COMP2001J Computer Networks

Lecture 10 – Network Layer
(DHCP, IP Forwarding, SDN)

Dr. Shen WANG (王燊)

shen.wang@ucd.ie

Outline

- DHCP
  - Dynamic Host Configuration Protocol
- IP Forwarding
- SDN
  - Software-Defined Networking

DHCP – Motivation

- ARP (as well as other Internet protocols) makes the assumption that hosts are configured with some basic information, such as their own IP addresses.
- How do hosts get this information?
- It is possible to manually configure each computer, but that is tedious and error-prone.
- There is a better way, and it is called **DHCP** (**Dynamic Host Configuration Protocol**).

DHCP – Initialization

- With DHCP, every network must have a **DHCP server** that is responsible for configuration.

- When a computer is started, it has a built-in Ethernet or other link layer address embedded in the NIC, but:

- This computer doesn’t have an IP address yet.
- This computer doesn’t know who to ask for an IP address.

DHCP- Discover

- Much like ARP, the computer **broadcasts** a request for an IP address on its network.
- It does this by using a **DHCP DISCOVER packet**.
- This packet must reach the DHCP server.

- If that server is not directly attached to the network, the router will be configured to receive DHCP broadcasts and relay them to the DHCP server, wherever it is located.

DHCP – Offer

- When the server receives the request, it allocates a free IP address and sends it to the host in a **DHCP OFFER packet** (which again may be relayed via the router).

- To be able to do this work even when hosts do not have IP addresses, the server identifies a host using its Ethernet address (which is carried in the DHCP DISCOVER packet)

DHCP – Leasing

- How long an IP address should be allocated?
  - If a host leaves the network and does not return its IP address to the DHCP server, that address will be permanently lost.
  - After a period of time, many addresses may be lost.

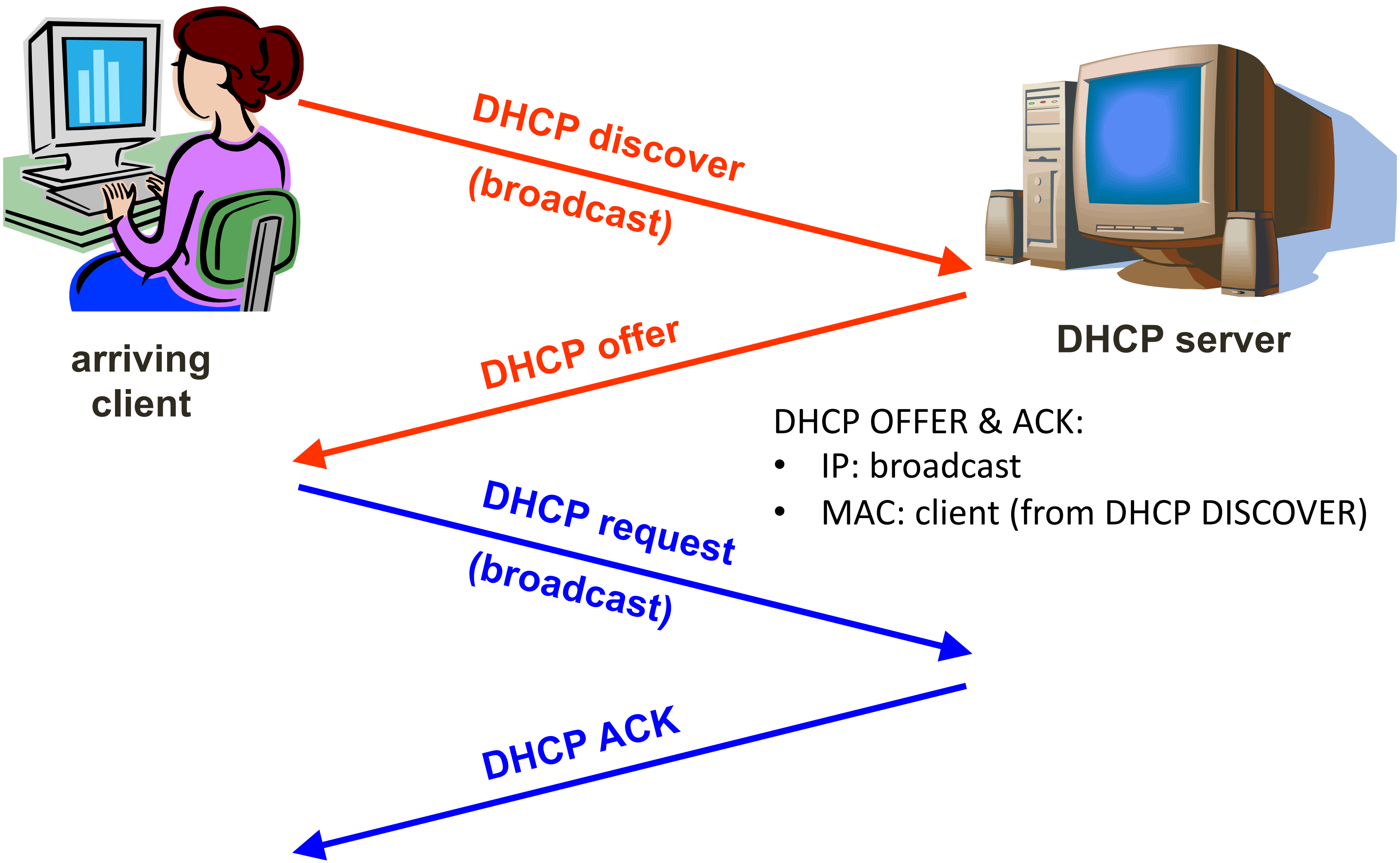
- To prevent that from happening, IP address assignment may be for a fixed period of time, a technique called **leasing**.
  - Just before the lease expires, the host must ask for a DHCP renewal.
  - If it fails to make a request or the request is denied, the host may no longer use the IP address it was given earlier.

