Netzwerktechnik und IT-Netze

Chapter 4: Network Layer

Vorlesung im WS 2016/2017

Bachelor Informatik

(3. Semester)

Prof. Dr. rer. nat. Andreas Berl

Fakultät für Elektrotechnik, Medientechnik und Informatik

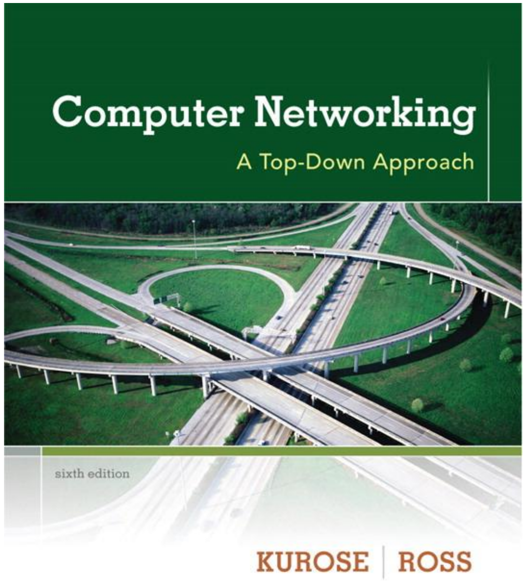
Overview

- Introduction
- Computer Networks and the Internet
- Application Layer
  - WWW, Email, DNS, and more
  - Socket programming
  - Web service
- Transport Layer
- Network Layer
- Link Layer

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Introduction

- A note on the use of these power point slides:
  -
  - Do not copy or distribute this slide set!



Computer Networking: A Top Down Approach
6th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2012

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Chapter 4: outline

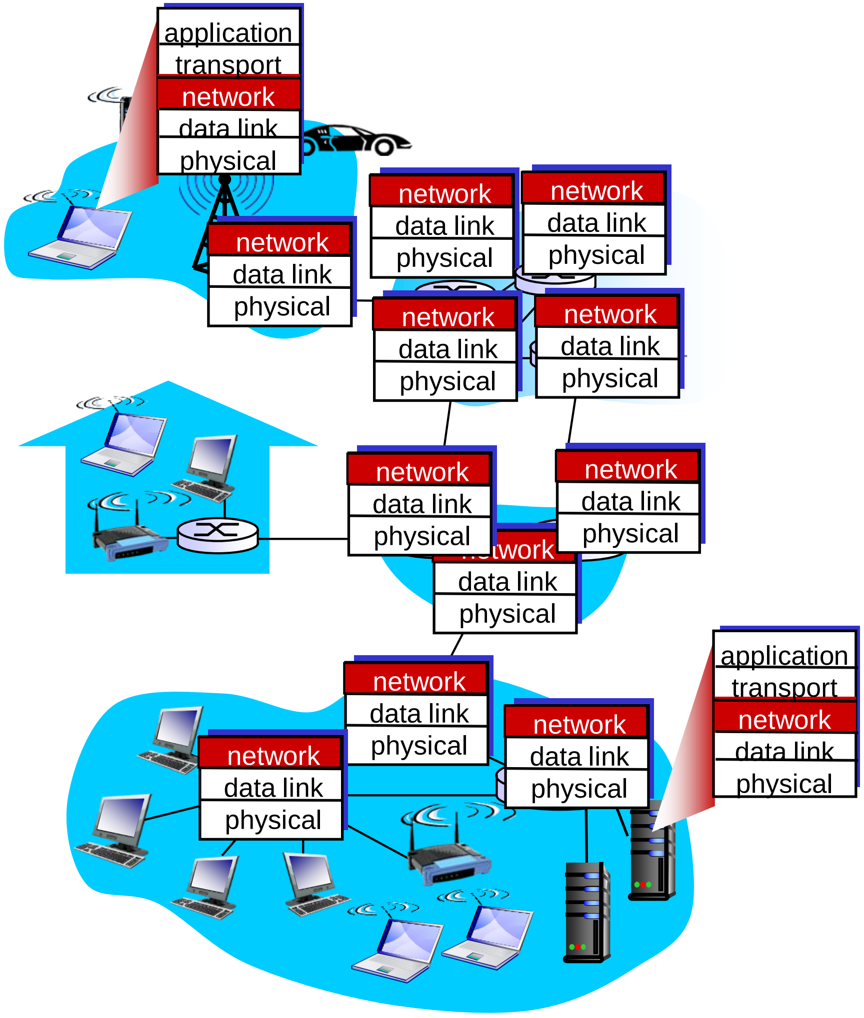
- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6

- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Network layer

- Transport segment from sending to receiving host
- On sending side encapsulates segments into datagrams
- On receiving side, delivers segments to transport layer
- Network layer protocols in every host, router
- Router examines header fields in all IP datagrams passing through it

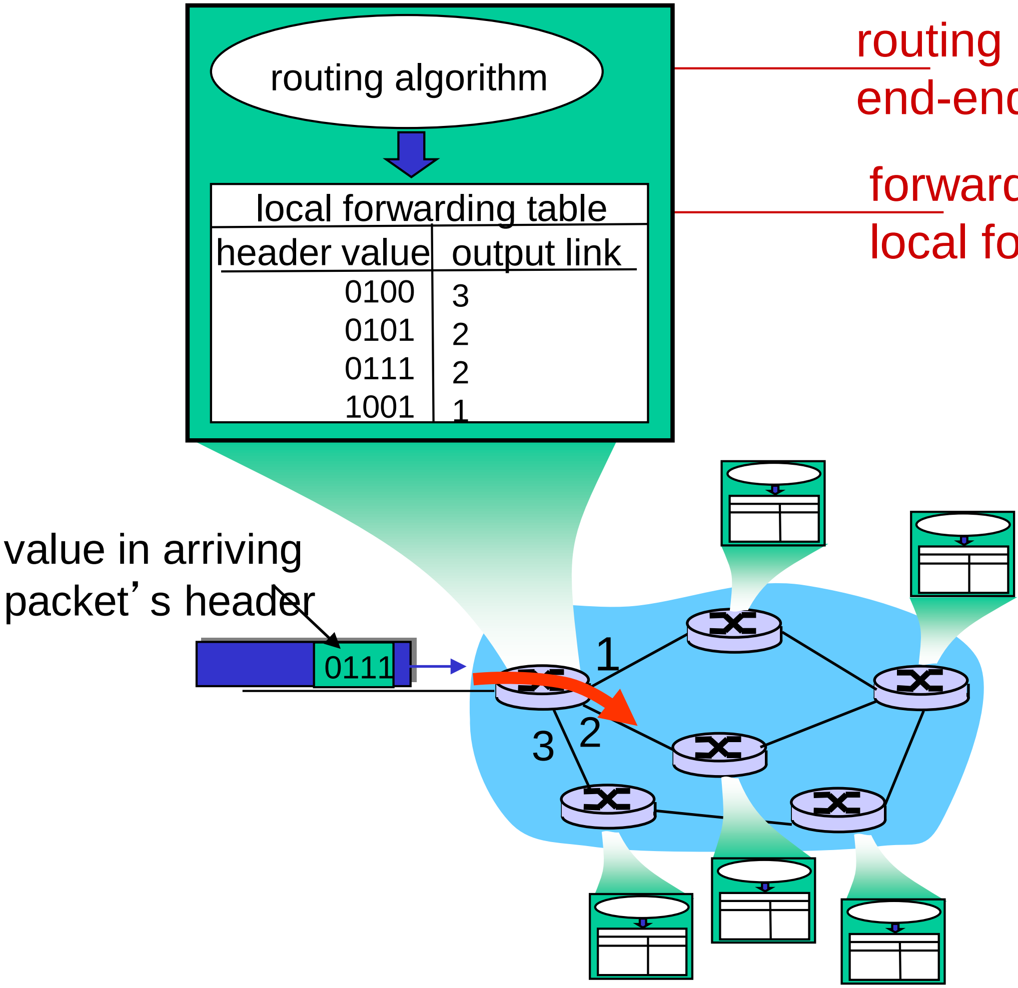


Two key network-layer functions

- Forwarding:
  - Move packets from router's input to appropriate router output
- Routing:
  - determine route taken by packets from source to destination
  - Routing algorithms
- Analogy:
  - Routing:
    - process of planning trip from source to destination
  - Forwarding:
    - process of getting through single interchange

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Interplay between routing and forwarding

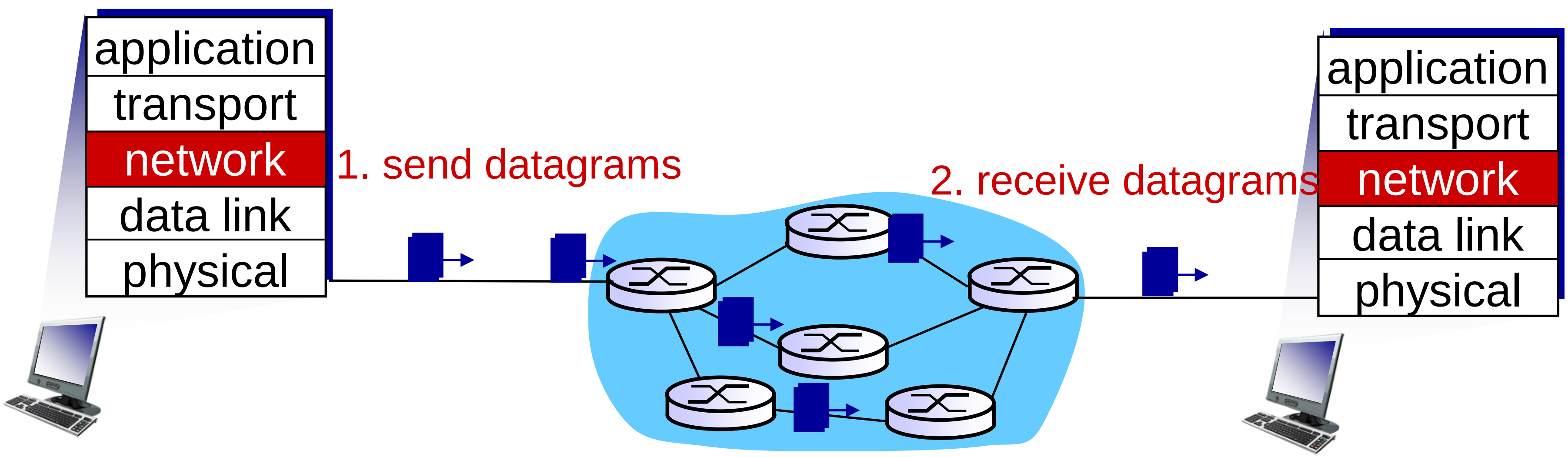


routing algorithm determines end-end-path through network

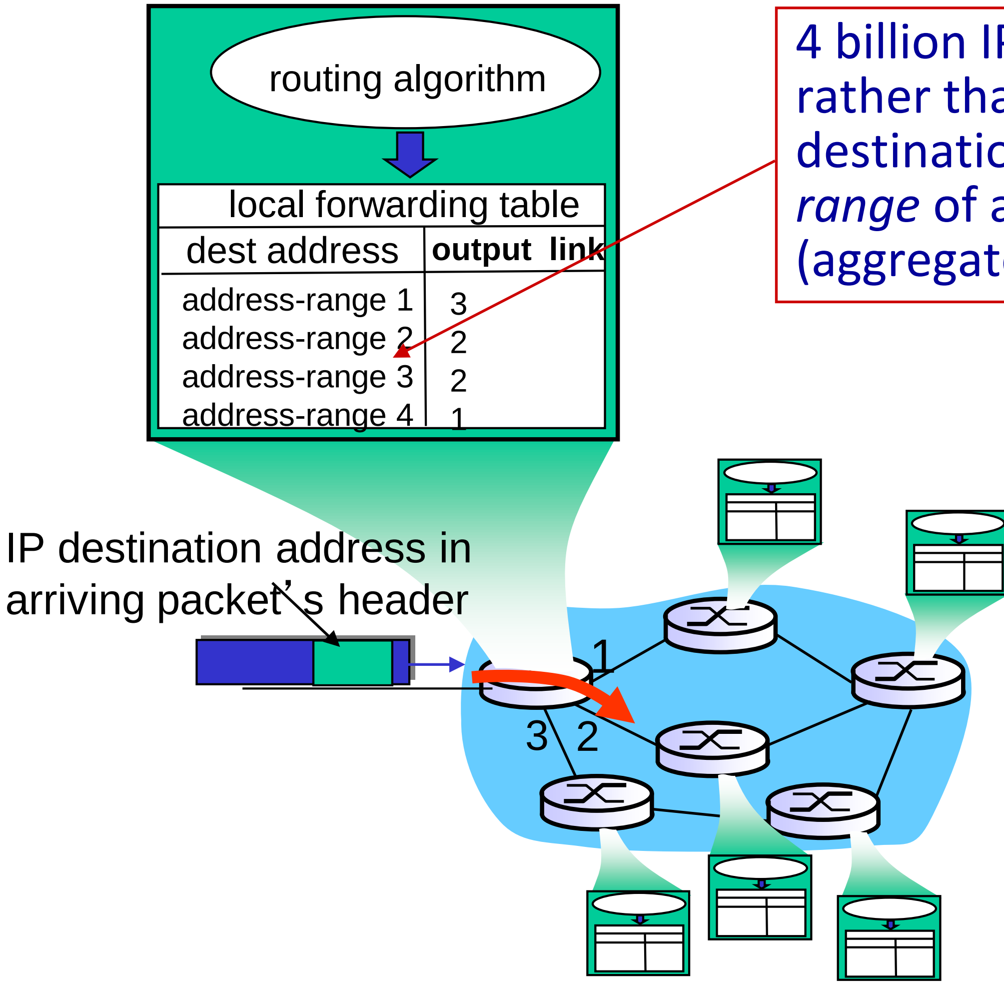
forwarding table determines local forwarding at this router

Datagram networks

- No call setup at network layer
- Routers: no state about end-to-end connections
- No network-level concept of “connection”
- Packets forwarded using destination host address



Datagram forwarding table



4 billion IP addresses, so rather than list individual destination address list *range* of addresses (aggregate table entries)

Datagram forwarding table

Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

Q: But what happens if ranges don't divide up so nicely?

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Longest prefix matching

- Longest prefix matching
  - When looking for forwarding table entry for given destination address, use longest address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
otherwise	3

Examples:

DA: 11001000 00010111 00010110 10100001 which interface?

