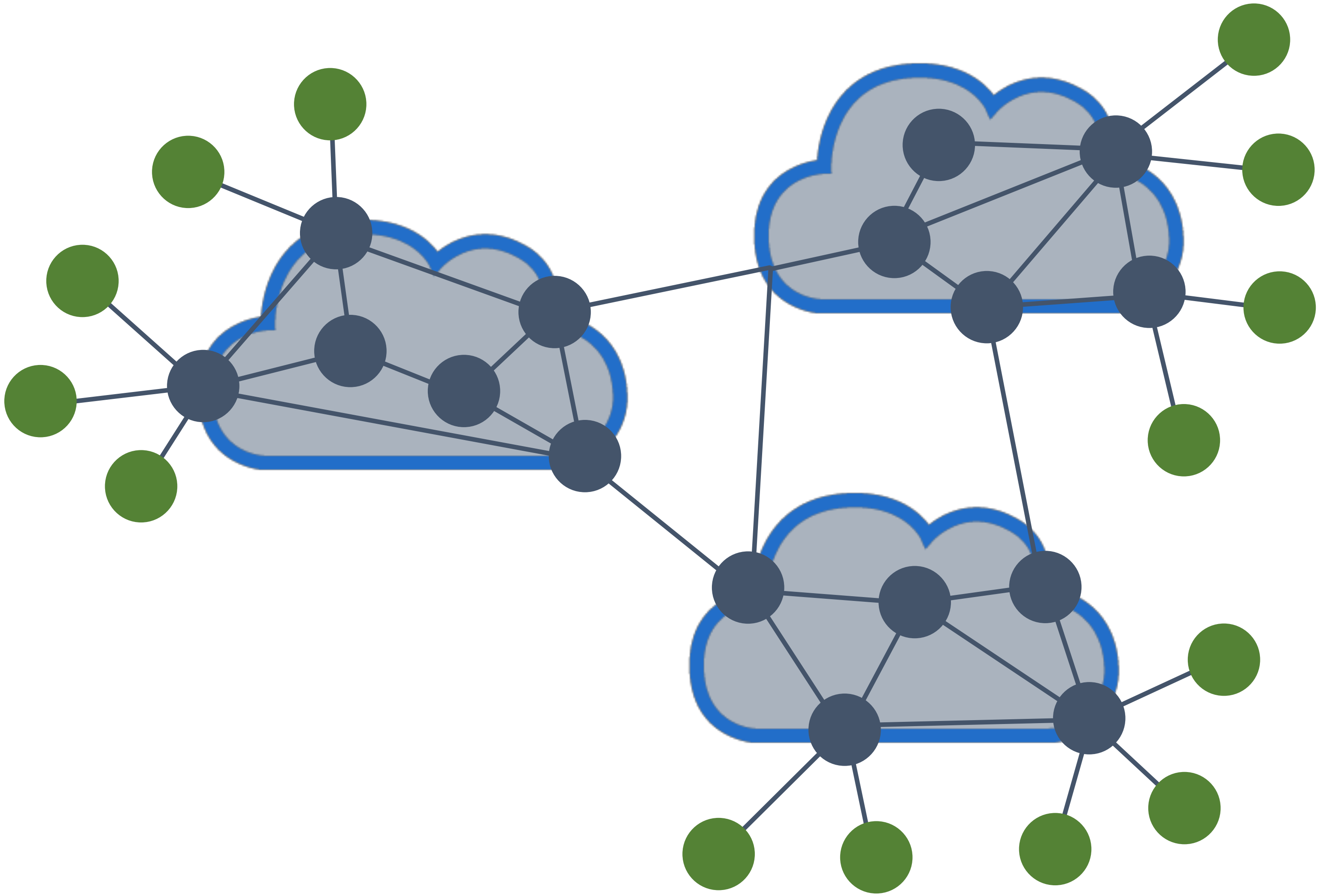
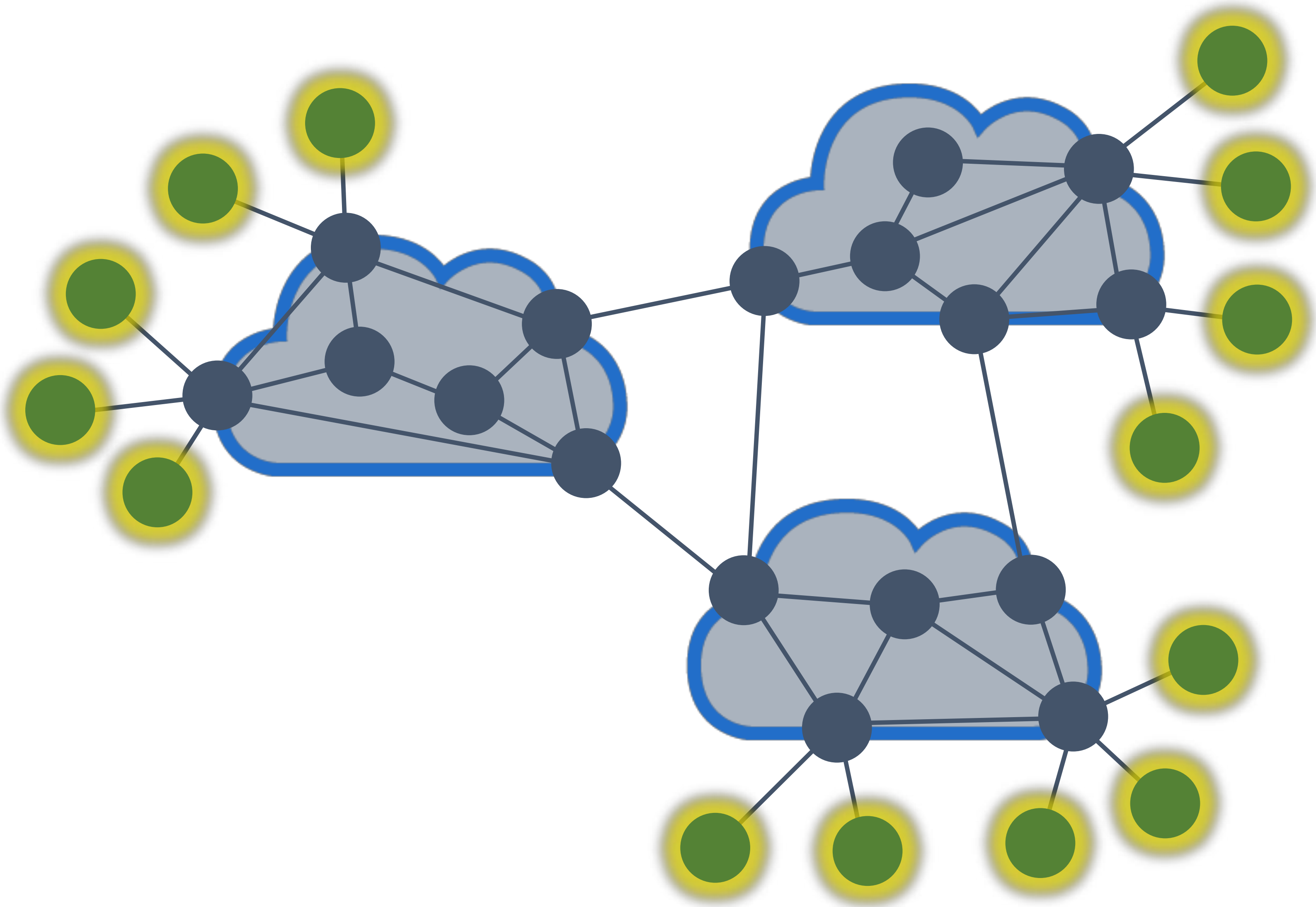
Classical Networking Introduction