DHCP – Multiple servers

- Multiple servers may respond
  - Multiple servers on the same broadcast media
  - Each may respond with an offer
  - The client can decide which offer to accept

- Accepting one of the offers
  - Client sends a **DHCP request** echoing the parameters
  - The DHCP server responds with an **ACK** to confirm
  - … and the other servers see they were not chosen

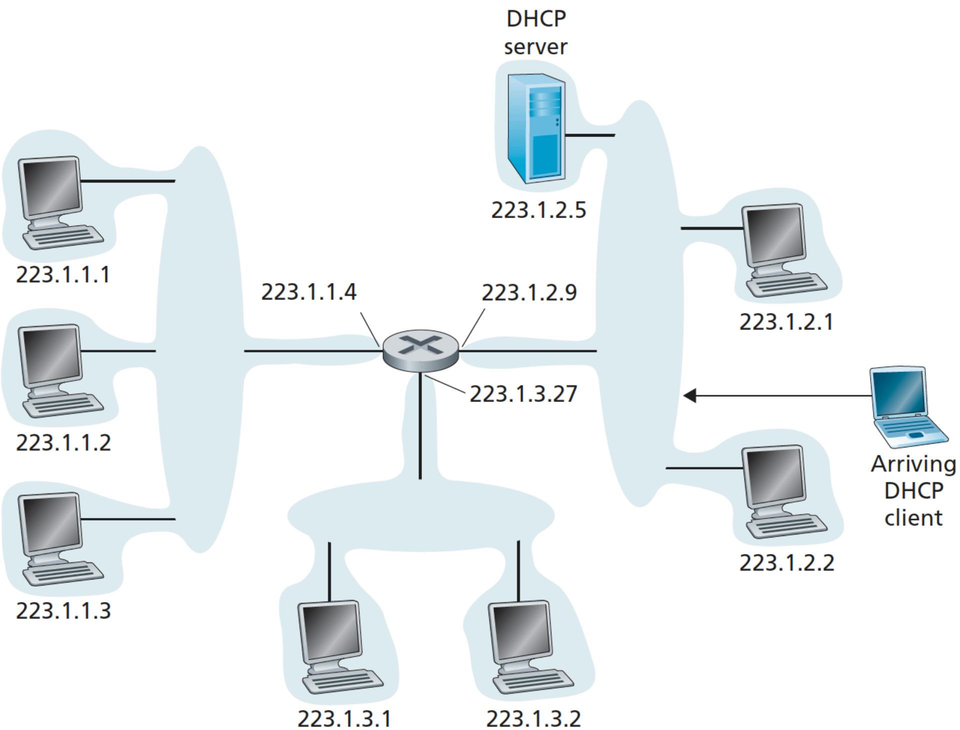
Dynamic Host Configuration Protocol

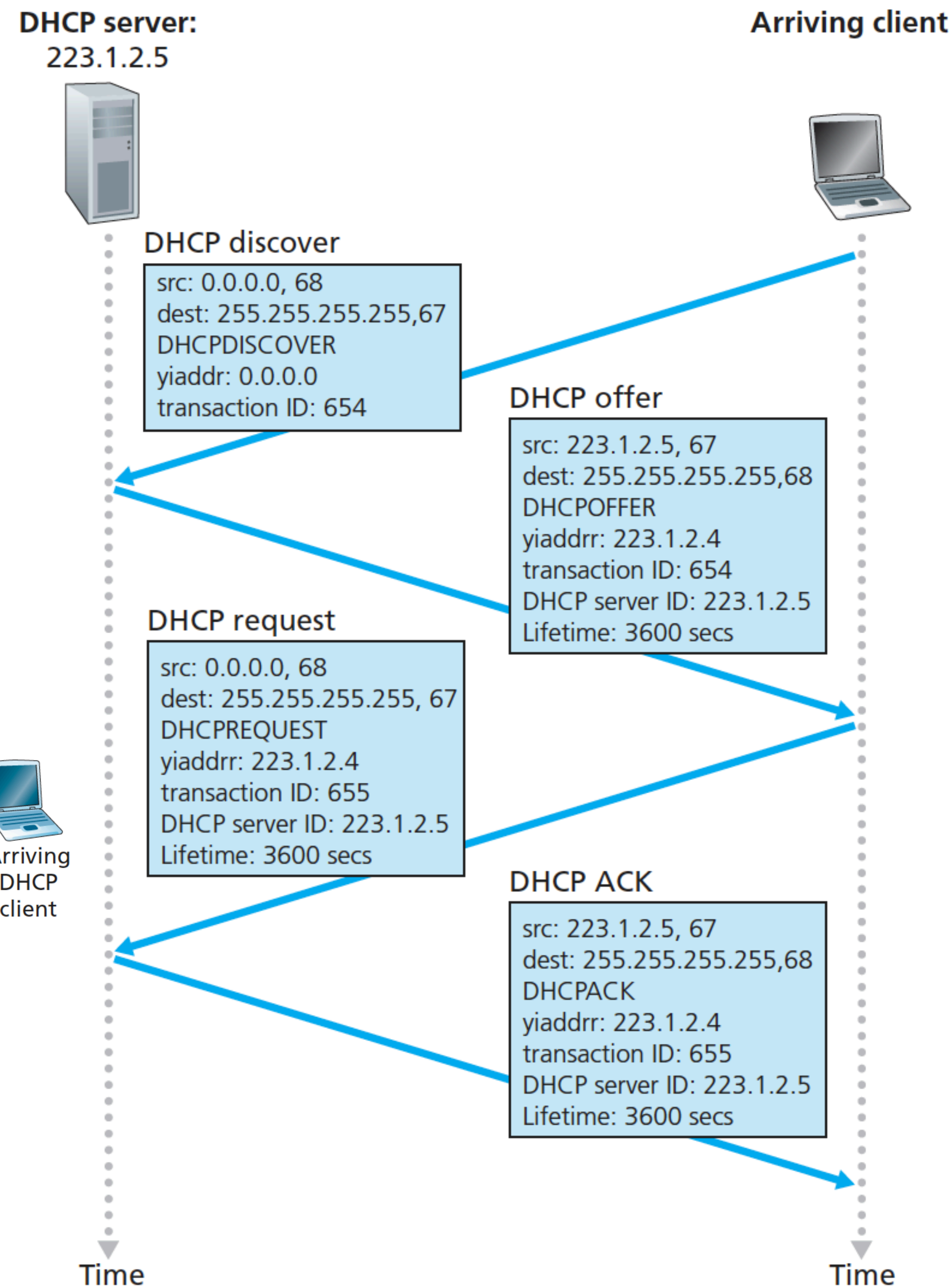


DHCP OFFER & ACK:

- IP: broadcast
- MAC: client (from DHCP DISCOVER)

