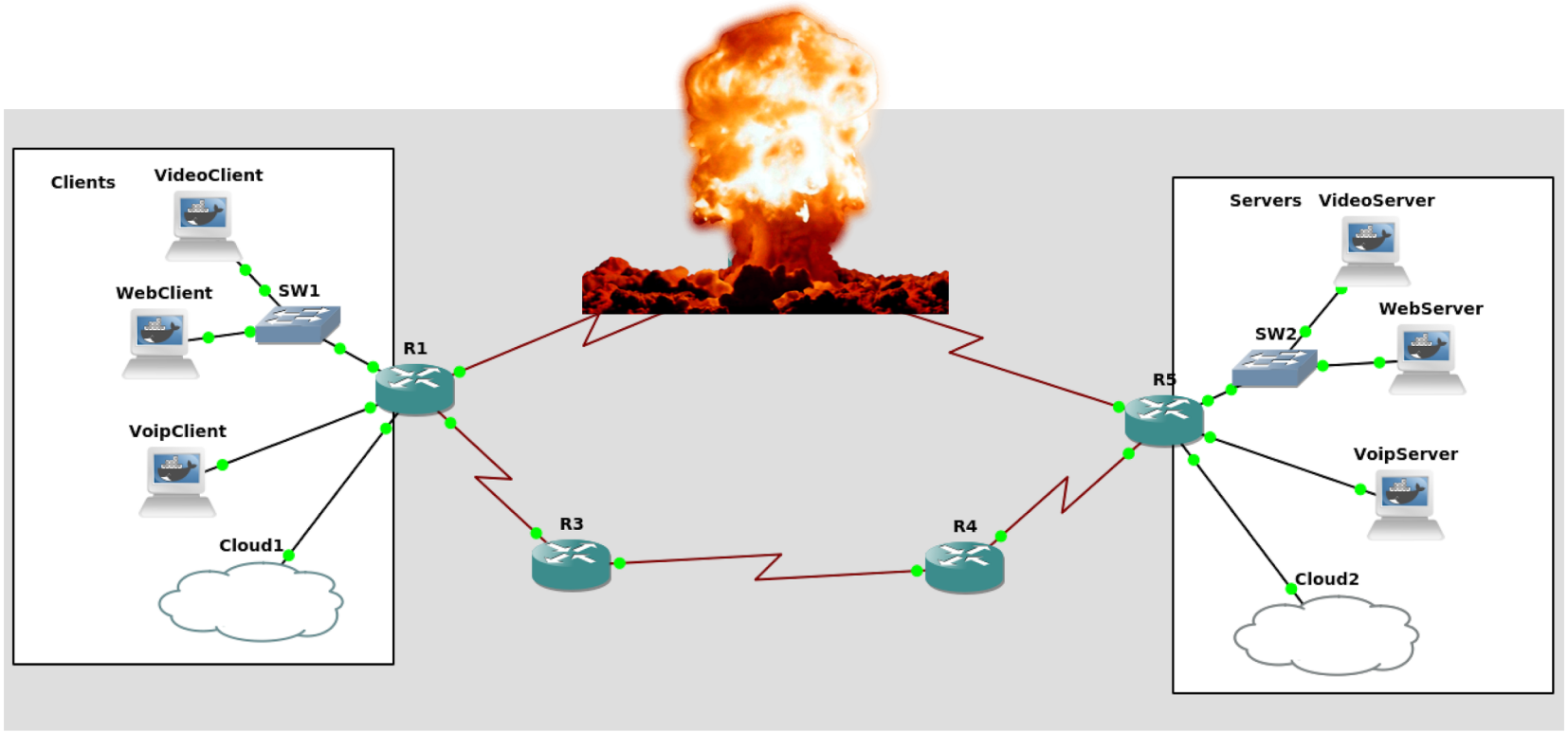
Software Defined Networking

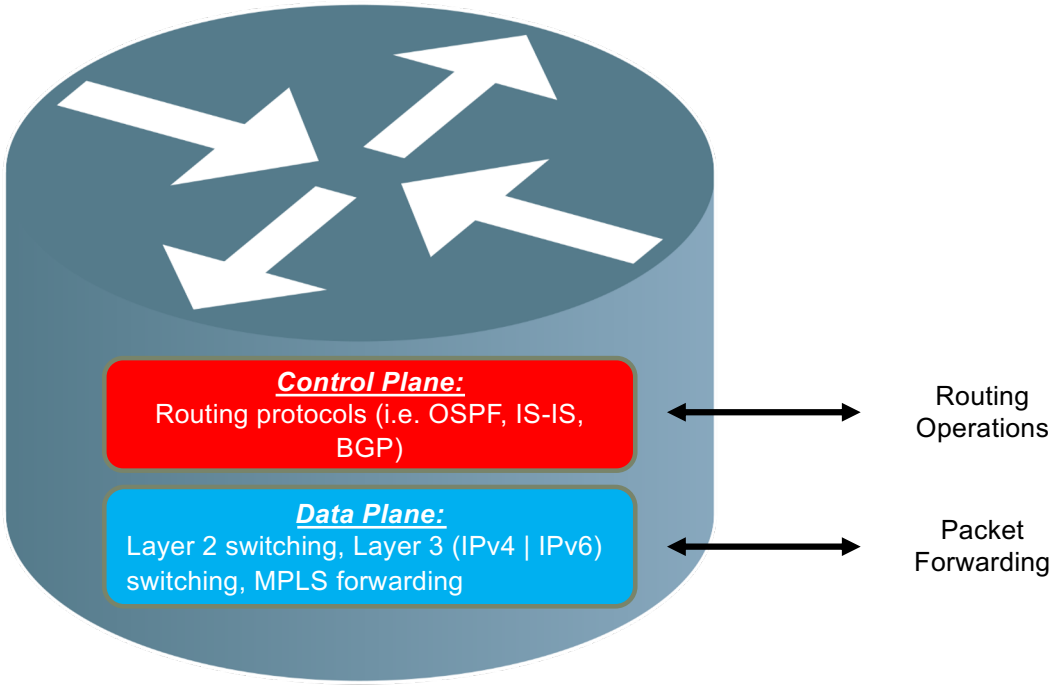
Refs: Software Defined Networks: A Comprehensive Approach