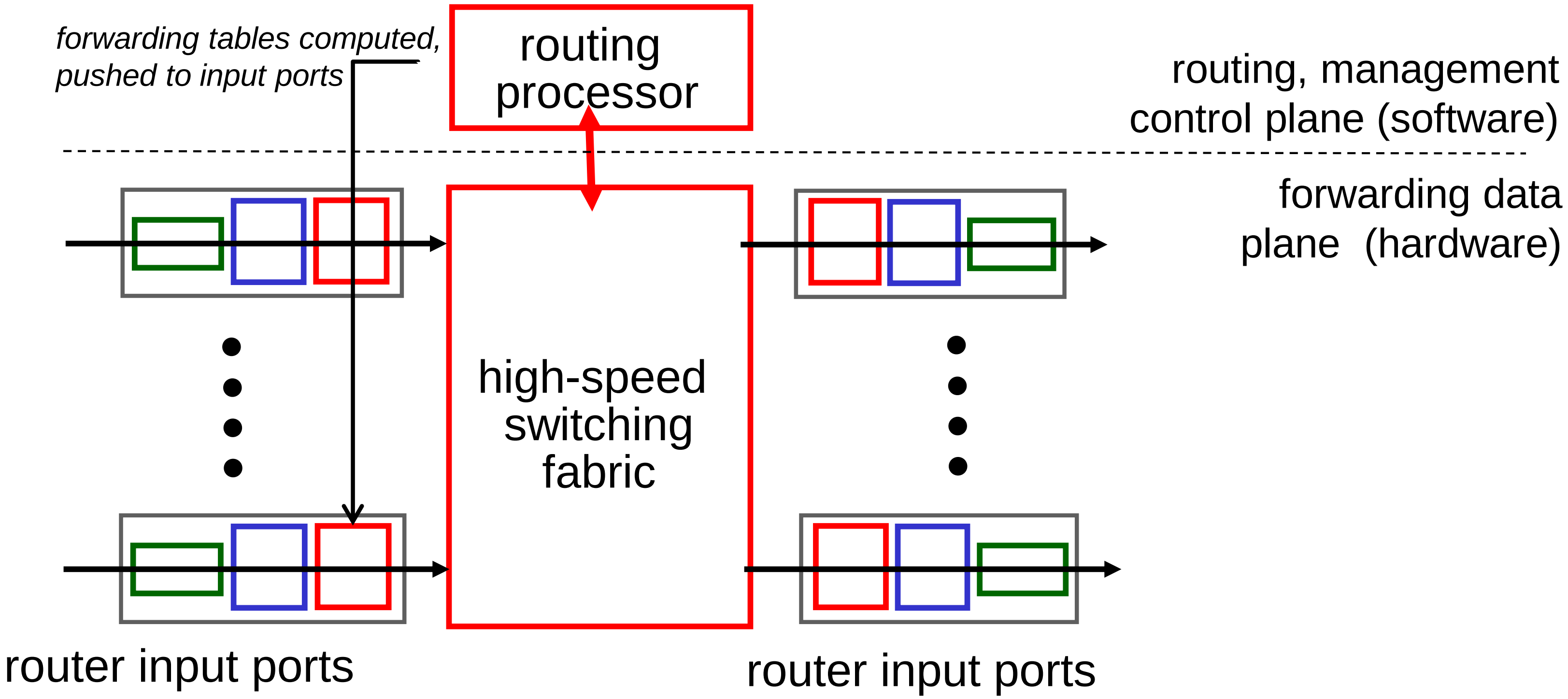
DA: 11001000 00010111 00011000 10101010 which interface?

Chapter 4: outline

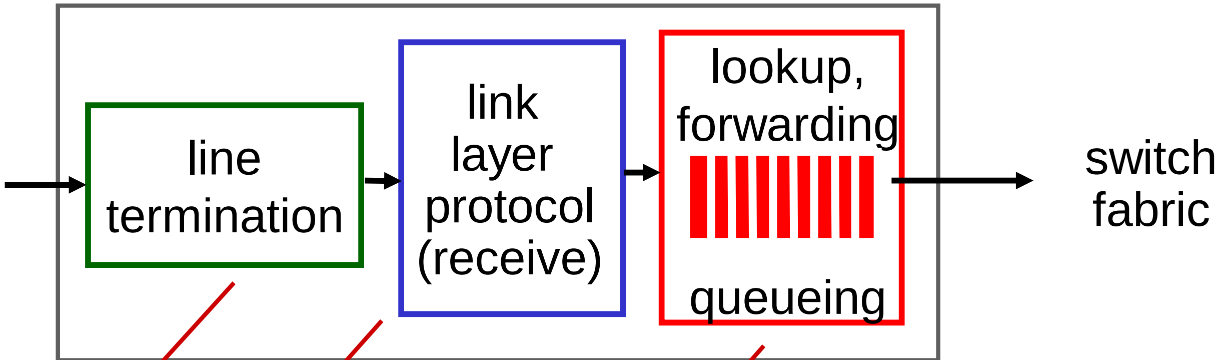
- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Router architecture overview

- Two key router functions:
  - Run routing algorithms/protocol (RIP, OSPF, BGP)
  - Forwarding datagrams from incoming to outgoing link



Input port functions



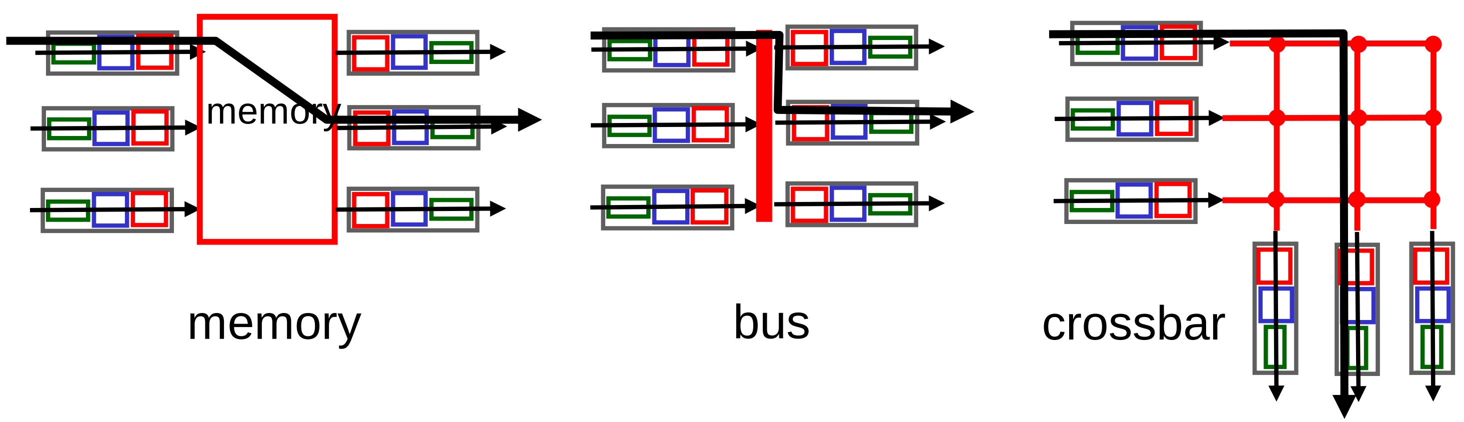
physical layer:
bit-level reception
data link layer:
e.g., Ethernet
see chapter 5

Decentralized switching:

- Given datagram dest, lookup output port using forwarding table in input port memory ("match plus action")
- goal: complete input port processing at 'line speed'
- queuing: if datagrams arrive faster than forwarding rate into switch fabric

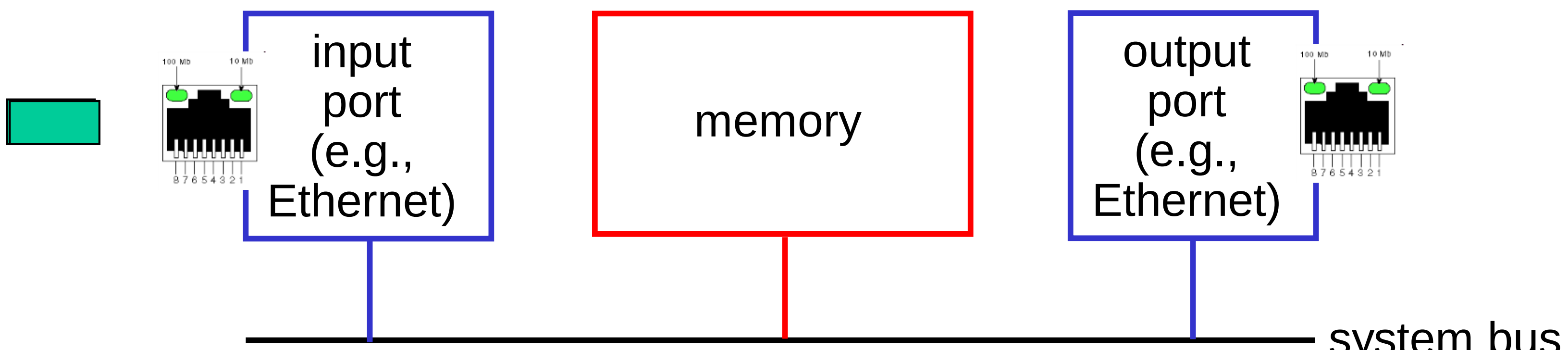
Switching fabrics

- Transfer packet from input buffer to appropriate output buffer
- Switching rate: rate at which packets can be transferred from inputs to outputs
  - Often measured as multiple of input/output line rate
  - N inputs: switching rate N times line rate desirable
- Three types of switching fabrics



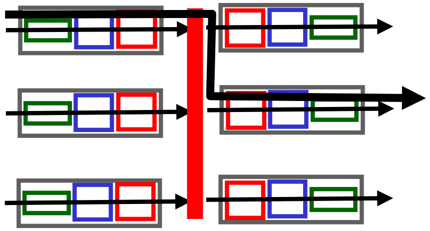
Switching via memory

- First generation routers:
  - Traditional computers with switching under direct control of CPU
  - Packet copied to system's memory
  - Speed limited by memory bandwidth (2 bus crossings per datagram)



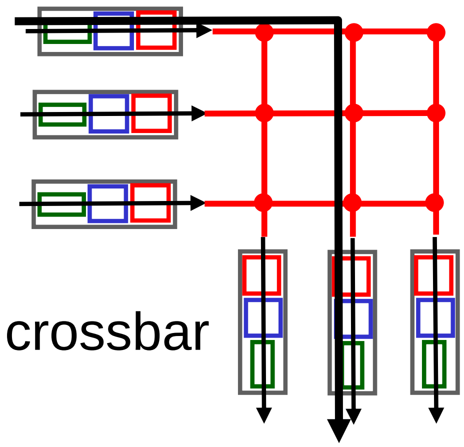
Switching via a bus

- Datagram from input port memory to output port memory via a shared bus
- Bus contention: switching speed limited by bus bandwidth
- 32 Gbps bus, Cisco 5600: sufficient speed for access and enterprise routers

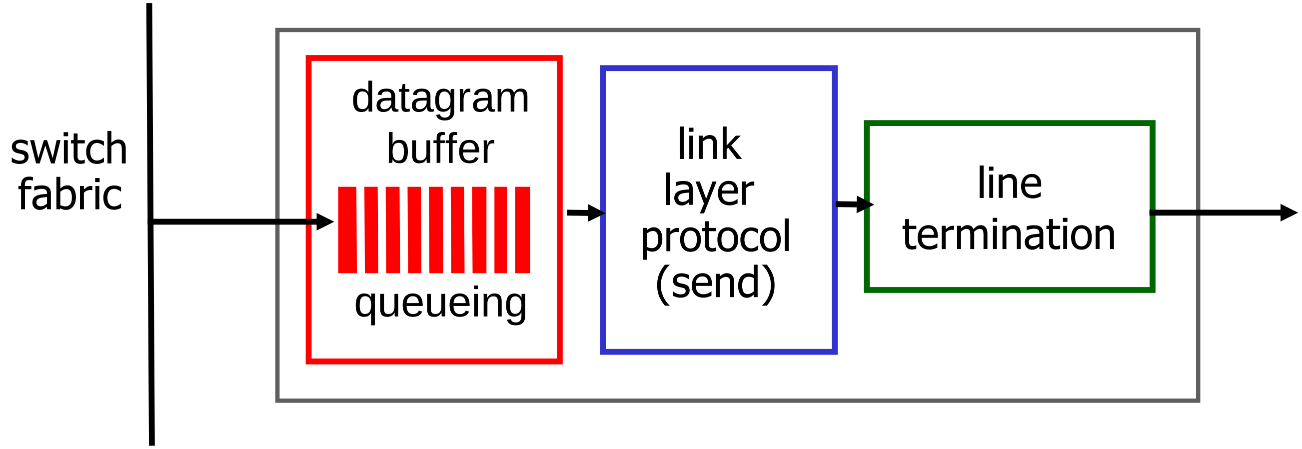


Switching via interconnection network

- Overcome bus bandwidth limitations
- Banyan networks, crossbar, other interconnection nets initially developed to connect processors in multiprocessor
- Advanced design: fragmenting datagram into fixed length cells, switch cells through the fabric.
- Cisco 12000: switches 60 Gbps through the interconnection network



Output ports: (HUGELY important)



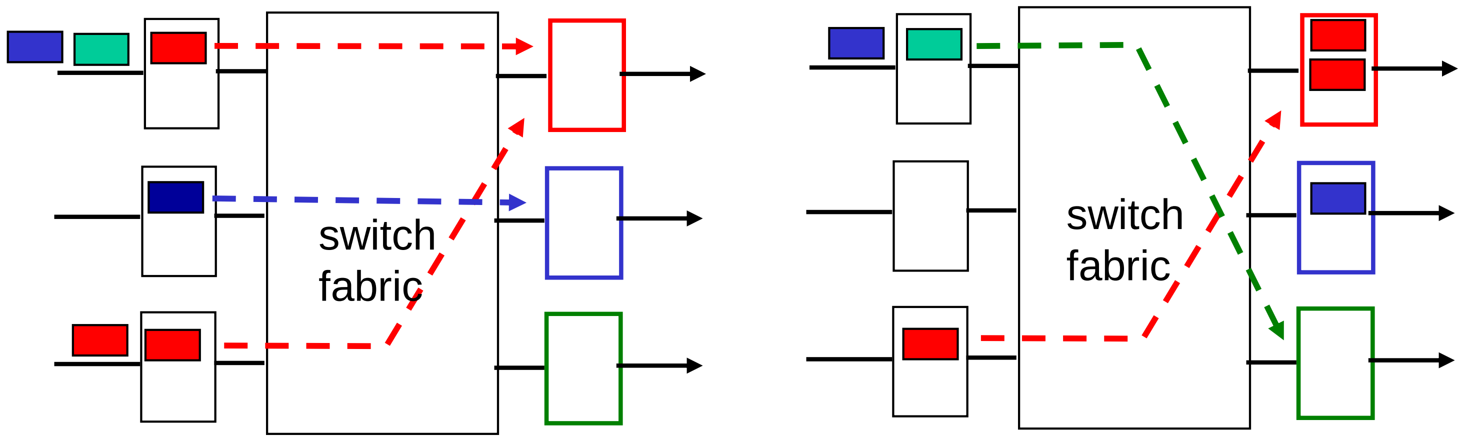
- Buffering required when datagrams arrive from fabric faster than the transmission rate