DHCP: Example



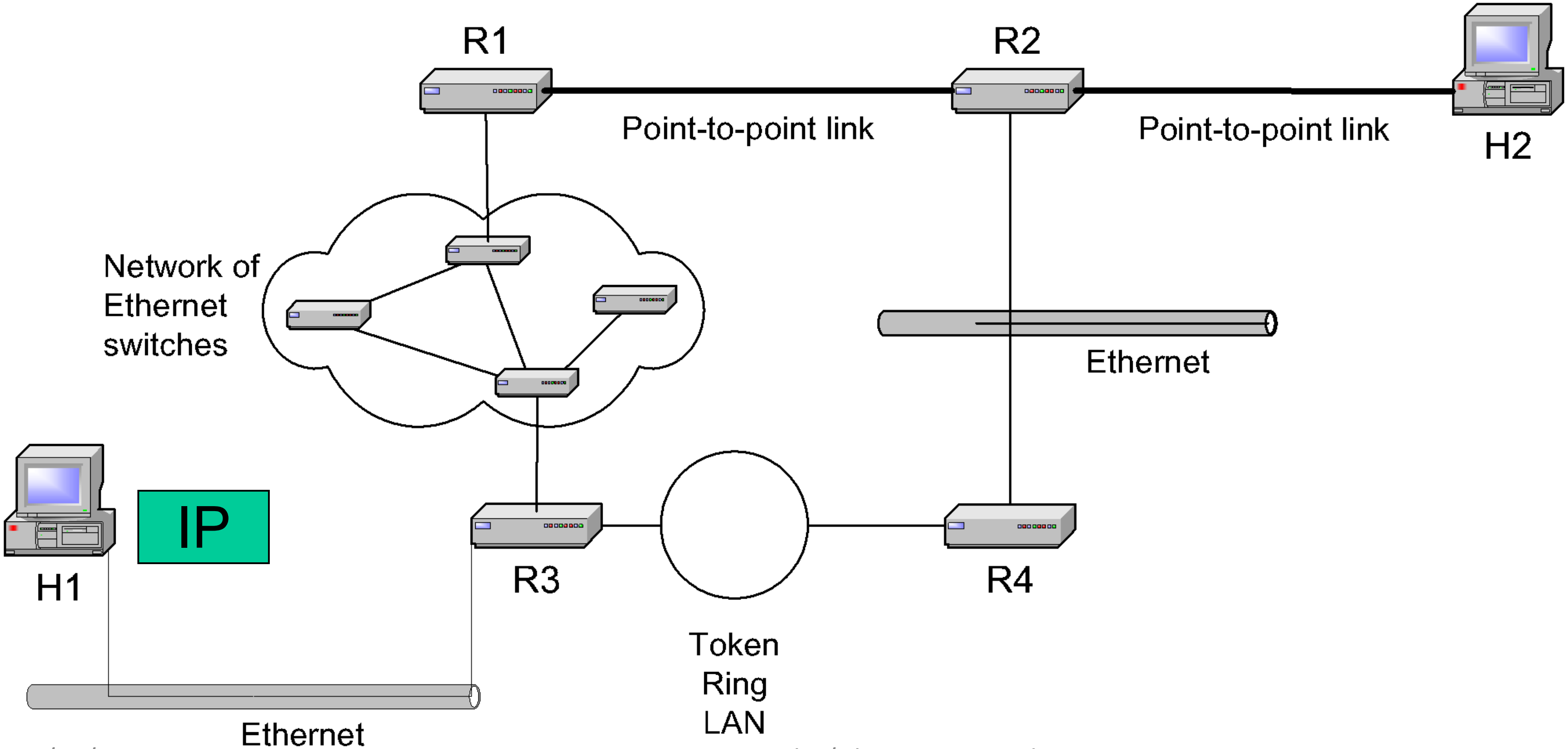


DHCP – Applications

- DHCP is widely used in the Internet to configure all sorts of parameters in addition to providing hosts with IP addresses.
- As well as in business and home networks, DHCP is used by ISPs to set the parameters of devices over the Internet access link, so that customers do not need to phone their ISPs to get this information.
- Common examples of the information that is configured include the network mask, the IP address of the default gateway, and the IP addresses of DNS and time servers.

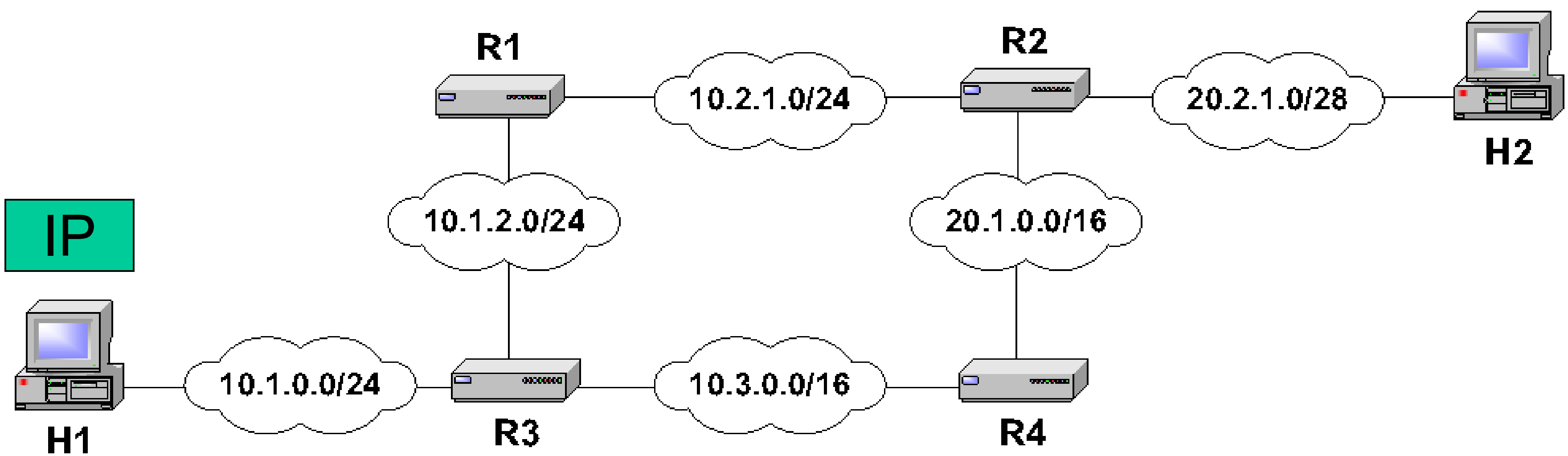
Delivery of an IP datagram

- View at the data link layer:
  - Internetwork is a collection of LANs or point-to-point links or switched networks that are connected by routers



Delivery of an IP datagram

- View at the IP layer:
  - An IP network is a logical entity with a network number
  - We represent an IP network as a “cloud”
  - The IP delivery service takes the view of clouds, and ignores the data link layer view



Tenets of end-to-end delivery of datagrams

The following conditions must hold so that an IP datagram can be successfully delivered

1. The network prefix of an IP destination address must correspond to a unique data link layer network (=LAN or point-to-point link or switched network).
2. Routers and hosts that have a common network prefix must be able to exchange IP datagrams using a data link protocol (e.g., Ethernet, PPP)
3. Every data link layer network must be connected to at least one other data link layer network via a router.