Antonio Virdis
Assistant Professor@ University of Pisa
antonio.virdis@unipi.it

Traditional Networks

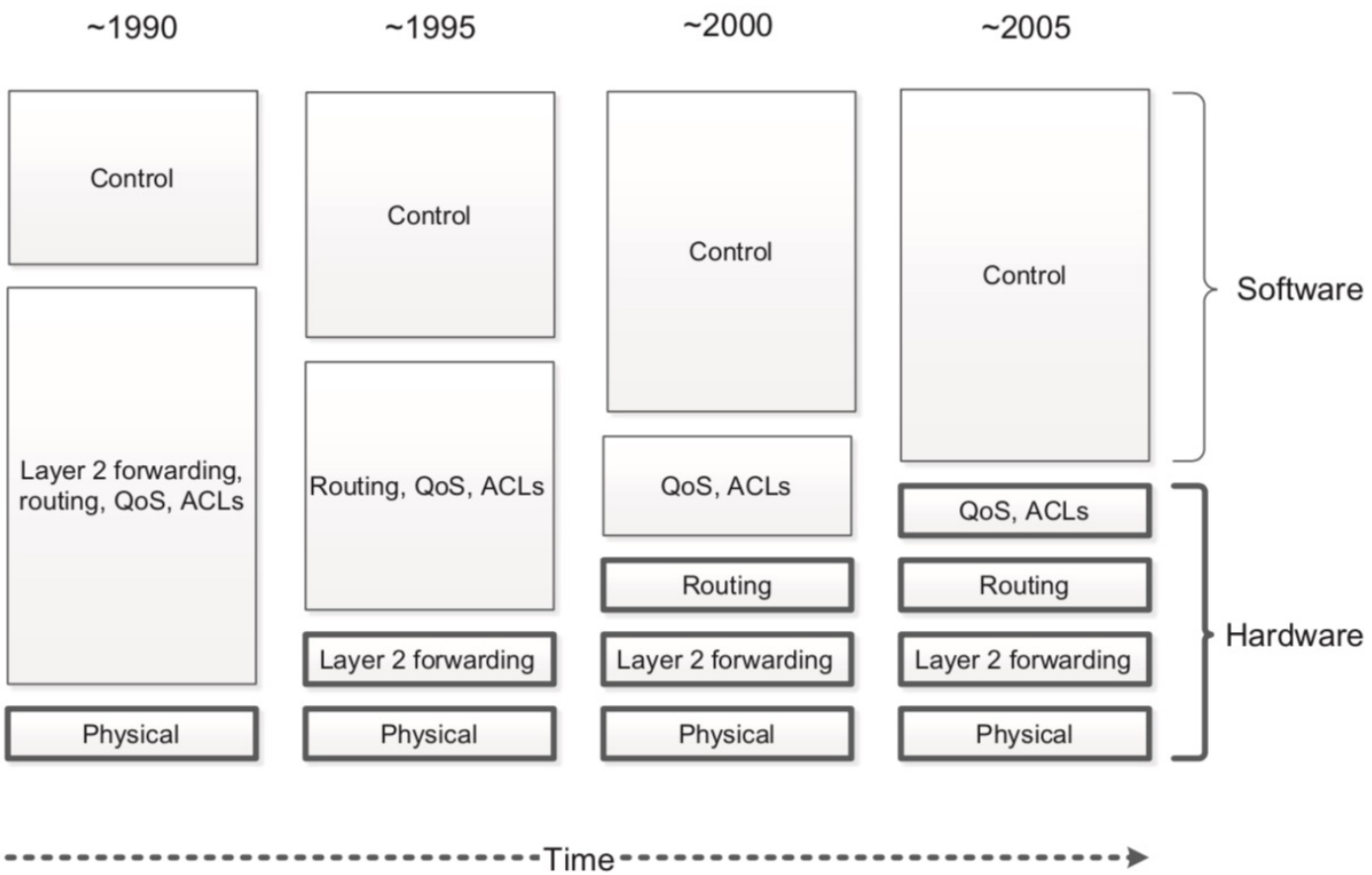


Traditional Networks

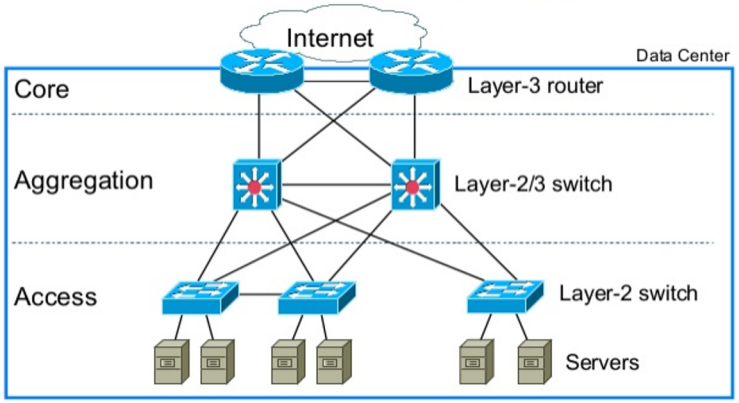


All Operations are implemented on the same device

From Software to Hardware

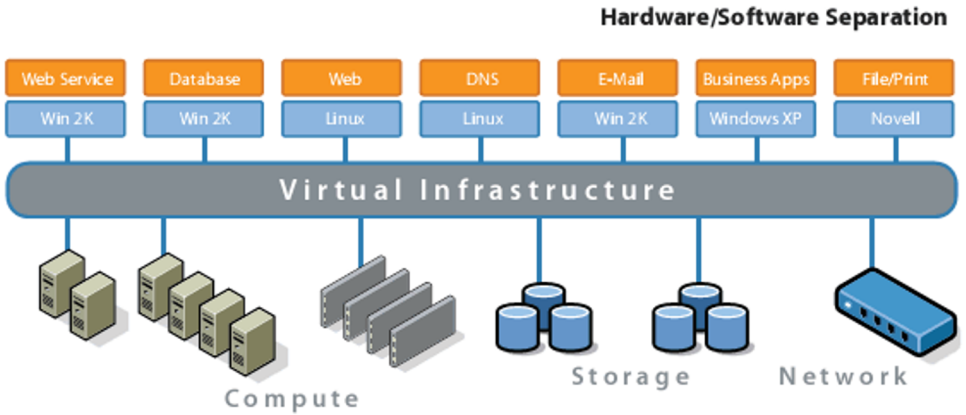


Use Case: Datacenters