Datagram (packets) can be lost due to congestion, lack of buffers

- Scheduling discipline chooses among queued datagrams for transmission

Priority scheduling – who gets best performance, network neutrality

Output port queueing



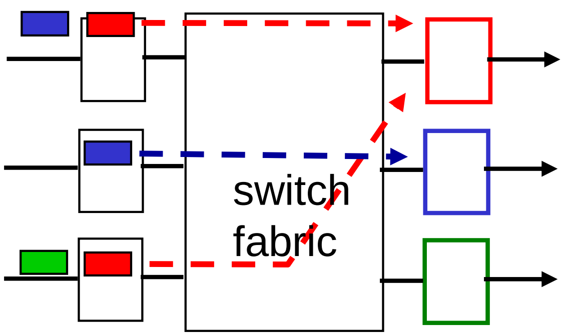
at *t*, packets more from input to output

one packet time later

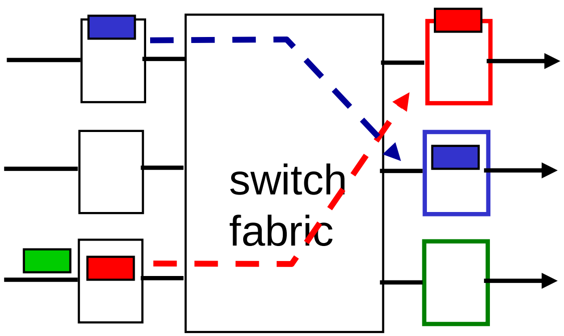
- Buffering when arrival rate via switch exceeds output line speed
- *Queueing (delay) and loss due to output port buffer overflow!*

Input port queuing

- Fabric slower than input ports combined -> queueing may occur at input queues
  - Queueing delay and loss due to input buffer overflow!
- Head-of-the-Line (HOL) blocking: Queued datagram at front of queue prevents others in queue from moving forward



output port contention:
only one red datagram can be transferred.
lower red packet is blocked



one packet time later: green packet experiences HOL blocking

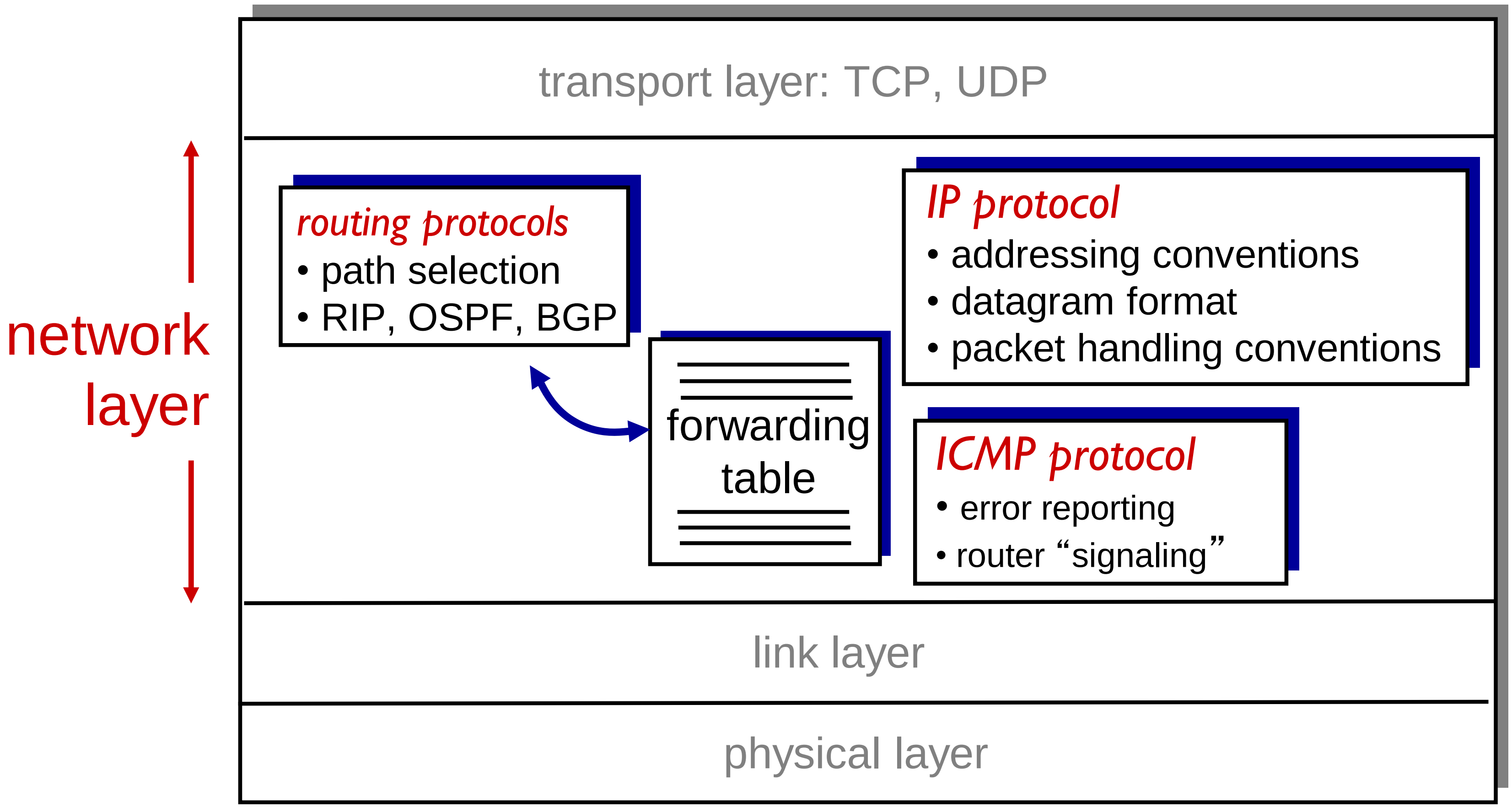
Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

The Internet network layer

- Host, router network layer functions:



IP datagram format

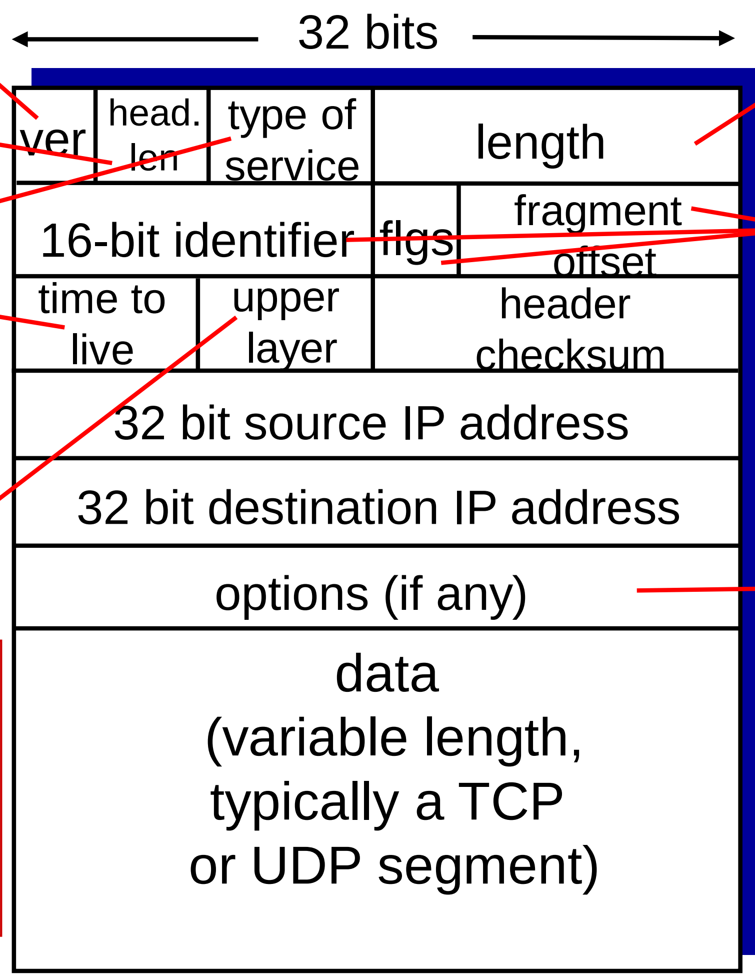
IP protocol version number

header length (bytes)

"type" of data

max number remaining hops (decremented at each router)

upper layer protocol to deliver payload to



total datagram length (bytes)

for fragmentation/ reassembly

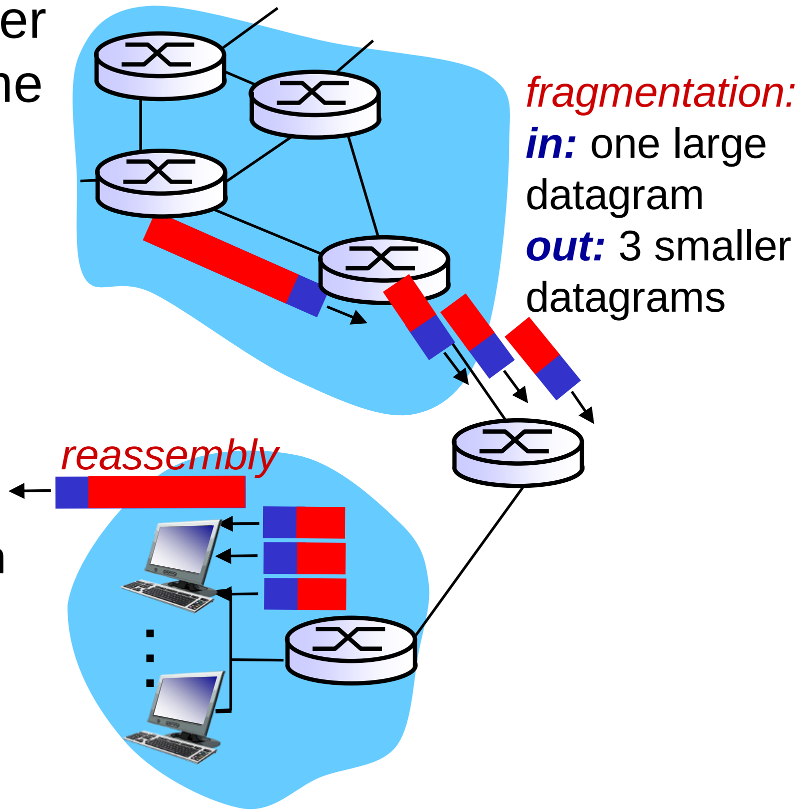
e.g. timestamp, record route taken, specify list of routers to visit.

how much overhead?

- 20 bytes of TCP
- 20 bytes of IP
- = 40 bytes + app layer overhead

IP fragmentation, reassembly

- Network links have MTU (max.transfer size) - largest possible link-level frame
  - different link types, different MTUs
- Large IP datagram divided ("fragmented") within net
  - One datagram becomes several datagrams
  - "Reassembled" only at final destination
  - IP header bits used to identify, order related fragments



IP fragmentation, reassembly

example:

- 4000 byte datagram
- MTU = 1500 bytes

	length =4000	ID =x	fragflag =0	offset =0	

one large datagram becomes several smaller datagrams

1480 bytes in data field

offset = 1480/8

	length =1500	ID =x	fragflag =1	offset =0	
	length =1500	ID =x	fragflag =1	offset =185	
	length =1040	ID =x	fragflag =0	offset =370	

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

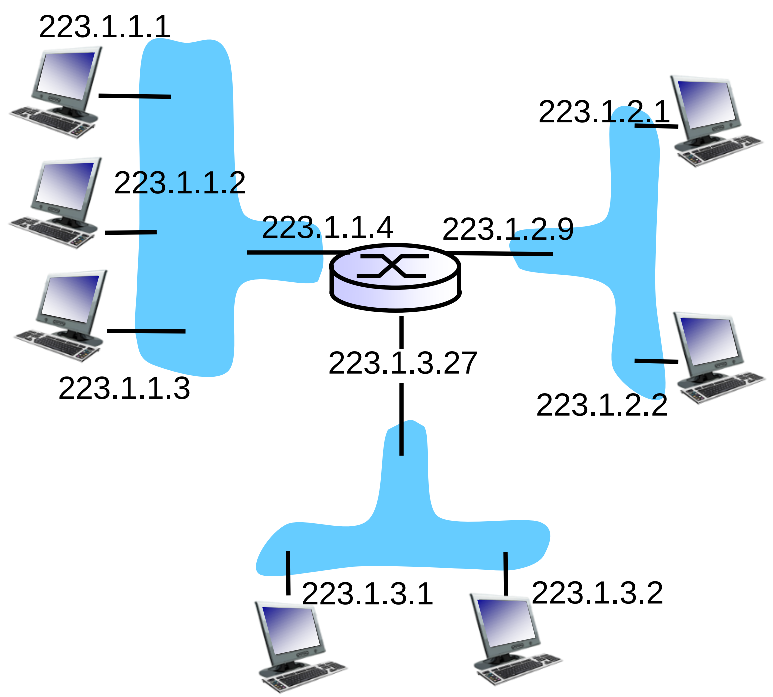
Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

IP addressing: introduction

- IP address: 32-bit identifier for host, router interface
- Interface: connection between host/router and physical link
  - Router's typically have multiple interfaces
  - Host typically has one or two interfaces (e.g., wired Ethernet, wireless 802.11)
- IP addresses associated with each interface

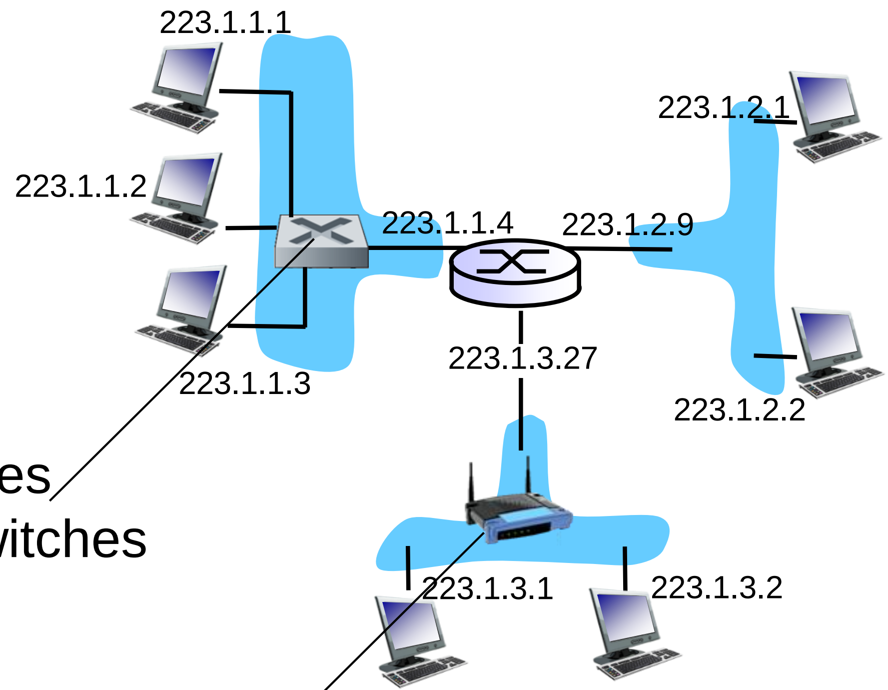


$$223.1.1.1 = \underbrace{11011111}_{223}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}$$

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

IP addressing: introduction

- Q: how are interfaces actually connected?
- A: we'll learn about that in chapter 5, 6.