Routing tables

- Each router and each host keeps a **routing table** which tells the router how to process an outgoing packet
- Main columns:
  1. **Destination address:** where is the IP datagram going to?
  2. **Next hop:** how to send the IP datagram?
  3. **Interface:** what is the output port?
- Next hop and interface column can often be summarized as one column
- Routing tables are set so that datagrams gets closer to the its destination

Routing table of a host or router

IP datagrams can be directly delivered ("direct") or is sent to a router ("R4")

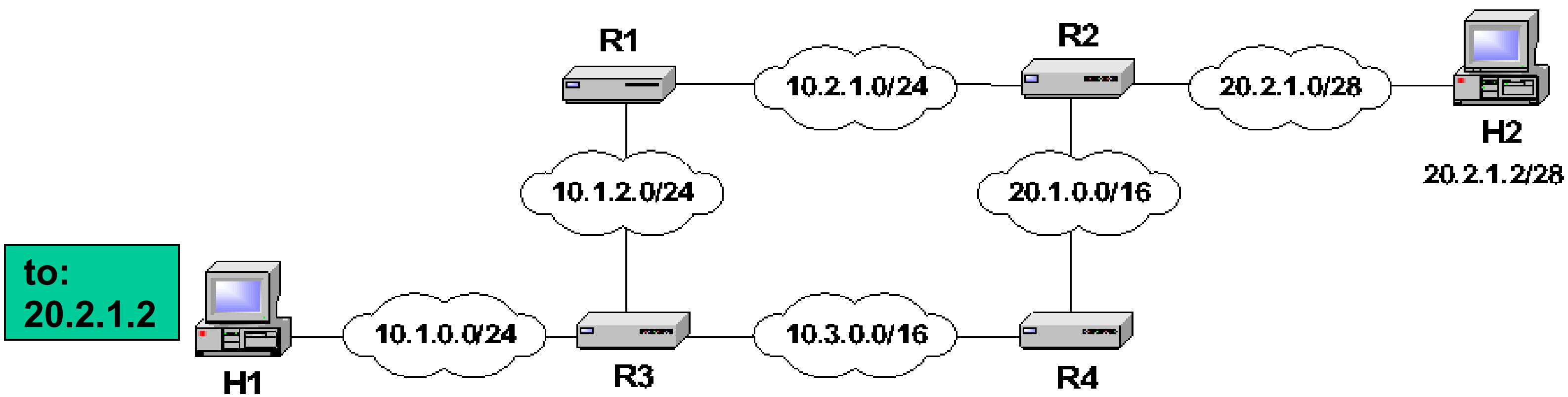
Destination	Next Hop	Interface
10.1.0.0/24	direct	eth0
10.1.2.0/24	direct	eth0
10.2.1.0/24	R4	serial0
10.3.1.0/24	direct	eth1
20.1.0.0/16	R4	eth0
20.2.1.0/28	R4	eth0

Delivery with routing tables

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.2.0.0/16	R2
30.1.1.0/28	R2

Destination	Next Hop
10.1.0.0/24	R1
10.1.2.0/24	R1
10.2.1.0/24	direct
10.3.1.0/24	R4
20.1.0.0/16	direct
20.2.1.0/28	direct

Destination	Next Hop
10.1.0.0/24	R2
10.1.2.0/24	R2
10.2.1.0/24	R2
10.3.1.0/24	R2
20.1.0.0/16	R2
20.2.1.0/28	direct



Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	R3
10.2.1.0/24	R3
10.3.1.0/24	R3
20.1.0.0/16	R3
20.2.1.0/28	R3

Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	direct
10.2.1.0/24	R4
10.3.1.0/24	direct
20.1.0.0/16	R4
20.2.1.0/28	R4

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	R3
10.2.1.0/24	R2
10.3.1.0/24	direct
20.1.0.0/16	direct
20.2.1.0/28	R2