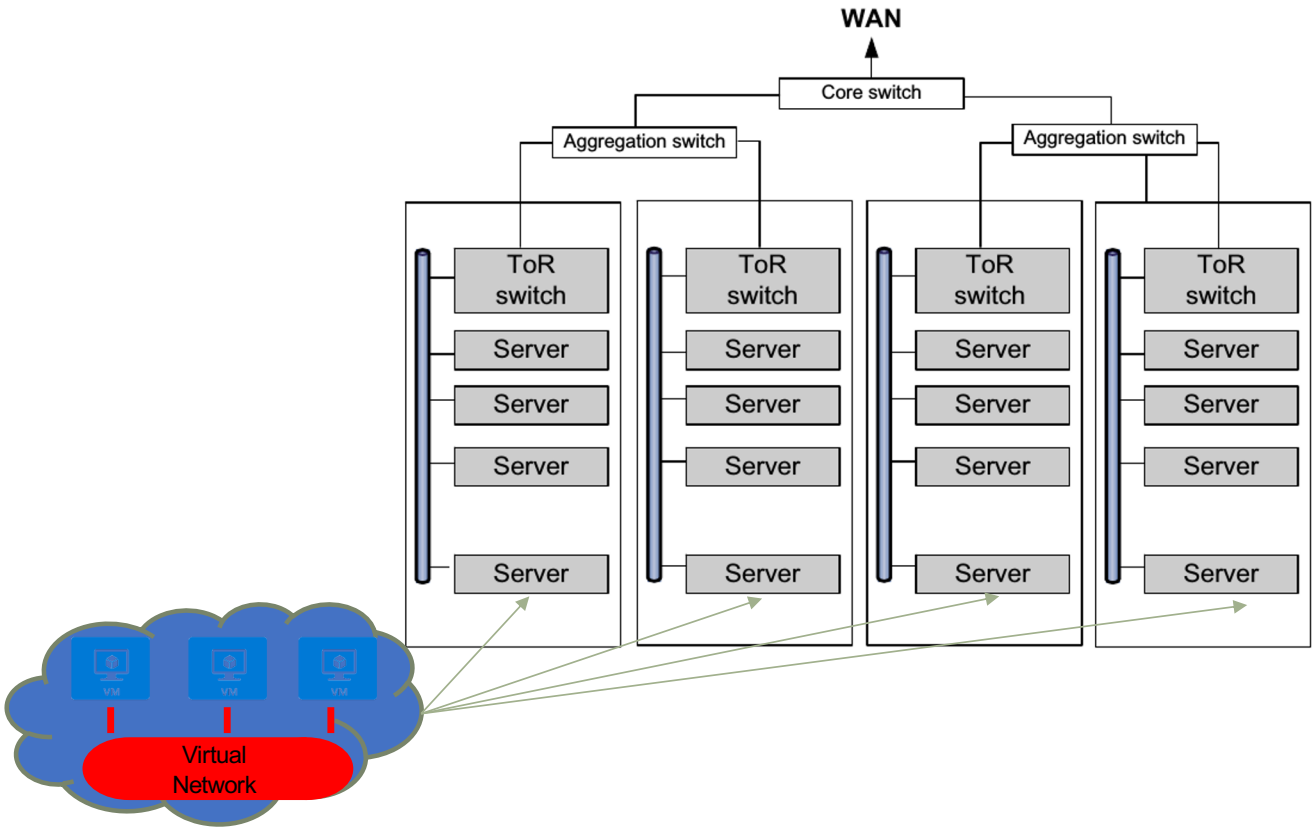


Traditional Datacenter architecture

Datacenter using Virtualization



Datacenter Networking



Data-center needs

- Automation
- Scalability
- Multipathing
- Multitenancy

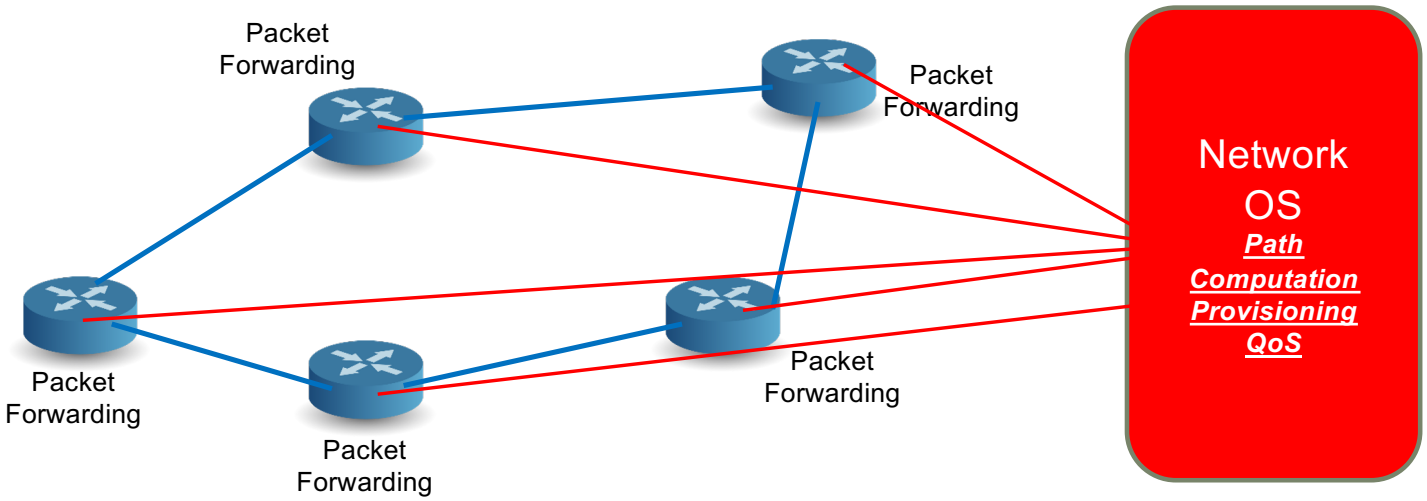
Why SDN (in short)

- Too **many** control plane targeting **many** networking problems in distributed fashion