QuTech, Delft

The Internet: a network of networks



Hosts: the end-point devices

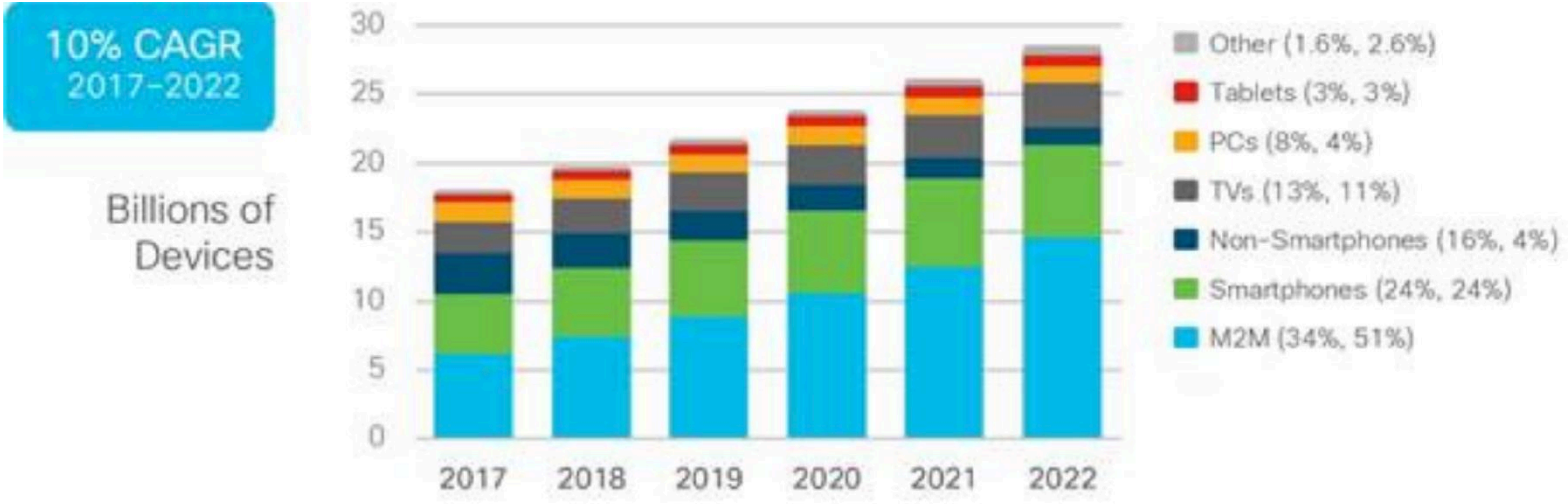


User devices

Servers

IoT devices

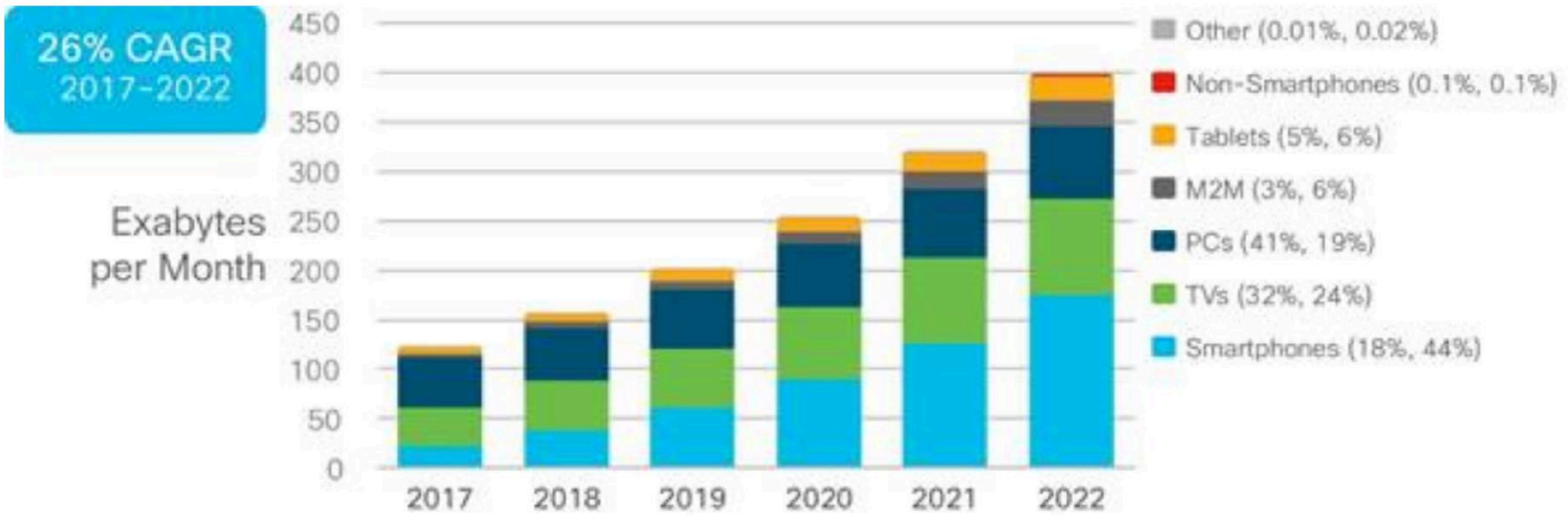
Tens of billions of end-point devices



* Figures (n) refer to 2017, 2022 device share

Source: Cisco VNI Global IP Traffic Forecast, 2017-2022

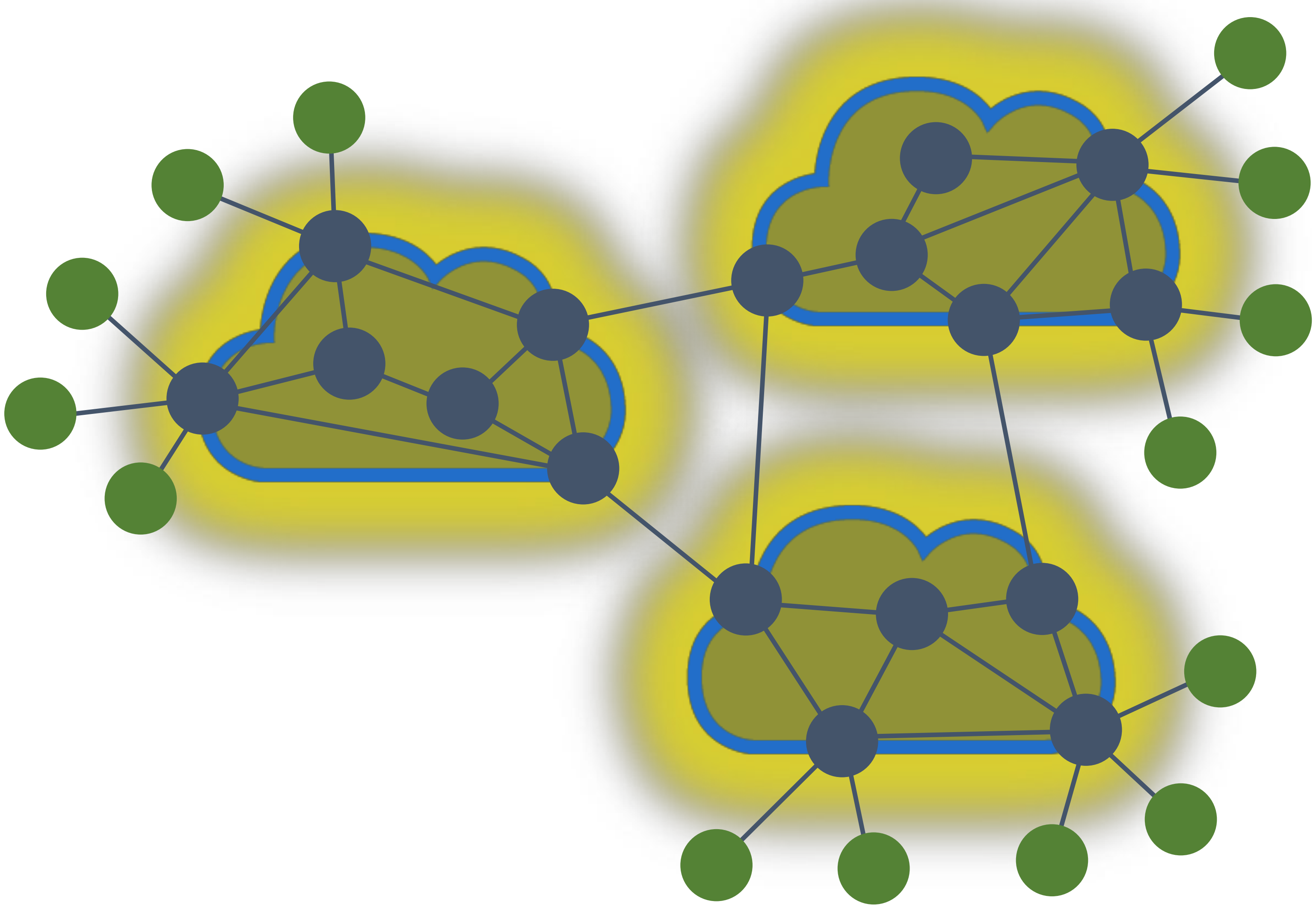
Hundreds of exabytes (10^{18} bytes) per month