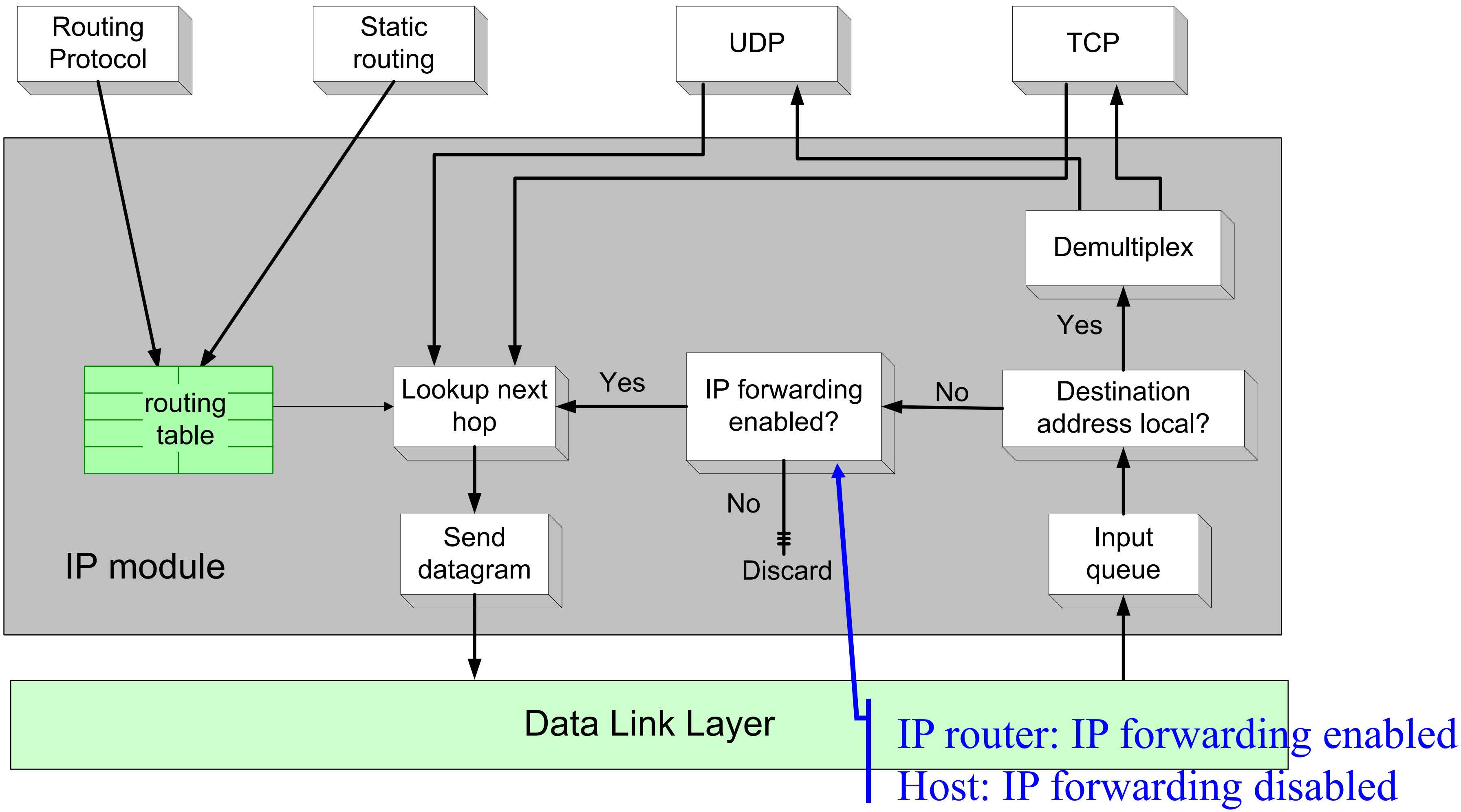
Delivery of IP datagrams

- There are two distinct processes to deliver IP datagrams:

  1. **Forwarding:** How to pass a packet from an input interface to the output interface?
  2. **Routing:** How to find and setup the routing tables?

- Forwarding must be done as fast as possible:
  - on routers, is often done with support of hardware
  - on PCs, is done in kernel of the operating system
- Routing is less time-critical
  - On a PC, routing is done as a background process

Processing an IP datagram



Processing an IP datagram

- Processing IP datagrams on IP router is very similar with the one on a host
- **Main difference:**
  "IP forwarding" is enabled on router and disabled on host

- **IP forwarding enabled**
  → if a datagram is received, but it is not for the local system, the datagram will be sent to a different system
- **IP forwarding disabled**
  → if a datagram is received, but it is not for the local system, the datagram will be dropped

Processing of an IP datagram at a router

Receive an IP datagram →

1. IP header validation
2. Process options in IP header
3. Parsing the destination IP address
4. Routing table lookup
5. Decrement TTL
6. Perform fragmentation (if necessary)
7. Calculate checksum
8. Transmit to next hop

Routing table lookup

- When a router or host need to transmit an IP datagram, it performs a routing table lookup
- **Routing table lookup:** Use the IP destination address as a key to search the routing table.
- Result of the lookup is the IP address of a next hop router, and/or the name of a network interface

Destination address	**Next hop/ interface**
network prefix *or* host IP address *or* loopback address *or* default route	IP address of next hop router *or* Name of a network interface

Type of routing table entries

- **Network route**
  - Destination addresses is a network address (e.g., 10.0.2.0/24)
  - Most entries are network routes
- **Host route**
  - Destination address is an interface address (e.g., 10.0.1.2/32)
  - Used to specify a separate route for certain hosts
- **Default route**
  - Used when no network or host route matches
  - The router that is listed as the next hop of the default route is the **default gateway (for Cisco: "gateway of last resort")**
- **Loopback address**
  - Routing table for the loopback address (127.0.0.1)
  - The next hop lists the loopback (lo0) interface as outgoing interface

Routing table lookup: Longest Prefix Match

- **Longest Prefix Match:** Search for the routing table entry that has the longest match with the prefix of the destination IP address

1. Search for a match on all 32 bits
2. Search for a match for 31 bits

   .....

32. Search for a match on 0 bits

Host route, loopback entry
→ 32-bit prefix match

Default route is represented as 0.0.0.0/0
→ 0-bit prefix match

128.143.71.21

↓

Destination address	Next hop
10.0.0.0/8	R1
128.143.0.0/16	R2
128.143.64.0/20	R3
128.143.192.0/20	R3
128.143.71.0/24	R4
128.143.71.55/32	R3
default	R5

↓

The longest prefix match for 128.143.71.21 is for 24 bits with entry 128.143.71.0/24
Datagram will be sent to R4

Route Aggregation

- Longest prefix match algorithm permits to aggregate prefixes with **identical next hop address** to a single entry
- This contributes significantly to reducing the size of routing tables of Internet routers

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
192.168.0.0/24	R2
192.168.1.0/24	R2
……	…
192.168.255.0/24	R2

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
192.168.0.0/16	R2

How do routing tables get updated?

- Adding an interface:
  - Configuring an interface eth2 with 10.0.2.3/24 adds a routing table entry:

Destination	Next Hop/ interface
10.0.2.0/24	eth2

- Adding a default gateway:
  - Configuring 10.0.2.1 as the default gateway adds the entry:

Destination	Next Hop/ interface
0.0.0.0/0	10.0.2.1

- Static configuration of network routes or host routes
- Update of routing tables through routing protocols