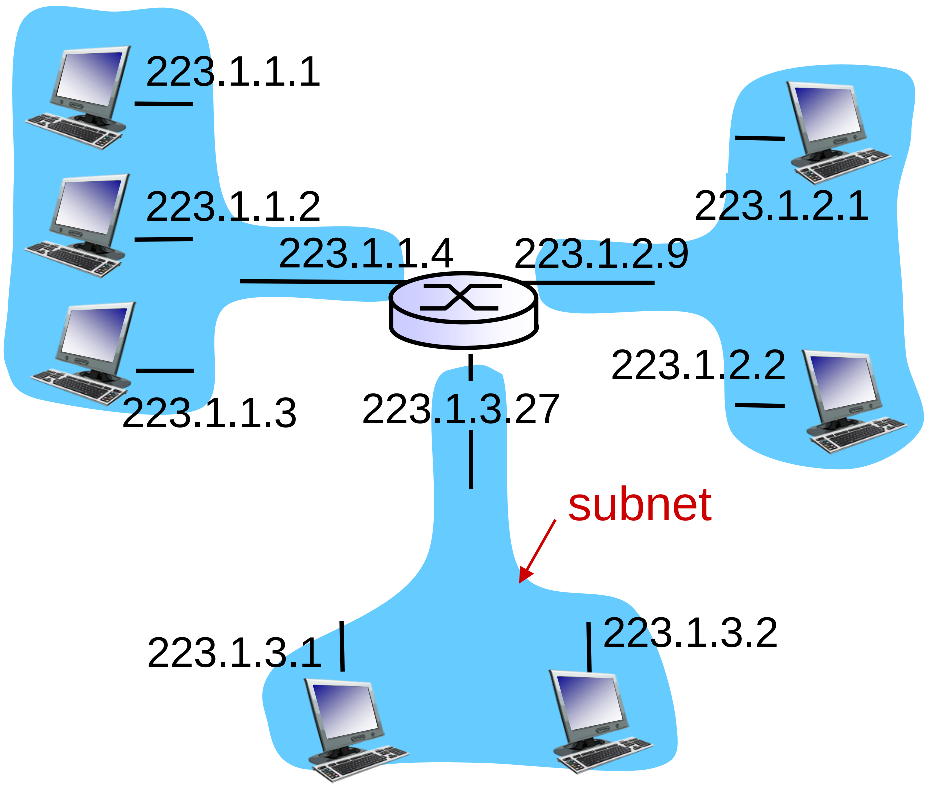
A: wired Ethernet interfaces connected by Ethernet switches

For now: don't need to worry about how one interface is connected to another (with no intervening router)

A: wireless WiFi interfaces connected by WiFi base station

Subnets

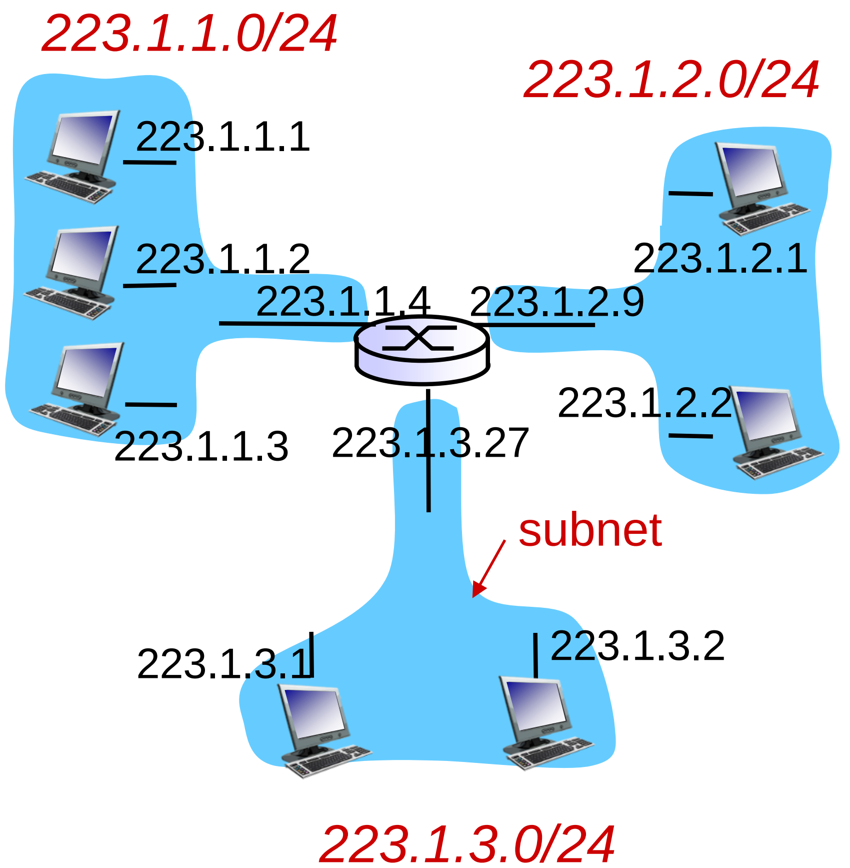
- IP address:
  - Subnet part - high order bits
  - Host part - low order bits
- What's a subnet ?
  - Devides interfaces with same subnet part of IP address
  - Can physically reach each other without intervening router



network consisting of 3 subnets

Subnets

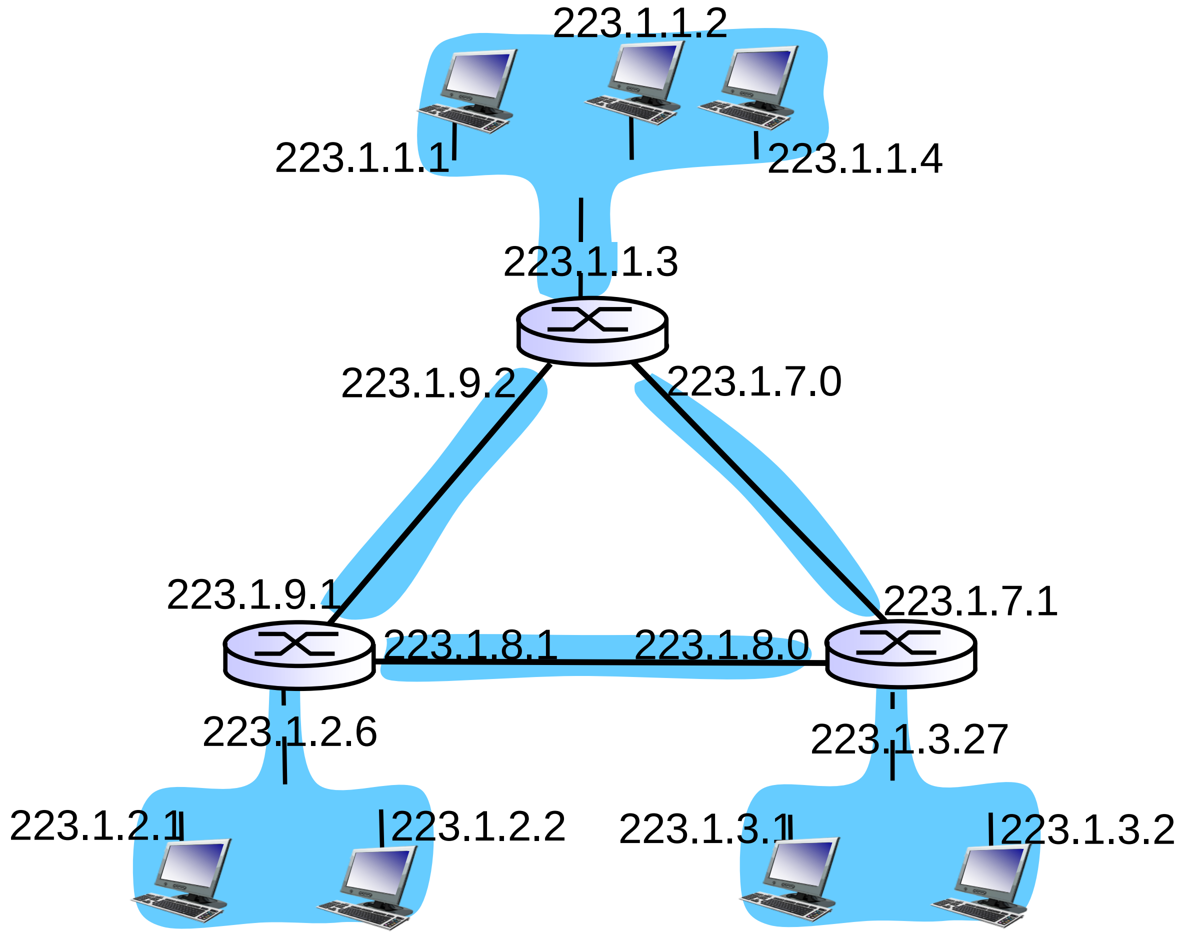
- Recipe
  - To determine the subnets, detach each interface from its host or router, creating islands of isolated networks
  - Each isolated network is called a subnet



subnet mask: /24

Subnets

- How many?



IP addressing: CIDR

- CIDR: Classless Inter Domain Routing
  - Subnet portion of address of arbitrary length
  - Address format: a.b.c.d/x, where x is # bits in subnet portion of address

subnet part | host part

11001000 00010111 00010000 00000000

200.23.16.0/23

IP addressing: Special addresses

CIDR-Adressblock	Adressbereich	Beschreibung
0.0.0.0/8	0.0.0.0 bis 0.255.255.255	aktuelles Netz (nur als Quelladresse gültig)
10.0.0.0/8	10.0.0.0 bis 10.255.255.255	Netzwerk für den privaten Gebrauch
100.64.0.0/10	100.64.0.0 bis 100.127.255.255	Mehrfach benutzter Adressbereich für Provider-NAT (siehe Carrier-grade NAT)
127.0.0.0/8	127.0.0.0 bis 127.255.255.255	Localnet
169.254.0.0/16	169.254.0.0 bis 169.254.255.255	Zeroconf
172.16.0.0/12	172.16.0.0 bis 172.31.255.255	Netzwerk für den privaten Gebrauch
192.0.0.0/24	192.0.0.0 bis 192.0.0.255	reserviert, aber zur Vergabe vorgesehen
192.0.0.0/29	192.0.0.0 bis 192.0.0.7	Dual-Stack Lite (DS-Lite), IPv4- und IPv6 Übergangsmechanismus mit globaler IPv6-Adresse und Provider-NAT für IPv4
192.0.2.0/24	192.0.2.0 bis 192.0.2.255	Dokumentation und Beispielcode (TEST-NET-1)
192.88.99.0/24	192.88.99.0 bis 192.88.99.255	6to4-Anycast-Weiterleitungspräfix
192.168.0.0/16	192.168.0.0 bis 192.168.255.255	Netzwerk für den privaten Gebrauch
198.18.0.0/15	198.18.0.0 bis 198.19.255.255	Netz-Benchmark-Tests
198.51.100.0/24	198.51.100.0 bis 198.51.100.255	Dokumentation und Beispielcode (TEST-NET-2)
203.0.113.0/24	203.0.113.0 bis 203.0.113.255	Dokumentation und Beispielcode (TEST-NET-3)
224.0.0.0/4	224.0.0.0 bis 239.255.255.255	Multicasts (früheres Klasse-D-Netz)
240.0.0.0/4	240.0.0.0 bis 255.255.255.255	reserviert (früheres Klasse-E-Netz)
255.255.255.255	255.255.255.255	Broadcast