- Closed system, having few to none open projects
- Manufacturers do not (need to) innovate quickly

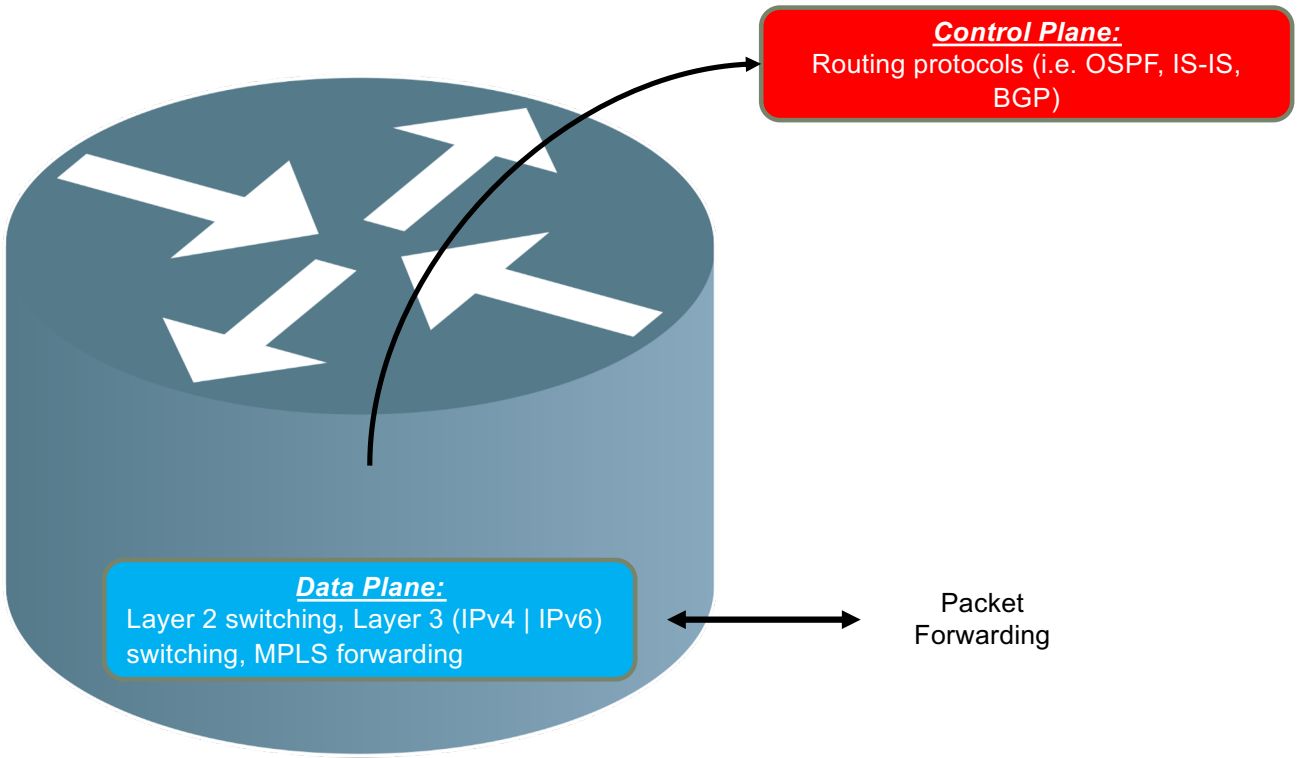
Slower evolution of networking technologies

Concept

- ***Software defined networking***: Physical ***separation*** of the network **control** plane from the **forwarding** plane, where control plane controls several devices
- Centralization of control