- ICMP messages – testing links

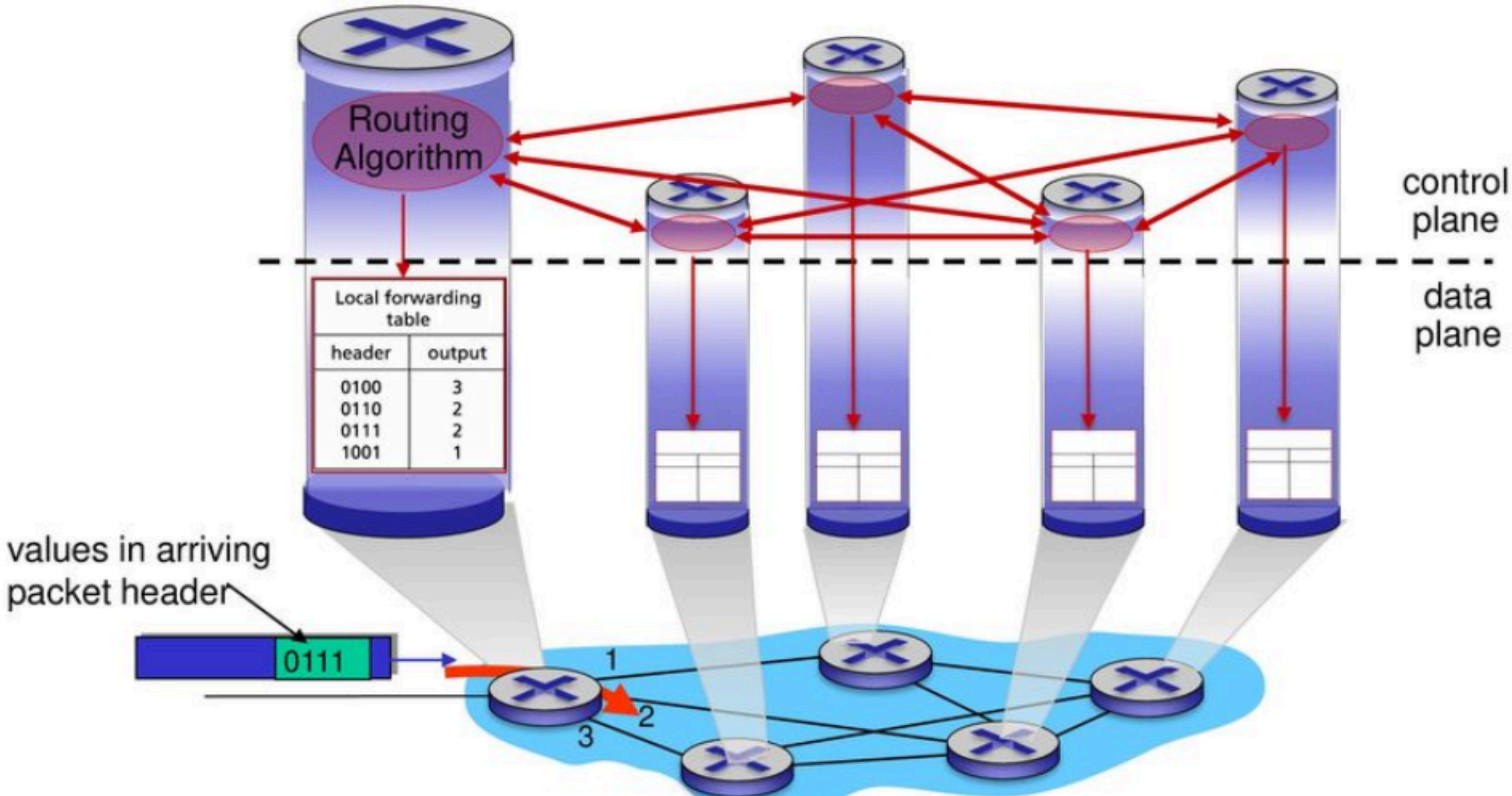
What is SDN?

- Software-defined networking (SDN) technology is an approach to cloud computing that
  - facilitates network management;
  - enables programmatically efficient network configuration;
  - in order to improve network performance and monitoring.

- SDN attempts to centralize network intelligence in one network component by *disassociating* the *forwarding process (data plane)* of network packets from the *routing process (control plane)*.

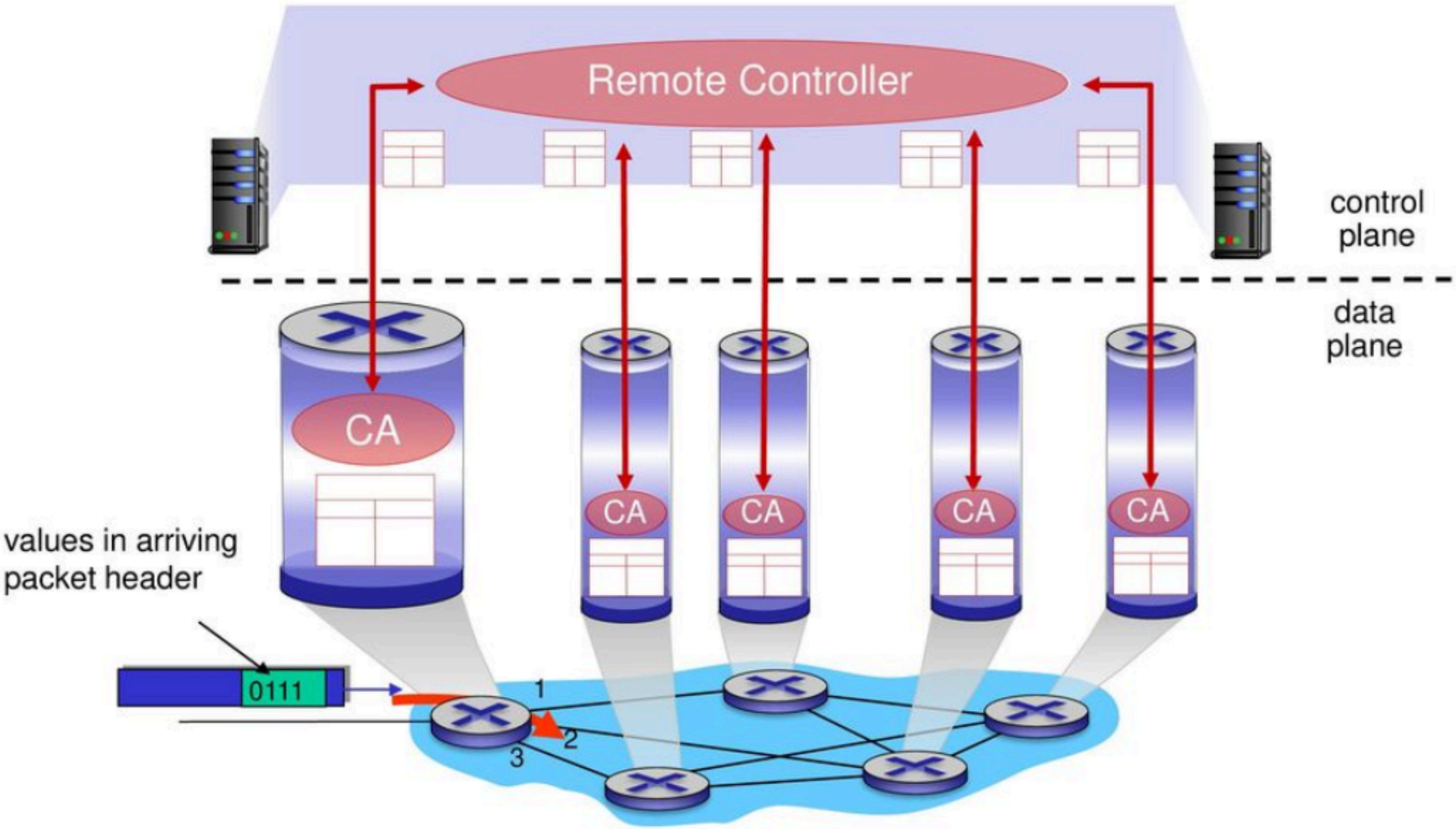
Per-router control plane

- Individual routing algorithm components ***in each and every router*** interact in the control plane



“Logically centralized” control plane

- A distinct (typically remote) controller interacts with local control agents (CAs).