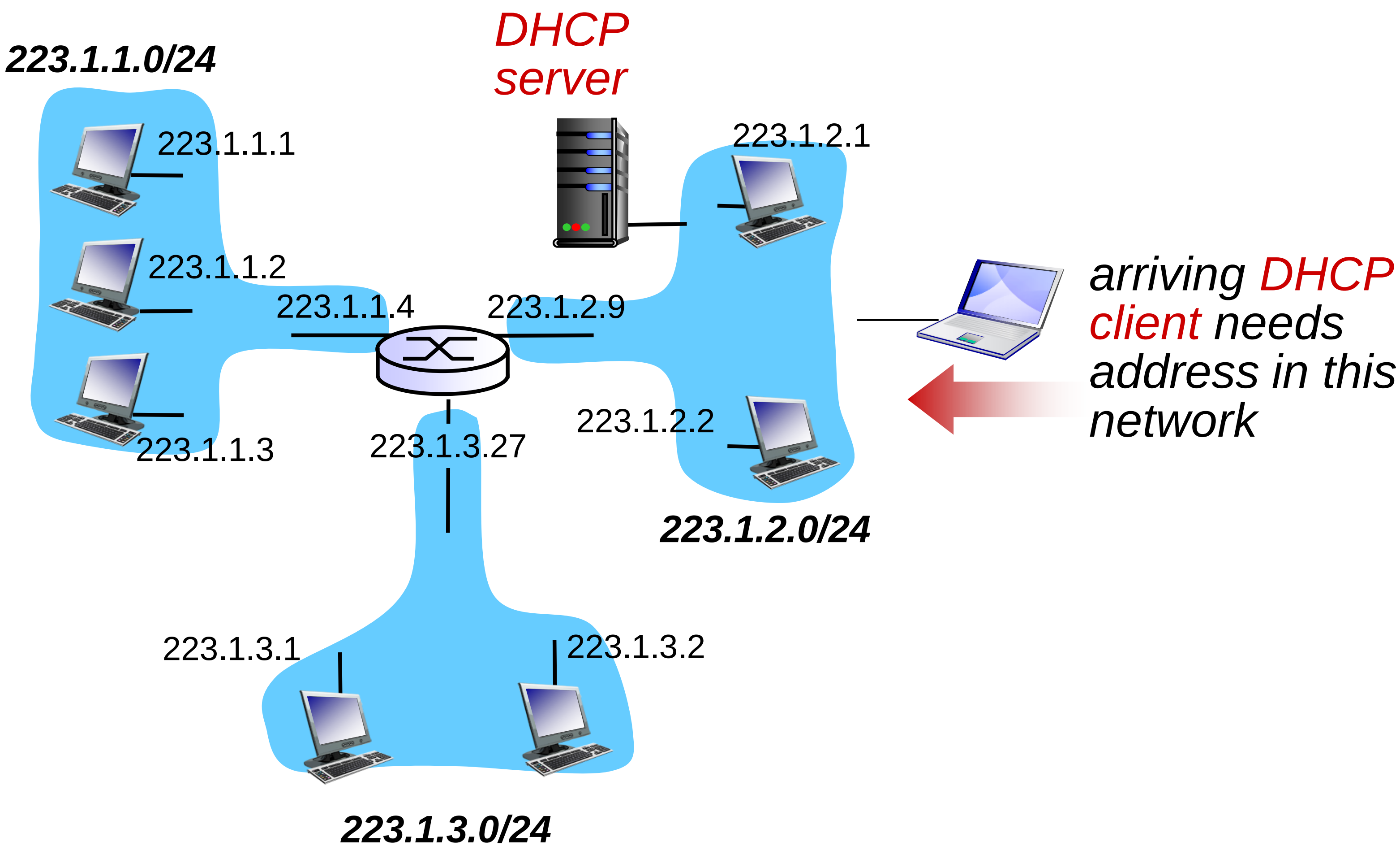
IP addresses: how to get one?

- Q: How does a host get IP address?
  - Hard-coded by system admin in a file
    - Windows: control-panel->network->configuration->TCP/IP->properties
    - UNIX: /etc/rc.config
  - DHCP: Dynamic Host Configuration Protocol: dynamically get address from as server
    - “Plug-and-play”

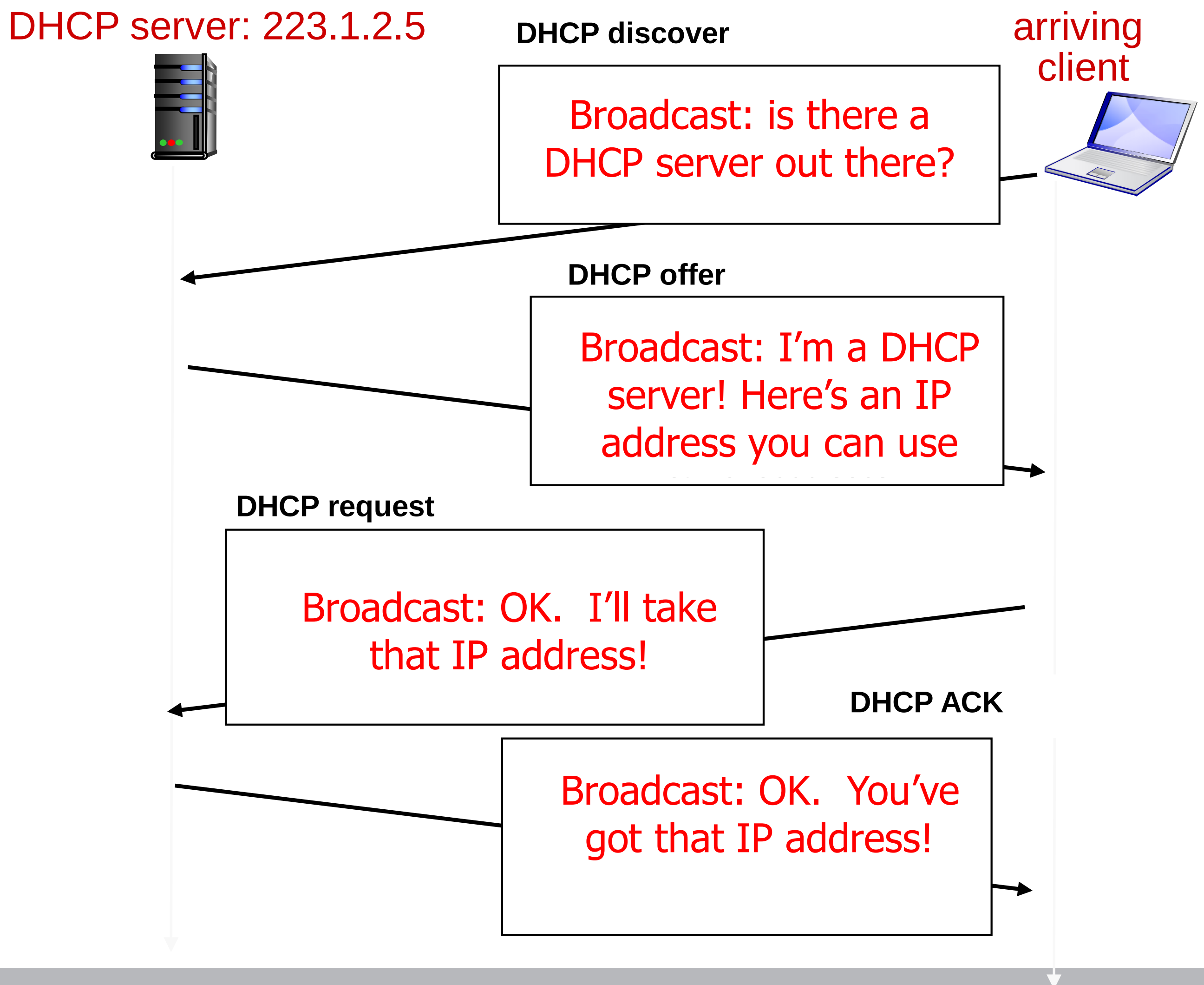
DHCP: Dynamic Host Configuration Protocol

- Goal: allow host to dynamically obtain its IP address from network server when it joins network
  - Can renew its lease on address in use
  - Allows reuse of addresses (only hold address while connected/“on”)
  - Support for mobile users who want to join network (more shortly)
- DHCP overview:
  - Host broadcasts “DHCP discover” msg [optional]
  - DHCP server responds with “DHCP offer” msg [optional]
  - Host requests IP address: “DHCP request” msg
  - DHCP server sends address: “DHCP ACK” msg

DHCP client-server scenario



DHCP client-server scenario

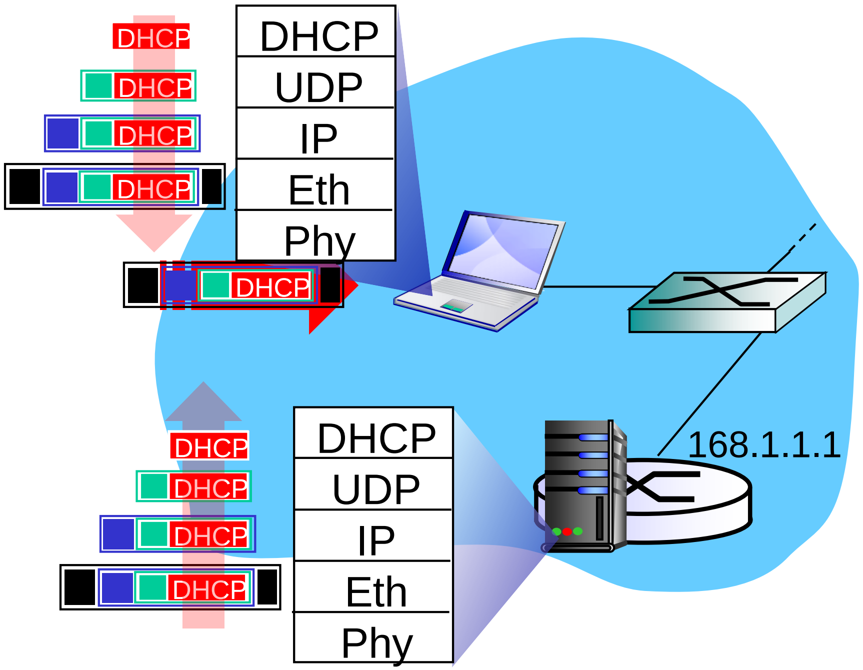


DHCP: more than IP addresses

- DHCP can return more than just allocated IP address on subnet:
  - Address of first-hop router for client
  - Name and IP address of DNS sever
  - Network mask (indicating network versus host portion of address)

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

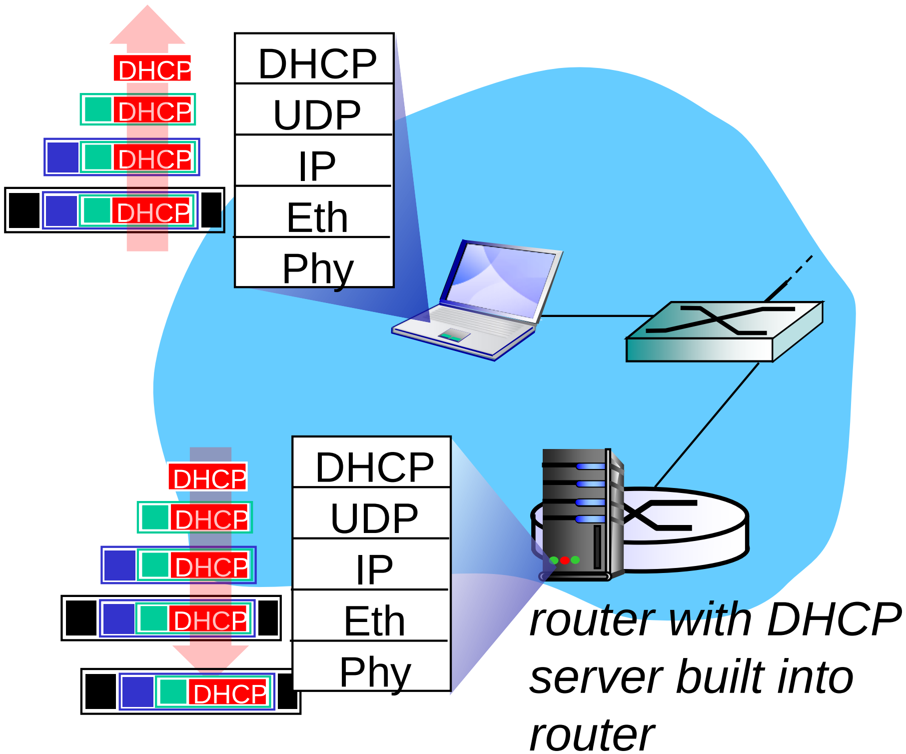
DHCP: example



- Connecting laptop needs its IP address, addr of first-hop router, addr of DNS server:
- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.1 Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

DHCP: example



- DCP server formulates DHCP ACK containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
- Encapsulation of DHCP server, frame forwarded to client, demuxing up to DHCP at client
- Client now knows its IP address, name and IP address of DSN server, IP address of its first-hop router

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

IP addresses: how to get one?

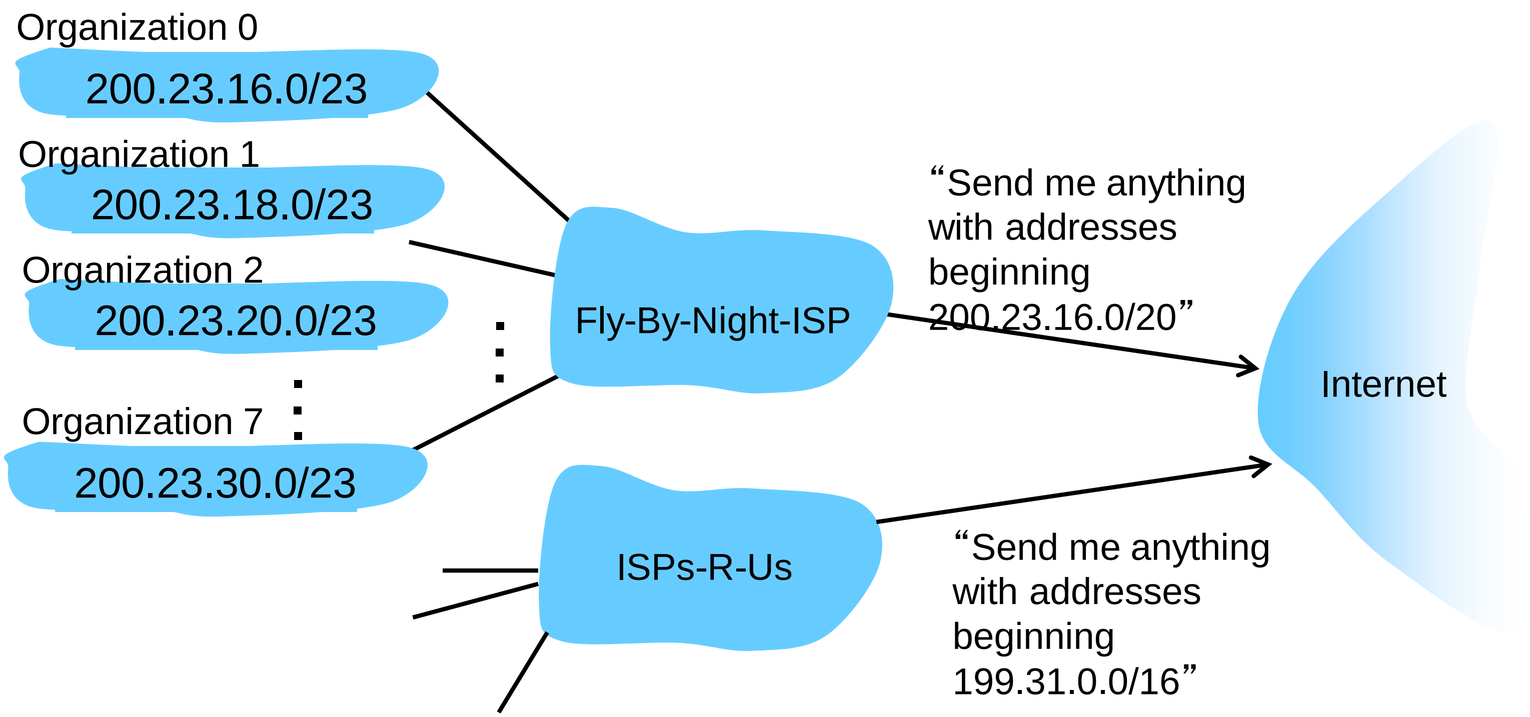
- Q: How does network get subnet part of IP addr?
- A: Gets allocated portion of its provider ISP's address space

