大阪大学
2015年度後期 情報ネットワーク学演習II

SDN Controller Development for Commercial Cloud Services

Jan. 6, 2016

Yasunobu Chiba

NEC Corporation

Scope of talk

- Explain a production SDN controller development use case to provide hints for the final exam

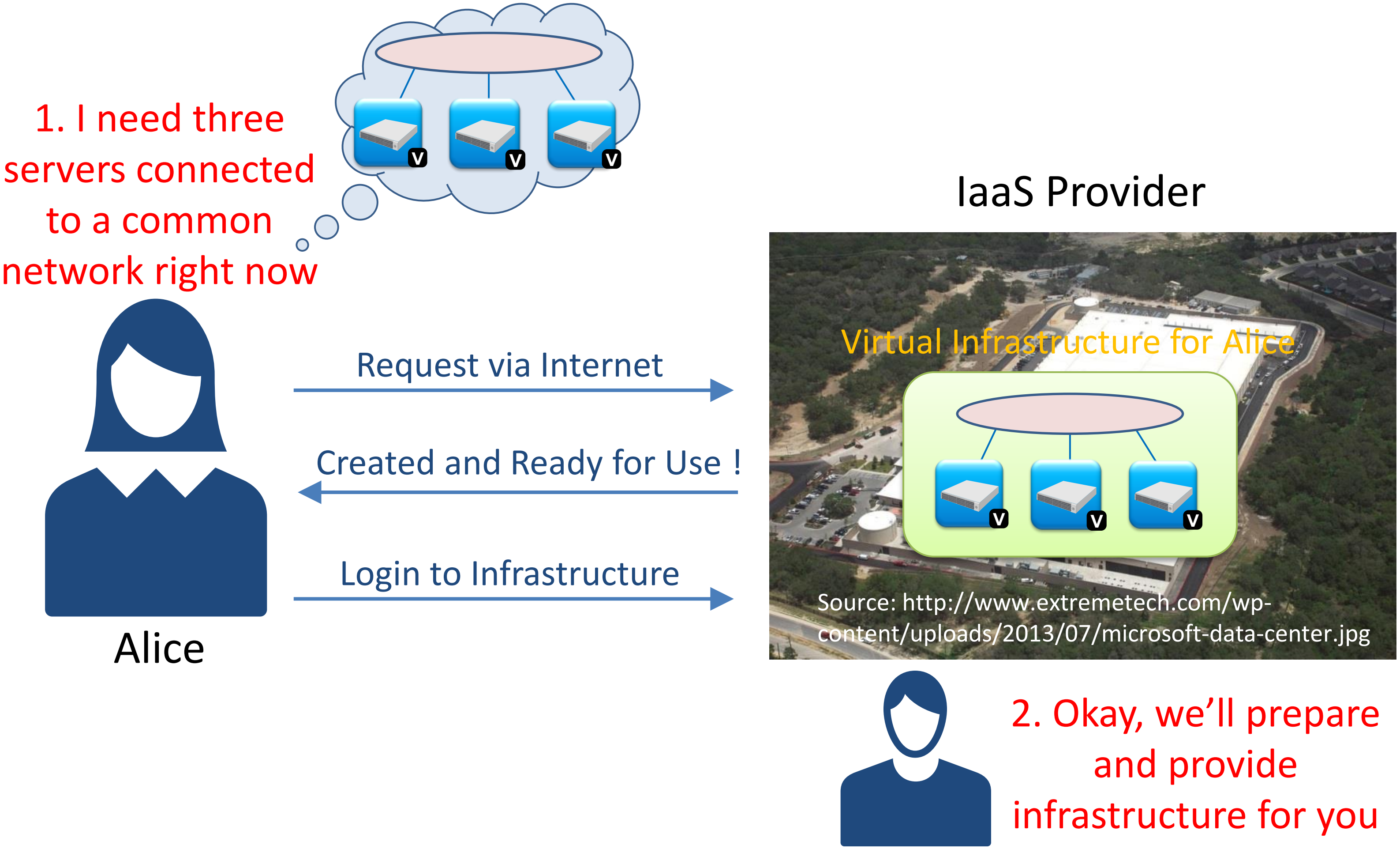
Agenda

- Background
- Requirements
- Design strategy
- Architecture and components
- Evaluation

Background

- Customer of SDN controller:
  - Cloud service provider providing Infrastructure as a Service (IaaS)
- Problems:
  - Need to provide a large number of virtual networks for tenants but VLAN does not scale in terms of # of virtual networks (limited to 4094)
  - Take some time to set up virtual networks while servers (virtual machines) can be deployed instantly

Infrastructure as a Service (IaaS)



1. I need three servers connected to a common network right now

Alice

Request via Internet

Created and Ready for Use !

