

# An Introduction to Software-Defined Networking (SDN)

Zhang Fu

# Roadmap

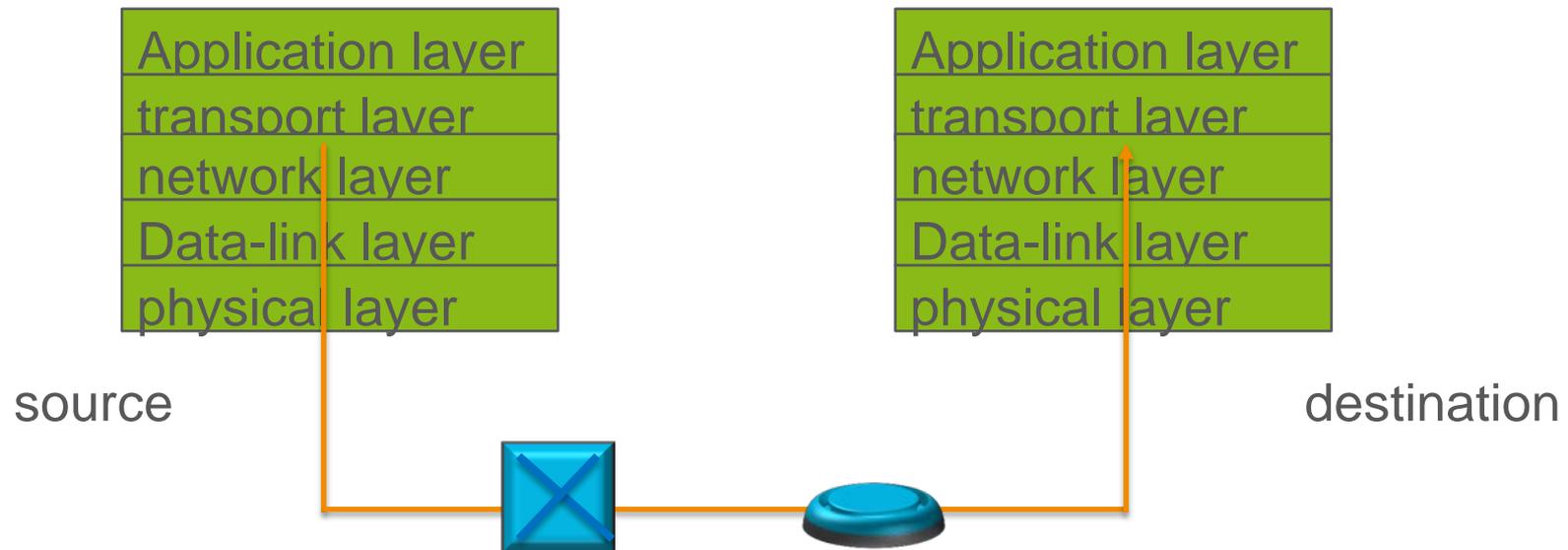


- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › Use cases
- › Challenges and research problems
- › Little bite of Openflow

# Reviewing traditional networking



## › Network layers



Why layers? Good abstraction, transparency...