ISP's block	<u>11001000 00010111 00010000</u>	00000000	200.23.16.0/20
Organization 0	<u>11001000 00010111 0001000</u>0	00000000	200.23.16.0/23
Organization 1	<u>11001000 00010111 0001001</u>0	00000000	200.23.18.0/23
Organization 2	<u>11001000 00010111 0001010</u>0	00000000	200.23.20.0/23
...	…..	….	….
Organization 7	<u>11001000 00010111 0001111</u>0	00000000	200.23.30.0/23

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

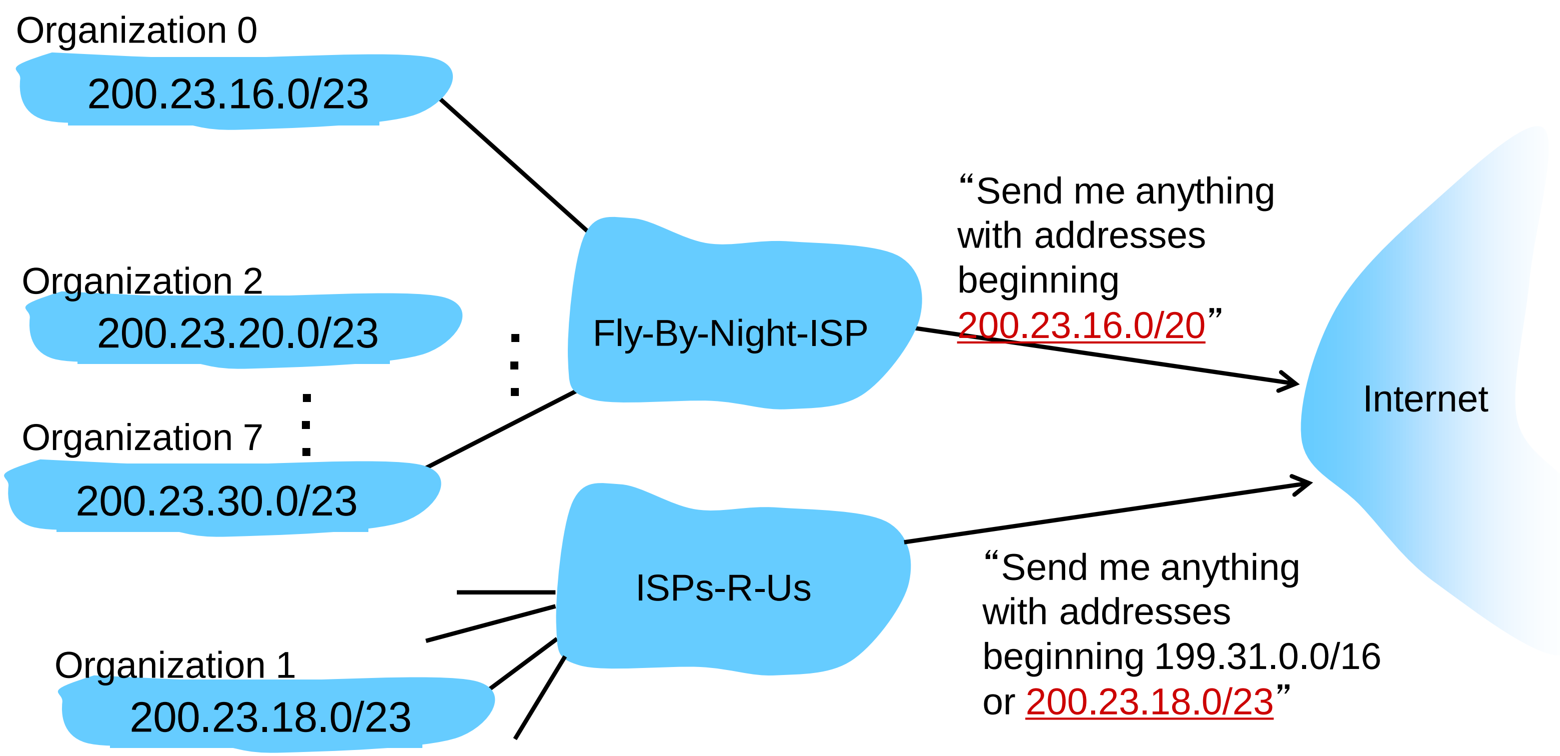
Hierarchical addressing: route aggregation

- Hierarchical addressing allows efficient advertisement of routing
- Information:



Hierarchical addressing: more specific routes

- ISPs-R-Us has a more specific route to Organization 1



Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

ICMP: internet control message protocol

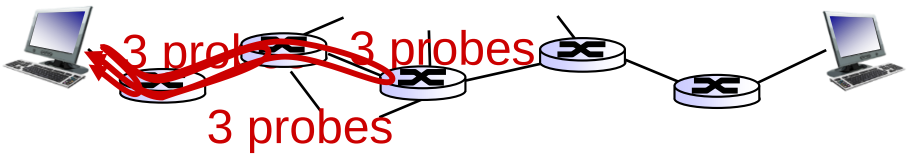
- Used by hosts & routers to communicate network-level information
  - Error reporting: unreachable host, network, port, protocol
  - Echo request/reply (used by ping)
- Network-layer "above" IP:
  - ICMP msgs carried in IP datagrams
- ICMP message: type, code plus first 8 bytes of IP datagram causing error

Type	Code	description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

Traceroute and ICMP

- Source sends series of UDP segments to dest
  - First set has TTL =1
  - Second set has TTL=2, etc.
  - Unlikely port number
- When nth set of datagrams arrives to nth router:
  - Router discards datagrams
  - And sends source ICMP messages (type 11, code 0)
  - ICMP messages includes name of router & IP address

- When ICMP messages arrives, source records RTTs
- Stopping criteria:
  - UDP segment eventually arrives at destination host
  - Destination returns ICMP "port unreachable" message (type 3, code 3)
  - Source stops



IPv6: motivation

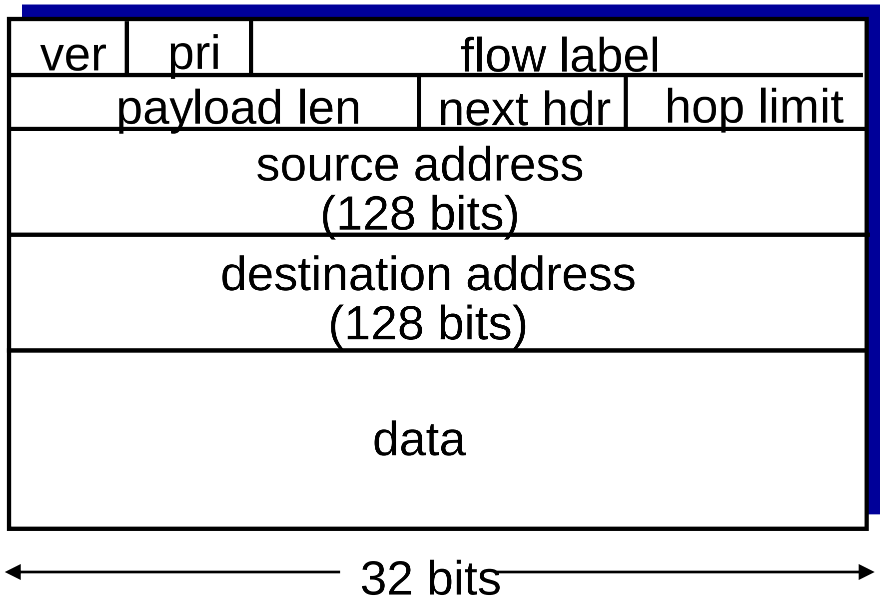
- Initial motivation: 32-bit address space soon to be completely allocated.
- Additional motivation:
  - Header format helps speed processing/forwarding
  - Header changes to facilitate QoS

- IPv6 datagram format:
  - Fixed-length 40 byte header
  - No fragmentation allowed

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

IPv6 datagram format

- Priority: identify priority among datagrams in flow
- Flow Label: identify datagrams in same “flow”.
  - (concept of“flow” not well defined).
- Next header: identify upper layer protocol for data



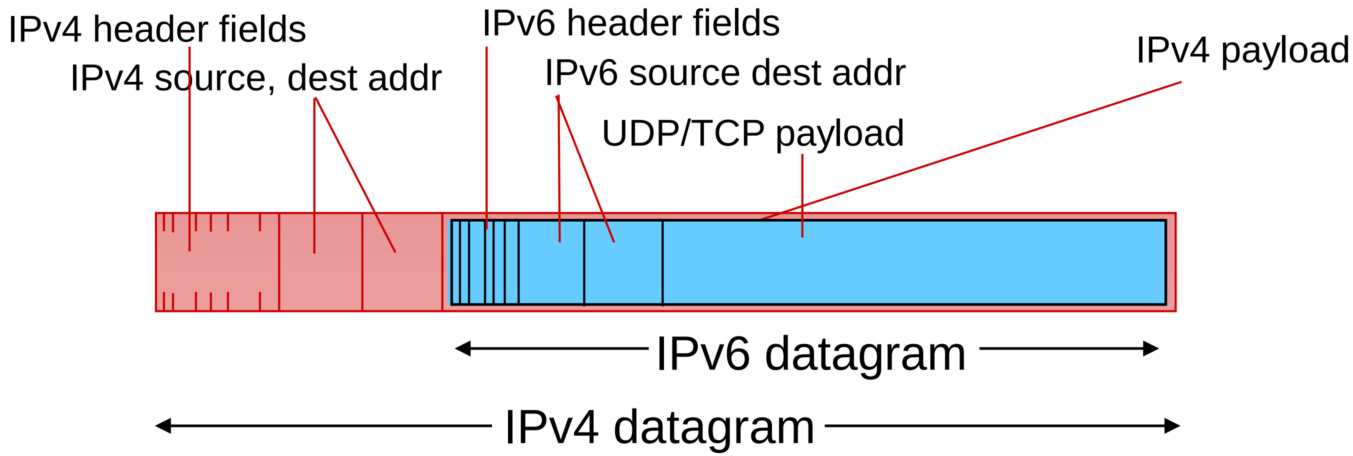
Other changes from IPv4

- Checksum: removed entirely to reduce processing time at each hop
- Options: allowed, but outside of header, indicated by “Next Header” field
- ICMPv6: new version of ICMP
  - Additional message types, e.g. “Packet Too Big”
  - Multicast group management functions

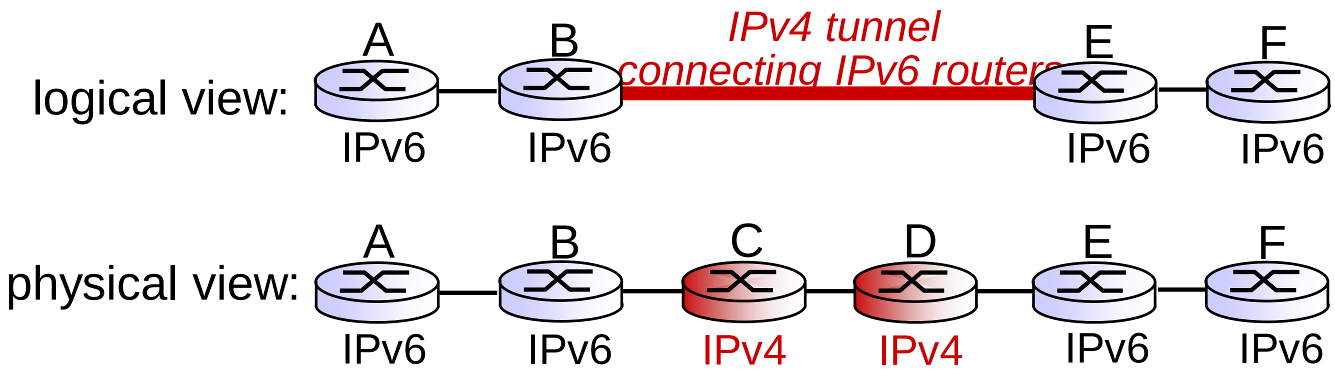
Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Transition from IPv4 to IPv6

- Not all routers can be upgraded simultaneously
  - No “flag days”
  - How will network operate with mixed IPv4 and IPv6 routers?
- Tunneling: IPv6 datagram carried as payload in IPv4 datagram among IPv4 routers

