# 5617, Fall 2022
# computer networking and communication

anduo wang, Temple University

TTLMAN 305, T 17:30-20:00

# to do 1

paper presentation, select your paper and date

- sign up ASAP — first come, first serve
- https://docs.google.com/document/d/11XbK2V0gMRwYanmjyQXshJvo9j7uqVyxsRe0PF7tfbc/edit?usp=sharing
- deadline: Sept 06
  - you will be assigned a date/paper after the deadline, if you have not signed up yet

# how to write a review

3-4 paragraphs

- summarize the (content of contribution of the) paper in your own words
- strength?
- weakness?

check the course website for detailed instructions and an example (template)

IMPORTANT: reviews not conforming to the requirements will be graded 0.

# to do 2

watch Scott Shenker's talk on "The Future of Networking, and the Past of Protocols" https://youtu.be/YHeyuD89n1Y

# to do — (optional) reading

## Brief history of the Internet

https://www.internetsociety.org/internet/history-internet/brief-history-internet/

# The Design Philosophy of the DARPA Internet Protocols

http://ccr.sigcomm.org/archive/1995/jan95/ccr-9501-clark.pdf

# why the protocol is as it is?

determine the motivation and reasoning which led to the design

- why IP is based on a connectionless or datagram model of service
- why layering of the architecture into the TCP/IP layers
- ...

# Internet architecture is still evolving

new extension changes the design principle

- SDN
- understanding of the history provides a necessary context

# (top level) goal of the Internet

an effective technique for multiplexed utilization of existing interconnected networks

- 1970s, assumption: ARPA packet radio network, but the assumption was: there would be other sorts of networks to interconnect
- 1980s: development of local area networks (LAN
- today: LAN (Ethernet) is (probably) the dominant network in the Internet

# (top level) goal of the Internet

an effective technique for multiplexed utilization of existing interconnected networks

unified architecture which incorporates a variety of predefined media transmission?

# (top level) goals of the Internet

an effective technique for multiplexed utilization of existing interconnected networks

unified architecture which incorporates a variety of predefined media transmission?

an open architecture for integrating separately administrated n entities into a common unity

# (top level) goals of the Internet

an effective technique for multiplexed utilization of existing interconnected networks

unified architecture which incorporates a variety of predefined media transmission?

circuit switching

an open architecture for integrating separately administrated n entities into a common unity

packet switching

# (top level) goals of the Internet

an effective technique for multiplexed utilization of existing interconnected networks

unified architecture which incorporates a variety of predefined media transmission?

circuit switching

an open architecture for integrating separately adminstrated n entities into a common unity

packet switching

"Leonard Kleinrock at MIT published the first paper on packet switching theory in July 1961, … of the theoretical feasibility of communications using packets rather than circuits"

# (top level) assumption

packet switching accepted as a fundamental component of the Internet architecture

- the glue that pulls together the various evolving network technologies
- the WAIST: a central, largely unchanging API above and below which innovation is enabled

# from the assumption

(the fundamental structure of the Internet) a packet switched communications facility in which a number of distinguishable networks are connected together using gateways which implement a store and forward packet forwarding algorithm

# (top level) goal

an effective technique for multiplexed utilization of existing interconnected networks

# (second level) goals

an effective technique for multiplexed utilization of existing interconnected networks

- survivability in the face of failure
- type of services to support
- varieties of networks to incorporate and utilize

# survivability

entities communicating can continue without having to reestablish or reset the high level state of their conversation

- synchronization never lost unless no physical path available
- at the top of transport: only one failure — total partition

# survivability

entities communicating can continue without having to reestablish or reset the high level state of their conversation

- synchronization never lost unless no physical path available
- at the top of transport: only one failure — total partition

the Internet architecture was to mask any transient failure

# achieving survivability

entities communicating can continue without having to reestablish or reset the high level state of their conversation

- in some architecture, the state is stored in the intermediate nodes

complex: to protect the information loss, the intermediate nodes must replicate state

# achieving survivability

entities communicating can continue without having to reestablish or reset the high level state of their conversation

- fate sharing: move the information (in the network) to the end point of the network, at the entity which is utilizing the networking services

# survivability by fate sharing

information about transmission stored in the host

- protect against any number of intermediate failures
- more trust in the host
- simple: easy to engineer

# types of services (TOS)

## Ist attempt

- (traditional type of service) virtual circuit service — bi-directional reliable data deliver (rdt)
- rdt is the first service recognized and implemented
  - TCP (transmission control protocol)

# reliable data delivery with TCP?

## this service has multiple variants

- remote login
  - low delay, low requirement for bandwidth
- file transfer
  - less concerned with delay, but requires high throughput

# reliable data delivery with TCP?

services *not* fit TCP well

- realtime teleconferencing
  - less concerned with reliability, but requires minimizing and smoothing the delay
- network debugger
  - when the network is already under stress and failures

# delay with TCP

the most serious source of delay in the network is the (TCP) mechanism to provide reliable data transmission

- the delay due to retransmission can be many times the round trip delivery time (RTT)
- completely disrupt the packet delivery (reassembly algorithm)

# delay with TCP

the most serious source of delay in the network is the (TCP) mechanism to provide reliable data transmission

- the delay due to retransmission can be many times the round trip delivery time (RTT)
- completely disrupt the packet delivery (reassembly algorithm)

but occasionally missing packets does not impair the intelligibility of the speech

- even if it does, listeners can ask the speaker to repeat

# reliable data delivery with TCP?

more services *not* fit TCP well

- network control

discussion …

# flexible transport services

the architecture must be prepared to tolerate (admit) simultaneously transports which wish to constrain reliability, delay, or bandwidth, at a minimum

- TCP splits into TCP/IP

# TCP/IP layering

TCP: reliable sequenced data stream

IP: basic building block out of which a variety of TOS could be created

- datagram
- best effort
- UDP (user datagram protocol), an application level interface

# type of services, recap

the architecture did not wish to assume that the underlying network themselves support multiple TOS

- the hope was that multiple TOS can be constructed out of the basic datagram building block using algorithms within the host and the gateway

# varieties

the architecture operates over a wide variety of networks

# varieties by minimal assumption

about the function the network will provide

- can transport a datagram or a packet
- suitable form of addressing

# varieties by minimal assumption

about the function the network will provide

- can transport a datagram or a packet
- suitable form of addressing

explicitly not assumed

- reliable sequenced data transfer
- broadcast/multicast
- differentiated services
- internal knowledge of failures, speed, delay

*IP!*

# other goals — distributed management

## partial success: two-tiered routing algorithms

- the Internet divided into domains
- each domain manages a subset of the gateways
- gateway from different administrations exchange routing tables
- private routing algorithms used among the gateways in a single administration

*BGP*

# other goals — distributed management

## partial success: two-tiered routing algorithms

- the Internet divided into domains
- each domain manages a subset of the gateways
- gateway from different administrations exchange routing tables
- private routing algorithms used among the gateways in a single administration

- lack of sufficient tool, even in the area of routing
- management of resources in the context of multiple administrations?

# other goals — cost inefficiency

## header of (IP) packet is long

- typically 40 bytes
- remote login: 1 byte of data + 40 bytes of header!