Login to Infrastructure

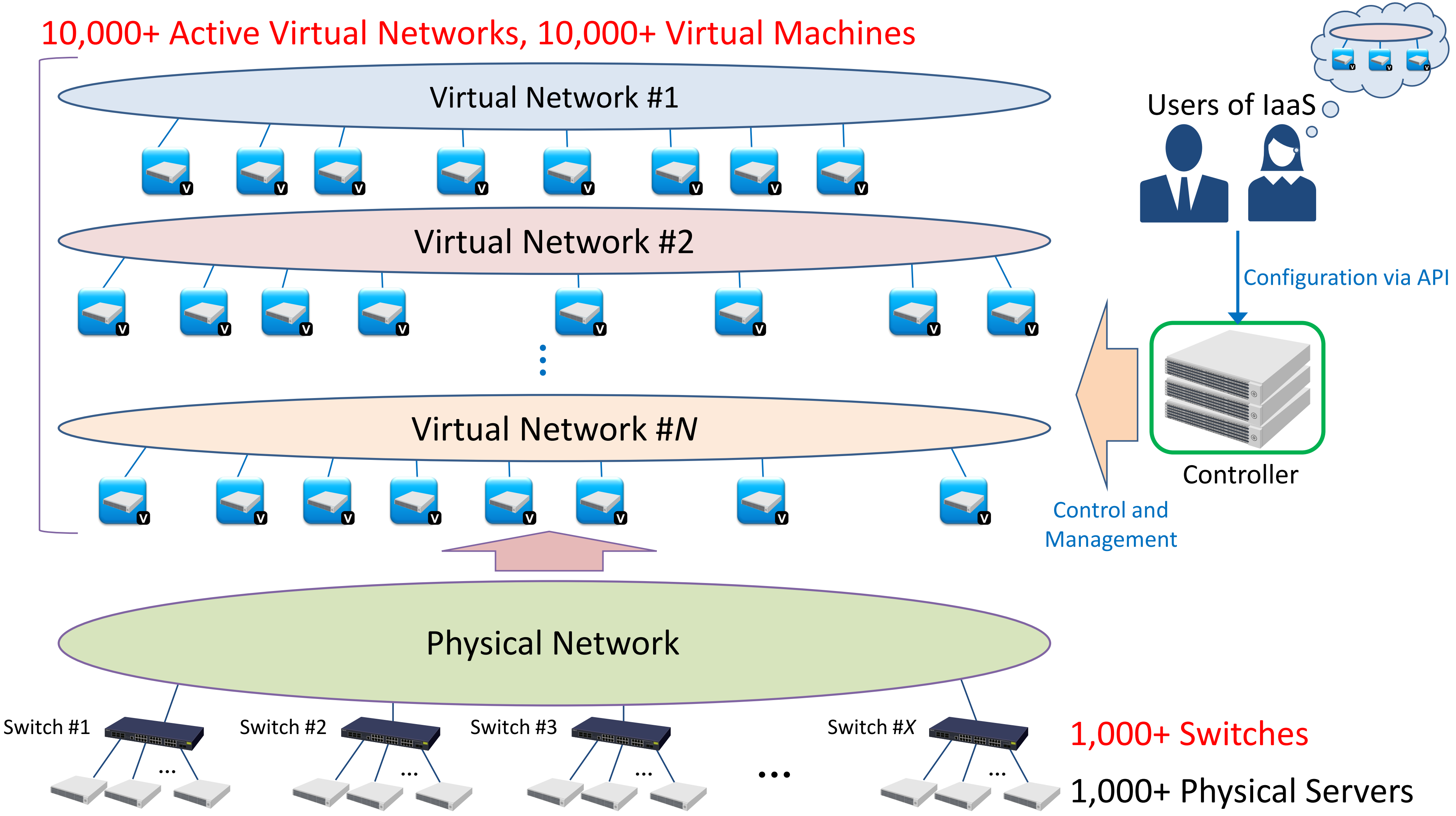
IaaS Provider

Virtual Infrastructure for Alice

Source: http://www.extremetech.com/wp-content/uploads/2013/07/microsoft-data-center.jpg

2. Okay, we'll prepare and provide infrastructure for you

Requirements in a nutshell



10,000+ Active Virtual Networks, 10,000+ Virtual Machines

Virtual Network #1

Virtual Network #2

Virtual Network #*N*

Physical Network

Switch #1

Switch #2

Switch #3

Switch #*X*

1,000+ Switches

1,000+ Physical Servers

Users of IaaS

Configuration via API

Controller

Control and Management

Detailed requirements for SDN Controller

- Functional Requirements
  - Provide virtual layer 2 networks for tenants (as well as virtual machines)
  - Manage association among virtual networks and virtual machines/switch ports
  - Associate a switch port with MAC addresses located on the switch port
  - All operations above can be done via Representational State Transfer (REST) interface
  - All operations can be done within a few seconds

- Non-functional Requirements
  - 1K+ switches must be managed
  - 10K+ active virtual networks must be managed
  - 10K+ virtual machines must be connected to virtual networks

10,000+ Virtual Machines ???