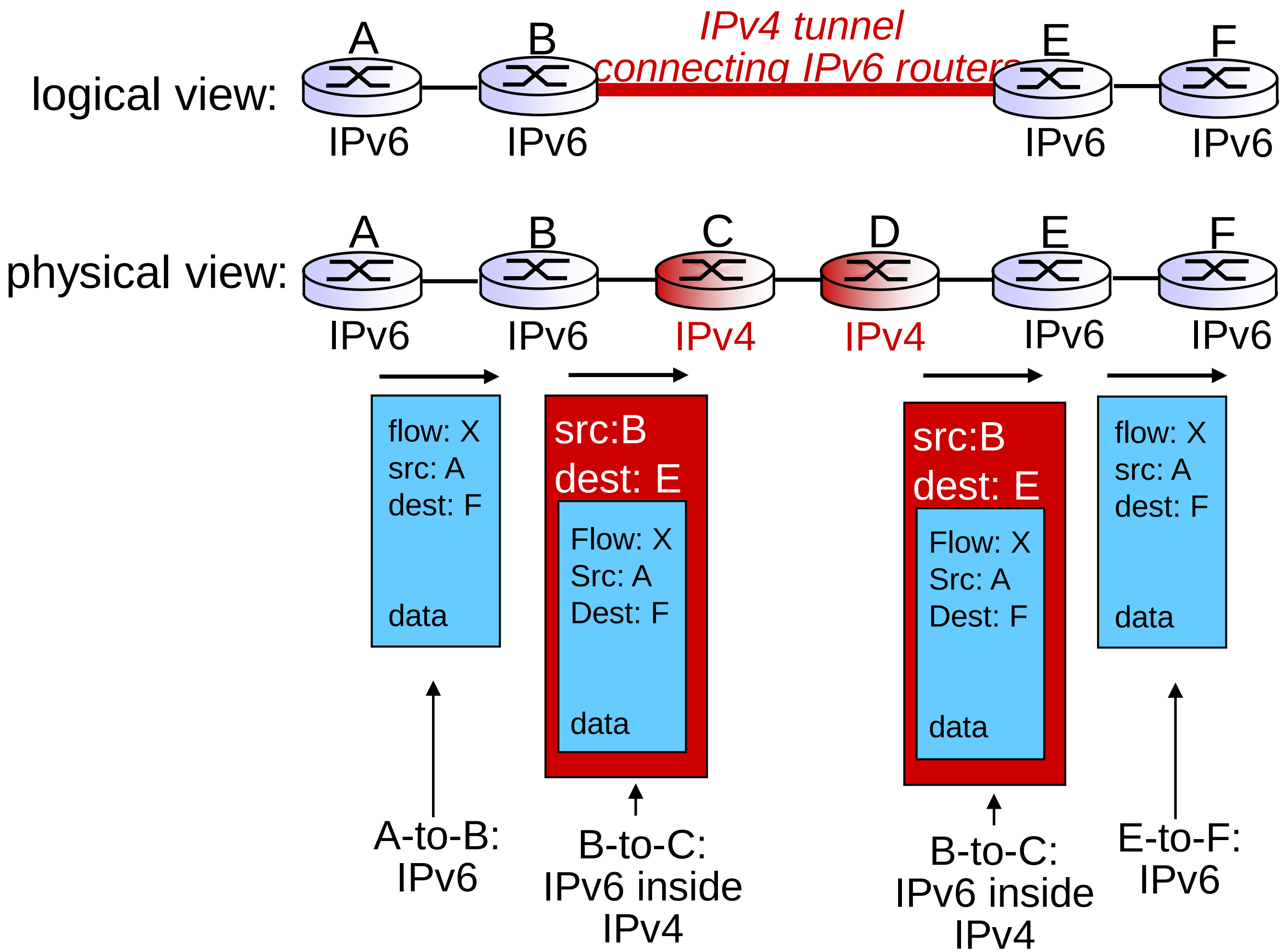


Tunneling



logical view:
A
B
E
F
IPv4 tunnel
connecting IPv6 routers
IPv6
IPv6
IPv6
IPv6
physical view:
A
B
C
D
E
F
IPv6
IPv6
IPv4
IPv4
IPv6
IPv6

Tunneling



Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

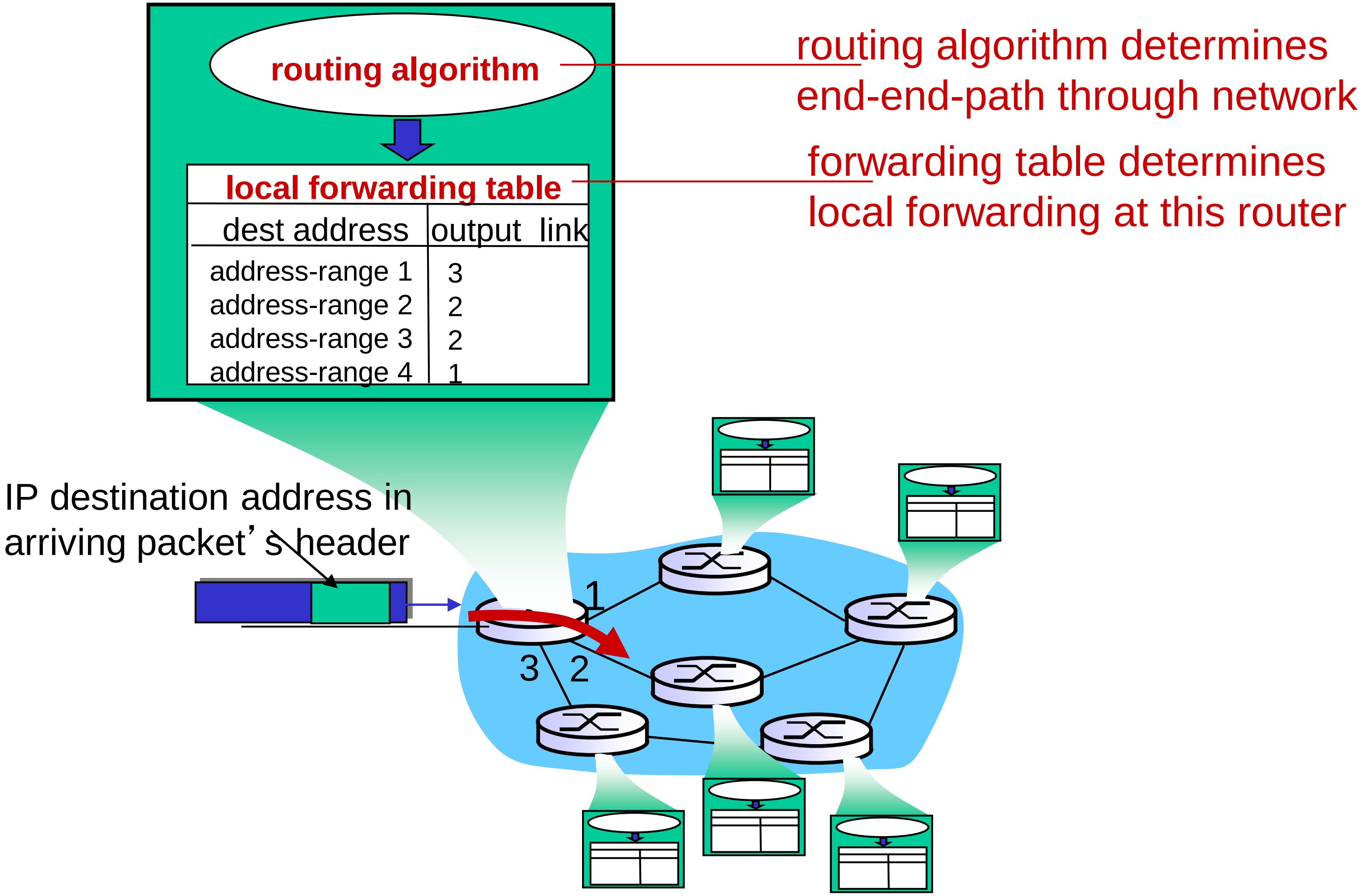
IPv6: adoption

- US National Institutes of Standards estimate [2013]:
  - ~3% of industry IP routers
  - ~11% of US gov't routers
- Long (long!) time for deployment, use
  - 20 years and counting!
  - think of application-level changes in last 20 years: WWW, Facebook, …
  - Why?

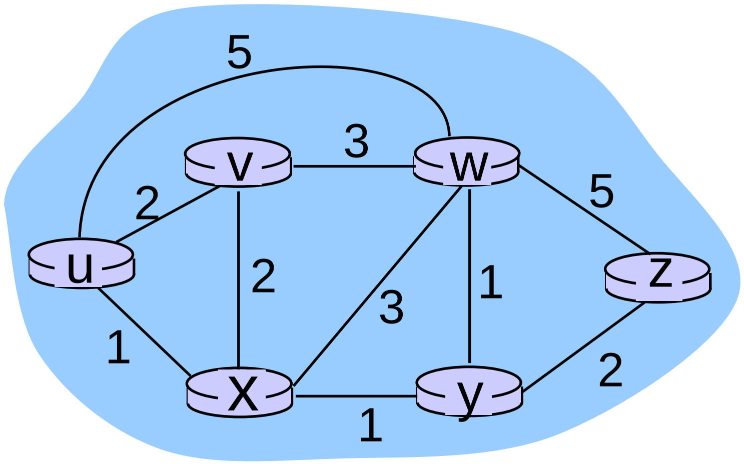
Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Interplay between routing, forwarding



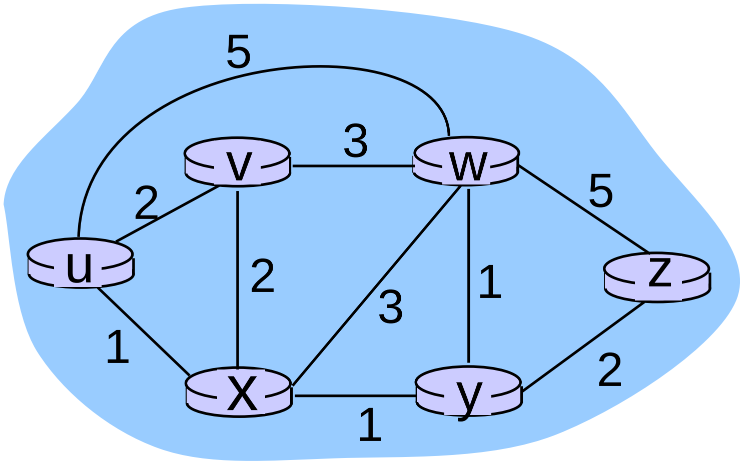
Graph abstraction



- graph: G = (N,E)
- N = set of routers = { u, v, w, x, y, z }
- E = set of links ={ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) }

aside: graph abstraction is useful in other network contexts, e.g., P2P, where *N* is set of peers and *E* is set of TCP connections

Graph abstraction: costs



- c(x,x') = cost of link (x,x')
- e.g., c(w,z) = 5
- cost could always be 1, or
- inversely related to bandwidth,
- or inversely related to congestion

Cost of path (x1, x2, x3,…, xp) = c(x1,x2) + c(x2,x3) + … + c(xp-1,xp)

Key question: what is the least-cost path between u and z ?
Routing algorithm: algorithm that finds that least cost path

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Routing algorithm classification

- Q: Global or decentralized information?
- global:
  - All routers have complete topology, link cost info
  - “Link state” algorithms
- decentralized:
  - Router knows physically-connected neighbors, link costs to neighbors
  - Iterative process of computation, exchange of info with neighbors
  - “Distance vector” algorithms

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

A Link-State Routing Algorithm

- Dijkstra's algorithm
  - Net topology, link costs known to all nodes
    - Accomplished via "link state broadcast"
    - All nodes have same info
  - Computes least cost paths from one node ("source") to all other nodes
    - Gives forwarding table for that node
  - Iterative: after k iterations, know least cost path to k dest.'s

- Notation:
  - c(x,y): link cost from node x to y; = ∞ if not direct neighbors
  - D(v): current value of cost of path from source to dest. v
  - p(v): predecessor node along path from source to v
  - N': set of nodes whose least cost path definitively known

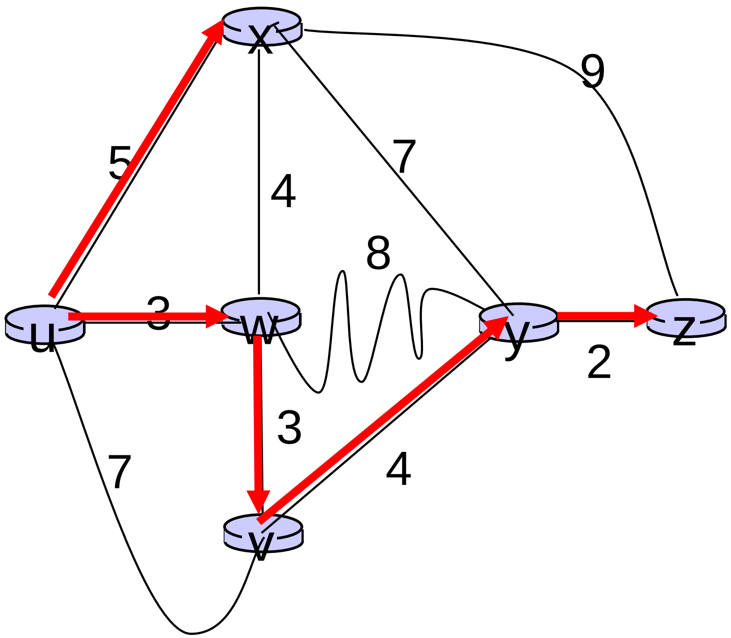
Dijsktra’s Algorithm

- Initialization:
  - N' = {u}
  - for all nodes v
  - if v adjacent to u
  - then D(v) = c(u,v)
  - else D(v) = ∞
  -
  - Loop
  - find w not in N' such that D(w) is a minimum
  - add w to N'
  - update D(v) for all v adjacent to w and not in N' :
  - D(v) = min(D(v), D(w) + c(w,v))
  - /* new cost to v is either old cost to v or known
  - shortest path cost to w plus cost from w to v */
  - until all nodes in N'

Dijkstra's algorithm: example

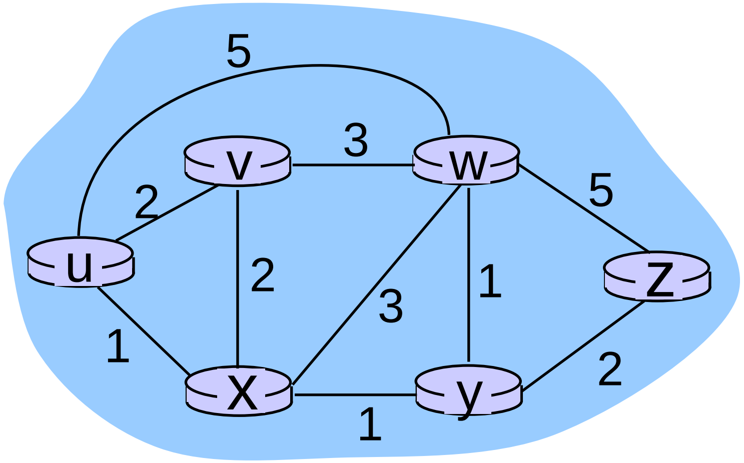
Step	N'	D(v) p(v)	D(w) p(w)	D(x) p(x)	D(y) p(y)	D(z) p(z)
0	u	7,u	3,u	5,u	∞	∞
1	uw	6,w		5,u	11,w	∞
2	uwx	6,w			11,w	14,x
3	uwxv				10,v	14,x
4	uwxvy					12,y
5	uwxvyz					