# Reviewing traditional networking

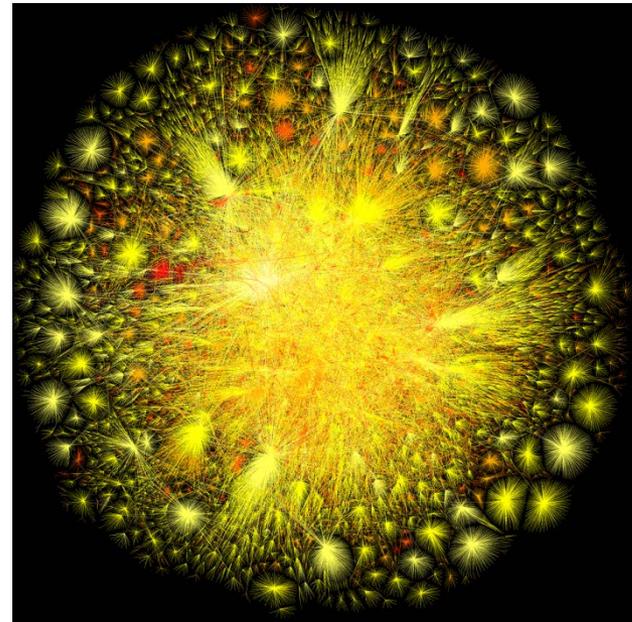


## › Design principles of Internet

- Simple
- Intelligent end-points
- Distributed control

## › Resulting in huge complex network and hard to manage

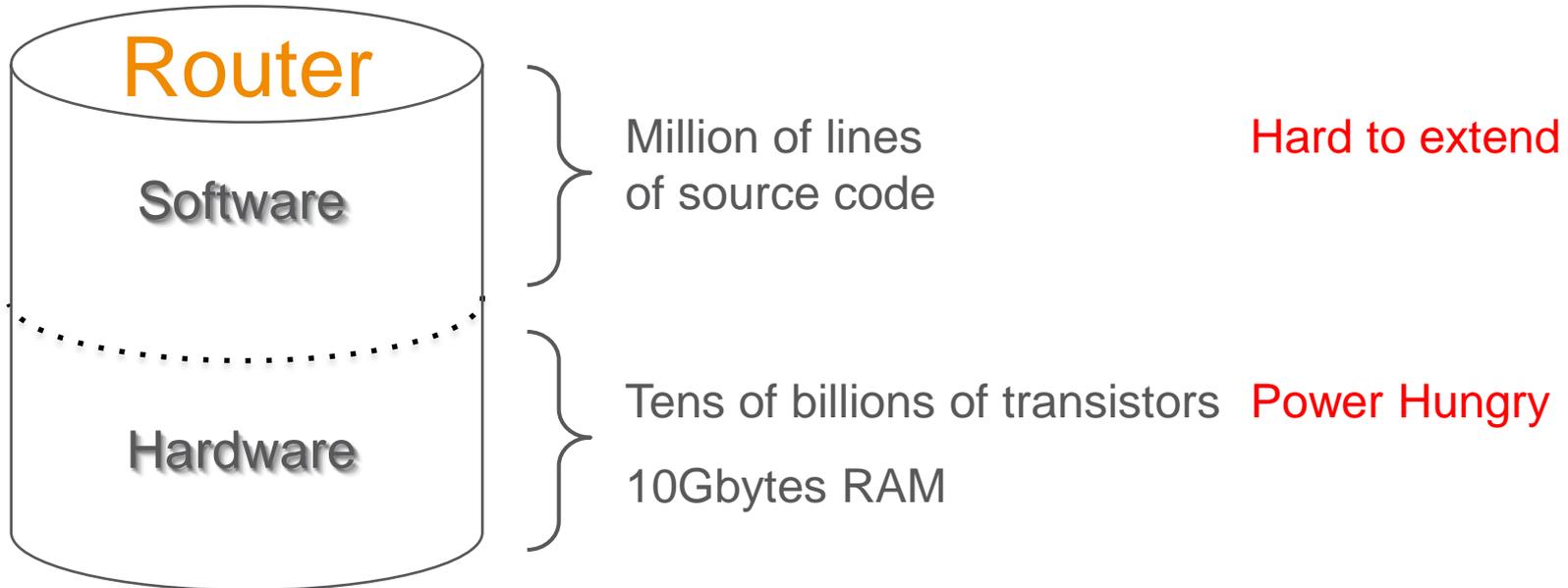
- Billions of computers
- Tens of thousands of ASes
- Great business for selling routers





# Reviewing traditional networking

## › Complex routers



Vertically integration with many complex functions: *OSPF, BGP, multicast, QoS, Traffic Engineering, NAT, firewalls, MPLS...*

# Roadmap

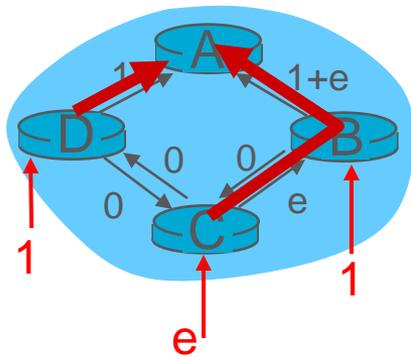


- › Reviewing traditional networking
- › **Examples for motivating SDN**
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › Use cases
- › Challenges and research problems
- › Little bite of Openflow

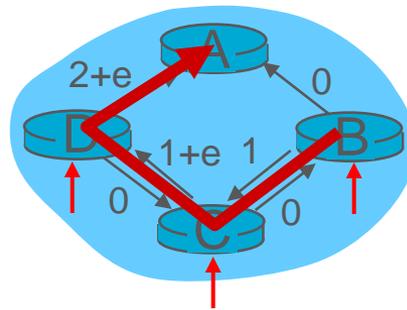


# Example: oscillation problem

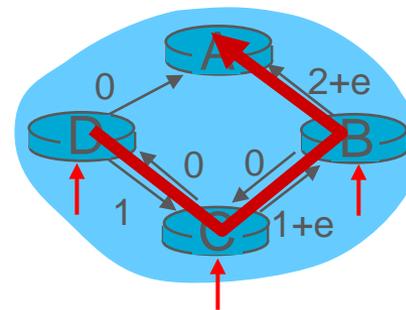
- › Link cost equals the amount of carried traffic