## (end points) retransmission of lost packets

- may cross several intervening nets a second time
- OK if retransmission is low (<1%)

## cost of attaching a host to the network

- all mechanisms (acks, retransmission) implemented in the host
- higher than other alternatives

# remaining goals

## accountability

- few tools for accounting for packets flow
- but consumers concerned with understanding and monitoring the usage within the network

## guidance

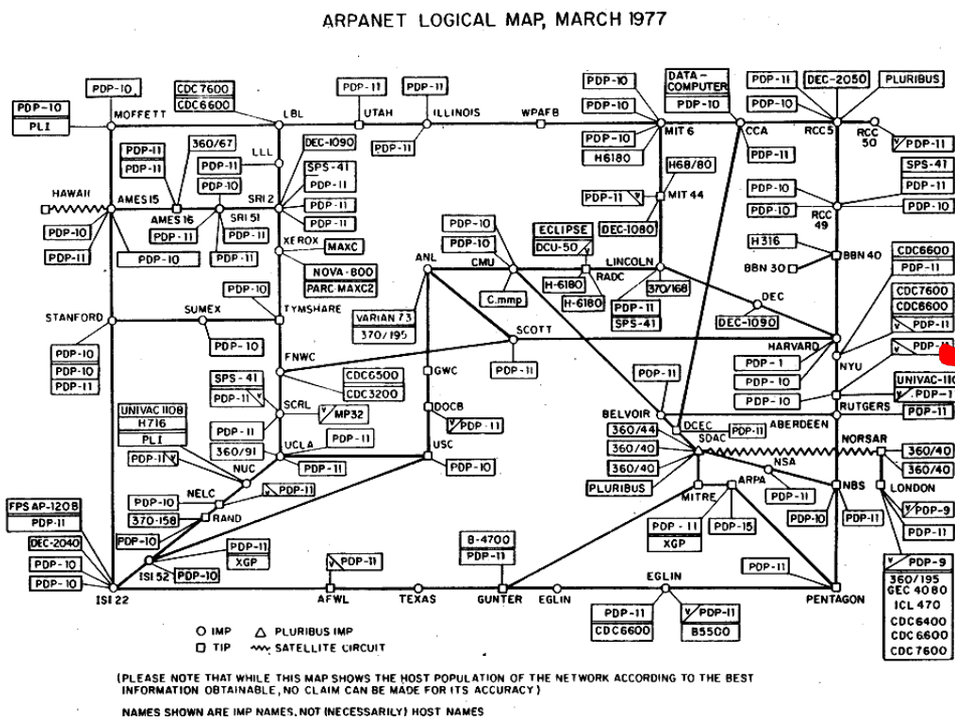
- the present architecture permits wide flexibility (service)
- but how to give guidance to a particular network
  - few guidance that would relate the engineering of the network to the type of services which would result

# software defined networking (SDN)

# the Internet: a wonderful success

## a remarkable story

- from research experiment to global infrastructure




ARPANET, 1977

today

*source*: https://en.wikipedia.org/wiki/Internet

# the Internet: a wonderful success

## innovations for everyday life

- Web, P2P, VoIP, social networking …

# the Internet: a wonderful success

## innovations take rapid transitions

Ahmed Khurshid., et al. "VeriFlow: Verifying Network-Wide Invariants in Real Time"
source: https://www.usenix.org/conference/nsdi13/technical-sessions/presentation/khurshid
NSDI **2013**

3 years, $8.2 million

**Veriflow Nabs $8.2 Million For Clever Ideas About Network Outage Prevention**

JULY 19, 2016 BY DREW CONRY-MURRAY

Startup Veriflow Networks has landed $8.2 million in series A funding. The A round was led by Menlo Ventures, along with its existing investor New Enterprise Associates.

http://packetpushers.net/veriflow-nabs-8-2-million-clever-ideas-network-outage-prevention/

# inside the 'Net': a different story

vendor lock-in

- specialized hardware
- protocols/software bundled with hardware
- slow innovation, deployment
- *$$$$$*

increasingly complex

- operators today are *masters of complexity*

# discipline for networking?

operating systems

- time sharing

programming languages

- data abstractions

database management systems

- data independence

networking

- lack of discipline, but bags of protocols …

# networking needs …

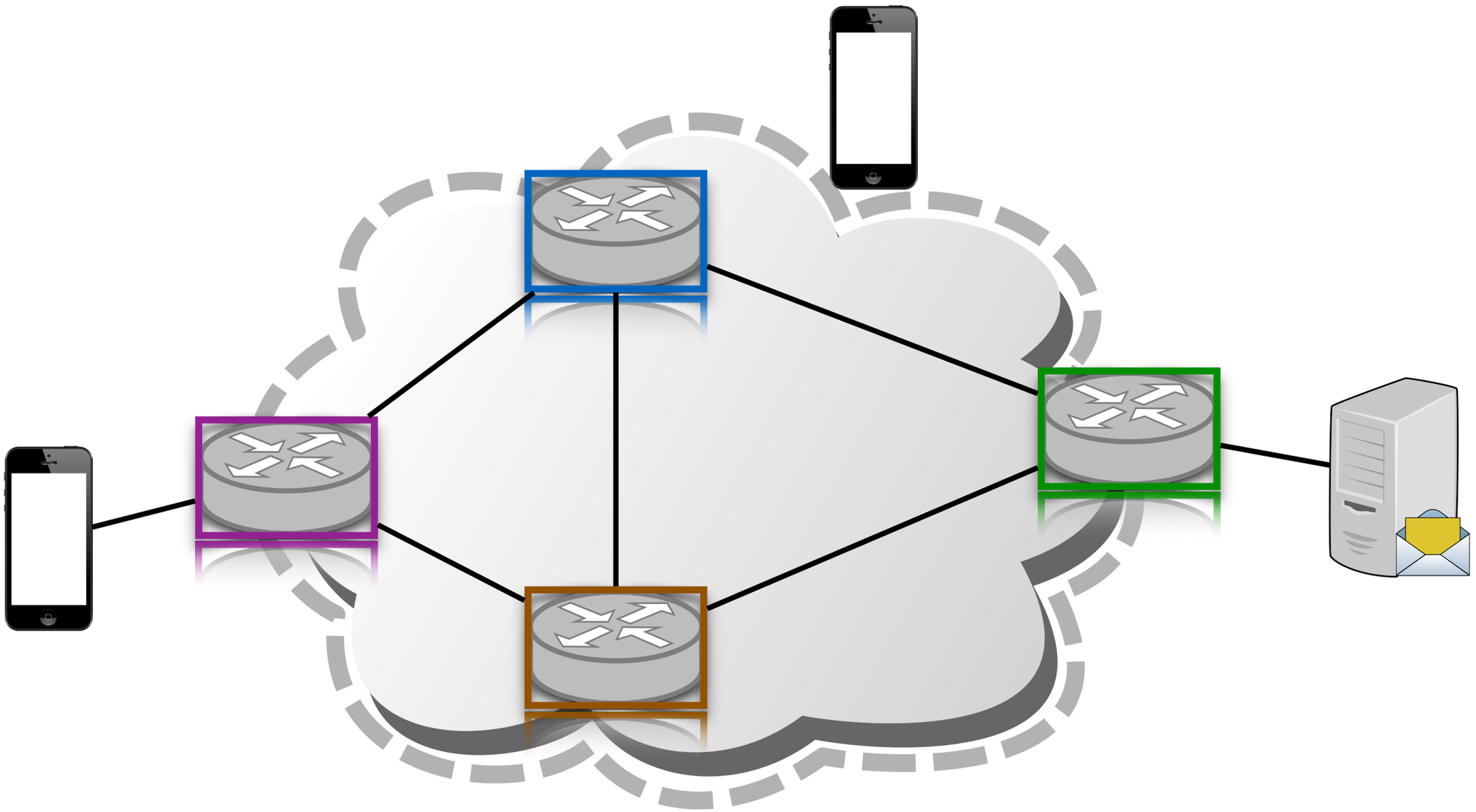
## break vendor lock-in

- freedom from suppliers
- freedom from low-level box by box configuration
- freedom of adding new services

