SDN, NFV and Cloud
An Overview of Current Trends in the Networking Industry

Luis MartinGarcia
luis@luismg.com
June 2016
1. SDN

Software Defined Networking
THE PROBLEM
What is SDN really trying to solve?
Two Main Issues

- Networks are Device-Centric.
- Network Devices are Hard to Configure
Networks are **Device-Centric**

- **Routers Today...**
  - Compute routes in CPU (RIB)
  - Push best route to the hardware (FIB)
  - Switch packets very fast based on destination address

<table>
<thead>
<tr>
<th>Control Plane</th>
<th>Data Plane</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF, EIGRP, IS-IS, BGP, PIM, SNMP...</td>
<td>L2 &amp; L3 Forwarding, VRF, QoS, ACLs, MPLS forwarding</td>
</tr>
</tbody>
</table>
Networks are **Device-Centric**

- Routers Today...
  - Every device has its own view of the network
  - Every device makes an independent forwarding decision
Networks are **Device-Centric**

**PROs**
- Scale very well (BGP)
- Are self-healing
- Are reliable and predictable

**CONs**
- Narrow hop-by-hop view
- Uncertainty beyond next hop
- Multiple levels of reconvergence
- All decisions based on “Destination IP”
- Very difficult to take into account other info
- Only fixed, non-adaptive metrics
- Difficult to extend or enhance
- In general, very hard to innovate
Devices are **Hard to Configure**

- Network Devices Today...
  - Networks are configured device by device
  - Configuration is manual
  - Configuration via Command-Line Interface (CLI)
  - Hard to keep configuration consistent
  - Hard to maintain software version consistency
Devices are **Hard to Configure**

- **Network Devices Today...**
  - Very difficult to automate
  - Lack of proper APIs and interfaces
  - Existing config mechanisms very hard to consume by software
    - SNMP is very “Read” oriented, hard to configure things
    - CLI inconsistent across software versions
    - NMS tools forced to use telnet & screen scraping
      - Ugly
      - Inefficient
      - Error prone
POSSIBLE SOLUTIONS

How are we solving the problem?
Software-Defined Networking

- What is it?
  - The latest cool thing in the networking industry
  - “SDN is a new approach to designing, building, and managing networks that separates the network’s control (brains) and forwarding (muscle) planes to better optimize each”.

Routers and Switches

Control Plane

Data Plane
Why **Separate** Both?

- **Faster Innovation**
  - Control logic is not tied to hardware.
  - HW and SW evolve independently.

- **Network-wide View**
  - Easier to observe the network and make decisions.

- **Flexibility**
  - If the HW manufacturer doesn’t want to implement the features I need, I can do it myself.
SDN Architecture

Network Controller

Network Devices
SDN Architecture

Network Controller

Southbound API Consumer

Network Devices

APIs
OS
Hardware

APIs
OS
Hardware

APIs
OS
Hardware

APIs
OS
Hardware
SDN Architecture

Business Applications

Network Controller

Network Devices
SDN Architecture

Business Applications

AT&T, Verizon, Telefonica, KPN, BT, DT...

Network Controller

APIC, APIC-EM, OpenDayLight, Vyatta, VAN...

Network Devices

OnePK, OpenFlow, JunOS API, NETCONF...

IOS, NxOS, ComWare, JunOS, SROS, EOS...

Cisco, HP, Juniper, Alcatel, Arista...
Common Operations Available

- **Configuration**
  - Pull or push configuration to the device

- **Statistics**
  - Obtain real-time statistics
  - Notice relevant events on the network
  - Polling devices / Async Notifications

- **Traffic**
  - Divert or copy packets to the controller
  - Inject or re-inject (same or different interface)
  - Drop
SDN Deployment Modes

Traditional Networking

- Configure: CLI
- Monitor: SNMP
- Control Plane
- Data Plane

Pure SDN

- Business Logic / Apps
- Northbound APIs
- Control Plane
- Data Plane
- Southbound Protocols & APIs
- Controller

Hybrid SDN

- Business Logic / Apps
- Extra Intelligence
- Controller
- Southbound Protocols & APIs
- Control Plane
- Data Plane
2. NFV
THE PROBLEM

Why do we need Network Function Virtualisation?
Two Main Issues

- Devices are heterogeneous and expensive.
- Devices are hard and slow to deploy.
Heterogenous and **Expensive Devices**

- **Heterogeneous**
  - Different vendors.
  - Different form factors.
  - Different deployment models.

- **Expensive**
  - Vendors charge a premium for their special-purpose hardware.
  - The fact that they are “physical” products increases cost per se.
Devices are Hard and Slow to Deploy

- New Services often require...
  - Racking new devices.
  - Laying out new cabling.
  - Performing initial config manually.