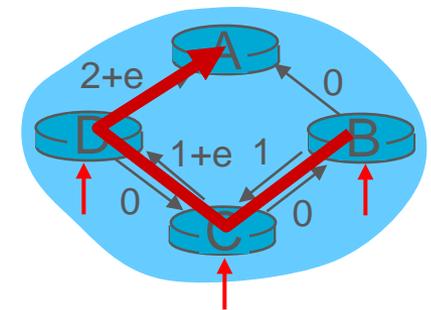
initially



given these costs,  
find new routing....  
resulting in new costs



given these costs,  
find new routing....  
resulting in new costs



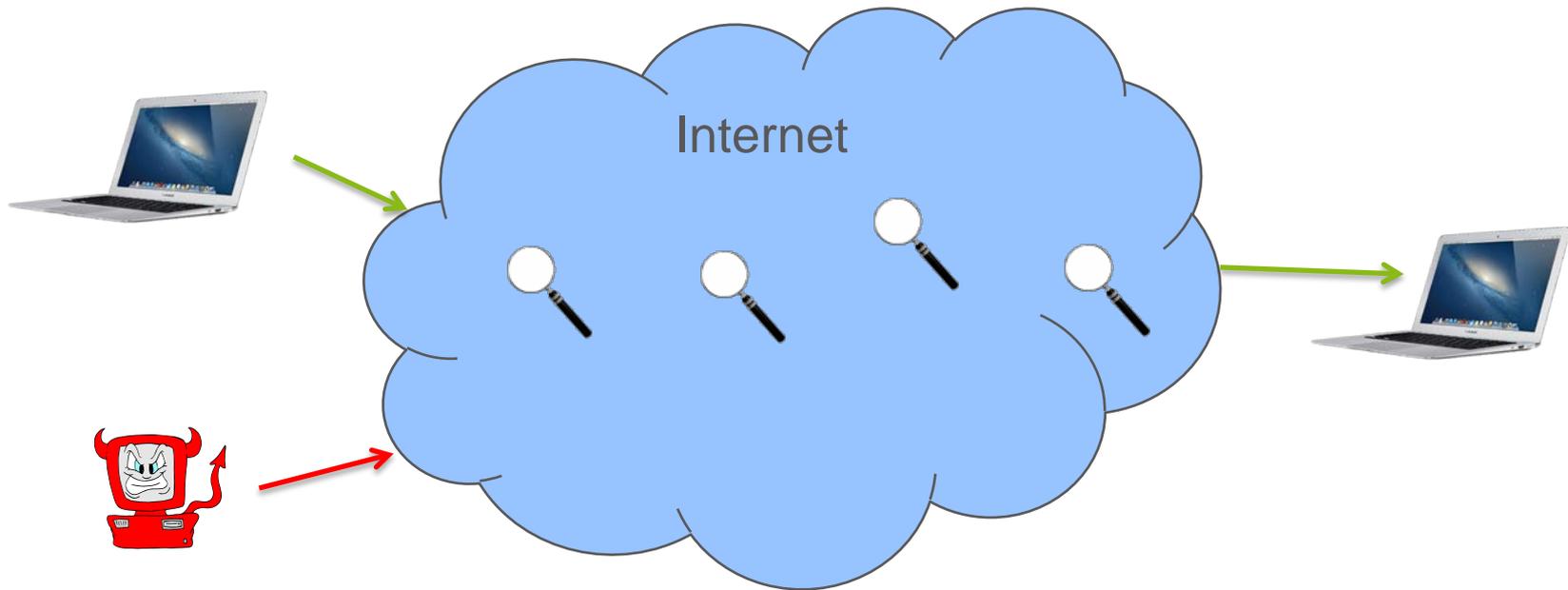
given these costs,  
find new routing....  
resulting in new costs

How to achieve optimal routing dynamically?

# Example: mitigating attacks



- › Checking the validity of packets by middle boxes



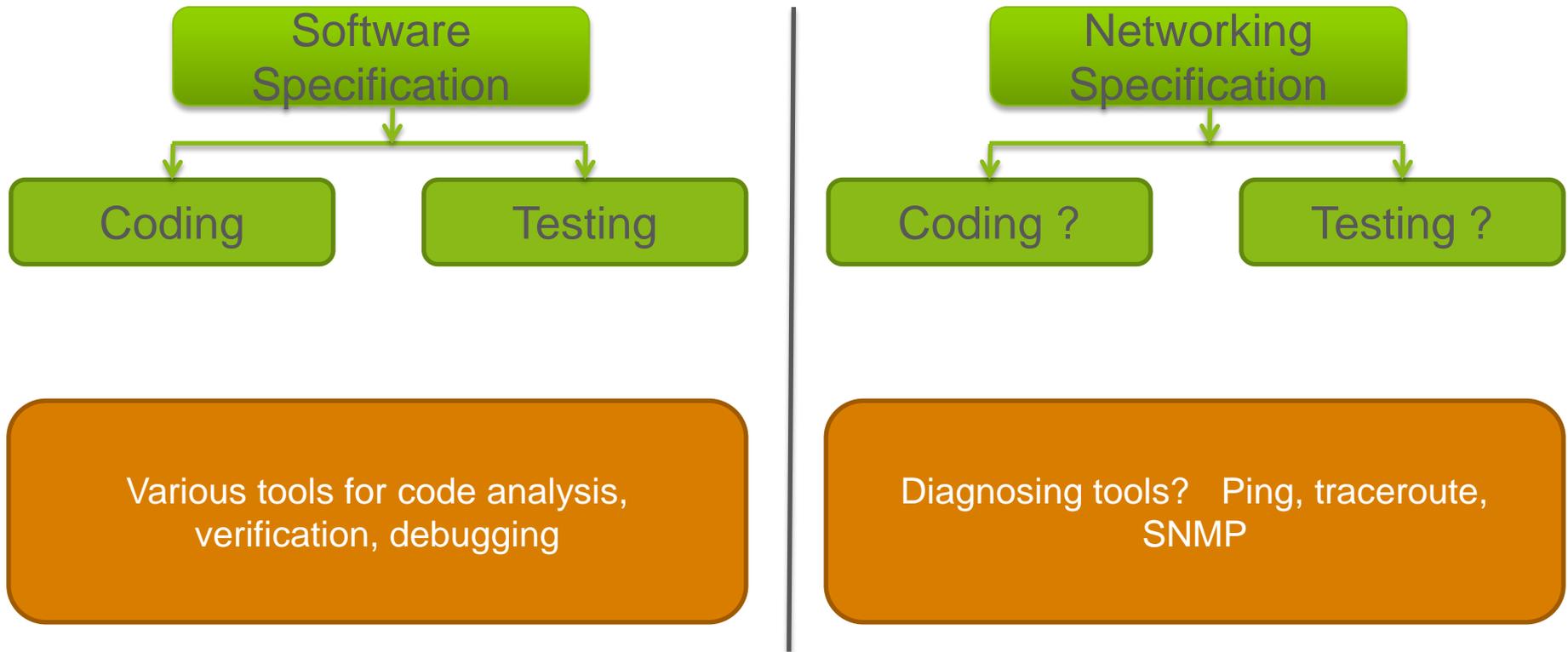
How to route the packets through a series of middle boxes?