SDN Networks



Not all the operations are implemented on the same device

Moving Control off the device

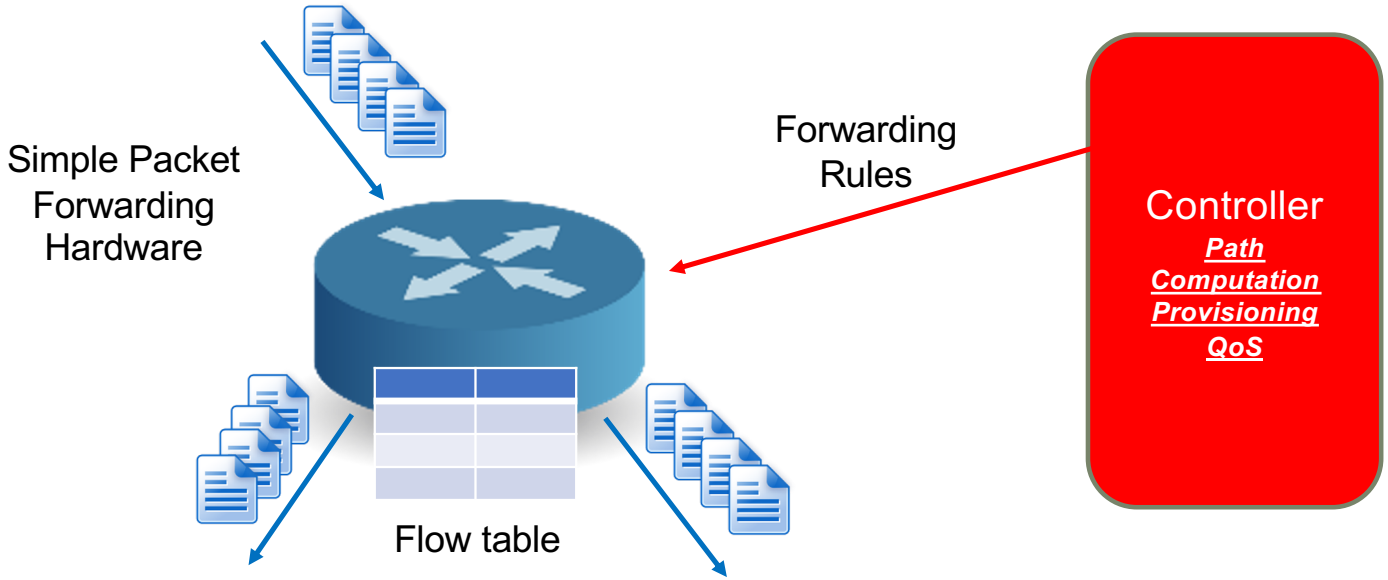
- Applications
- Control
- forwarding
  filtering
  prioritization

SDN traits

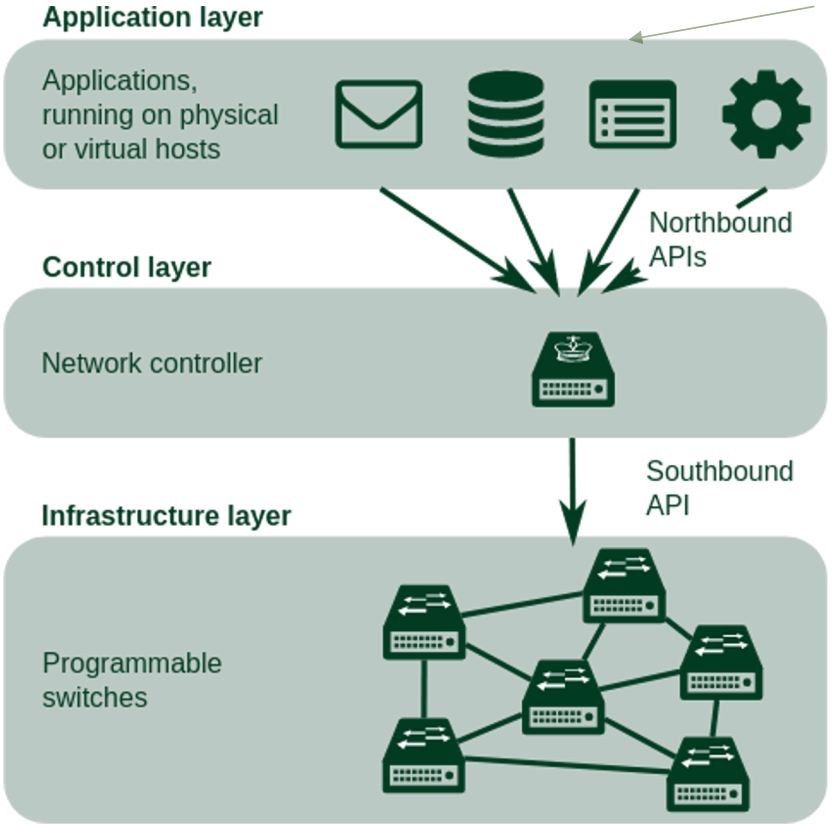
- Plane separation
- Simple device
- Decentralized control
- Net automation and virtualization
- Openness

SDN components

- Routers become ***simple hardware*** for packet forwarding (switch)
- A ***centralized controller*** is responsible for defining forwarding rules (controller)