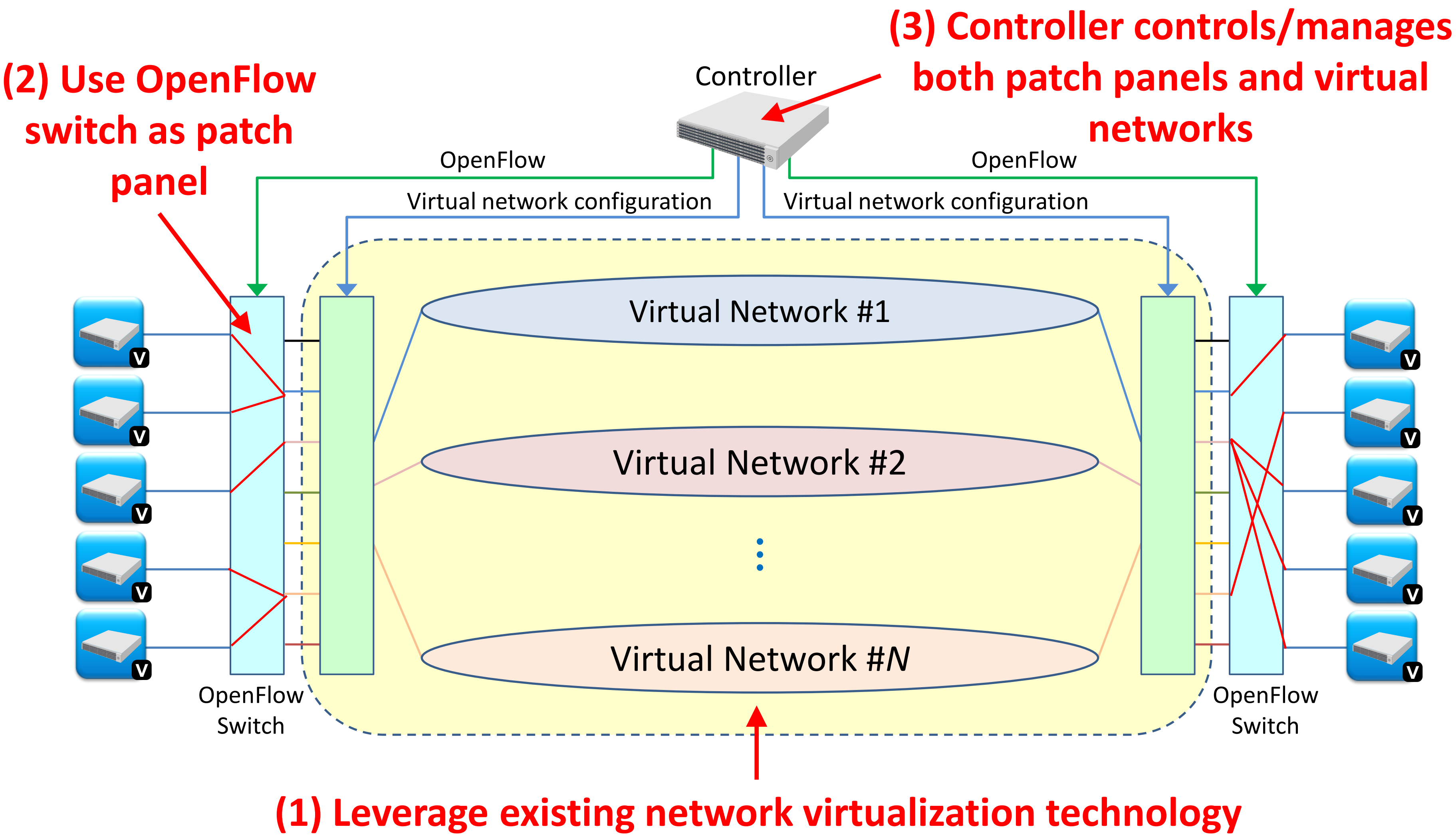
10,000+ Virtual Networks ???

Operation is done in a few seconds ???

1,000+ Switches ???

REST Interface ???

Design strategy


(3) Controller controls/manages both patch panels and virtual networks
Controller
(2) Use OpenFlow switch as patch panel
OpenFlow
OpenFlow
Virtual network configuration
Virtual network configuration
Virtual Network #1
Virtual Network #2
Virtual Network #N
V
OpenFlow Switch
OpenFlow Switch
(1) Leverage existing network virtualization technology