## introduce disciplines

- systematic principles that guide networking practice

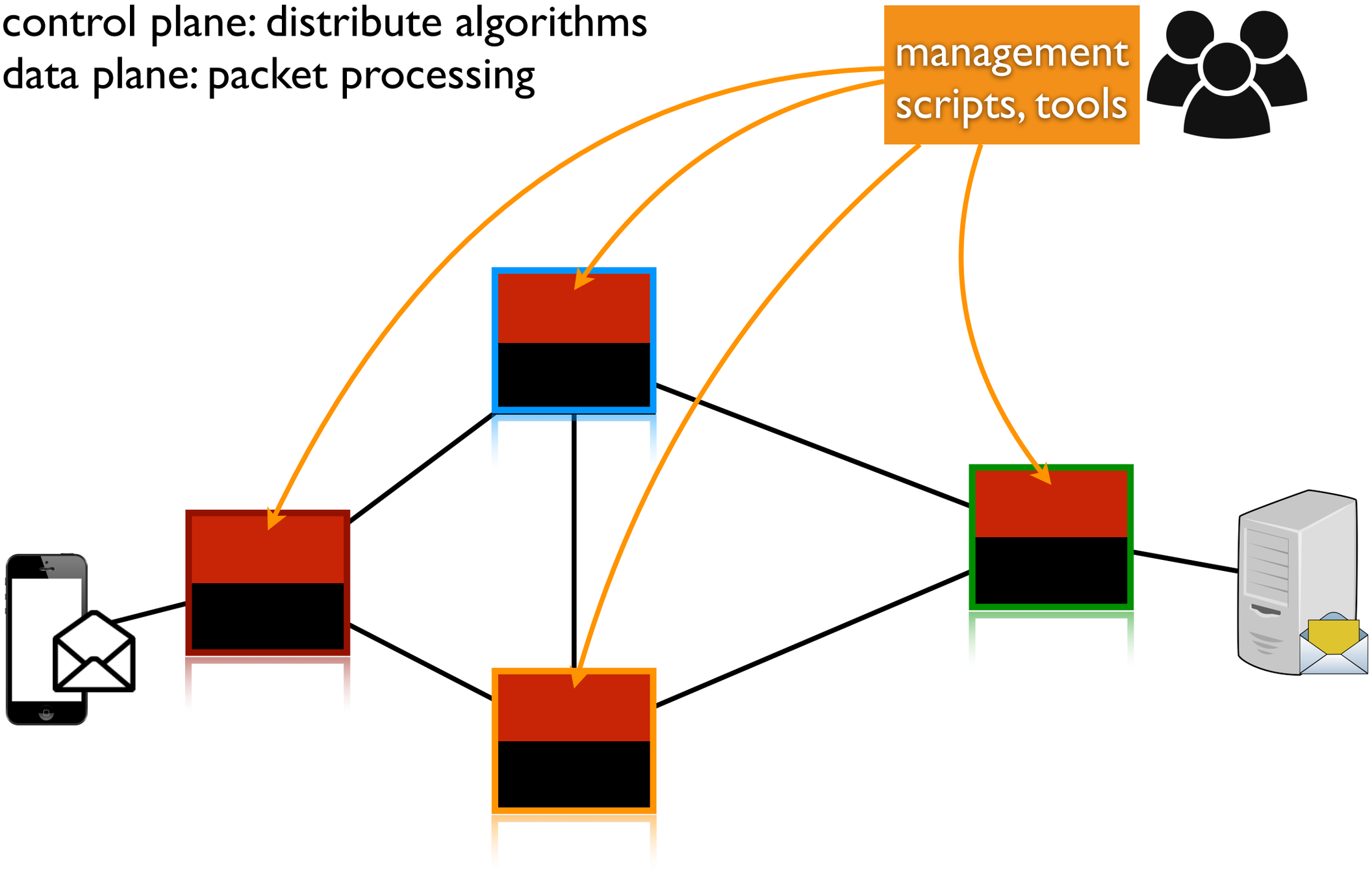
# software defined networks



# software defined networks

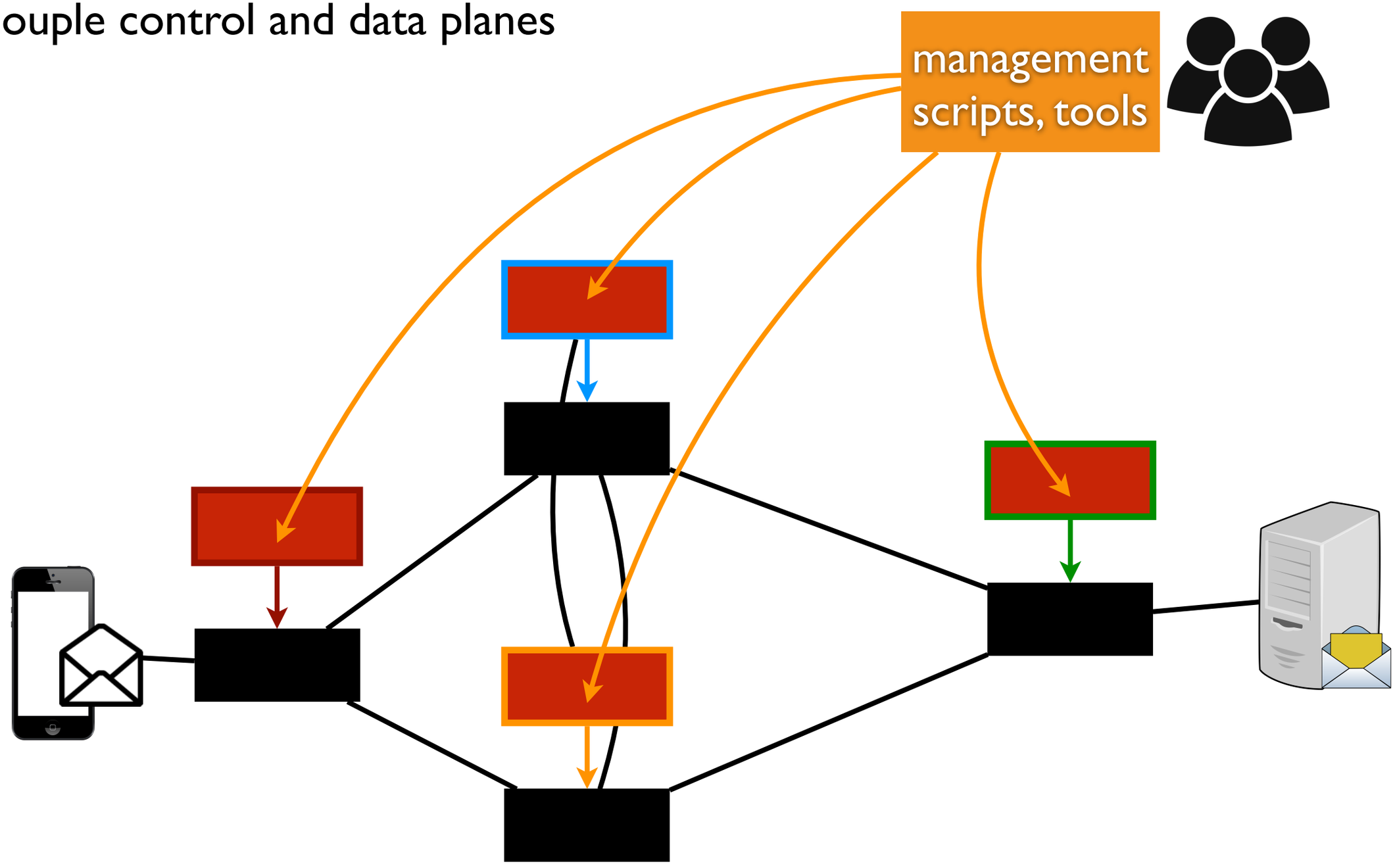
control plane: distribute algorithms

data plane: packet processing



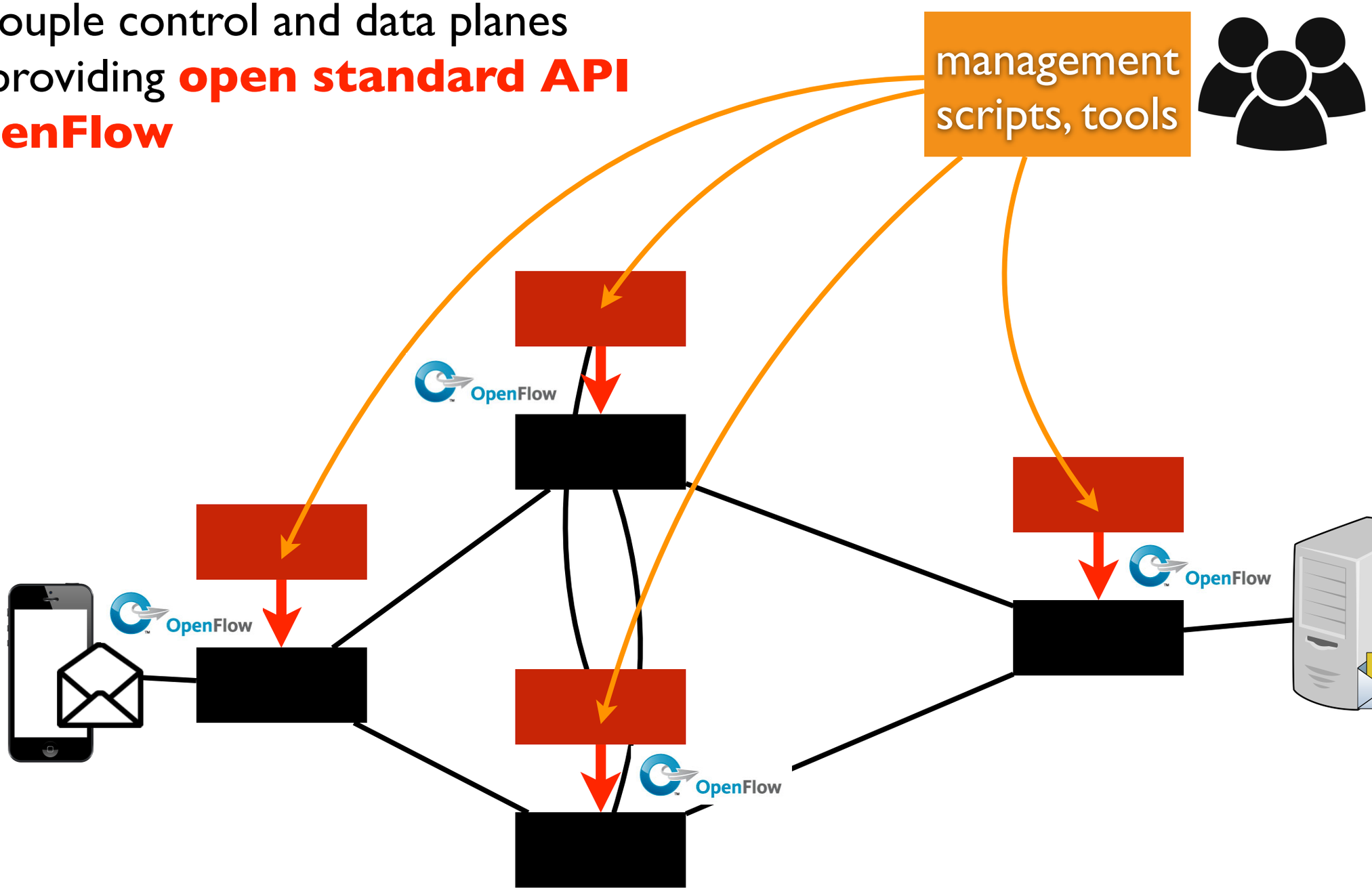
# software defined networks

decouple control and data planes



# open dataplane API

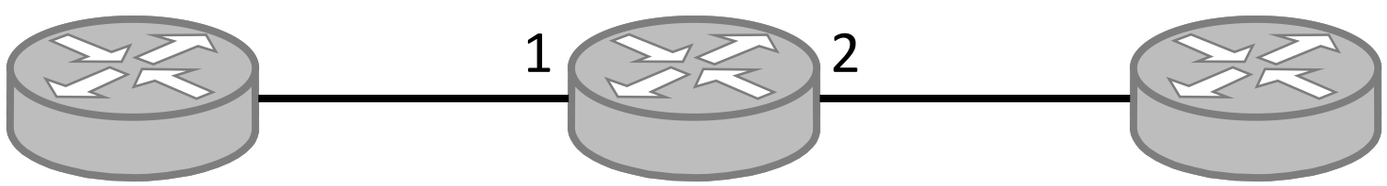
decouple control and data planes