SDN Architecture



applications

controller

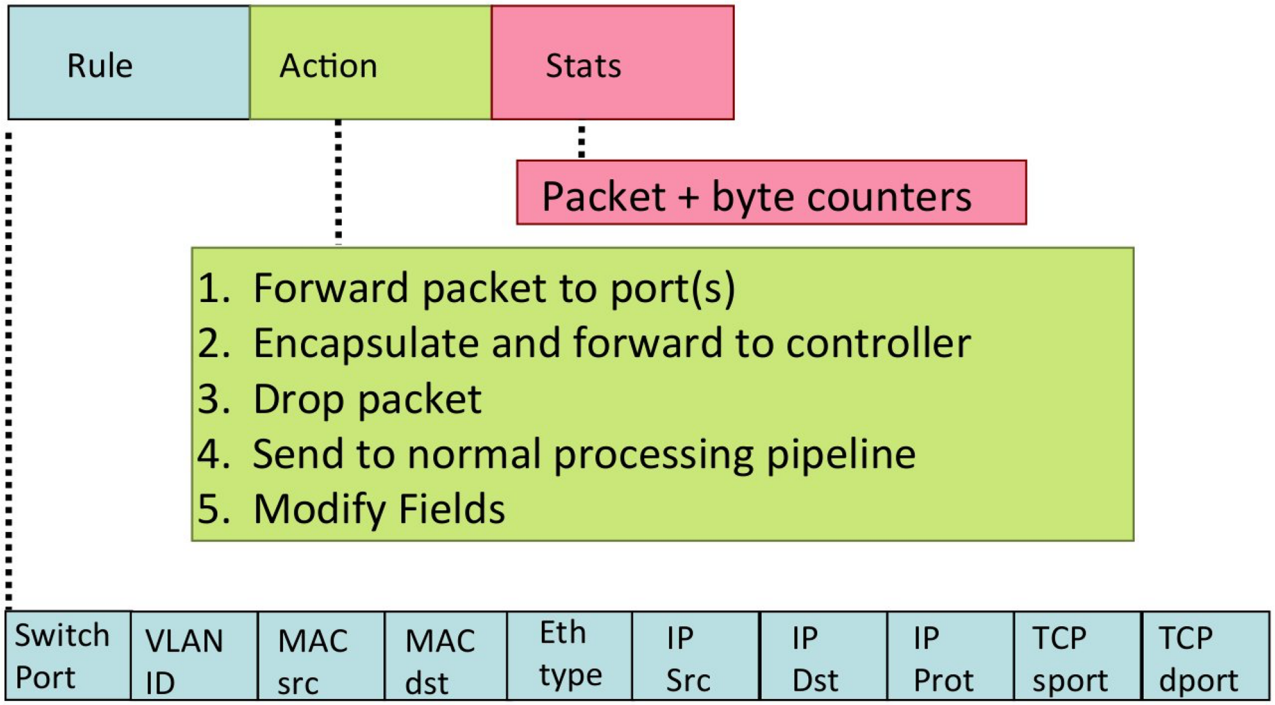
devices

SDN Devices

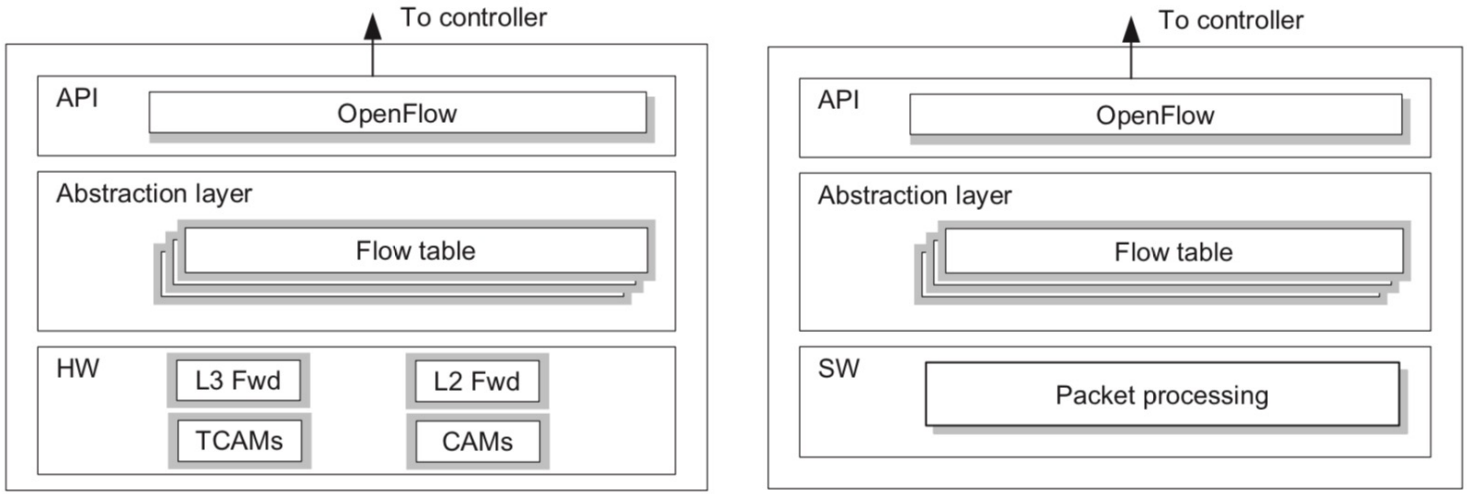
- Packet processing function
- Abstraction layer (flow table)
- API for communication with the controller

Flow Table

- Set of rules that determines the action to be performed for each packet



HW vs SW


To controller
API
OpenFlow
Abstraction layer
Flow table
HW
L3 Fwd
L2 Fwd
TCAMs
CAMs
To controller
API
OpenFlow
Abstraction layer
Flow table
SW
Packet processing