* Figures (n) refer to 2017, 2022 traffic share

Source: Cisco VNI Global IP Traffic Forecast, 2017-2022

Autonomous systems: administrative domains

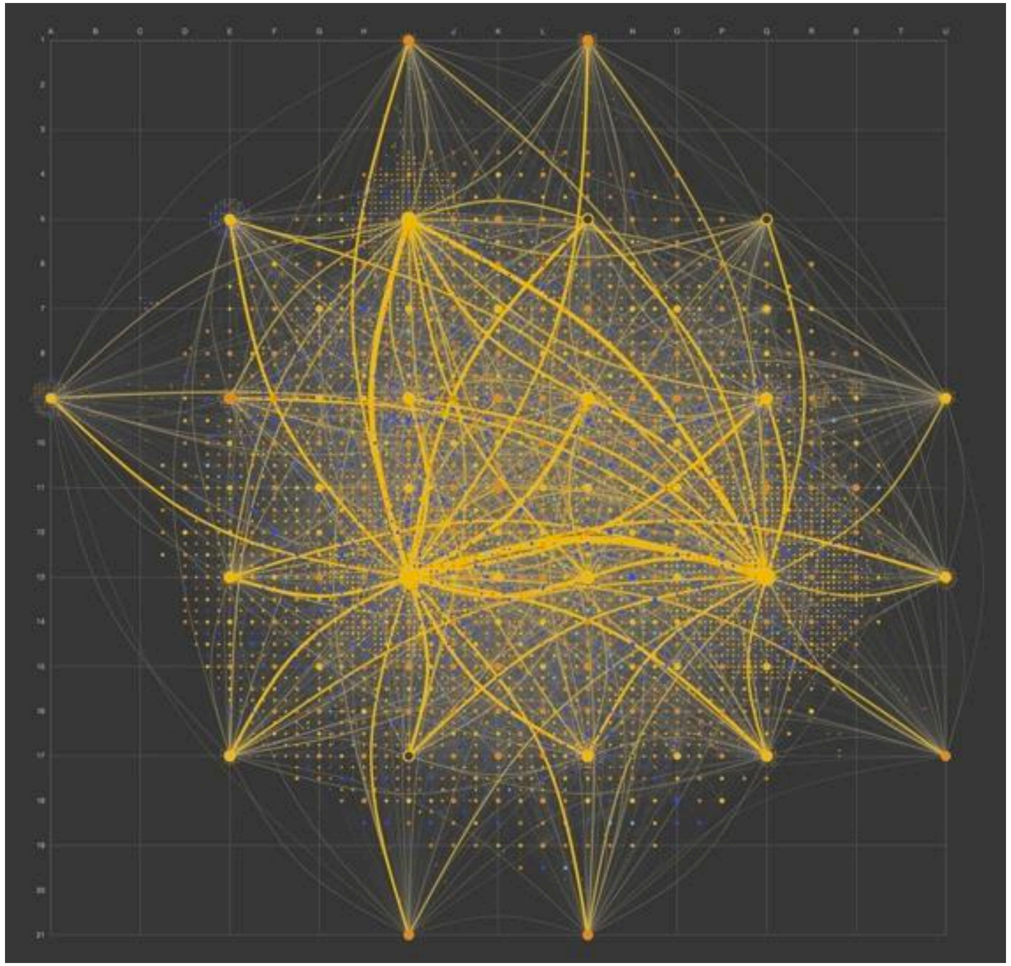


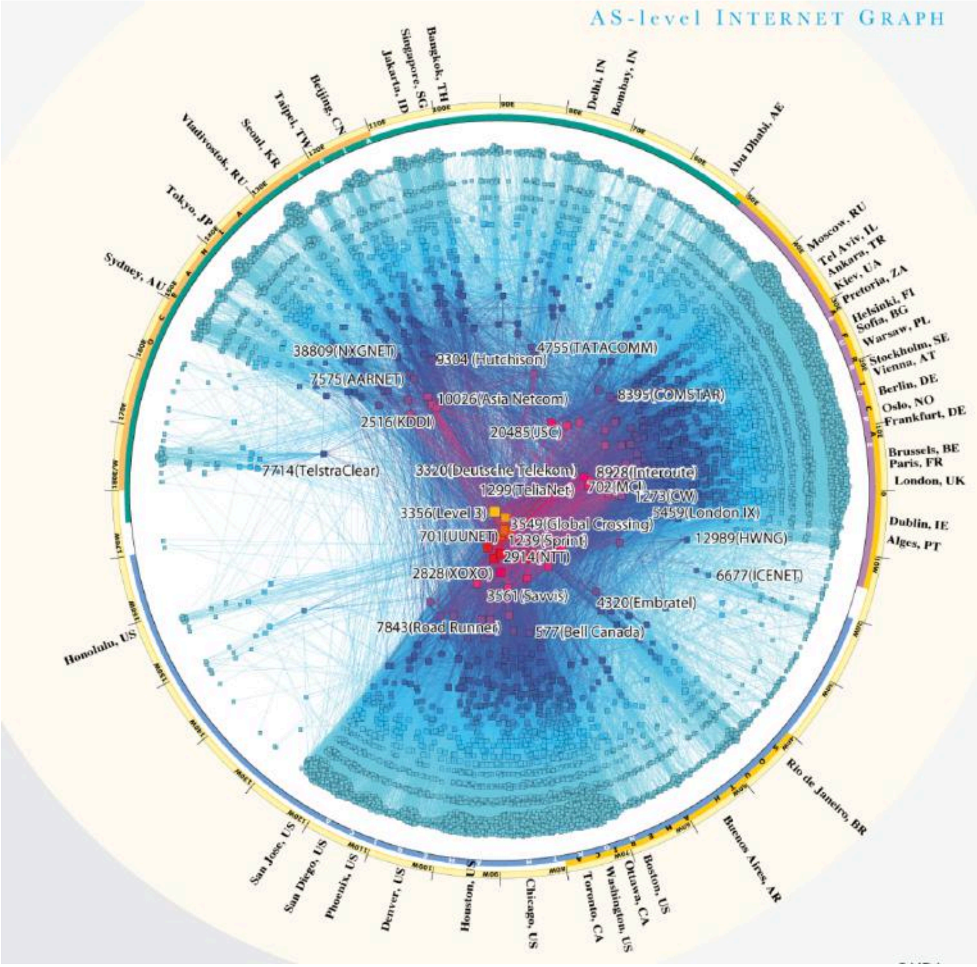
Service providers

Large companies

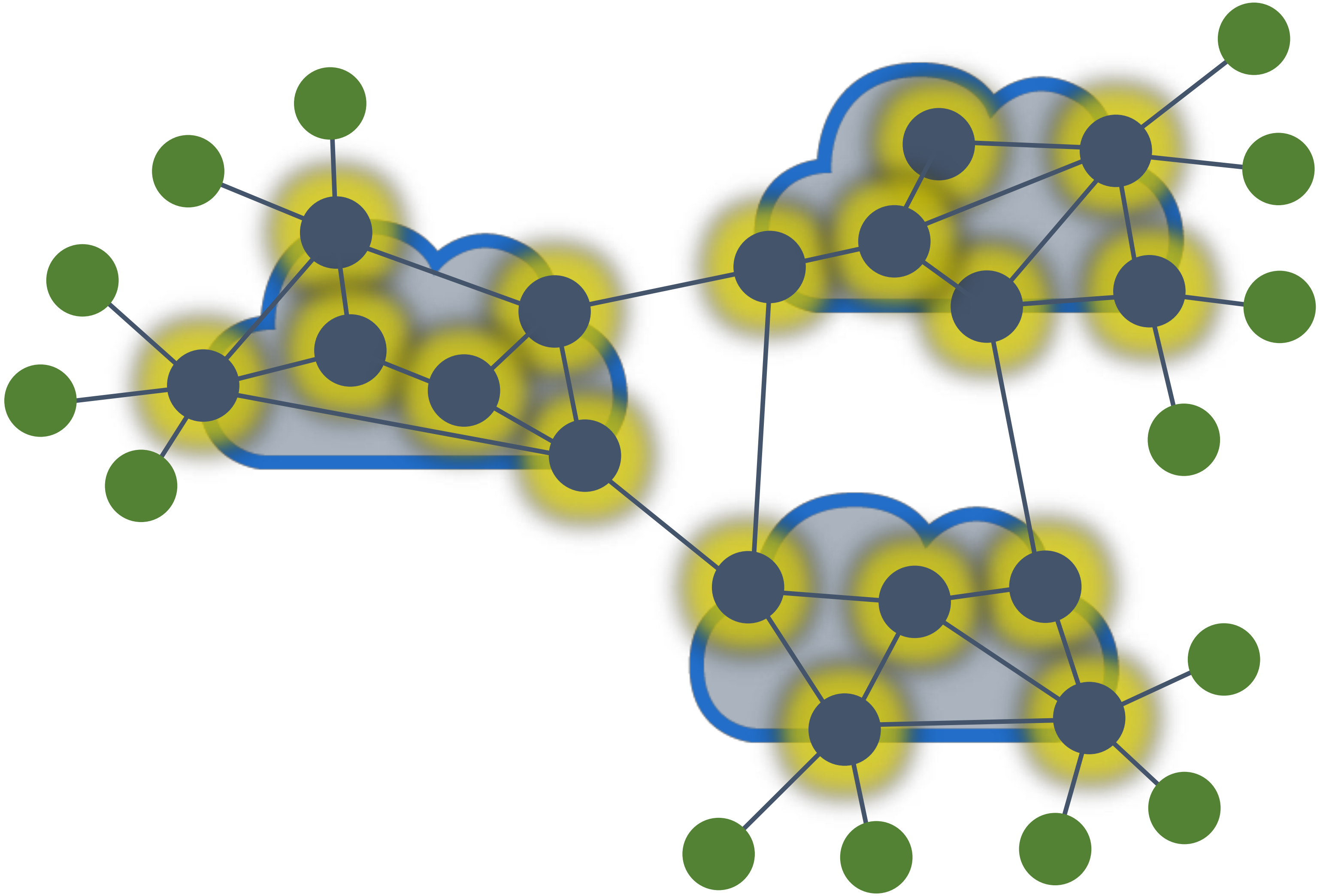
Total > 50,000

Maps of the internet (at AS level of detail)