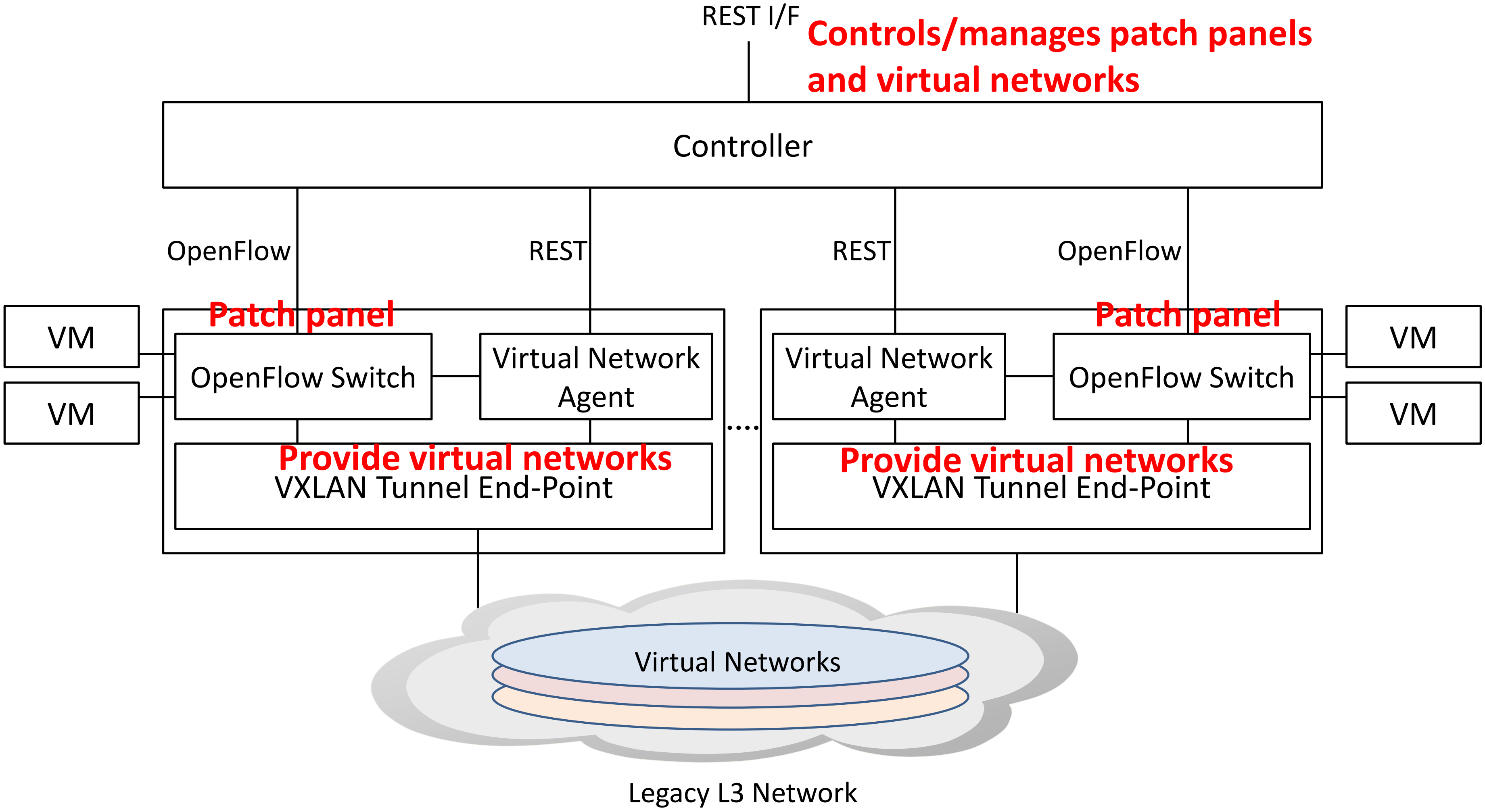
Network virtualization technologies

Technology	PDU	Underlay network	Connectivity	Maximum # of isolated networks / Maximum # of isolated links between a pair of hosts	Note
VLAN - 802.1Q	Ethernet	Physical	Any-to-Any	4094	# of networks is limited by switch implementation.
VLAN - 802.1ad (Q-in-Q)	Ethernet	Physical	Any-to-Any	16760836	# of networks is limited by switch implementation and it is typically not allowed to accommodate the maximum number.
VLAN - 802.1ah (MAC-in-MAC)	Ethernet	Ethernet	Point-to-Point	1	Only a single tunnel can be created for a pair of MAC addresses.
Pseudo-wire (PWE3)	Ethernet	MPLS	Point-to-Point	1048560+	# of links depends on router implementation and it is typically limited to fewer than the maximum number.
VPLS	Ethernet	MPLS	Any-to-Any	Unspecified	# of networks depends on topology and router implementation.
MPLS IP-VPN	IP	MPLS	Any-to-Any	Unspecified	

Network virtualization technologies – cont'd

Technology	PDU	Underlay network	Connectivity	Maximum # of isolated networks / Maximum # of isolated channels between a pair of hosts	Note
L2TP	Ethernet / IP	UDP/IP	Point-to-Point	65536 * tunnels * sessions	Multiple tunnels/sessions can be created for a pair of hosts.
EtherIP	Ethernet	IP	Point-to-Point	1	Only a single link can be created for a pair of IP addresses.
GRE	Ethernet / IP	IP	Point-to-Point	4294967296	2^32 tunnels can be created for a pair of IP addresses.
VXLAN	Ethernet	UDP/IP	Any-to-Any	16777216	
NVGRE	Ethernet	IP	Any-to-Any	16777216	
IP-in-IP	IP	IP	Point-to-Point	1	Only a single tunnel can be created for a pair of IP addresses.
IPsec Tunnel	IP	IP	Point-to-Point	4294967296	2^32 tunnels can be created for a pair of IP addresses. The number is limited by SPI.
LISP	IP	IP	Point-to-Point	1	Only a single tunnel can be created for a pair of IP addresses.
PPP	Ethernet / IP	Ethernet, UDP/IP, etc.	Point-to-Point	1	In PPPoE case, 65536 PPP sessions can be created for a pair of MAC addresses.