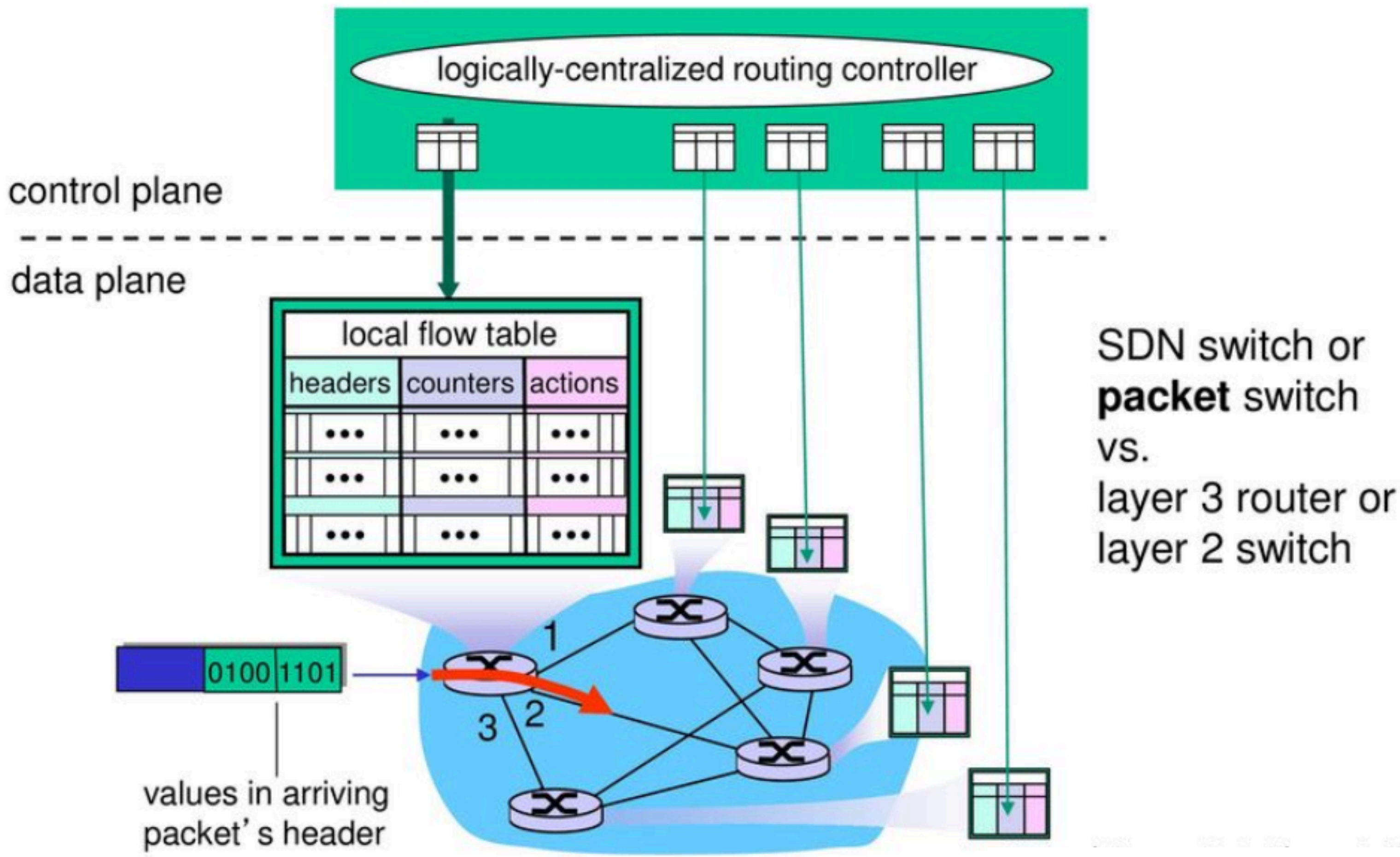


Motivation

- Proliferation of “middleboxes” that perform many layer-3 functions (except forwarding)
  - NAT: rewrite packet header’s IP address and port #
  - Firewall: block traffic based on header-field values
  - Load-balancer: forward packets requesting a given service to one of a set of servers
- Proliferation of **layer-2 switches** and **layer-3 routers**
- Each has its own specialized hardware, software, and management interface
- A **unified approach** to provide **many** network-layer functions and **certain** link-layer functions in an **integrated** manner