# Roadmap



- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › Use cases
- › Challenges and research problems
- › Little bite of Openflow

# Software development VS Network diagnosing



- The life cycle for network protocols is much longer than that for software
- Timely research does not find its way into practice

# Network substrate

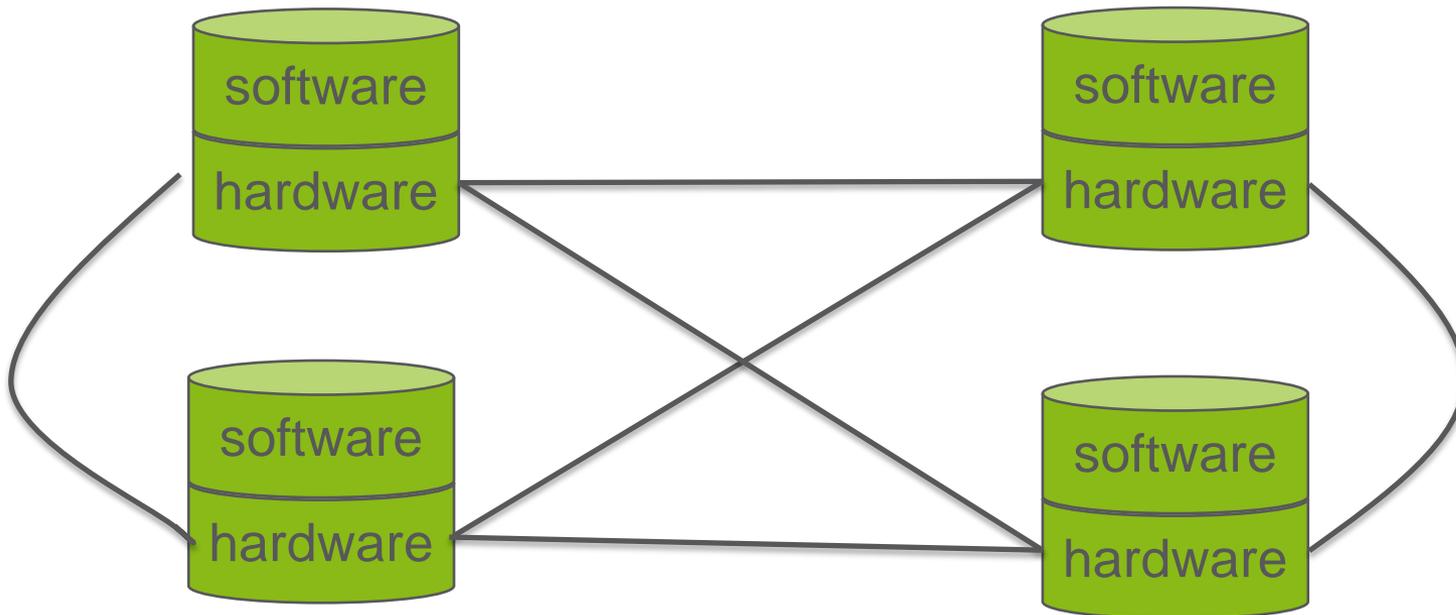


- › We want to mimic the success in software industry
  - Has simple common substrate
  - Building OS on top the hardware, which enables easy deployment of networking applications