by providing **open standard API**
**OpenFlow**



# OpenFlow: simple open dataplane API

## prioritized list of rules


- pattern → action
  - pattern: match packet header bits
  - actions: drop, forward, modify, send to controller
- priority: disambiguate overlapping patterns

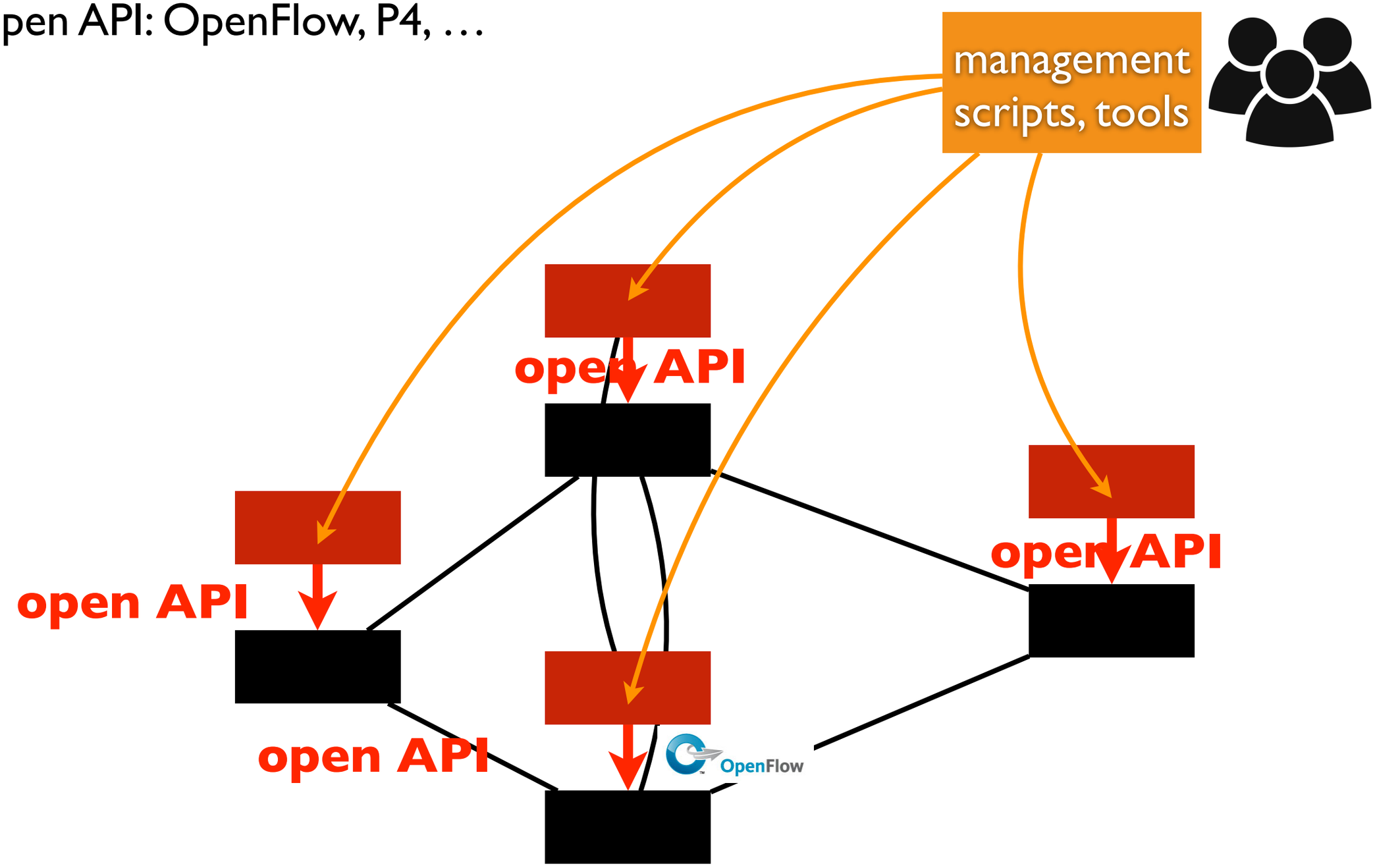



1. src=1.2.*.*, dest=3.4.5.* → drop
2. src = *.*.*.*, dest=3.4.*.* → forward(2)
3. src=10.1.2.3, dest=*.*.*.* → send to controller

Nick McKeown., et al. "OpenFlow: enabling innovation in campus networks"