Flowtables scaling

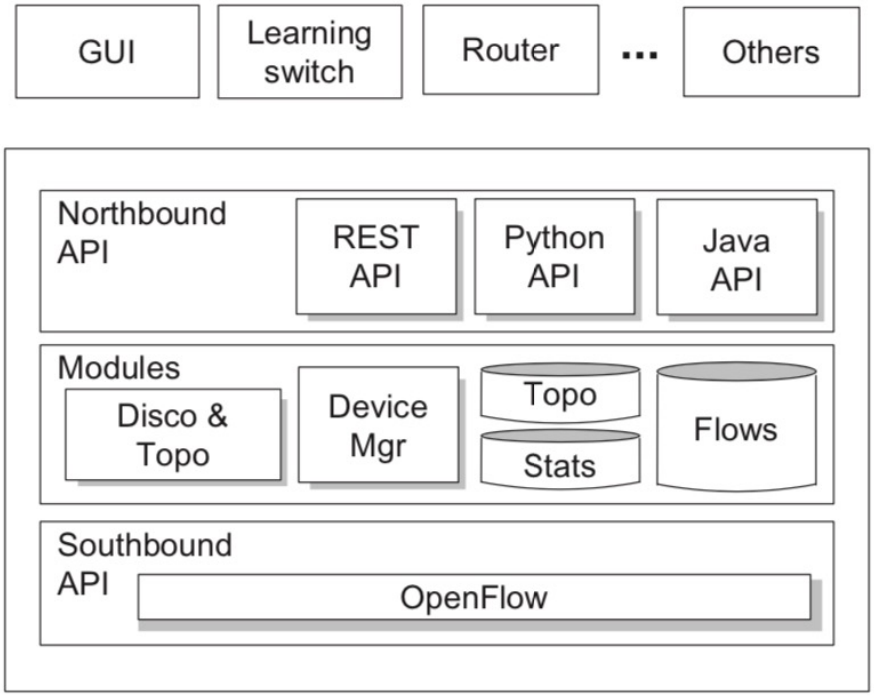
Examples of SDN devices

- Commercial examples: OVS (nicira), Indigo (Big Switch)
- OpenFlow support added to legacy switches
- Whiteboxes

SDN Controller

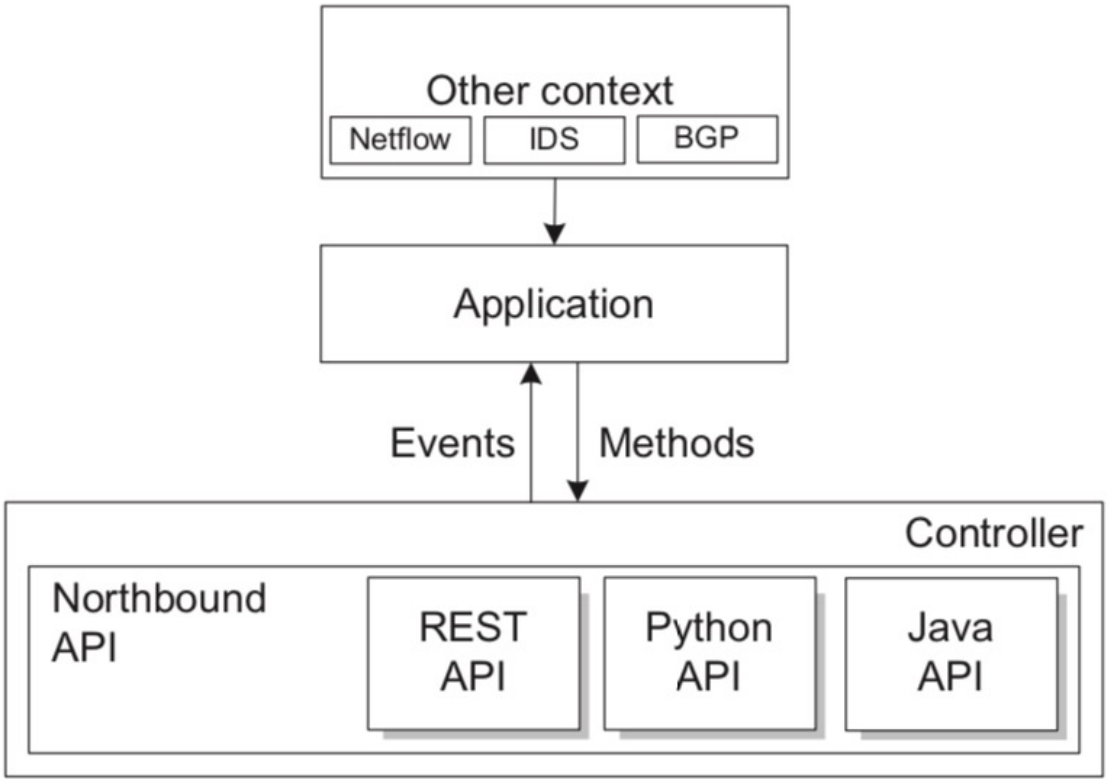
- View of the network
- Policy decision
- Control of SDN devices
- Northbound API