Generalized Forwarding and SDN

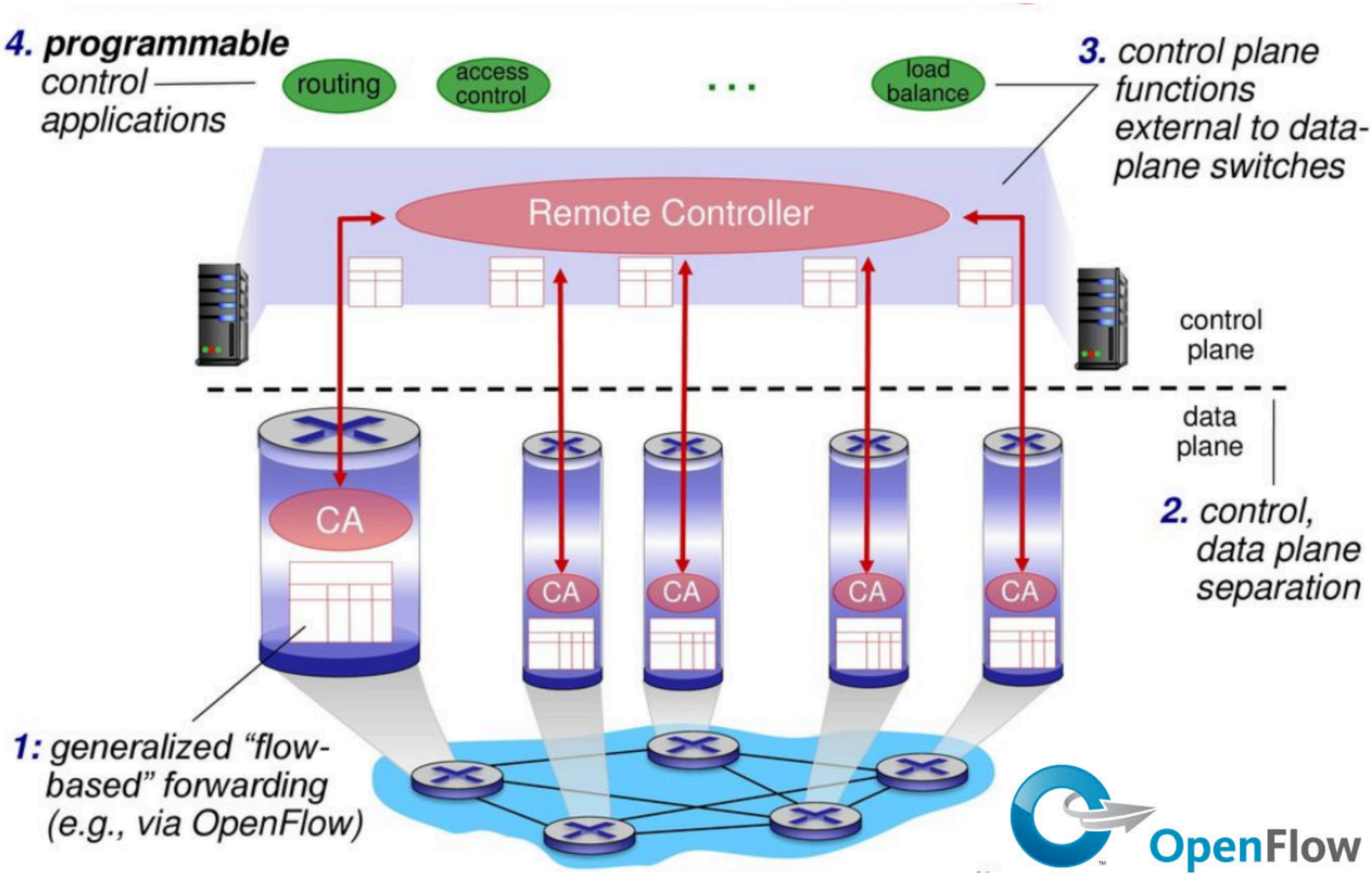
- Each router contains a ***flow (match + action) table*** that is computed and distributed by a ***logically centralized*** routing controller.



Why a logically centralized control plane?

- Easier network management: avoid router misconfigurations, greater flexibility of traffic flows
- Table-based forwarding allows "programmable" routers
  - Centralized "programming" easier: compute tables centrally and distribute flow tables
  - Distributed "programming" more difficult: compute tables as result of distributed algorithm (protocol) implemented in each and every router
- Open (non-proprietary) implementation of control plane

SDN


4. programmable control applications
routing
access control
...
load balance
3. control plane functions external to data-plane switches
Remote Controller
control plane
data plane
CA
CA
CA
CA
CA
2. control, data plane separation
1: generalized "flow-based" forwarding (e.g., via OpenFlow)
OpenFlow

A significant research work

- B4 at Google: Published at ACM Sigcomm 2013

SIGCOMM 2013 Hong Kong

B4: Experience with a Globally-Deployed Software Defined WAN

Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, Jonathan Zolla, Urs Hölzle, Stephen Stuart and Amin Vahdat
Google, Inc.
b4-sigcomm@google.com

ABSTRACT

We present the design, implementation, and evaluation of B4, a private WAN connecting Google's data centers across the planet. B4 has a number of unique characteristics: i) massive bandwidth requirements deployed to a modest number of sites, ii) elastic traffic demand that seeks to maximize average bandwidth, and iii) full control over the edge servers and network, which enables rate limiting and demand measurement at the edge. These characteristics led to a Software Defined Networking architecture using OpenFlow to control relatively simple switches built from merchant silicon. B4's centralized traffic engineering service drives links to near 100% utilization, while splitting application flows among multiple paths to balance capacity against application priority/demands. We describe experience with three years of B4 production deployment, lessons learned, and areas for future work.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing Protocols

Such overprovisioning delivers admirable reliability at the very real costs of 2-3x bandwidth over-provisioning and high-end routing gear.

We were faced with these overheads for building a WAN connecting multiple data centers with substantial bandwidth requirements. However, Google's data center WAN exhibits a number of unique characteristics. First, we control the applications, servers, and the LANs all the way to the edge of the network. Second, our most bandwidth-intensive applications perform large-scale data copies from one site to another. These applications benefit most from high levels of average bandwidth and can adapt their transmission rate based on available capacity. They could similarly defer to higher priority interactive applications during periods of failure or resource constraint. Third, we anticipated no more than a few dozen data center deployments, making central control of bandwidth feasible.

We exploited these properties to adopt a software defined networking (SDN) architecture for our data center WAN interconnect. We were most motivated by deploying routing and traffic engineering protocols customized to our unique requirements. Our de-

Some top conferences

- There are some top conferences in computer networks every year.

- ACM Sigcomm (http://conferences.sigcomm.org/sigcomm/2019/),Beijing
- IEEE INFOCOM (https://infocom2019.ieee-infocom.org/), Paris
- IEEE ICC (https://icc2019.ieee-icc.org/), Shanghai (Dublin next year)
- USENIX NSDI (https://www.usenix.org/conference/nsdi19)

- There are some interesting events (competitions, tutorials workshops, etc.) for students.

Summary

- In your Packet Tracer Labs:
  - Are you tired of configuring IP address, subnet mask, and default gateway manually for every single connected end device?
  - Are you tired of configuring dynamic routing protocols manually for every single router in your network?
- Any smarter ways existed?

- Tell me the whole story about forwarding a packet from one-end to another.

See you on...

- 6th May (Monday)
- Lab: Packet Tracer Review
- Bring your own laptop!
- Time: 15:25
- Venue: Room 218, TB-4