# open dataplane interface

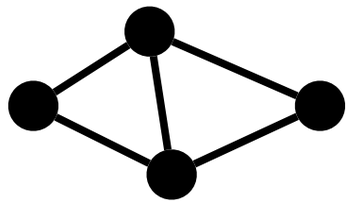
open API: OpenFlow, P4, …

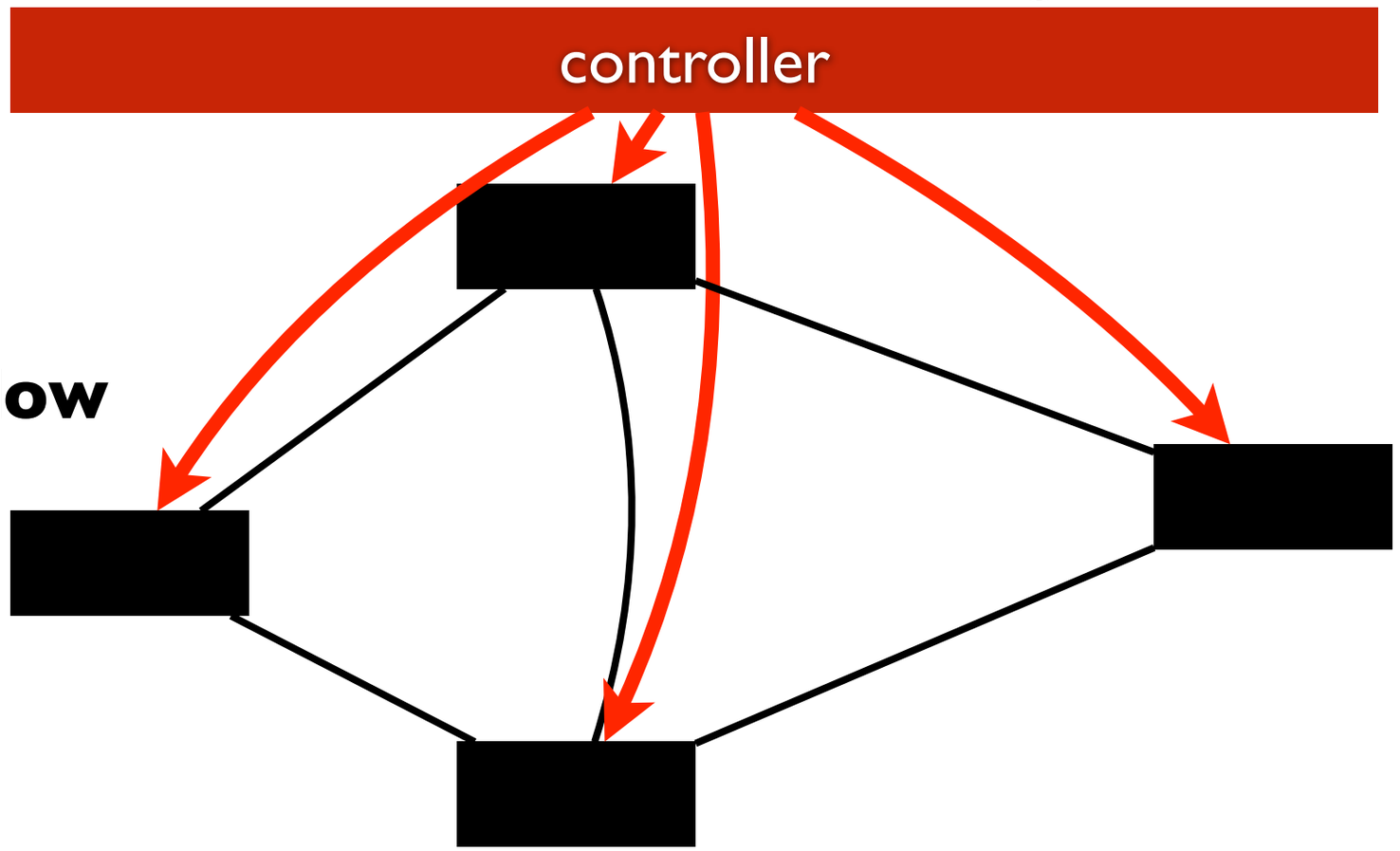



P. Bosshart., et al. P4: Programming Protocol-Independent Packet Processors

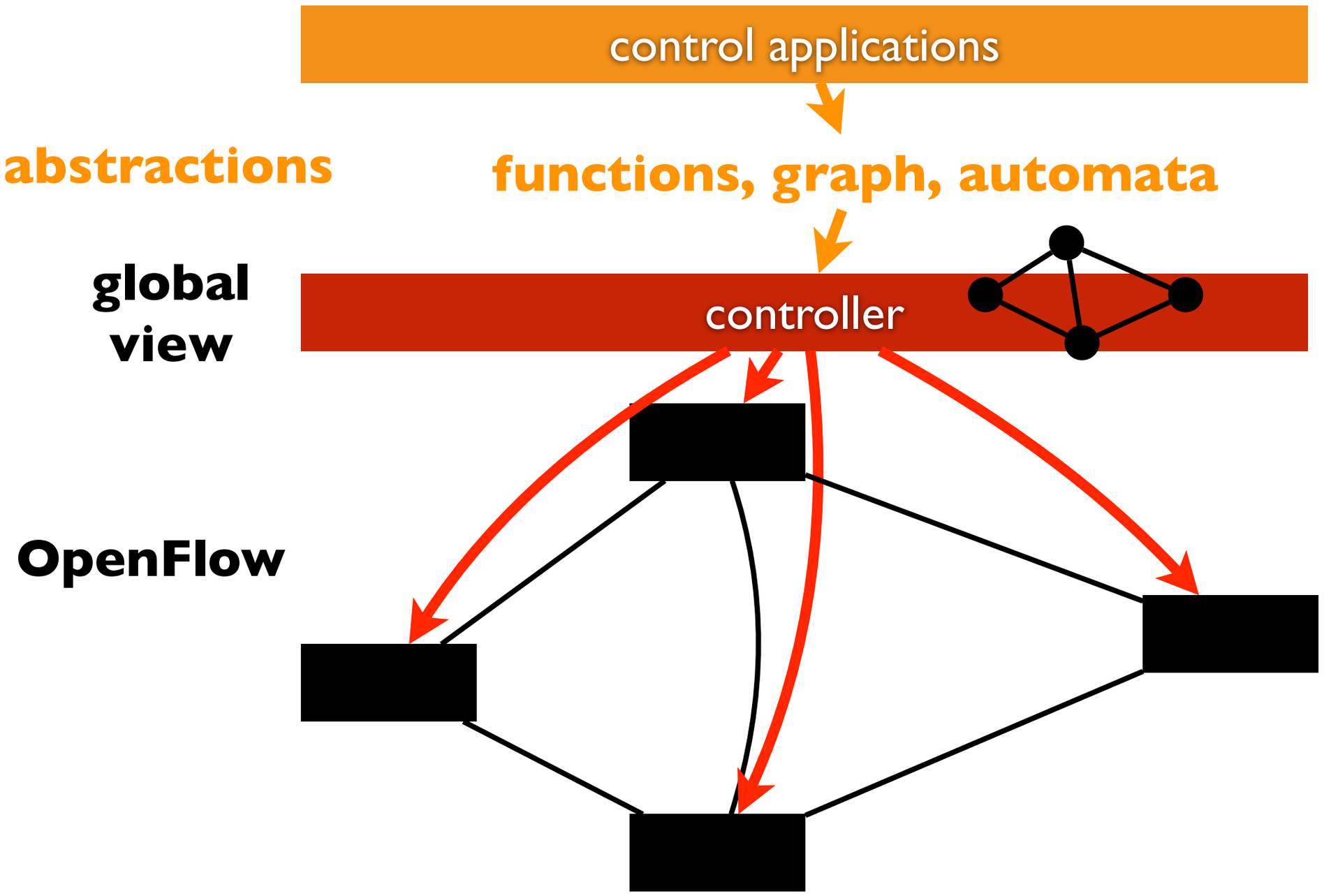
# (logically) centralized controller

**global network view**





# higher-level abstractions

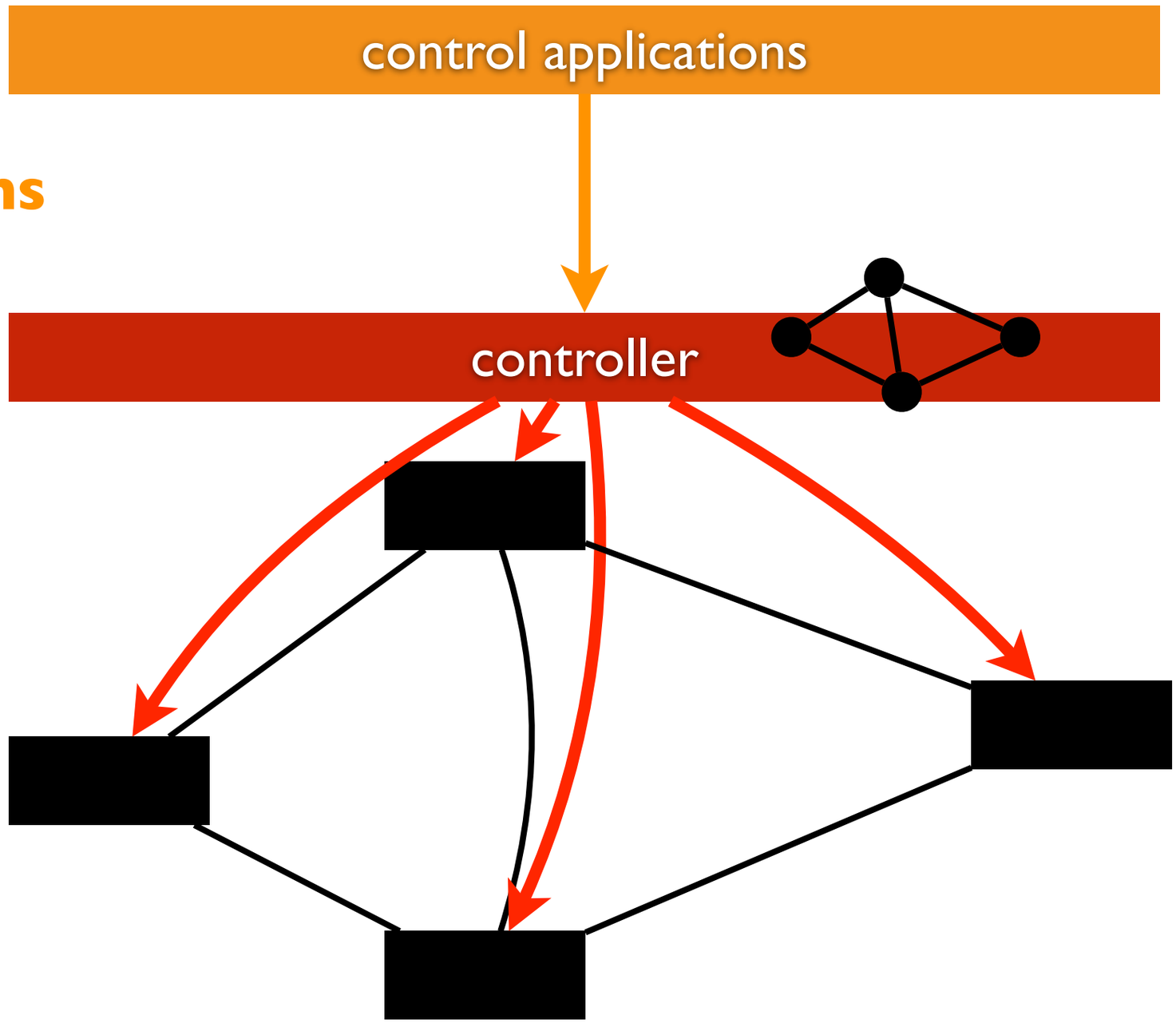