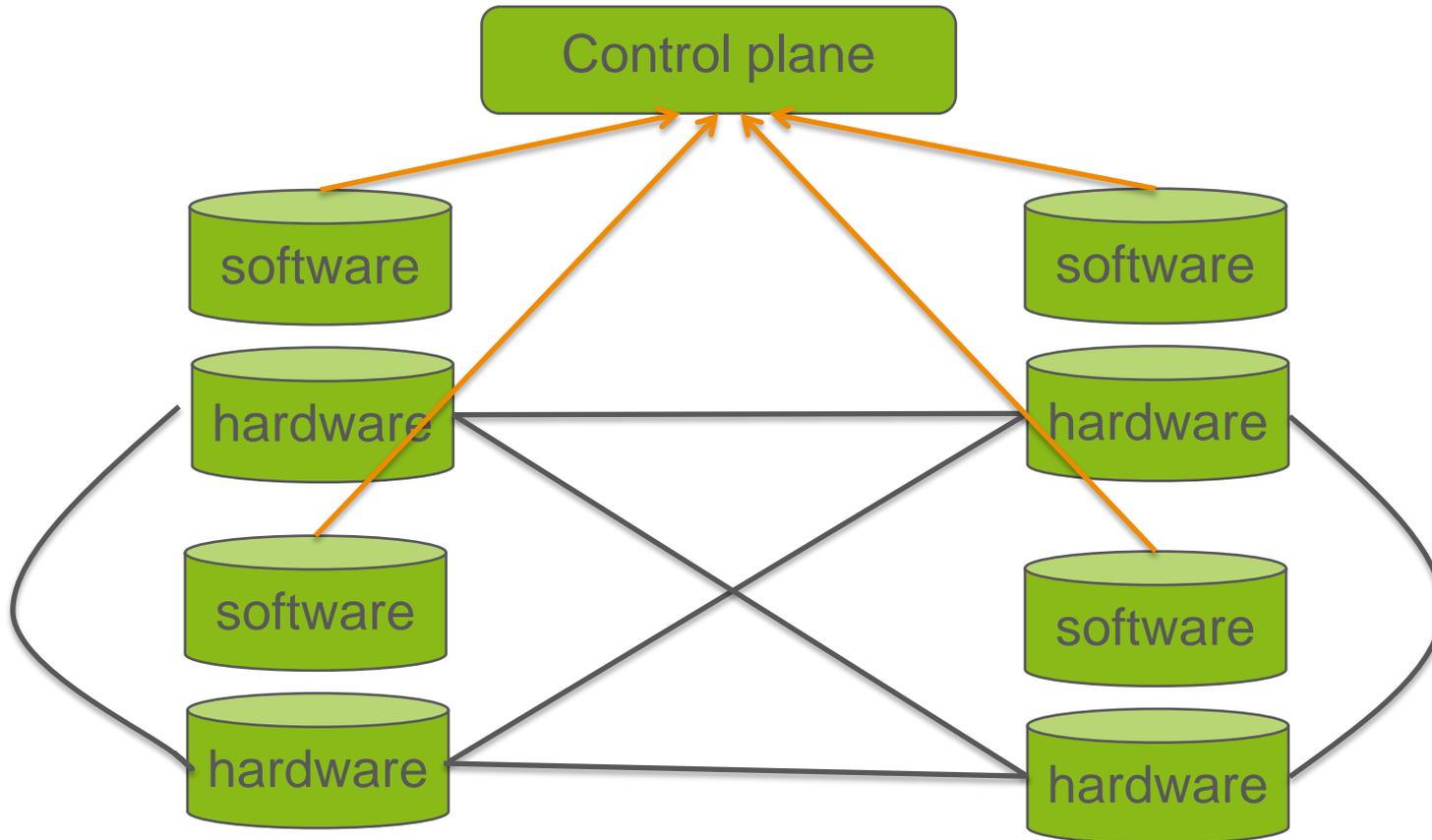
## *SDN*

- A network in which the control plane is physically separate from the data plane.
- A single control plane controls several forwarding devices.