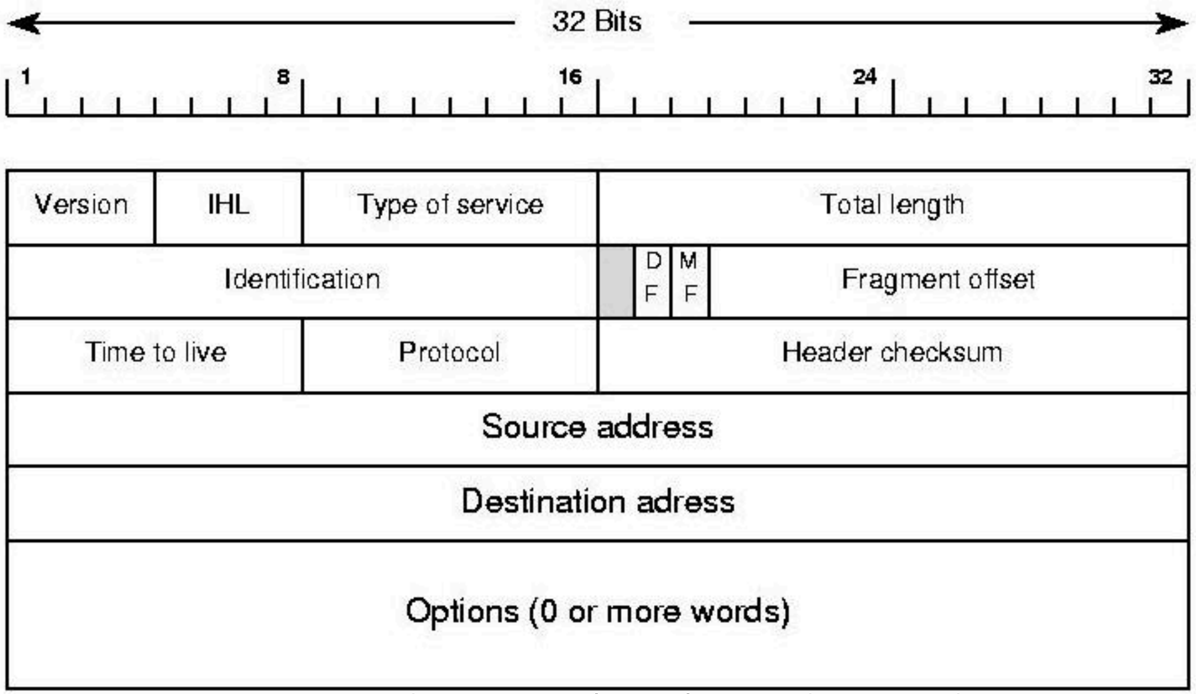


Routers: devices that forward "IP" packets

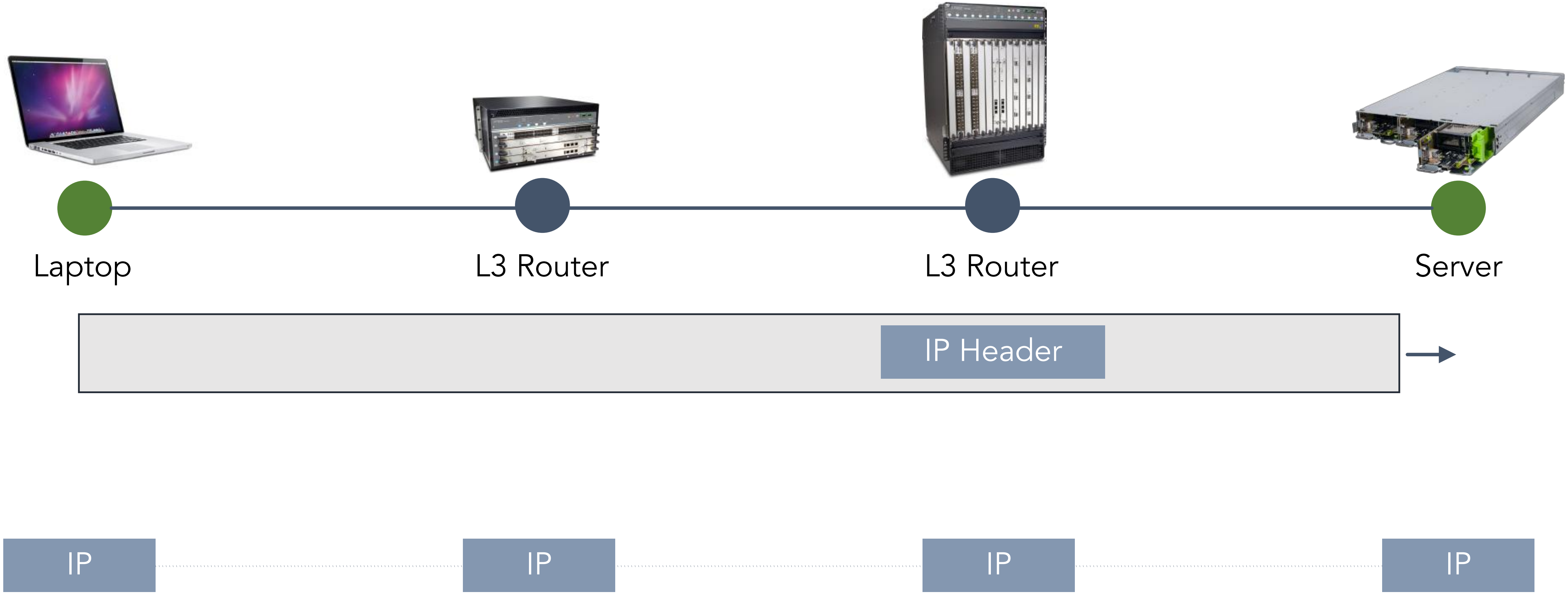


IPv4: unreliable datagram service

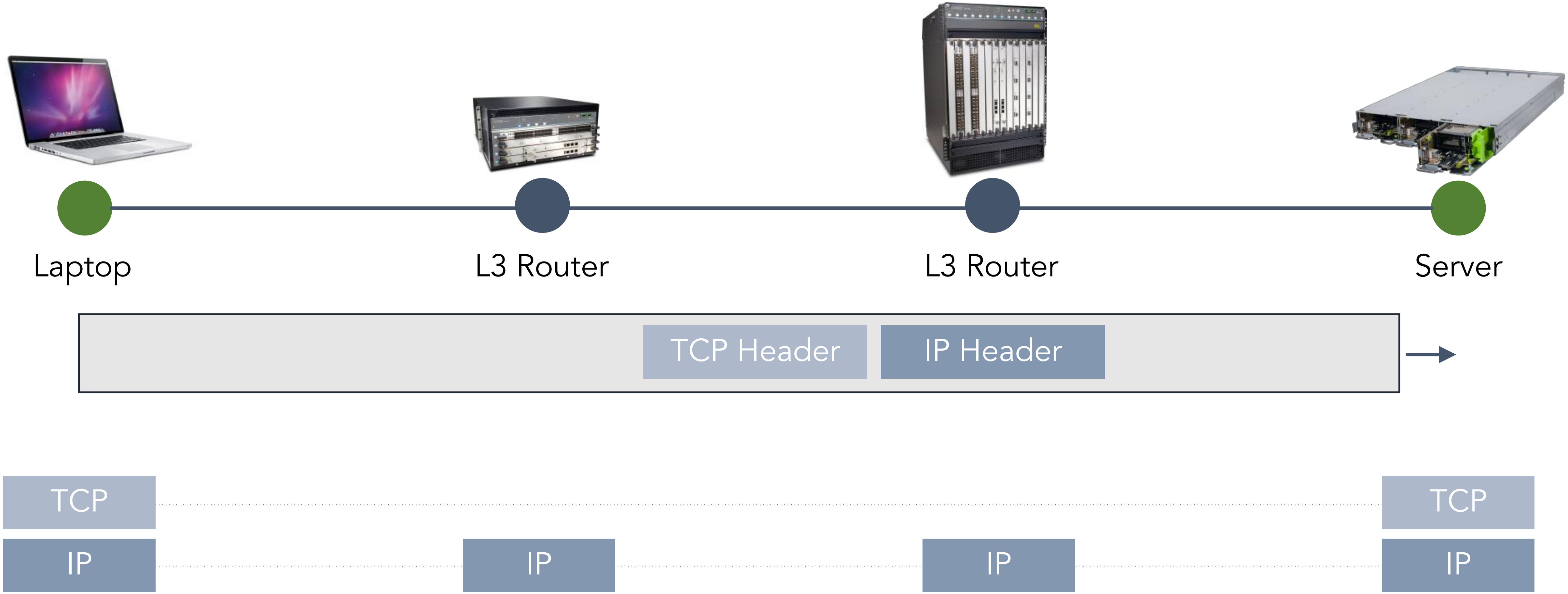
Packet by packet forwarding
Each packet has a destination IP address
Packets may be dropped, delayed, re-ordered, duplicated
End-points use higher-layer protocols (e.g. TCP) to make it reliable



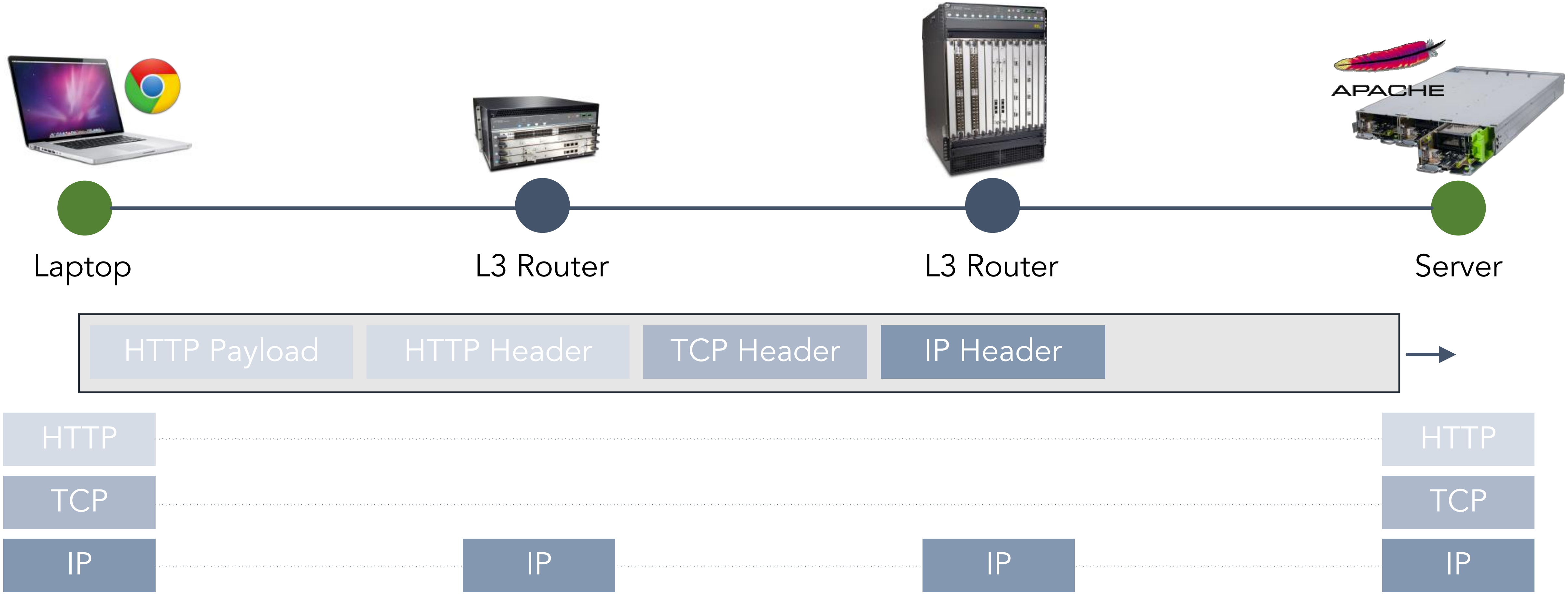
L3 Internet Protocol (IP): datagram service



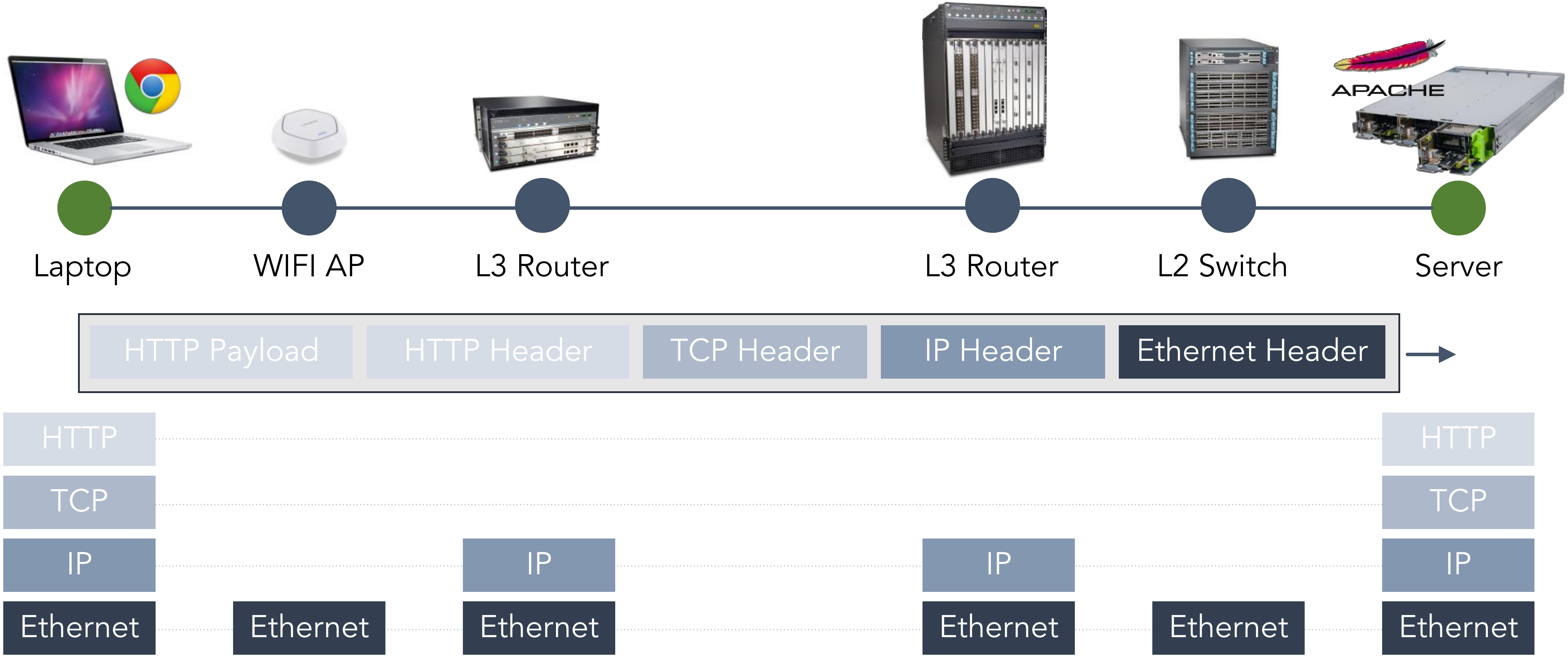
L4 TCP: reliable end-to-end connections



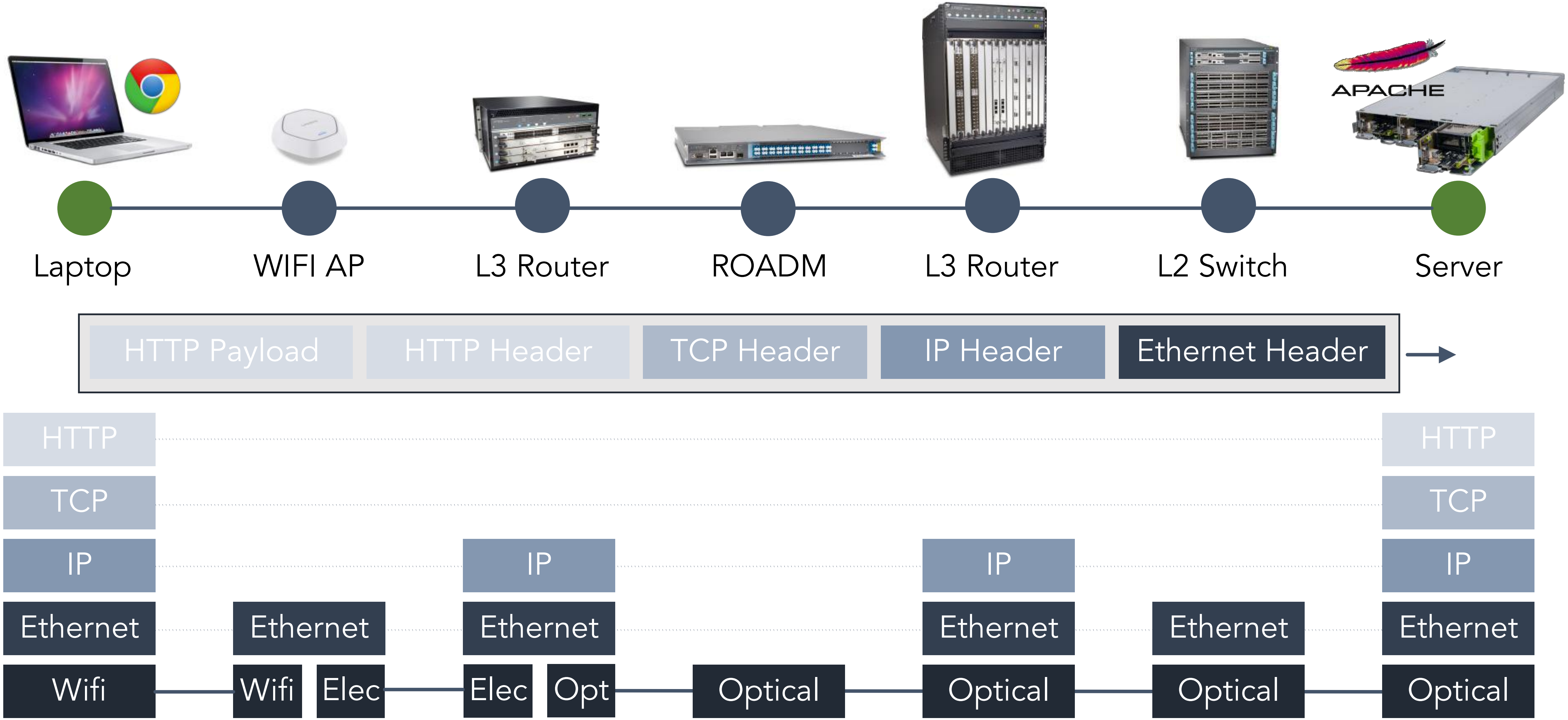
L7 application protocol: e.g. HTTP, FTP, …