# protocols ➔ applications

**abstractions**

**global view**

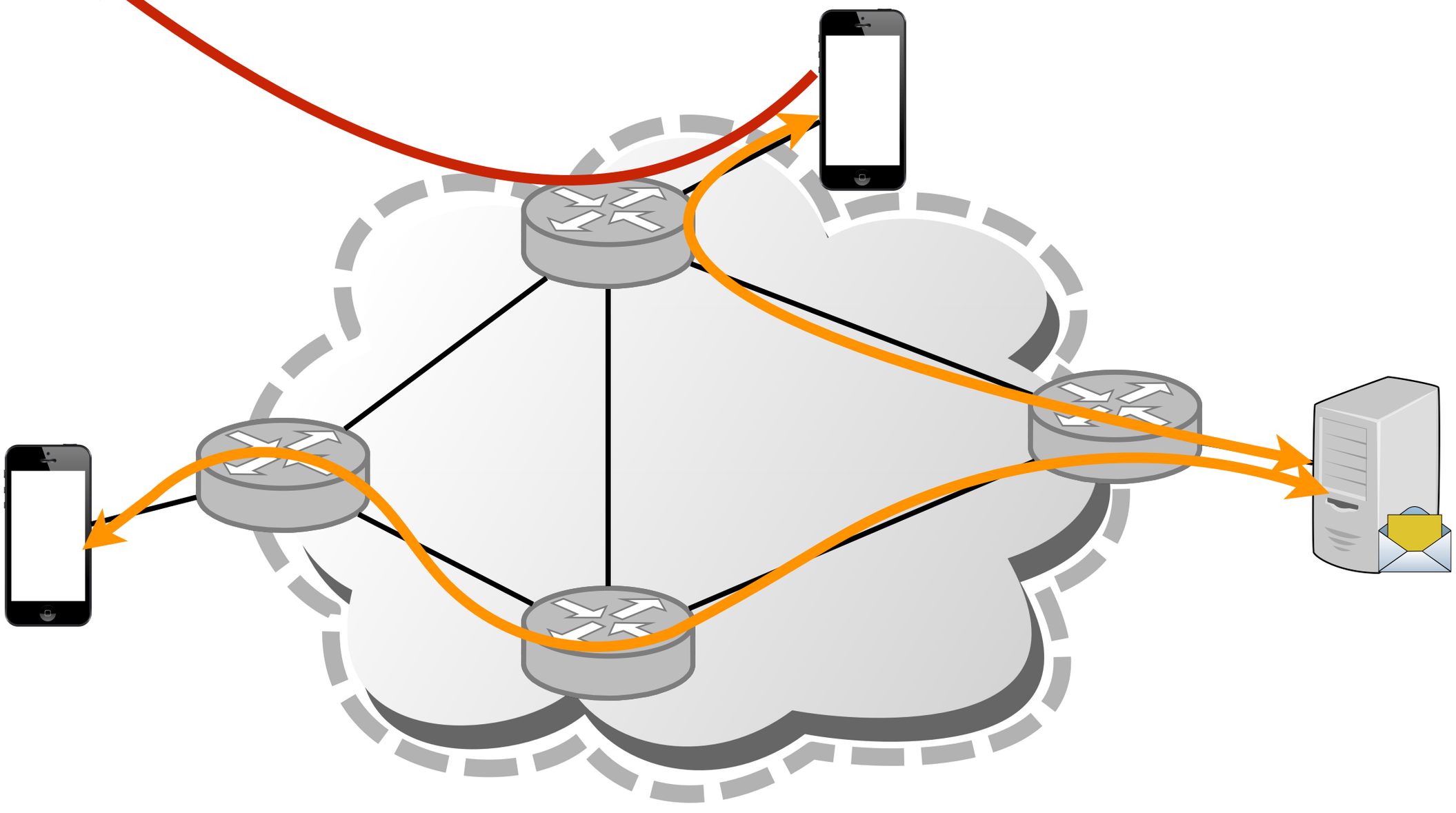
**OpenFlow**



# application: seamless mobility

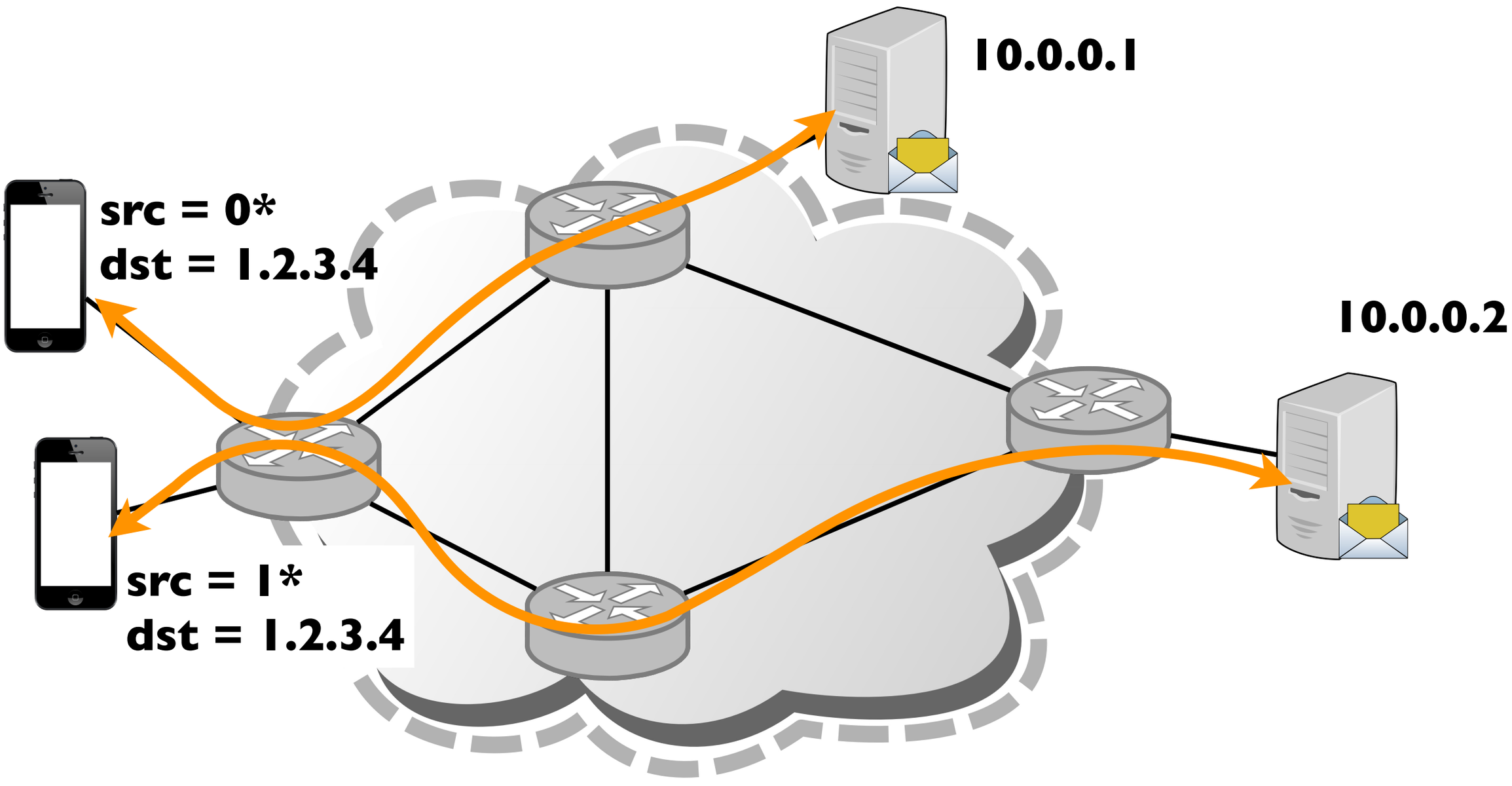
app

- See host sending traffic at new location
- Modify rules to reroute the traffic

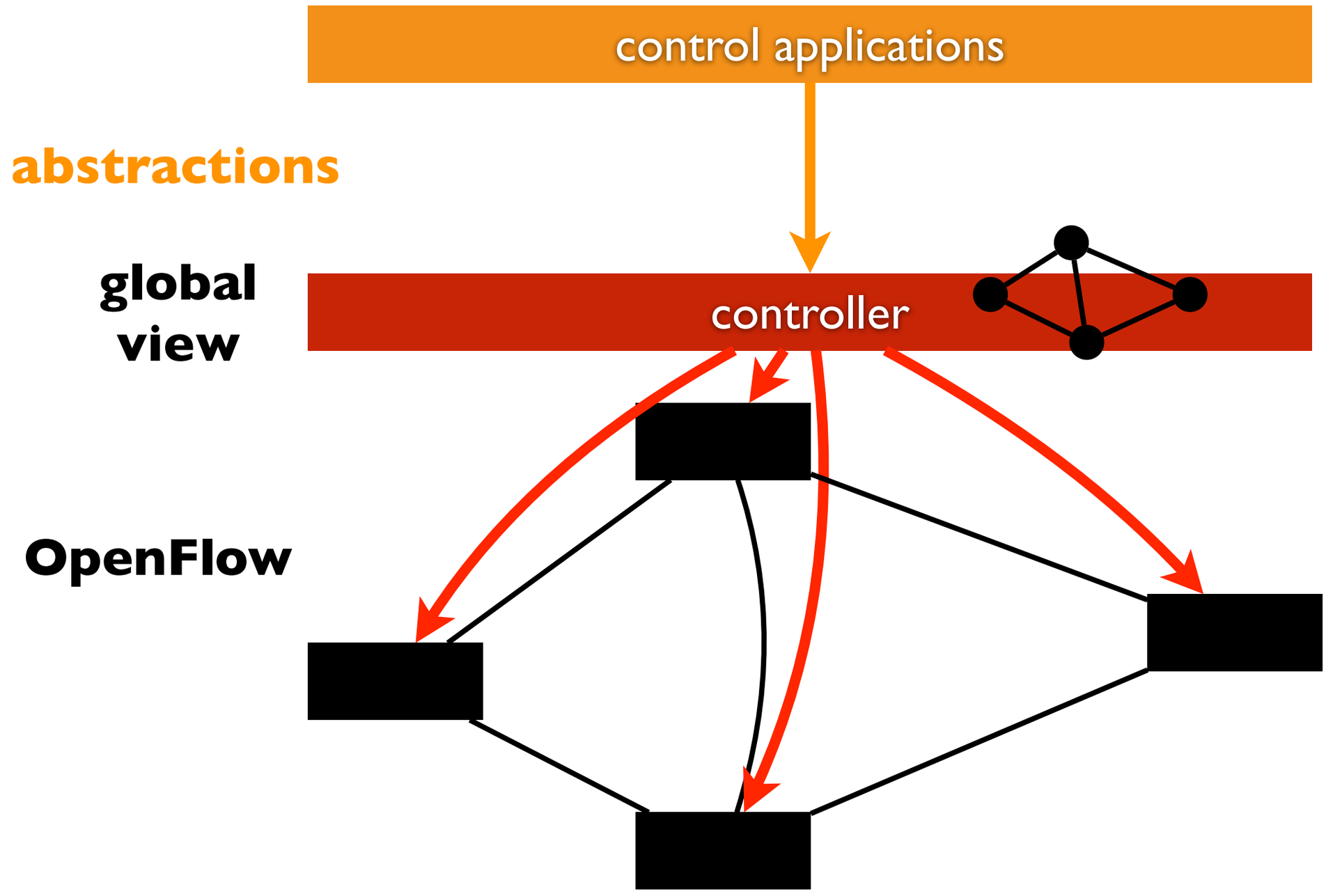


# application: server load balancing

- pre-install load-balancing policy