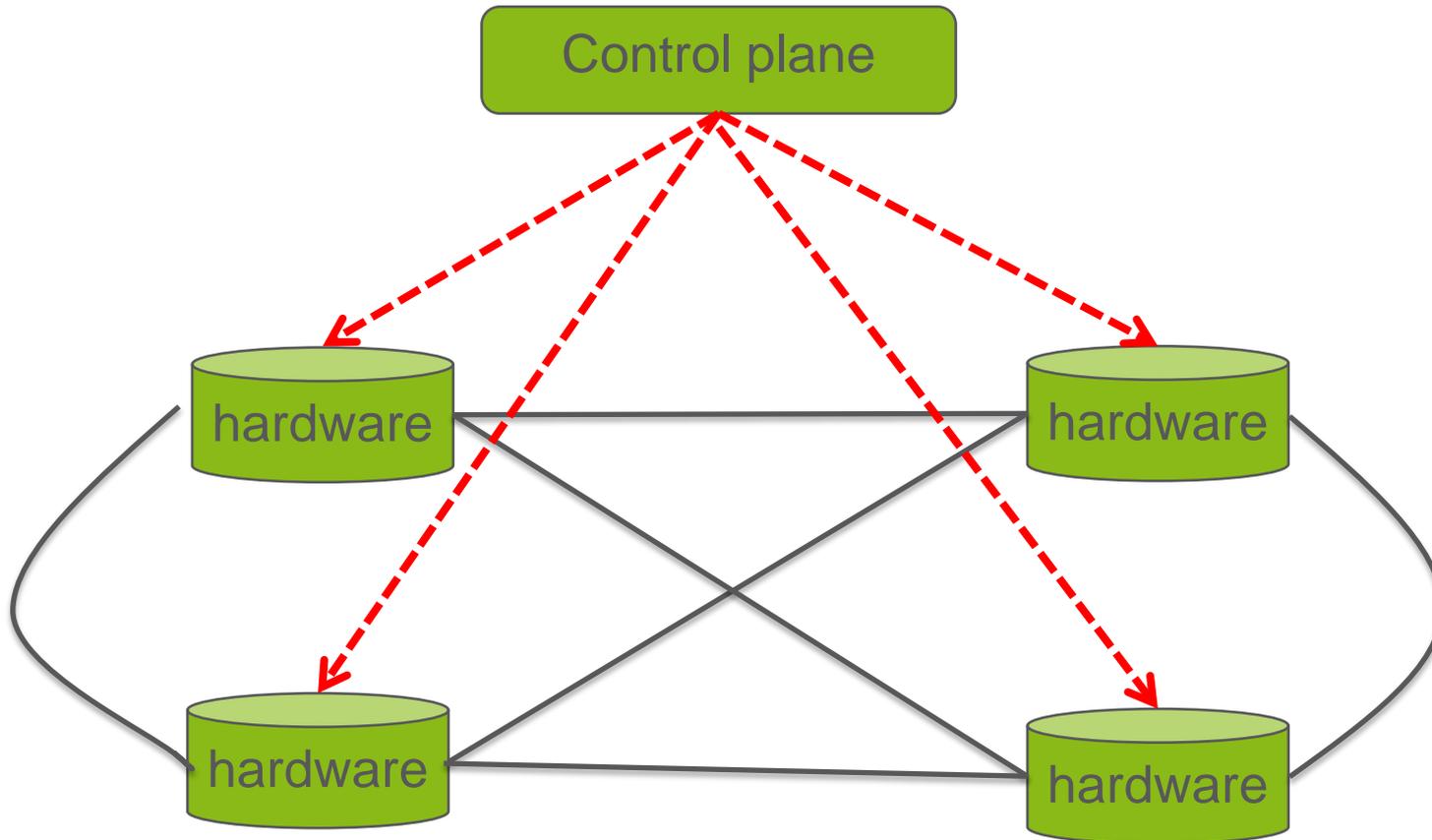
# Separate data and control plane



# Separate data and control plane



# Separate data and control plane

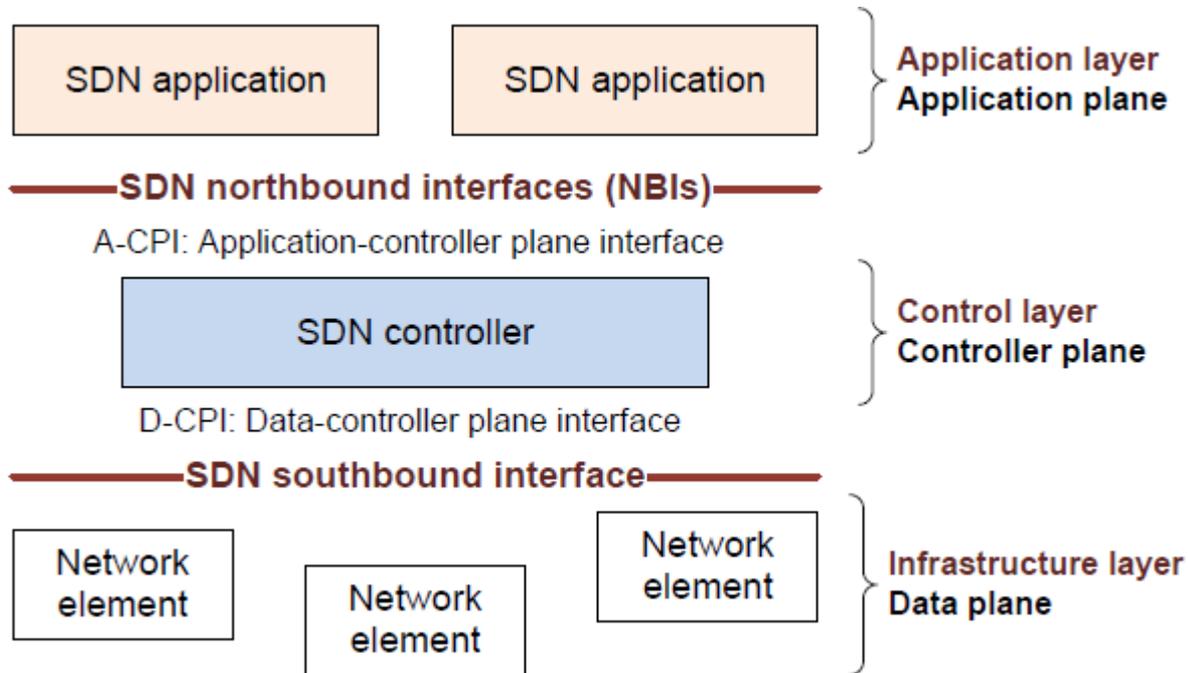


# Roadmap

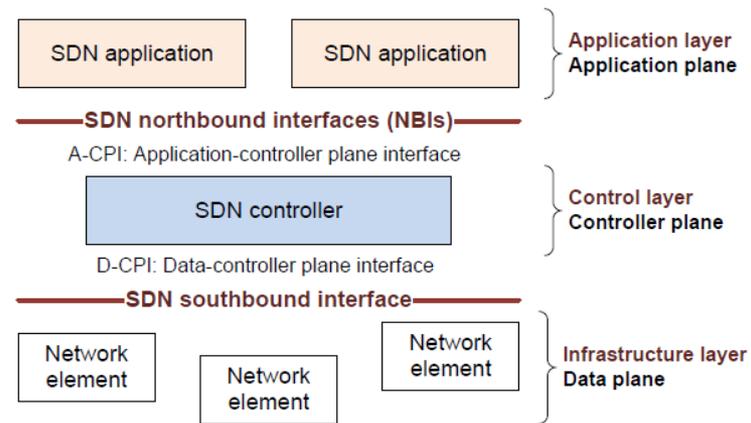


- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › **SDN architecture**
- › SDN components
- › Use cases
- › Challenges and research problems
- › Little bite of Openflow

# SDN architecture



# SDN architecture



- › The data plane consists of network elements, which expose their capabilities to the control plane via southbound interface
- › The SDN applications are in the application plane and communicate their network requirements toward the control plane via northbound interface
- › The control plane sits in the middle
  - translate the applications' requirements and exerts low-level control over the network elements
  - Provide network information to the applications
  - Orchestrate different applications

# Roadmap



- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › Use cases
- › Challenges and research problems
- › Little bite of Openflow

# Data-plane



- › Data sources and sinks
- › Traffic forwarding/processing engine
  - May have the ability to handle some types of protocols
- › Provide interfaces communicating to the control plane
  - Programmatic control of all functions offered by the network element
  - Capability advertisement
  - Event notification

# Control-plane



- › Logically centralized
- › Core functionality
  - Topology and network state information
  - Device discovery
  - Path computation
  - Security mechanism
- › Coordination among different controllers
- › Interfaces to the application plane

# Application-plane



- › Applications specify the resources and behaviors required from the network, with the context of business and policy agreement
- › It may need to orchestrate multiple-controllers to achieve the objectives
- › Programming languages help developing applications, e.g. Pyretic, FatTire, etc.