L2 Ethernet: Local Area Network (LAN)



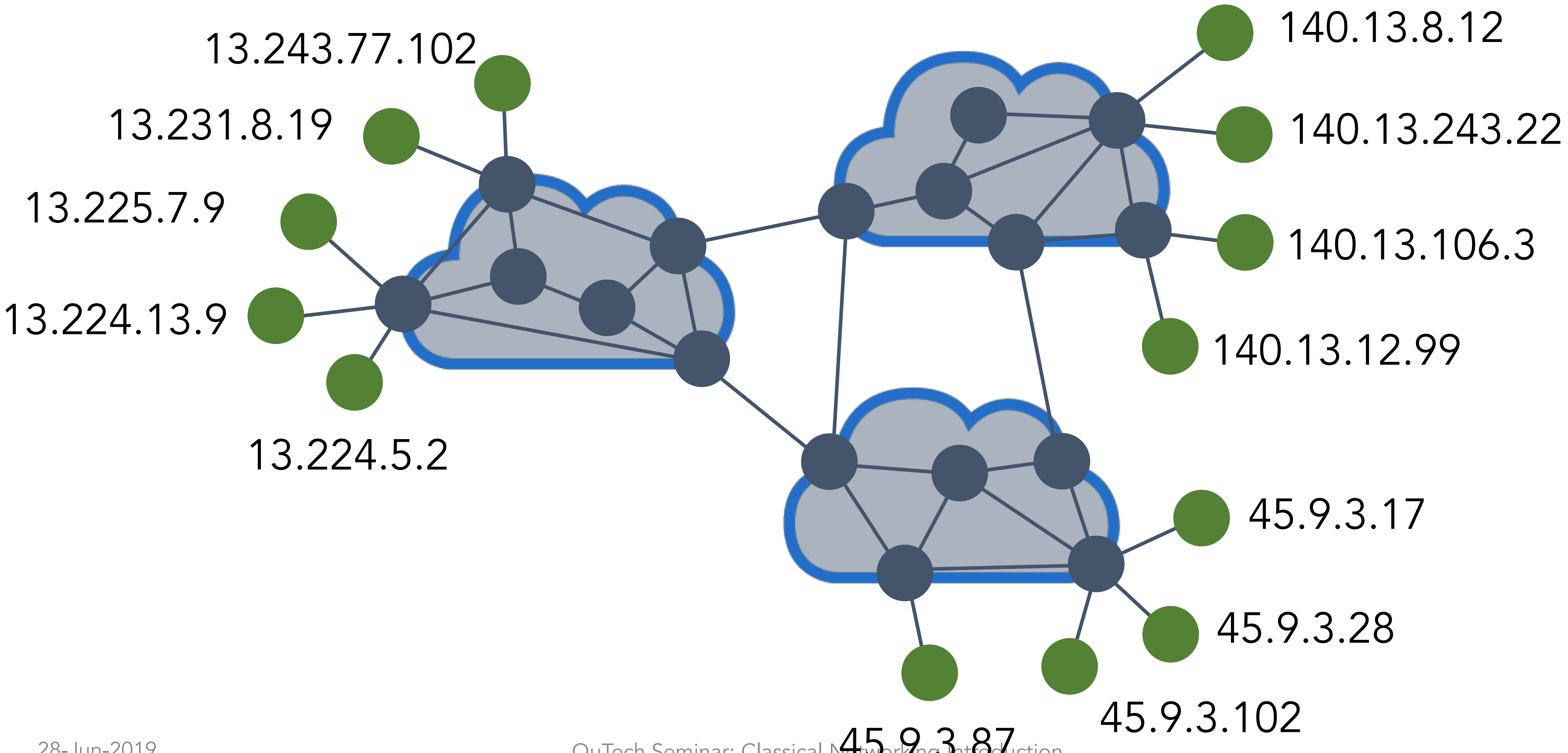
L1 physical layers: WIFI, electrical, optical…



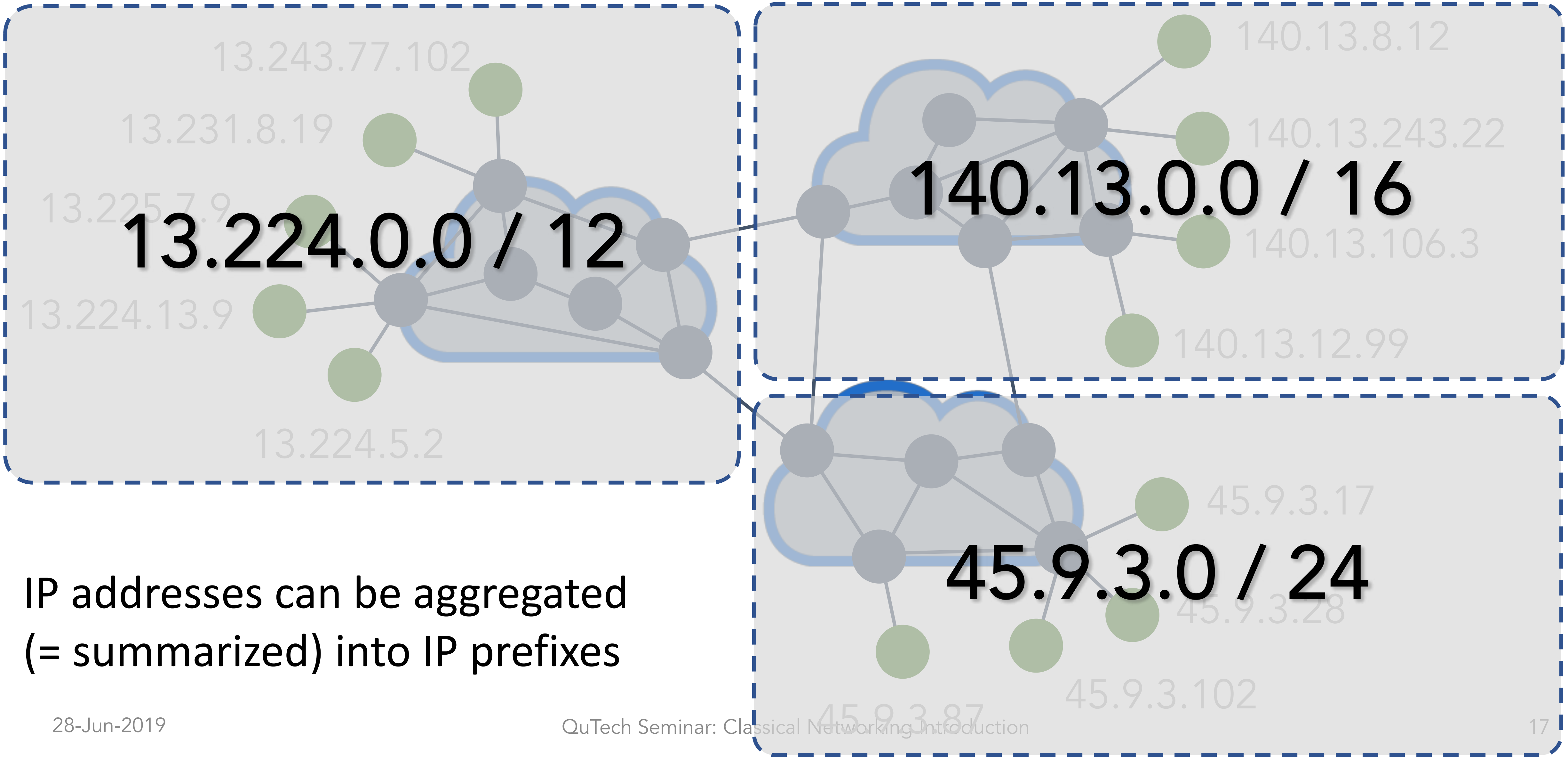
The OSI reference model

Layer number	Layer name	Responsibility	Examples
Layer 7	Application	End-user functionality	HTTP, FTP, SMTP
Layer 6	Presentation	Encoding	ASN.1, GPB, Thrift
Layer 5	Session	Dialogue control, checkpointing	-
Layer 4	Transport	Reliable end-to-end stream, end-to-end flow control, segmentation	TCP, QUIC, UDP
Layer 3	Network	Global addressing, routing, fragmentation, quality of service.	IPv4, IPv6, IPX
Layer 2	Datalink	MAC: Coordinate access to a shared medium. LLC: Group bits into frames, handle bit errors, point-to-point connections and flow-control	IEEE 802 series (Ethernet) PPP
Layer 1	Physical	Send and receive bits over physical medium.	Ethernet 1000BASE-T SONET/SDH

Each end-point has an IP address (IPv4 or IPv6)



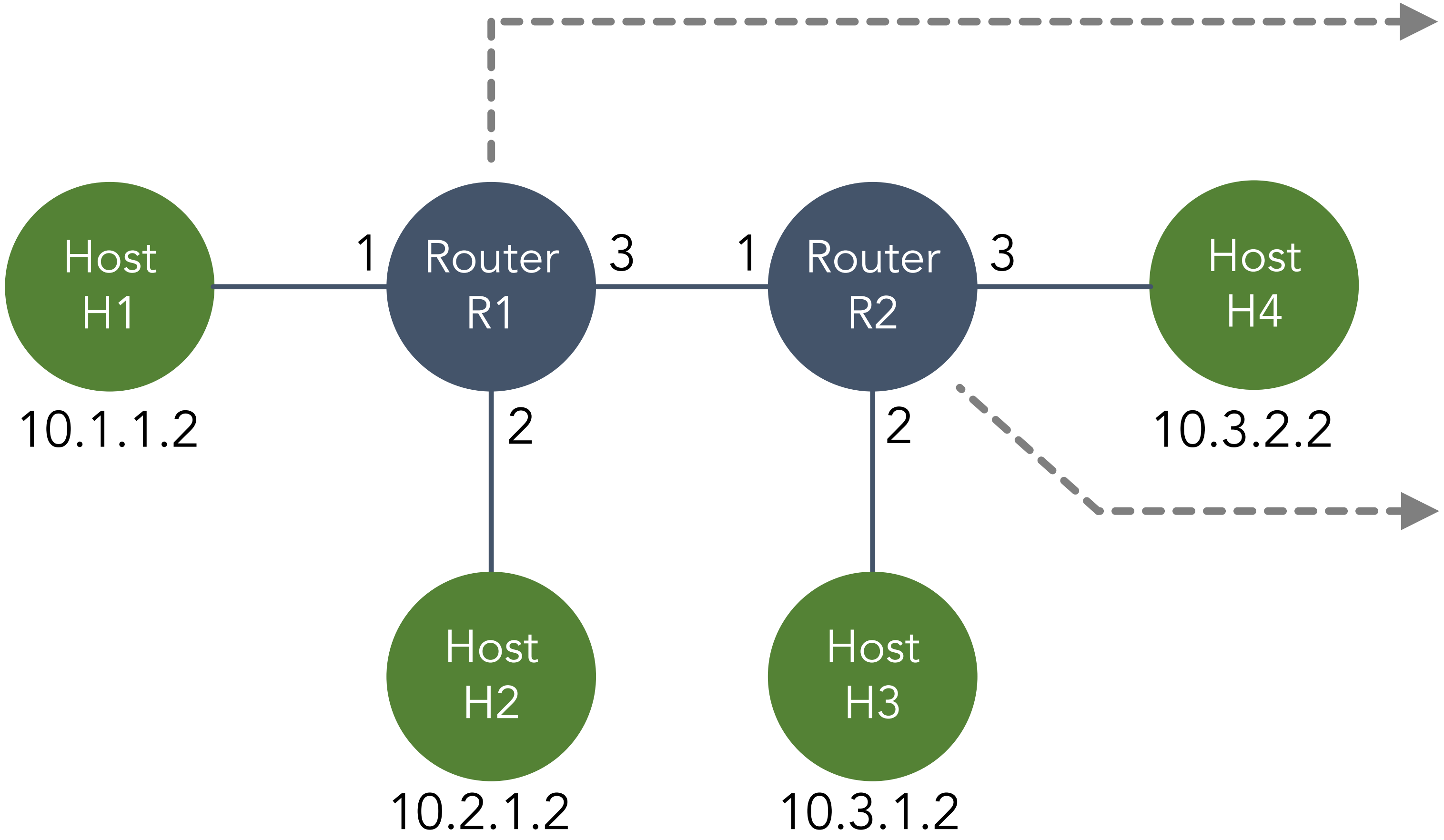
IP addresses have topological significance