- split traffic based on source IP



# protocols ➔ applications

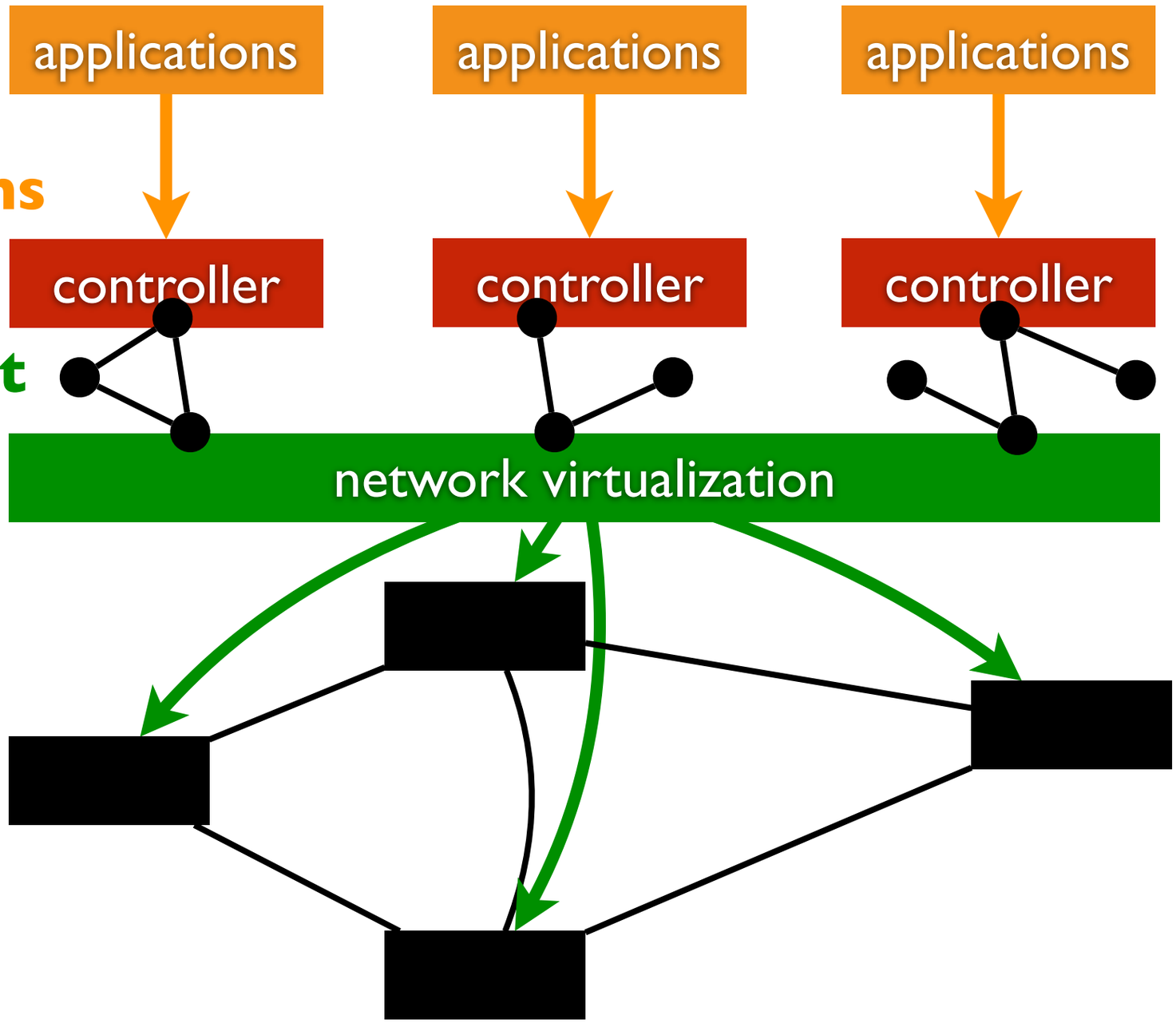


# network virtualization

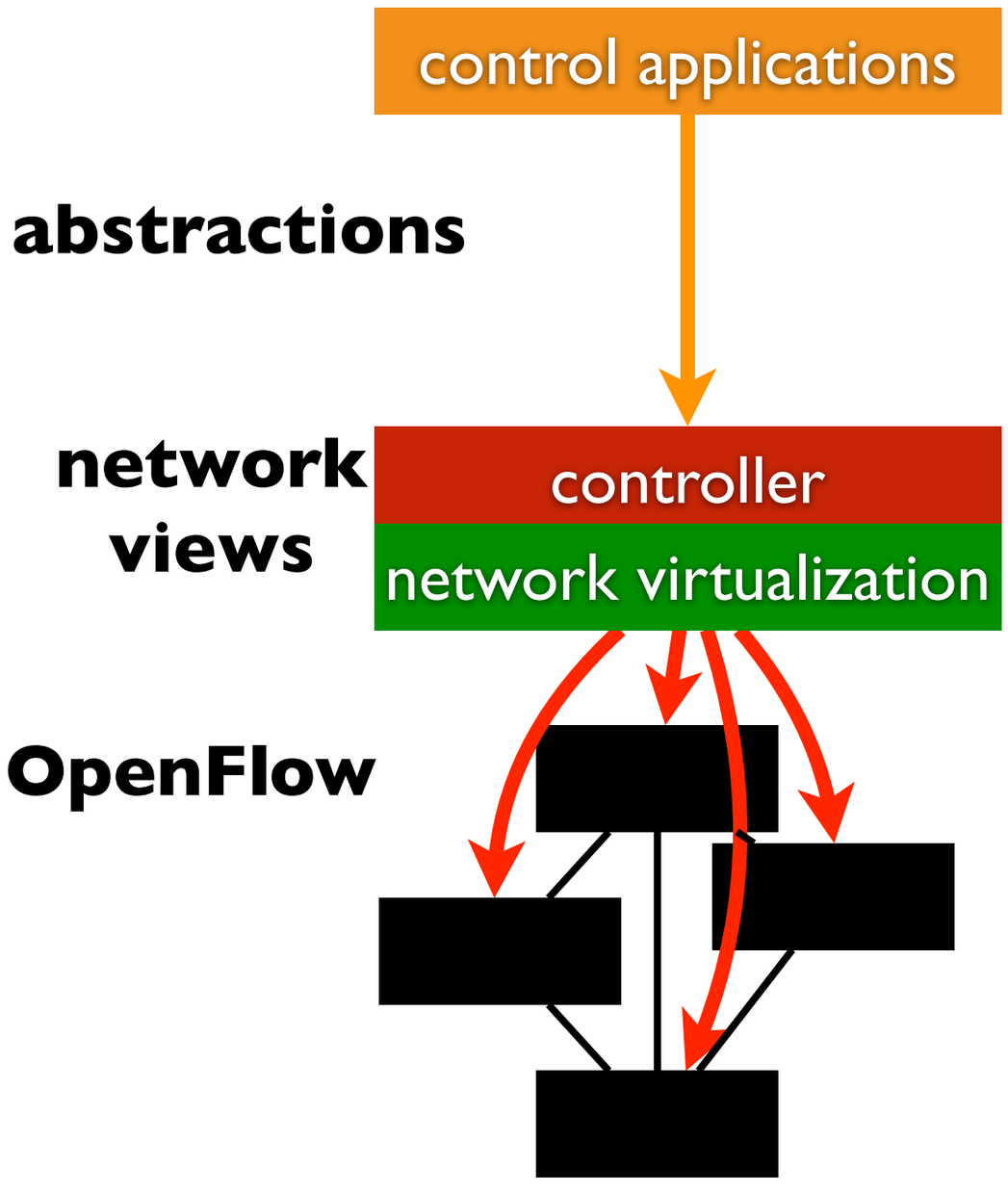
**abstractions**

**virtual Net**

**OpenFlow**



# recap: SDN technologies




## supporting technologies

- central network control
- programmability
- network virtualization