- Work at the Datacenter...
  - Requires physical presence.
  - Qualified Engineers (different profiles)
  - Security & Safety procedures.
  - Maintenance windows.
POSSIBLE SOLUTION

How are we solving the problem?
Separate Function from Hardware

- Run network functions in commodity HW
  - Network functions implemented in Software.
  - Running on top of standard x86 platforms.
    - As Virtual Machines
    - Inside containers
    - Directly on the baremetal

- Applicable for a number of functions
  - Firewalls, IDS, Routers, Load Balancers, Proxies...
Virtual **Network Functions**

**PROs**
- Homogenous Datacenter
- No need to deploy physically everytime.
- SW is generally cheaper than HW
- Faster to deploy new devices
- Cheaper redundancy
- Easier multi-tenancy

**CONs**
- Performance limitations
- More complex traffic flows
- Harder to implement security controls
3. Cloud

Public, Private & Hybrid Cloud Environments
THE PROBLEM

What do we need the Cloud for?
Two Main Issues

- Deploying IT services is difficult and slow.
- The cost of IT doesn’t necessarily match the growth of the business.
IT is Difficult and Slow

- Professional IT is complex
  - Requires highly skilled engineers, not always available.
  - In mature companies, there are a lot of politics involved (different departments, responsibilities, etc).

- Markets change faster than ever
  - Need to put new services on the market faster than the local IT can handle.
Cost of IT scales differently

- Big investments upfront
  - IT infrastructure is expensive and must be paid upfront, even when there is zero revenue.
  - Scaling up when there is growth is slow.
  - Scaling down is almost impossible.
  - Environment is always either overprovisioned or underprovisioned.

- IT needs to be more agile
  - Need to grow and shrink dynamically as needed.
  - Pay as you grow.
POSSIBLE SOLUTION
How are we solving the problem?
Cloud Environments

- Run my IT on someone else’s infrastructure
  - Rely on specialised companies to provide the IT infrastructure needed.
  - Rely on their know-how.
  - Rely on their 24/7 support services.
  - Concentrate on business applications, not the rest of the stack.
  - Pay IT as an utility bill.
Cloud Deployment Modes

On-Premise

Local Infrastructure \[\text{WAN/Internet}\] \[\text{Cloud Infrastructure}\]

Public Cloud

Local Infrastructure \[\text{WAN/Internet}\] \[\text{Cloud Infrastructure}\]

Hybrid Cloud

Local Infrastructure \[\text{WAN/Internet}\] \[\text{Cloud Infrastructure}\]
Cloud Offerings

SaaS
- Applications
  - Databases
  - Frameworks
  - Web Servers
  - Containers
  - Operating System
  - Virtual Machines
  - Hypervisor

PaaS
- Services (DNS, Load B, caching)
  - Compute
  - Storage
  - Network

IaaS
- Mgmt
- Identity & Security

- Hypervisor
- Containers
- Operating System
- Virtual Machines

- Web Servers
- Frameworks
- Databases

- Services (DNS, Load B, caching)

- Compute
- Storage
- Network
4. Use Cases
Programmable Forwarding: OpenFlow

Switch
- Switch Control Plane
- OpenFlow Agent
- Switch Forwarding Engine
  - OF Interface
  - OF Interface
  - OF Interface

OpenFlow Protocol

Controller
- Intelligence
- OpenFlow SDK
### Programmable Forwarding: OpenFlow

#### Headers Fields
- Ingress Port
- Source MAC
- Dest MAC
- Ether Type
- VLAN ID
- VLAN Priority
- IP SRC
- IP DEST
- IP Proto
- IP TOS
- TCP/UDP SRC
- TCP/UDP DEST

#### Match Tuple

#### Counter
- Packets / Bytes

#### Actions
- Forward
- Port
- Flood
- Normal
- Controller
- Drop
- Set

#### OpenFlow 1.0 Tuple

#### Flow Table

Flow Entry

Match Tuple

Flow Action
Experiment in Production
Large Scale Provisioning
Traffic Steering

Destination

Service Provider

Enterprise

Satellite Link

DSL

ISP

MPLS

Controller

Internet

Branch Office
IXP Enhancements
WAN Optimisation
VMWare NSX

Distributed Networking

ESXi Kernel

Physical Host

Overlay Network

Dummy Transport Network

Distributed Networking

ESXi Kernel

Physical Host
Cisco ACI

Control
- APIC Controller
- vCenter

Spines

Leaves

Compute & Services

VMs
Cloud Orchestration

End Goal:
Deploy a new three-tier application.
Cloud Orchestration

Cloud Orchestrator

- Amazon AWS
- Microsoft Azure
- Google Cloud Plat.

Network Devices
Compute Servers
Disk Arrays
Firewall Devices
Load Balancers
DNS Servers
Edge Devices
Whitebox Networking

Single Vendor

Traditional Device

Module
Operating System
Supervisor
ASICs

“Whitebox” Device

App
Operating System
x86 Server
Merchant Silicon

Vendor A
Vendor B
Vendor C
Other Tendencies

- Flow Visualization
- Self-Configuring Networks
- Self-Optimizing Networks
- Network abstractions closer to the application
- Convergence to Ethernet end-to-end.
- Network Engineering teams bringing in SW Developers.
- Industry creating the standards, not the IETF.
Questions?

Thanks!
CREDITS

Special thanks to the following individuals and organizations that made their templates, pictures and icons available under open licenses:

- Presentation template by SlidesCarnival
- Network Diagram Icons by Cisco Systems
- Title Slide Icon by FreePik
- Datacenter Rack picture by Clayton O’Neill
- Miscellaneous Icons (video, email…) by Google
SDN, NFV and Cloud
An Overview of Current Trends in the Networking Industry

Luis MartinGarcia
luis@luismg.com
June 2016