IP addresses can be aggregated (= summarized) into IP prefixes

Routers do lookup in forwarding table (FIB)

Looks up the destination IP address for each individual packet



Forwarding table (FIB) of router R1

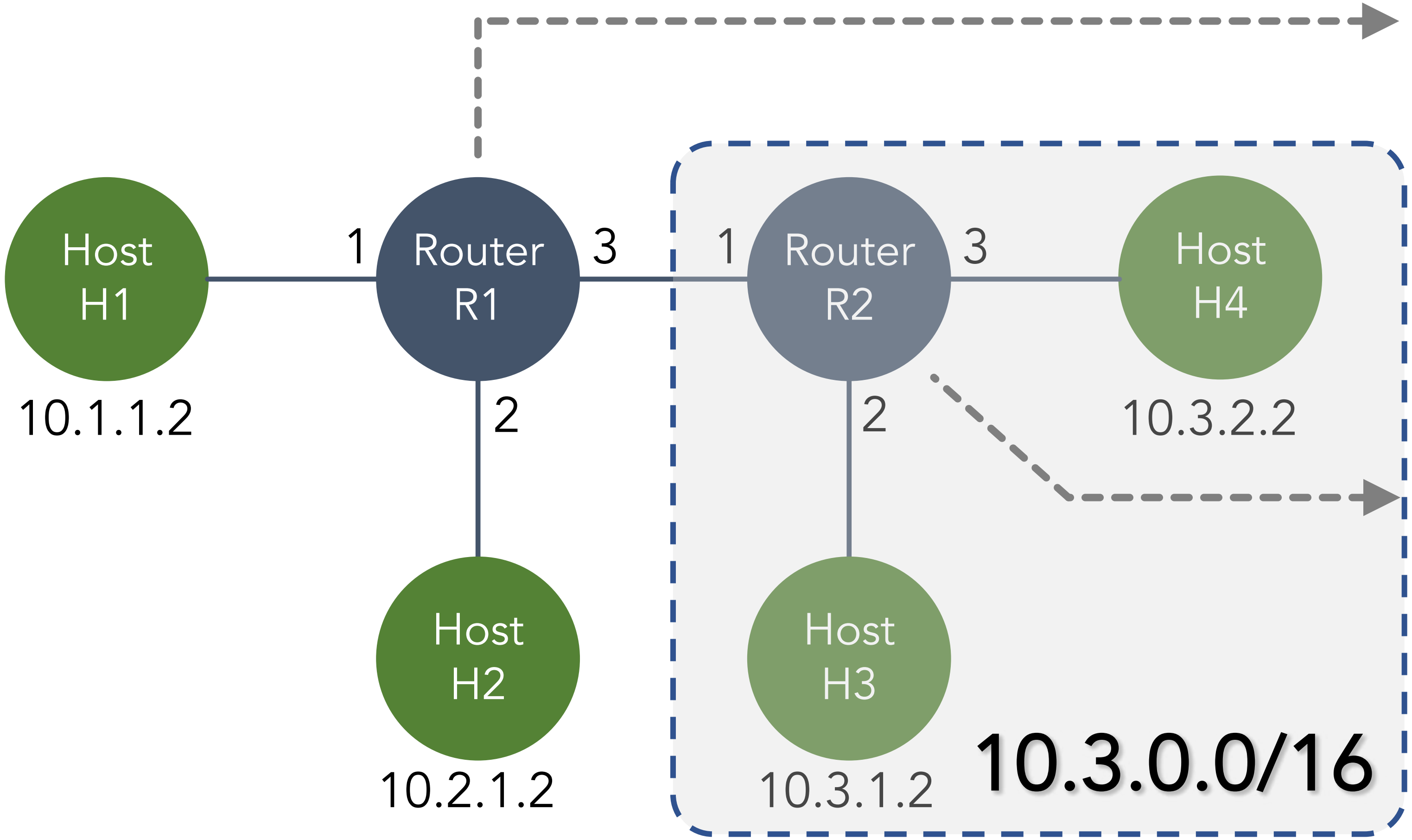
Destination	Next-hop
10.1.1.2 / 32	Interface 1
10.2.1.2 / 32	Interface 2
10.3.1.2 / 32	Interface 3
10.3.2.2 / 32	Interface 3

Forwarding table (FIB) of router R2

Destination	Next-hop
10.1.1.2 / 32	Interface 1
10.2.1.2 / 32	Interface 1
10.3.1.2 / 32	Interface 2
10.3.2.2 / 32	Interface 3

Aggregation and longest prefix match

Use aggregation to reduce forwarding table size
Default route is special case



Forwarding table (FIB) of router R1

Destination	Next-hop
10.1.1.2 / 32	Interface 1
10.2.1.2 / 32	Interface 2
10.3.0.0 / 16	Interface 3

Forwarding table (FIB) of router R2

Destination	Next-hop
0.0.0.0 / 0	Interface 1
10.3.1.2 / 32	Interface 2
10.3.2.2 / 32	Interface 3

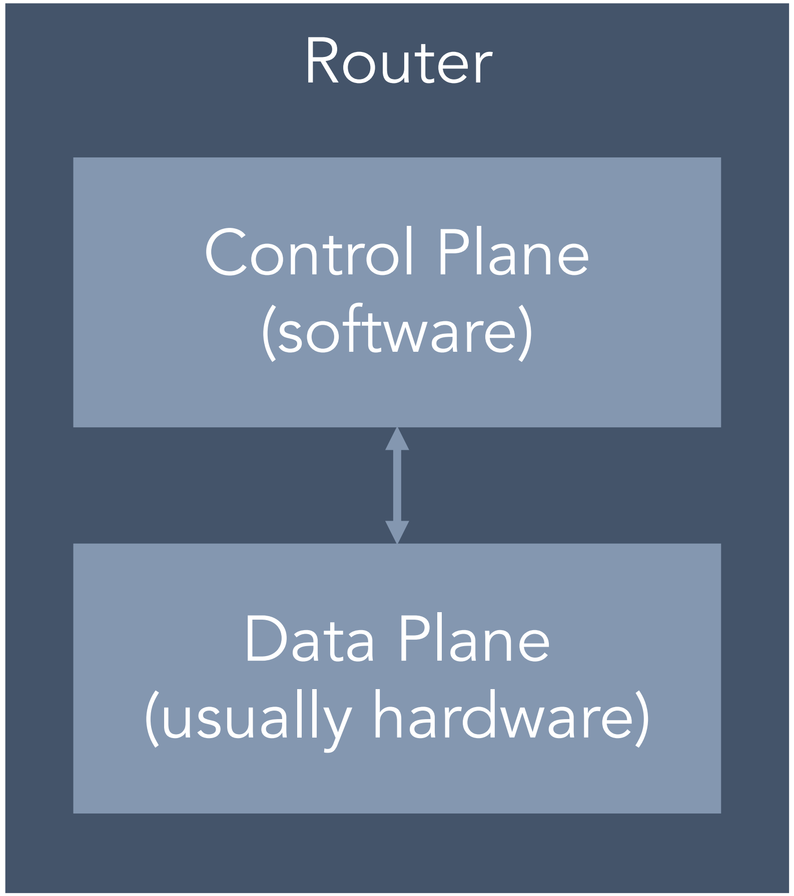
Addressing

- **Datalink: Ethernet address**
  - Just happens to be globally unique
  - No topological significance ("flat address space")
  - Not managed (burned into chip)
- **Network: IPv4 address (32 bits) or IPv6 address (128 bits)**
  - Originally intended to be unique on the Internet
  - Private IP addresses (e.g.10/8), NAT, and anycast complicate matters
  - Topological significance; allows for aggregation; makes the Internet scale
  - Very carefully managed (complex procedures for allocating and assigning IP addresses)
- **Transport: Port numbers**
  - Identify service ("socket") running on a particular server

Naming

- Domain name: human readable name
- Identifies a service
- Example: www.google.com
  (*not* https://www.google.com/somepage)
- The Domain Name System (DNS) protocol translates a name to an IP address
- One DNS name can map to multiple different IP addresses (for load balancing – other mechanisms exist as well)
- The mappings can change over time (for resilience)
- DNS is highly decentralized and scalable