System architecture



Controls/manages patch panels and virtual networks

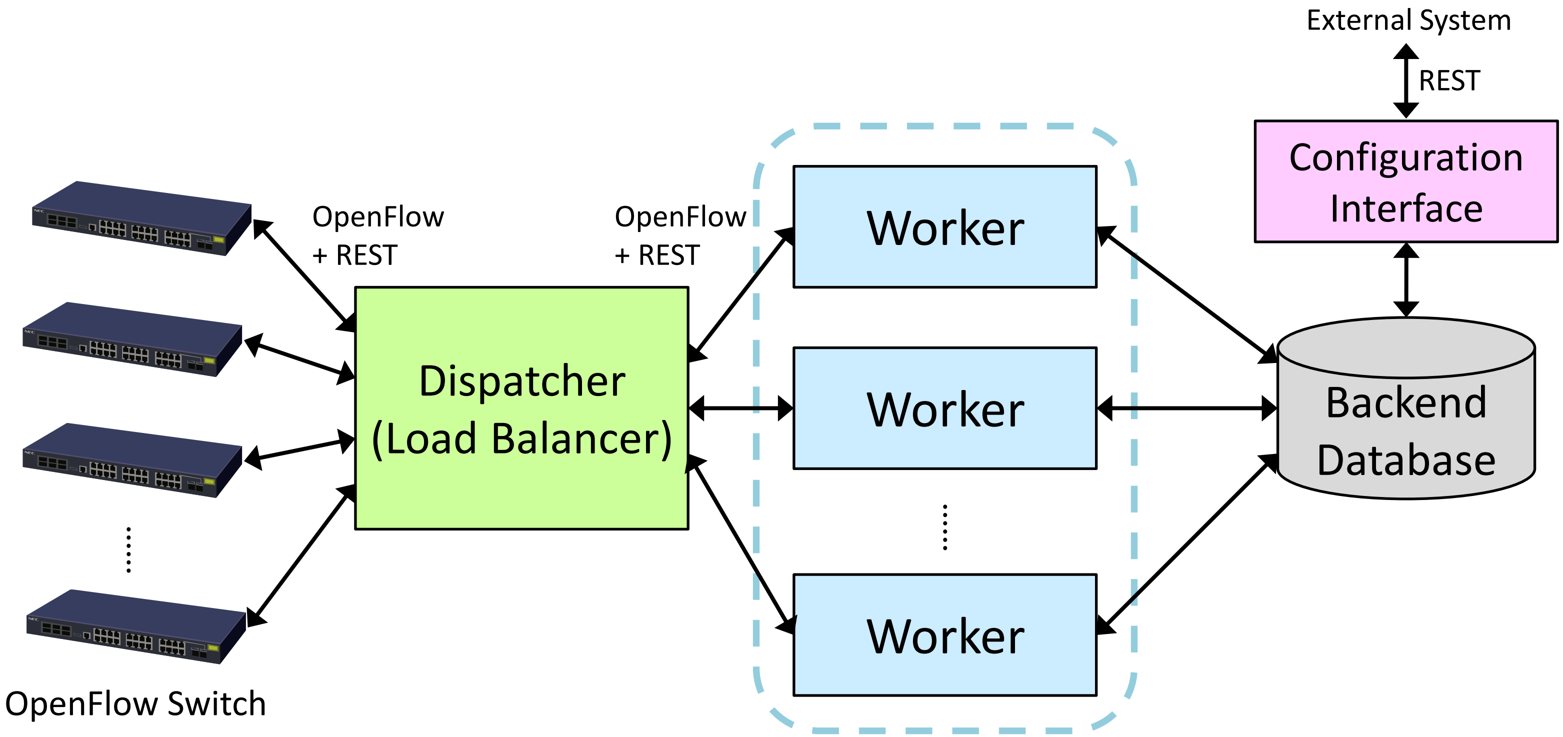
Patch panel

Patch panel

Provide virtual networks

Provide virtual networks

Controller design

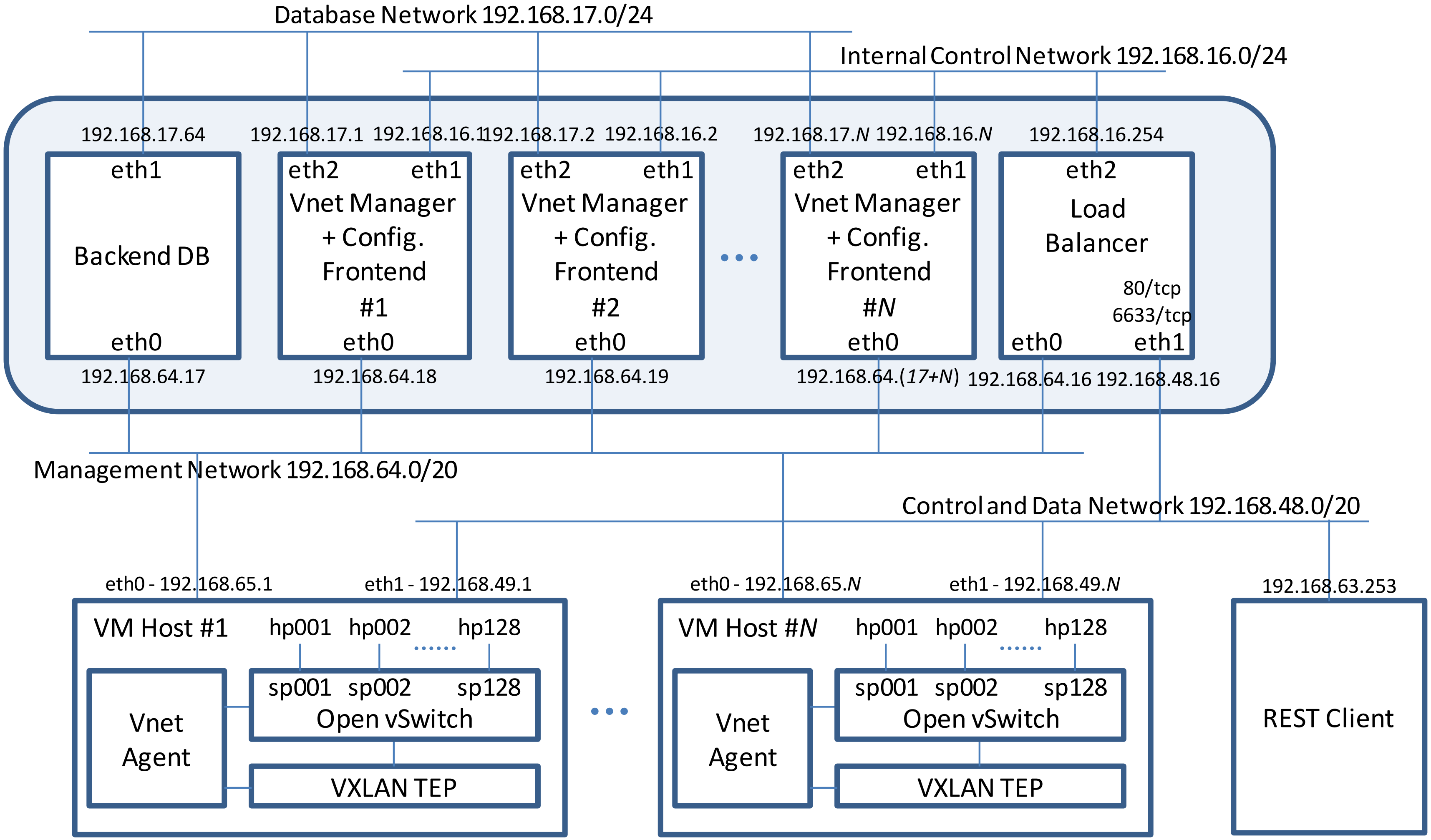


REST interface design

Path	Method	Request Parameters		Behavior
		Key	Description	
/networks	POST	id	A unique identifier of the network.	Create a new network associated.
		description	Description (text string) of the network.	
/networks/<net_id>	DELETE	-	-	Delete the network identified by net_id.
	POST	id	A unique identifier of the switch port.	Attach a switch port to the network identified by net_id.
		datapath_id	Datapath identifier of the switch which the switch port belongs.	
		number	Port number and name of the switch port. *number* and*name* are exclusive and one of them must be provided.	
		name		
		vid	VLAN identifier of the switch port. You can multiplex multiple networks on a single switch port with 802.1q VLAN.	
		description	Description (text string) of the switch port.	
/networks/<net_id>/ports/<port_id>	DELETE	-	-	Detach the switch port identified by port_id from the network identified by net_id.
/networks/<net_id>/ports/<port_id>/mac_addresses	POST	address	MAC addresses to be associated with the switch port.	Associate a MAC address to the switch port identified by port_id and net_id.
/networks/<net_id>/ports/<port_id>/mac_addresses/<mac_addresses>	DELETE	-	-	Detach the MAC address from the switch port.

Reference: https://rawgit.com/trema/virtual-network-platform/master/doc/api/api.html

