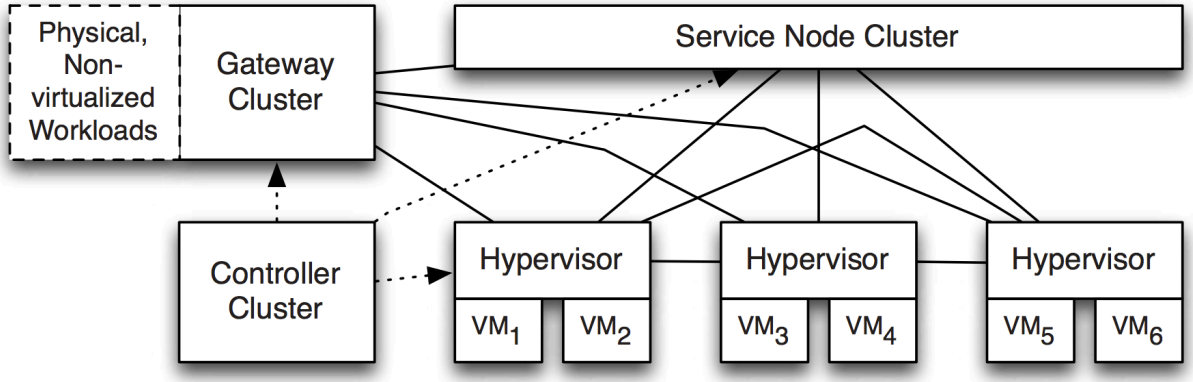


- Network hypervisor sets up tunnels between host hypervisors
  - Multicast implemented as an overlay service
- Physical network simply sees IP packets
- Centralized SDN controller configures the hosts' virtual switches
- Logical datapath implemented entirely on the sending host

Implementing the Logical Datapath

- Tunnel endpoints are virtual switches running on host hypervisors
  - Implemented with Open vSwitch
- Controller cluster can
  - Modify flow table entries
  - Set up tunnels

Controller Structure



- Hypervisors and physical gateways provide the controller with location and topology information
- Service providers configure the controller
- Forwarding state pushed to OVS via OpenFlow

Challenges

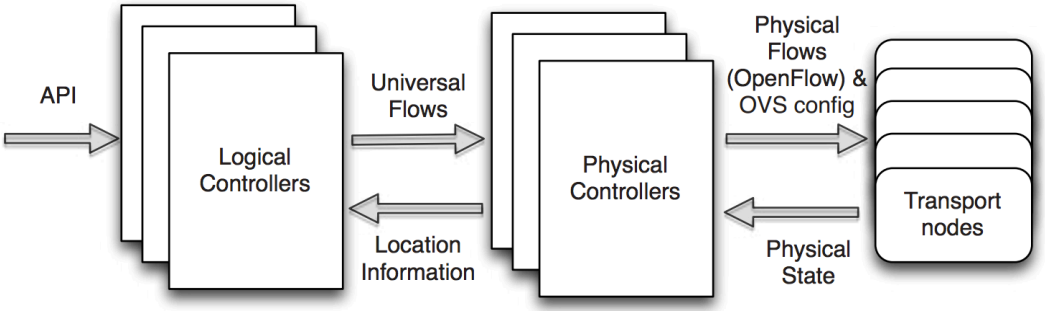
- **Datapath design:** Making software switching at end hosts fast

- **Scaling the computation:** computing the logical datapaths and tunnels

Making the Datapath Fast

- Exact-match flows in kernel
  - User-space program matches on full flow table, installs exact match in the kernel
  - Future packets for the same flow are matched in-kernel
- Hardware offloading of encapsulated packets requires some additional tricks.

Scaling Controller Computation



- Two-layer distributed controller
  - **Logical controllers:** Compute flows and tunnels for logical datapaths (and universal flows)
  - **Physical controllers:** Communicate with hypervisors, gateways, and service nodes
- Logical controller avoids dealing with the full mesh of tunnels.

Takeaways: The Role of SDN

- Network virtualization != SDN
  - Predates SDN
  - Doesn't require SDN
- Easier to virtualize an SDN switch
  - Run separate controller per virtual network
  - Partition the space of all flows
  - Use open interface to the hardware
- Network virtualization can also use software switches

Conclusion

- The rise of virtualization and multi-tenant datacenters has created a need for network virtualization
- SDN plays some role in configuring logical data paths and tunnels
- Interestingly, in the case of NVP, it all happens at the host!