Control plane versus data plane



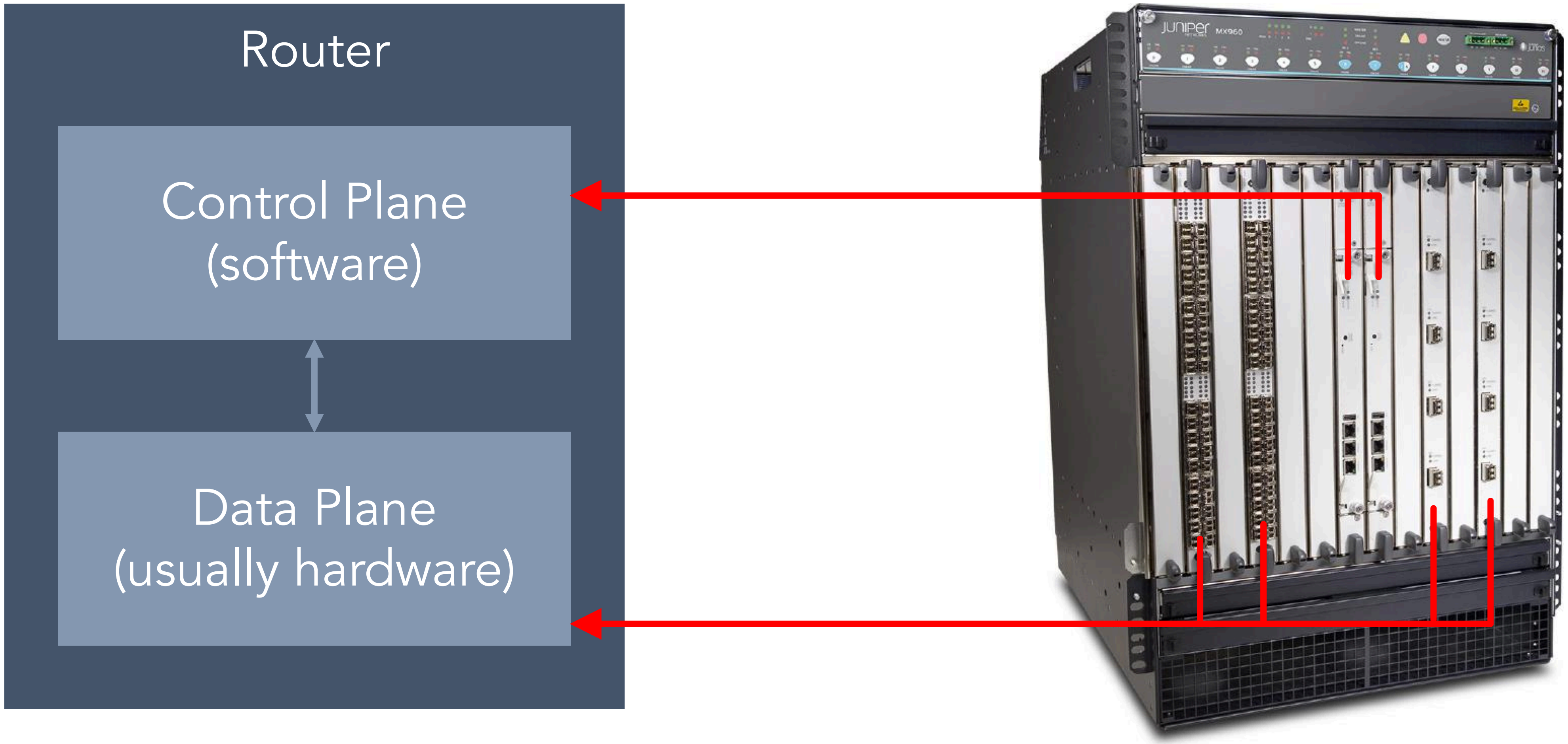
Control plane

- Discover network topology and resources
- Program the forwarding plane (e.g. forwarding table)
- Flexibility is main concern

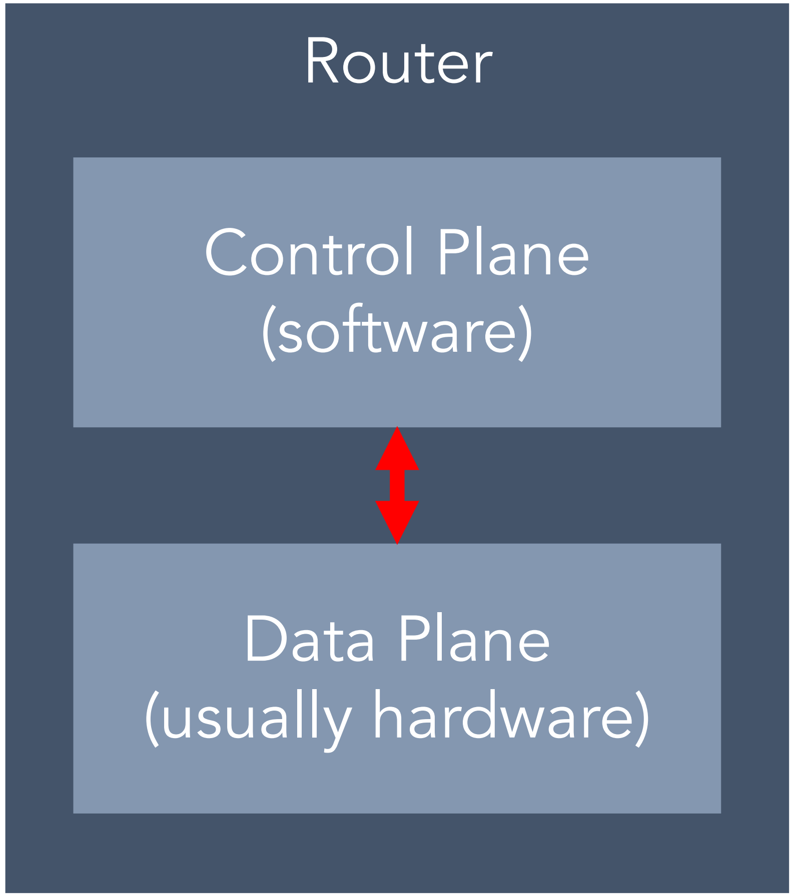
Data plane (aka forwarding plane)

- Forward packets
- Using the forwarding table populated by the control plane
- Speed is main concern

Control plane versus data plane



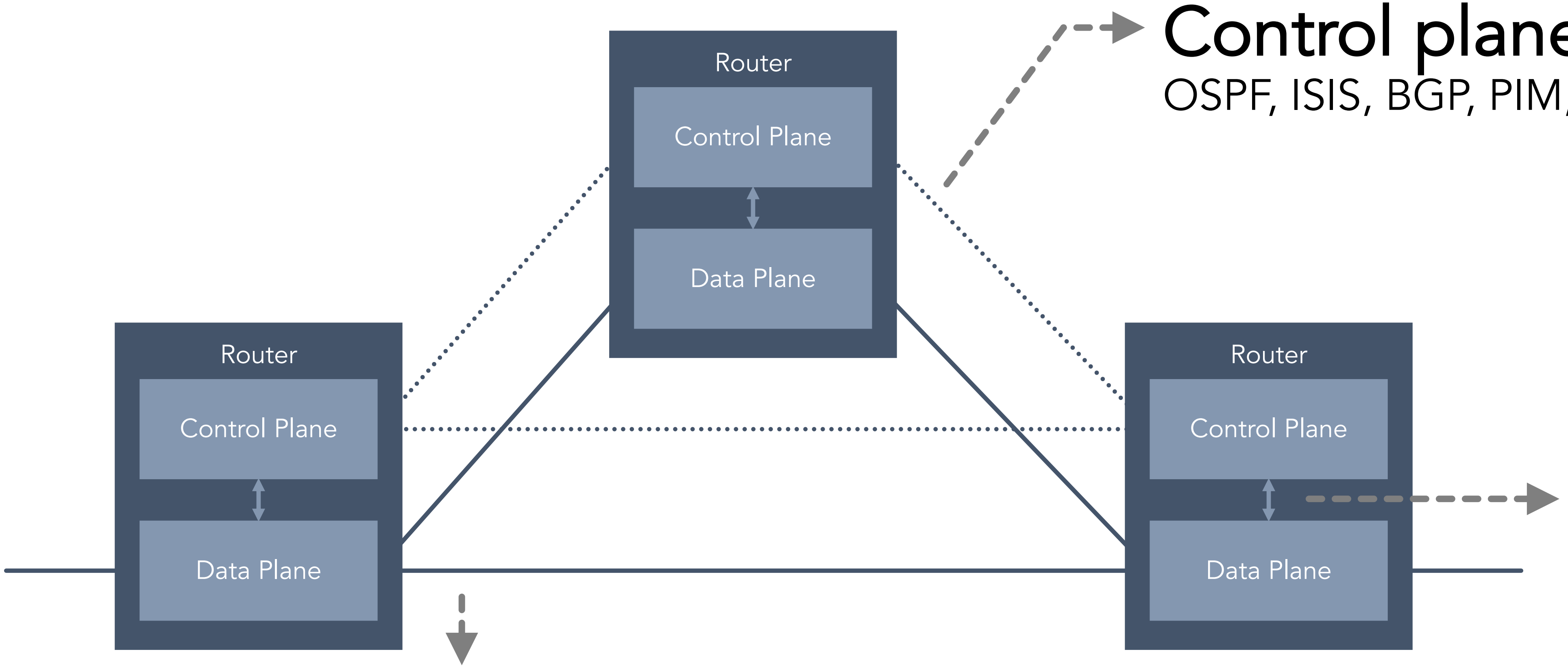
Control plane - data plane interface



Control plane – data plane interface

- Configure networking devices
  (e.g. set interface address)
- Program forwarding plane
  (e.g. add entry to forwarding table)
- Send and receive control plane packets ("punting")
- Retrieve statistics
- Historically proprietary (SDK provided by chip vendor)
- Standards: ForCES, OpenFlow, SAI, Netlink, P4 generated…

"Traditional" distributed control plane



Control plane protocols

OSPF, ISIS, BGP, PIM, RSVP, LDP, …

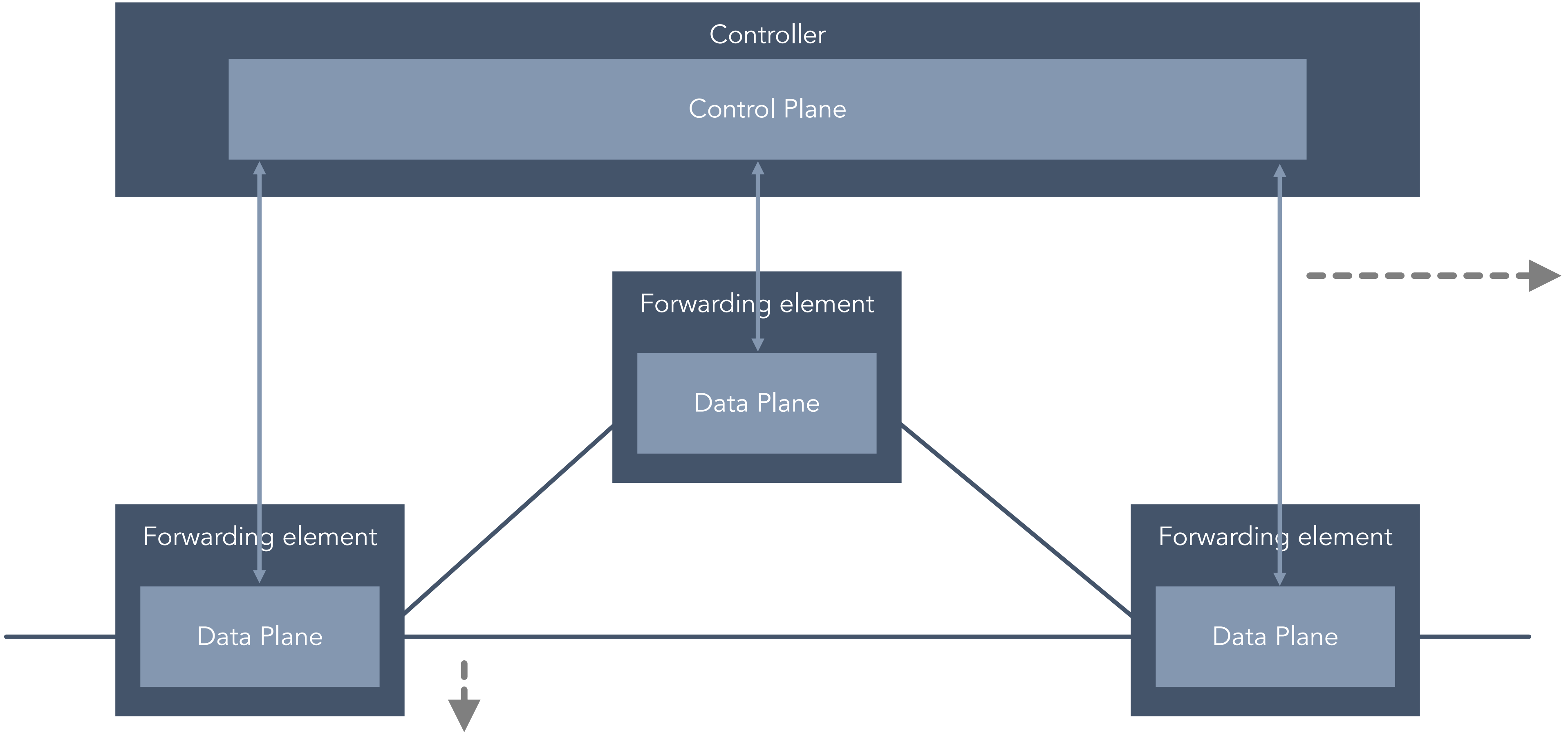
API

proprietary,
ForCES,
OpenFlow,
SAI,
P4 generated…

Data plane protocols

IPv4, IPv6, MPLS, …

SDN: centralized control plane (simplified)