Evaluation setup



Evaluation items and results

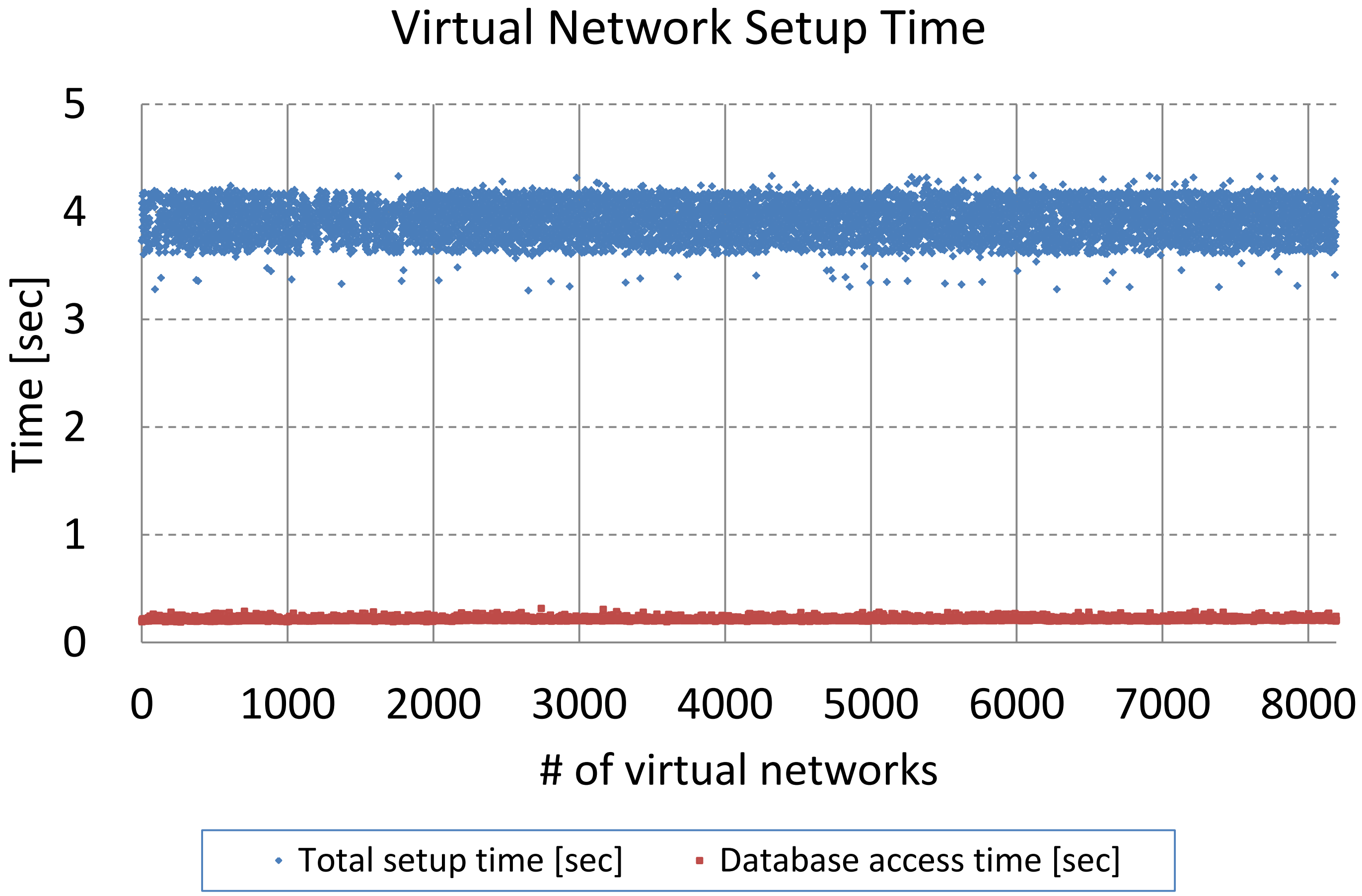
- # of switches that can be managed
  - 410 - 412 switches per a single Virtual Network Manager were connected and initialized properly
    - Switch daemons were not able to run due to insufficient memory (system memory was 2 GB)
  - 1024 switches were connected and initialized with three Virtual Network Managers

Evaluation items and results

- # of virtual networks that can be managed
  - 16384 virtual networks that have 8 ports (virtual machines) each were successfully created with 1024 switches and three Virtual Network Managers

- Virtual network setup time
  - Setup is done in several seconds and setup time did not increase even if we have a number of virtual networks
  - Database access time was constant and a minor factor

Evaluation result – Setup time



- Setup time does not increase even if we have a number of virtual networks
- Database access time is constant and a minor factor

Conclusion

- Explained actual virtual network deployment in a commercial data center
- Virtual networks are constructed and managed by leveraging existing virtual network technology and OpenFlow
- Confirmed the design is feasible and satisfies customer requirements