Main Functions of the controller



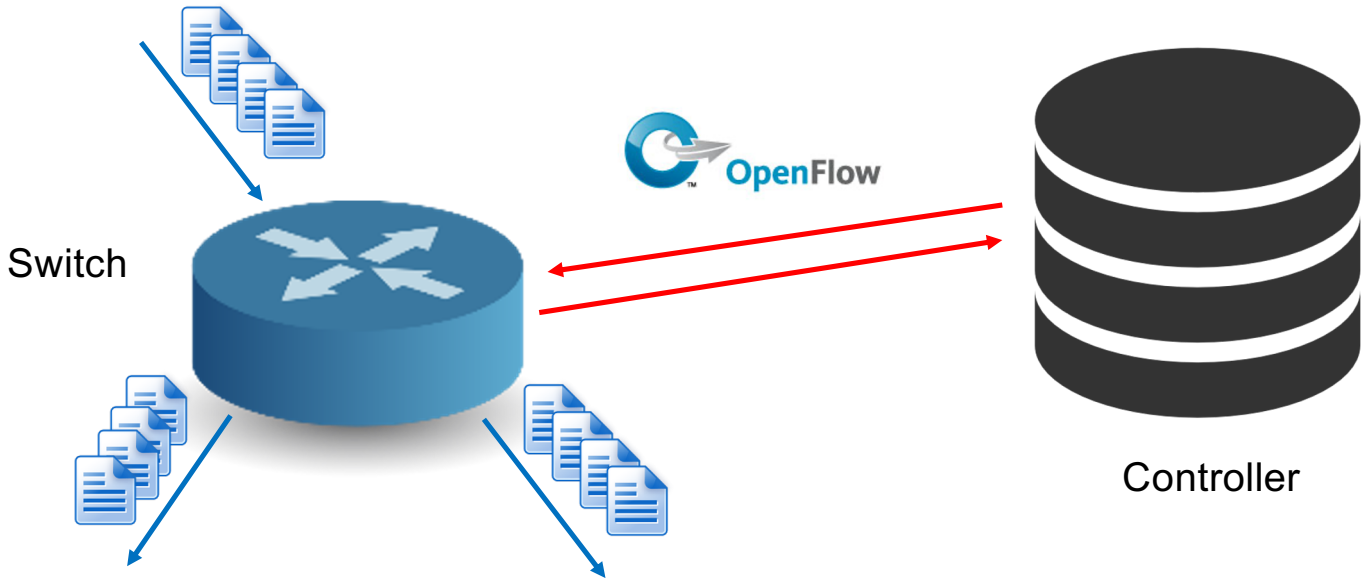
North/south Interfaces

Northbound


Other context
Netflow
IDS
BGP
Application
Events
Methods
Controller
Northbound API
REST API
Python API
Java API

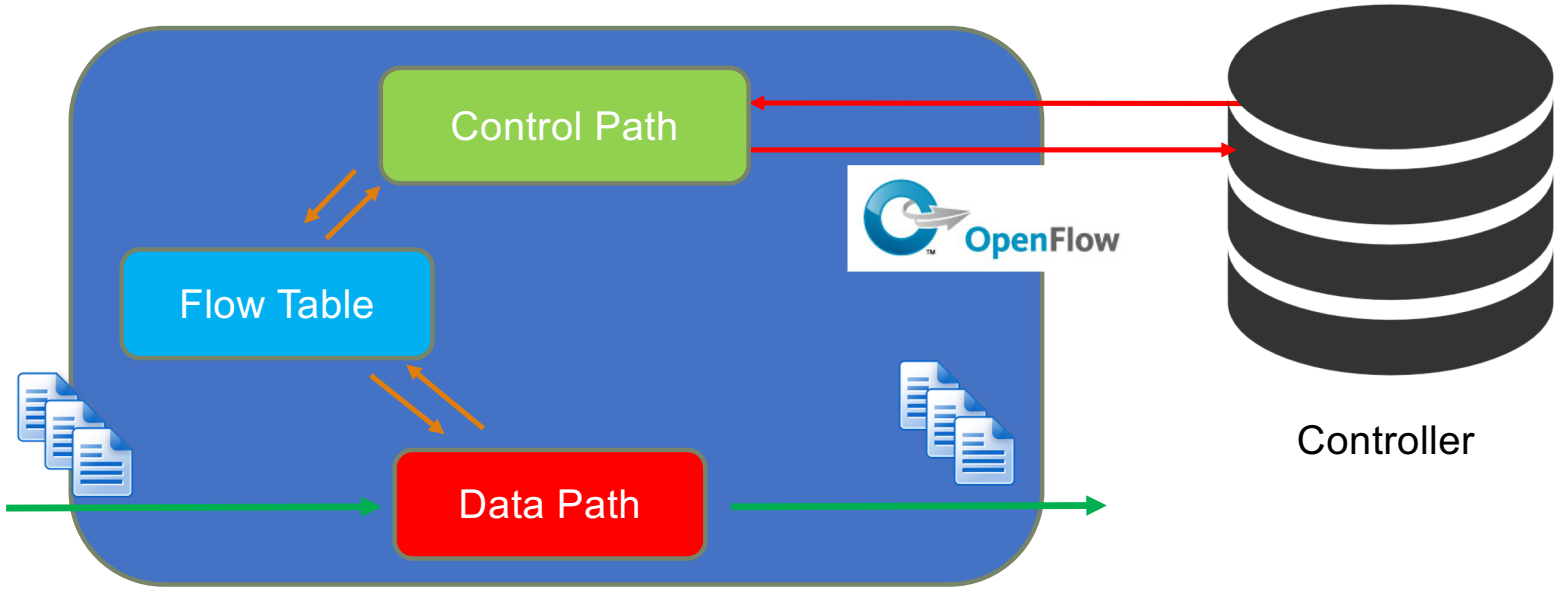
Southbound

Open Flow



Open Flow Switch

- Packets are forwarded according to a simple ***flow table***
- Controller uses the Open Flow protocol to populate the Flow Table



Applications

- Run on top of the controller
- Main possibilities given to apps:
  - Configure routing of flows
  - Balance traffic among available paths
  - React to topology changes
  - Redirect traffic for specific functions

Reactive vs Proactive

Multiple “SDN views”

SDN Architecture