API
proprietary,
ForCES,
OpenFlow,
SAI,
P4 generated…

Data plane protocols
IPv4, IPv6, MPLS, …

Example forwarding plane chip

Broadcom Tomahawk 3

- 12.8 terrabits per second (Tbps)
- 128 x 100G interfaces
- 8 billion packets per second
- 0.04 nano seconds to forward 1 packet
- Relatively small forwarding tables (480K)
- Shallow buffers (internal shared 64MB)
- Lean feature set

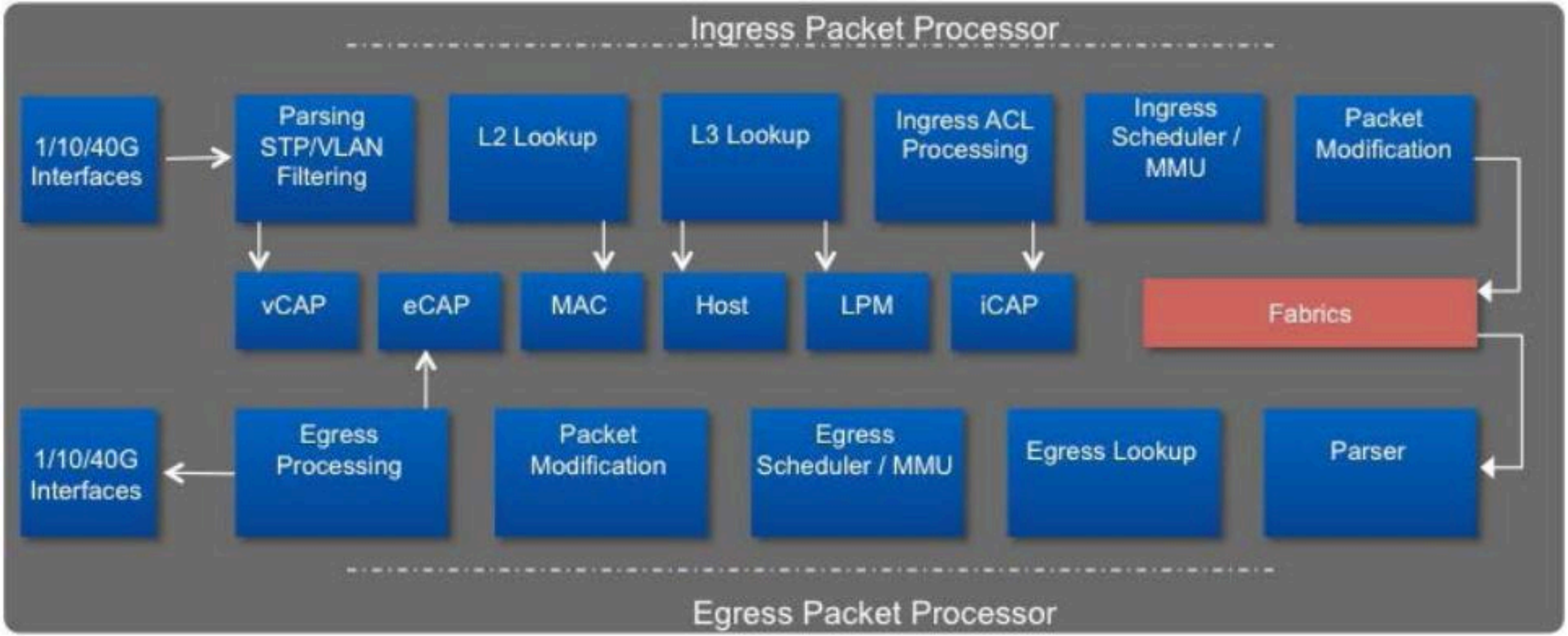
The responsibilities of the forwarding plane

For *each* packet: layer 3 longest prefix match lookup of destination IP address

Actually, a *lot* more than that:

- Layer 2 exact match lookup of destination Ethernet address
- Buffer packets in case of congestions
- Scheduling for Quality of Service (QoS)
- Lookup of source address for validation and learning
- Compute hash for Equal Cost Multipath (ECMP)
- Apply policies (ACLs)
- Track metrics (counters, thresholds, …)
- Make copies for multicast and taps
- … etc. etc. etc.

Example fixed forwarding spipeline



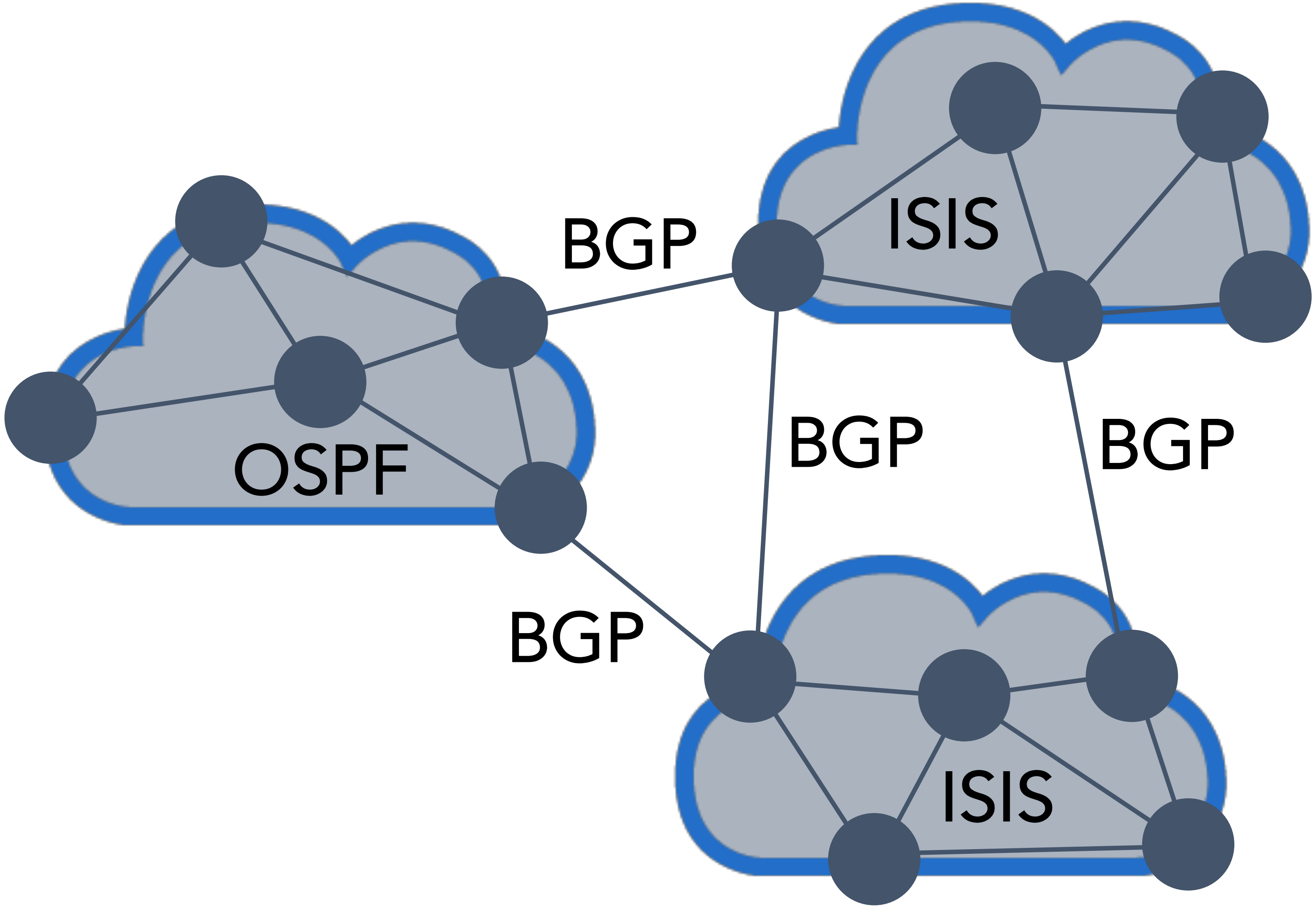
Traditional distributed routing protocols

- Intra-domain routing protocols
  - Exchange routes *inside* autonomous system
  - All routers owned and operated by same organization
  - Higher level of trust, more information available
  - Emphasis on shortest path
  - Example: OSPF, ISIS, EIGRP, RIP, RIFT, BGP (!), ….
- Inter-domain routing protocols:
  - Exchange routes *between* autonomous systems
  - Routers owners by different organizations
  - Lower level of trust, less information available, real money is involved
  - Emphasis on implementing business policies (customers, transit, peering, …)
  - Only one protocol: BGP version 4

Routing protocol algorithms

- **Link State Routing**
  - Every router discovers full topology and computes shortest path
  - The dominant algorithm for intra–domain routing
  - Open Shortest Path First (OSPF)
  - Intermediate-System to Intermediate-System (ISIS)
- **Vector Routing**
  - Distance-Vector or Path-Vector
  - Routers do not know full topology
  - Each router locally choses best path and propagates that path
  - Routing Information Protocol (RIP)
  - Border Gateway Protocol (BGP)

Intra-domain vs inter-domain (simplified)



Other control plane protocols

- Traffic engineering protocols:
  - Used to
  - RSVP, LDP, BGP-LU, ISIS-TE, OSPF-TE, Segment Routing
- Multicast protocols:
  - IGMP, MLD, PIM (several modes)
- Service protocols:
  - L3VPN, L2VPN, EVPN, …
- Management protocols:
  - SSH, SNMP, NETCONF, FTP, …
- Telemetry protocols
  - NetFlow, GPB, …

Next seminars

- Deeper dive into routing protocols
  - Open Shortest Path First (OSPF)
  - Intermediate System to Intermediate System (ISIS)
  - Border Gateway Protocol (BGP)
- Traffic engineering
  - Multi-Protocol Label Switching (MPLS)
  - Resource Reservation Protocol (RSVP)
  - Path Computation Element Protocol (PCEP)
- Software Defined Networking (SDN)
  - SDN motivations and applications
  - OpenFlow
  - Programming Protocol-independent Packet Processors (P4)