## benefits

- simplified operation with direct, network-wide control
- cost reduction with open hardware

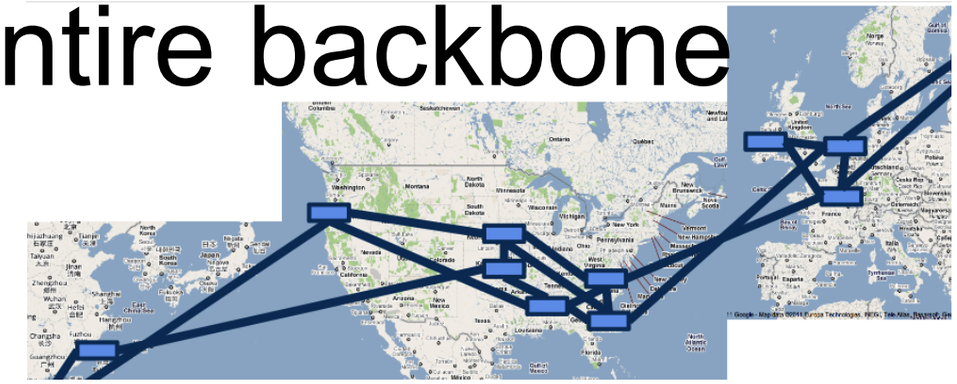
# a major trend in networking

OPEN NETWORKING FOUNDATION

Deutsche Telekom · facebook · Goldman Sachs · Google · Microsoft · NTT Communications · verizon · YAHOO!

Google

Entire backbone



runs on SDN

nicira

Bought for **$1.2 x $10^9$**
(mostly cash)