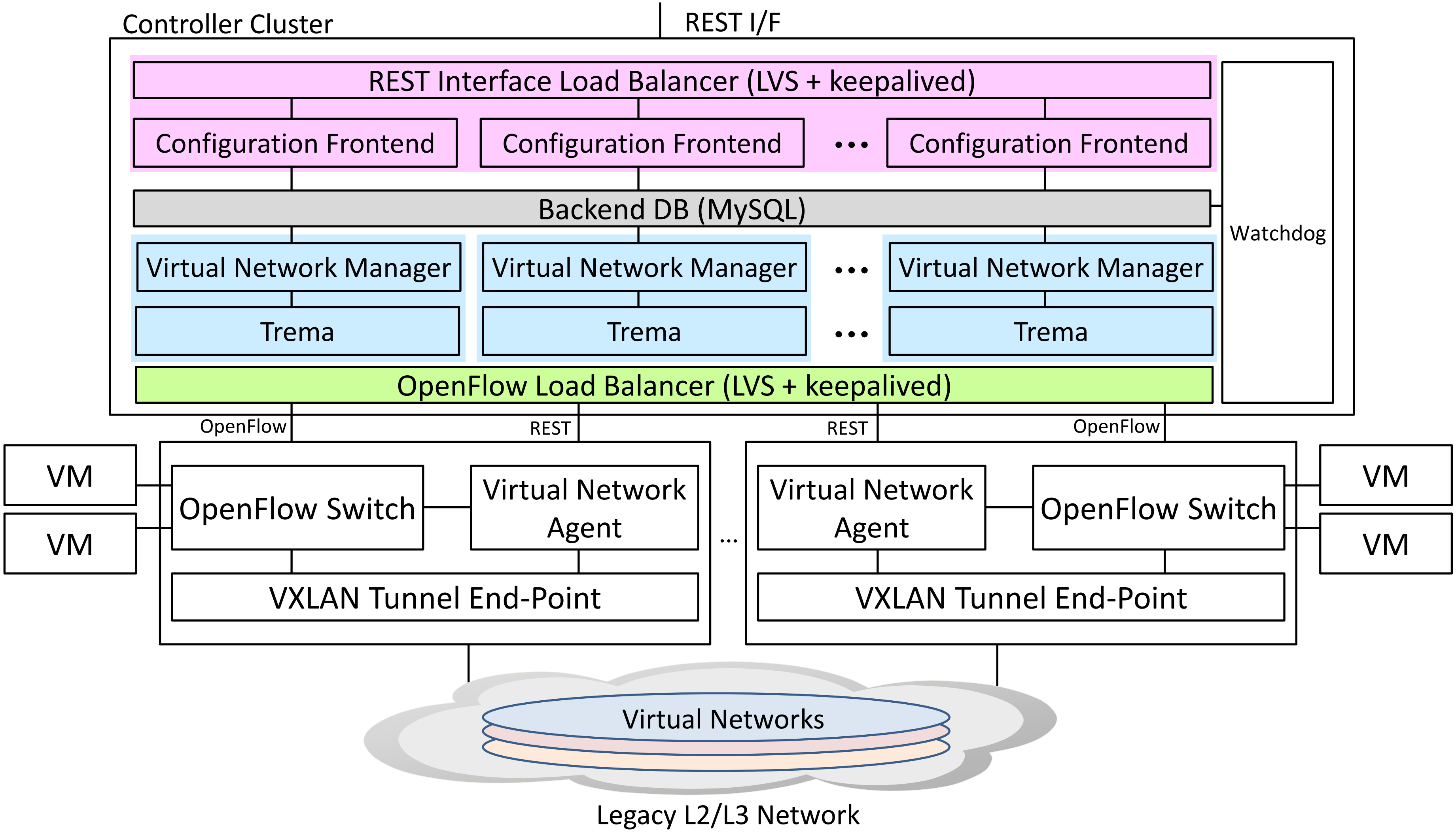
Keys to Successful Final Exam

- Clearly identify and state a single problem to be solved
- Study and leverage existing off-the-shelf technologies (Don't reinvent the wheel!)
- Design and develop a system combining off-the-shelf technologies and your unique idea

FIN

BACKUP

Implementation



Components

- Controller Cluster – software suite consists of
  - Virtual Network Manager
  - Trema
  - Backend DB
  - Configuration Frontend
  - OpenFlow Load Balancer (LVS + keepalived)
  - REST Interface Load Balancer (LVS + keepalived)
- Virtual Network Agent
- VXLAN Tunnel Endpoint
- OpenFlow Switch

Components

- Virtual Network Manager
  - Retrieves configuration from Backend DB and installs/removes flow entries to/on switches
  - Developed on top of Trema library
  - Multiple instances can be run at the same time for redundancy/performance

- Trema
  - Is unmodified Trema core modules (switch manager and daemons)

- Backend DB
  - Stores virtual network configuration
  - Stores operational states of switches and Virtual Network Manager
  - Implemented with MySQL
  - Can be clustered for redundancy/performance

Components

- Configuration Frontend
  - Provides REST interface
  - Receives requests from clients to update virtual network configuration
  - Implemented with Sinatra
  - Multiple instances can be run at the same time for redundancy/performance
- OpenFlow Load Balancer
  - Distributes control traffic between Virtual Network Managers and OpenFlow switches
  - Acts as a simple L4 load balancer
  - Implemented with Linux Virtual Server (LVS) and keepalived
  - Can be clustered for redundancy

Components

- REST Interface Load Balancer
    - Distributes traffic between Configuration Frontend and clients
    - Acts as a simple L4 load balancer
    - Implemented with Linux Virtual Server (LVS) and keepalived
    - Can be clustered for redundancy

Components

- Virtual Network Agent
  - Receives requests from Virtual Network Manager and configures VXLAN Tunnel Endpoint and OpenFlow switch
  - Notifies Virtual Network Manager if specific events (system reboot etc.) happened
  - Implemented with Sinatra

Components

- VXLAN Tunnel Endpoint
  - Is a VXLAN Tunnel Endpoint implementation defined in the VXLAN spec.
- OpenFlow Switch
  - Is unmodified Open vSwitch

References

- Virtual Network Platform
  - https://github.com/trema/virtual-network-platform
- Trema
  - https://github.com/trema/trema
- Linux Virtual Server
  - http://www.linuxvirtualserver.org/
- Keepalived
  - http://www.keepalived.org/
- Sinatra
  - http://www.sinatrarb.com/
- MySQL
  - http://www.mysql.com/
- VXLAN
  - http://www.ietf.org/id/draft-mahalingam-dutt-dcops-vxlan-06.txt
  - https://www.kernel.org/doc/Documentation/networking/vxlan.txt
- Open vSwitch
  - http://openvswitch.org/