# Roadmap



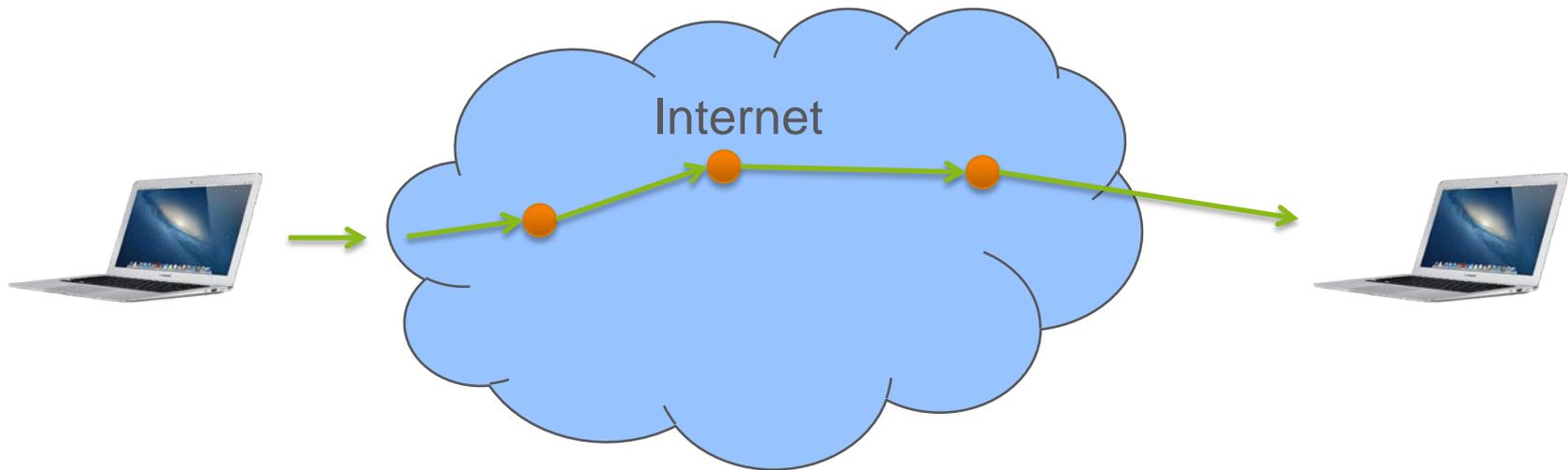
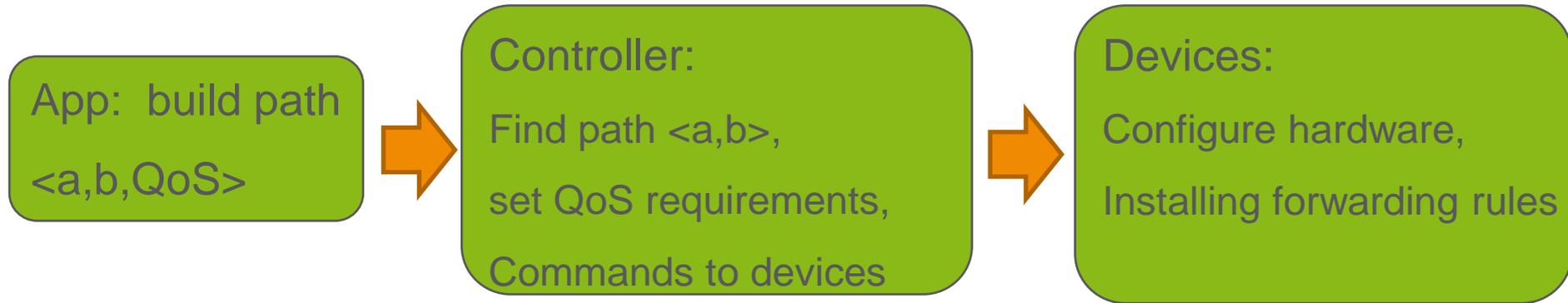
- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › **Use cases**
- › Challenges and research problems
- › Little bite of Openflow

# Use cases



- › Traffic engineering
- › Mobility and wireless
- › Security
- › Data center networking

# Example: routing



# Roadmap



- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › Use cases
- › **Challenges and research problems**
- › Little bite of Openflow

# Challenges and research problems



- › Switch design
  - Find common abstraction
  - Flow table capacity
  - Throughput
- › Controller platform
  - Distributed vs centralized
  - Flexibility
- › Dependability and security
  - Attack to data plane
  - Attack to control plane
  - Trust, privacy issues

# Roadmap



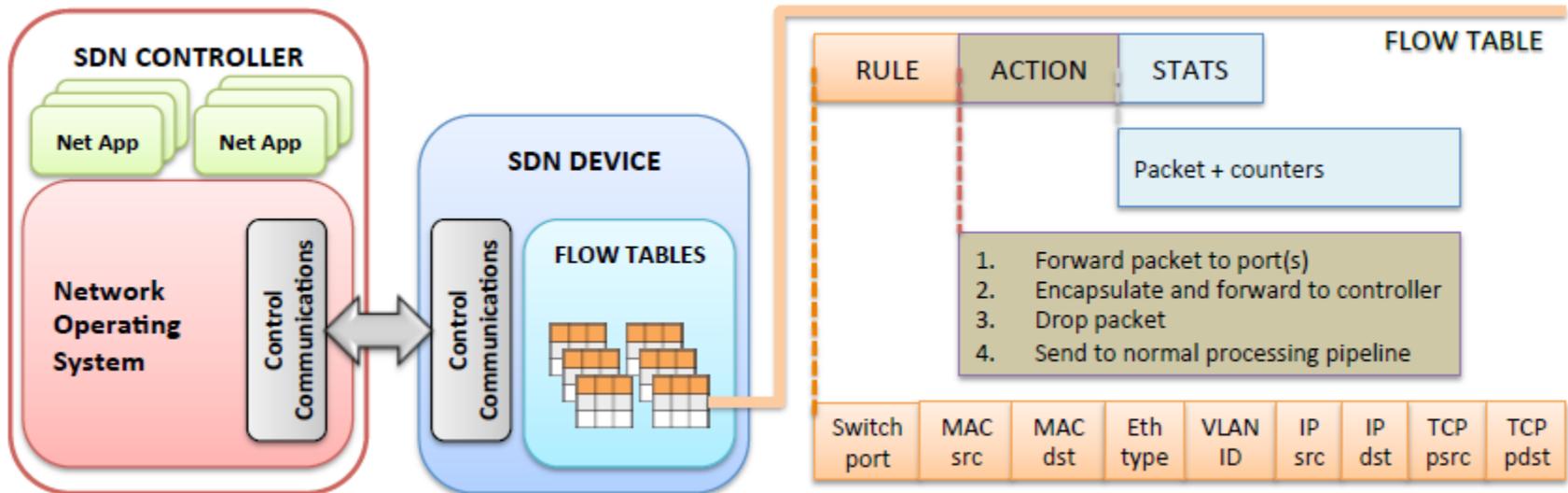
- › Reviewing traditional networking
- › Examples for motivating SDN
- › Enabling networking as developing softwares
- › SDN architecture
- › SDN components
- › Use cases
- › Challenges and research problems
- › Little bite of Openflow

# Openflow



An southbound standard:

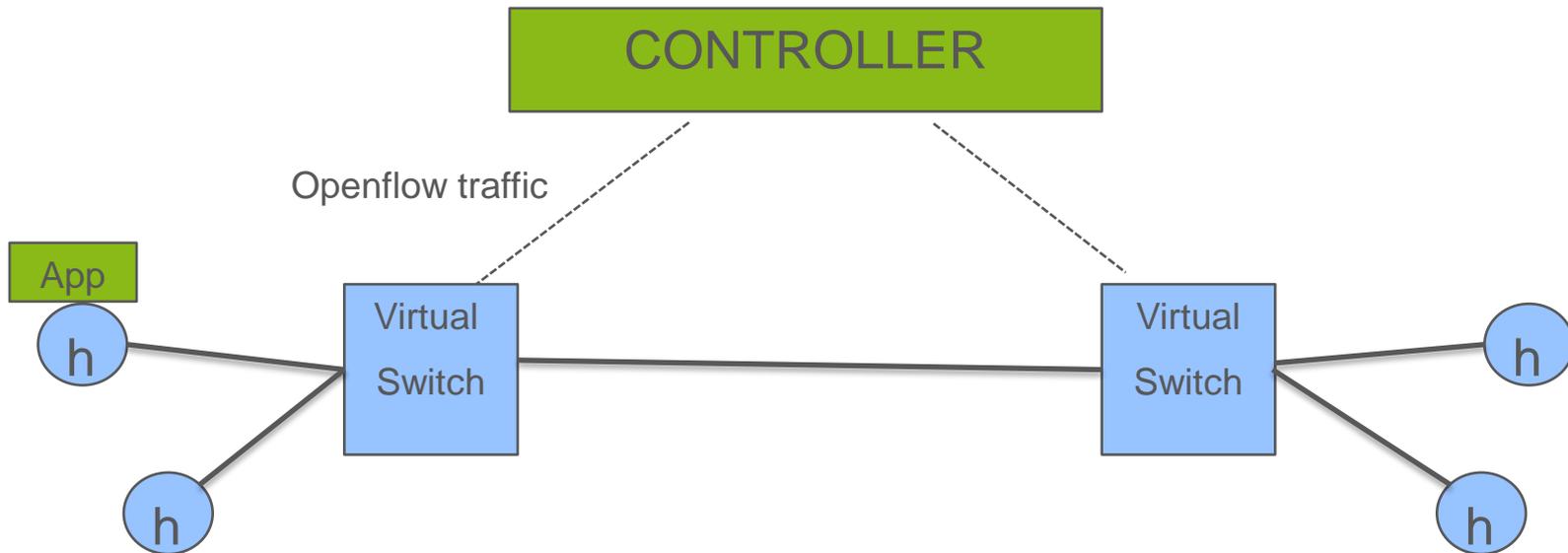
- Provide specification to implement Openflow-enabled forwarding devices
- Communication channel between data and control plane



# Mininet



- › Provide tools to create virtualized network with OVS
- › CLI for manipulating network dynamically
- › Virtualized hosts



# References



- › "The road to SDN" <http://queue.acm.org/detail.cfm?ref=rss&id=2560327>
- › Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *proceedings of the IEEE*, 103(1), 14-76.
- › "SDN Architecture 1.0 - Open Networking Foundation", [https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR\\_SDN\\_ARCH\\_1.0\\_06062014.pdf](https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf)
- › Openflow specification v1.0, <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.0.pdf>
- › Mininet, [mininet.org](http://mininet.org)
- › POX controller, <http://www.noxrepo.org/>



**ERICSSON**