- Notes:
  - Construct shortest path tree by tracing predecessor nodes
  - Ties can exist (can be broken arbitrarily)



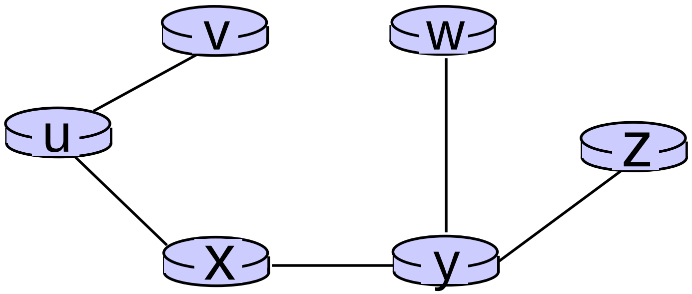
Dijkstra’s algorithm: another example

Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



Dijkstra's algorithm: example (2)

- Resulting shortest-path tree from



- Resulting forwarding table in u:

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Distance vector algorithm

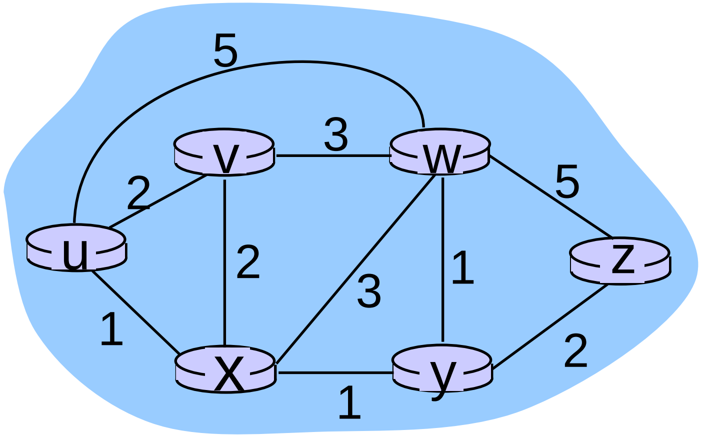
- Bellman-Ford equation (dynamic programming)
  - let
  - $d_x(y)$:= cost of least-cost path from x to y
  - Then
    - $d_x(y) = \min_v \{c(x,v) + d_v(y)\}$

cost from neighbor v to destination y

cost to neighbor v

min taken over all neighbors v of x

Bellman-Ford example



- clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$
- B-F equation says:
  - $d_u(z) = \min \{ c(u,v) + d_v(z),$
  - $c(u,x) + d_x(z),$
  - $c(u,w) + d_w(z) \}$
  - $= \min \{2 + 5,$
  - $1 + 3,$
  - $5 + 3\} = 4$
- Node achieving minimum is next
- Hop in shortest path, used in forwarding table

Distance vector algorithm

- $D_x(y)$ = estimate of least cost from x to y
  - x maintains distance vector $D_x = [D_x(y): y \in N]$
- Node x:
  - Knows cost to each neighbor v: $c(x,v)$
  - Maintains its neighbors' distance vectors. For each neighbor v, x maintains $D_v = [D_v(y): y \in N]$

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Distance vector algorithm

- Key idea:
  - From time-to-time, each node sends its own distance vector estimate to neighbors
  - When x receives new DV estimate from neighbor, it updates its own DV using B-F equation:
    - $D_x(y) \leftarrow \min_v\{c(x,v) + D_v(y)\}$ for each node $y \in N$
  - Under minor, natural conditions, the estimate $D_x(y)$ converge to the actual least cost $d_x(y)$

Distance vector algorithm

- Iterative, asynchronous: each local iteration caused by:
  - Local link cost change
  - DV update message from neighbor
- Distributed:
  - Each node notifies neighbors only when its DV changes
  - Neighbors then notify their neighbors if necessary

- Each node:

wait for (change in local link cost or msg from neighbor)

recompute estimates

if DV to any dest has changed, *notify* neighbors

node x table

from \ cost to	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

from \ cost to	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\}$
$= \min\{2+0 , 7+1\} = 2$

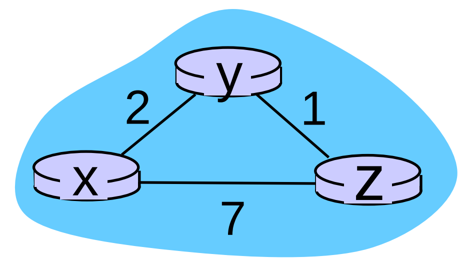
$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\}$
$= \min\{2+1 , 7+0\} = 3$

node y table

from \ cost to	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

node z table

from \ cost to	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0



time

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\}$$
$$= \min\{2+0 , 7+1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\}$$
$$= \min\{2+1 , 7+0\} = 3$$

node x table

from \ cost to	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

from \ cost to	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

from \ cost to	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

node y table

from \ cost to	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

from \ cost to	x	y	z
x	0	2	7
y	2	0	1
z	7	1	0

from \ cost to	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

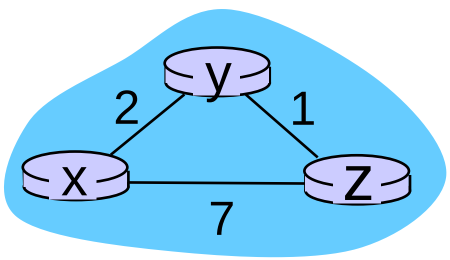
node z table

from \ cost to	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

from \ cost to	x	y	z
x	0	2	7
y	2	0	1
z	3	1	0

from \ cost to	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

time



Distance vector: link cost changes

- Link cost changes:
- Node detects local link cost change
- Updates routing info, recalculates distance vector
- If DV changes, notify neighbors

"good news travels fast"

t_0 : *y* detects link-cost change, updates its DV, informs its neighbors.

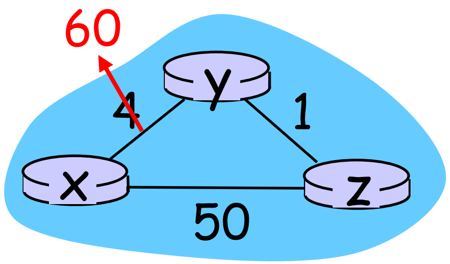
t_1 : *z* receives update from *y*, updates its table, computes new least cost to *x* , sends its neighbors its DV.

t_2 : *y* receives *z*'s update, updates its distance table. *y*'s least costs do *not* change, so *y* does *not* send a message to *z*.

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Distance vector: link cost changes

- Link cost changes:
  - Node detects local link cost change
  - Bad news travels slow - “count to infinity” problem!
  - 44 iterations before algorithm stabilizes: see text
- Poisoned reverse:
  - If Z routes through Y to get to X :
  - Z tells Y its (Z’s) distance to X is infinite (so Y won’t route to X via Z)
  - Will this completely solve count to infinity problem?



Comparison of LS and DV algorithms

- Message complexity
  - LS: With n nodes, E links, O(nE) msgs sent
  - DV: Exchange between neighbors only
    - convergence time varies
- Speed of convergence
  - LS: O(n2) algorithm requires O(nE) msgs
    - May have oscillations
  - DV: Convergence time varies
    - May be routing loops
    - Count-to-infinity problem

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Comparison of LS and DV algorithms

- Robustness: what happens if router malfunctions?
  - LS:
    - Node can advertise incorrect link cost
    - Each node computes only its own table
  - DV:
    - DV node can advertise incorrect path cost
    - Each node's table used by others
    - Error propagate thru network

Chapter 4: outline

- Introduction
- Datagram networks
- What’s inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6

- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Hierarchical routing

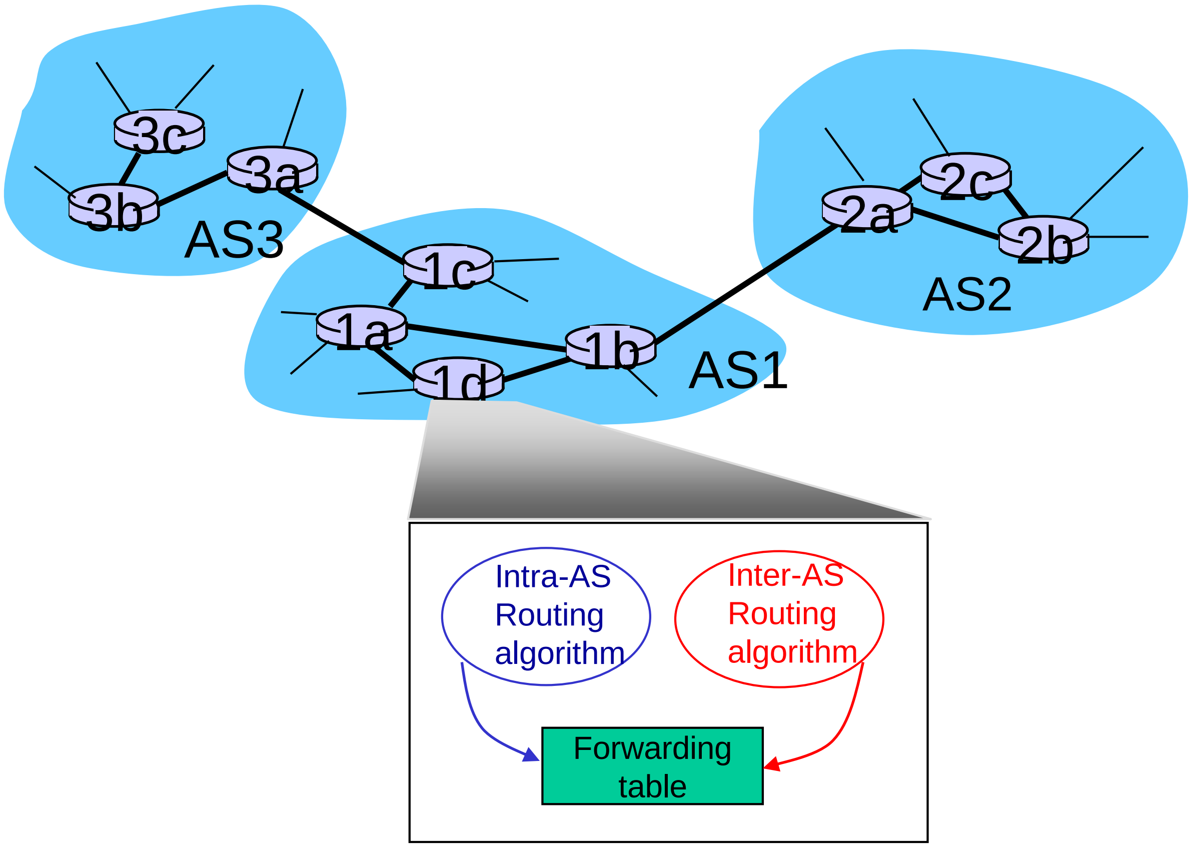
- Our routing study thus far - idealization
  - All routers identical
  - Network “flat”
  - … not true in practice
- Scale: with 600 million destinations:
  - Can’t store all dest’s in routing tables!
  - Routing table exchange would swamp links!
- Administrative autonomy
  - Internet = network of networks
  - Each network admin may want to control routing in its own network

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Hierarchical routing

- Aggregate routers into regions, "autonomous systems" (AS)
- Routers in same AS run same routing protocol
  - "Intra-AS" routing protocol
  - Routers in different AS can run different intra-AS routing protocol
- Gateway router:
  - At "edge" of its own AS
  - Has link to router in another AS

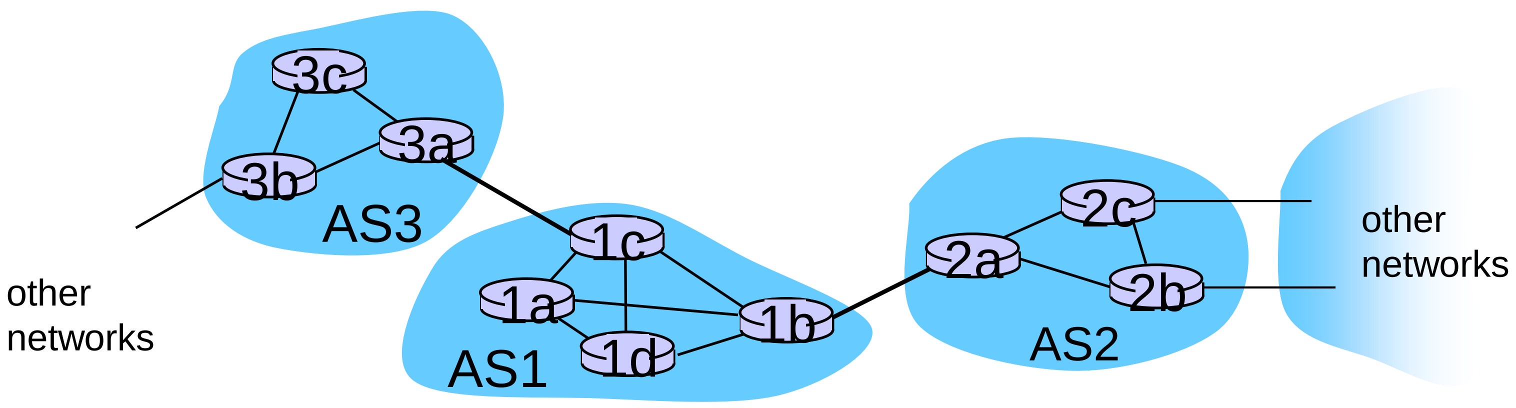
Interconnected ASes



- Forwarding table configured by both intra- and inter-AS routing algorithm
  - Intra-AS sets entries for internal dests
  - Inter-AS & intra-AS sets entries for external dests

Inter-AS tasks

- Suppose router in AS1 receives datagram destined outside of AS1:
  - Router should forward packet to gateway router, but which one?
- AS1 must:
  - Learn which dests are reachable through AS2, which through AS3
  - Propagate this reachability info to all routers in AS1
- Job of inter-AS routing!



Chapter 4: outline

- Introduction
- Datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- Routing algorithms
  - Link state
  - Distance vector
  - Hierarchical routing
- Routing in the Internet
  - RIP
  - OSPF
  - BGP

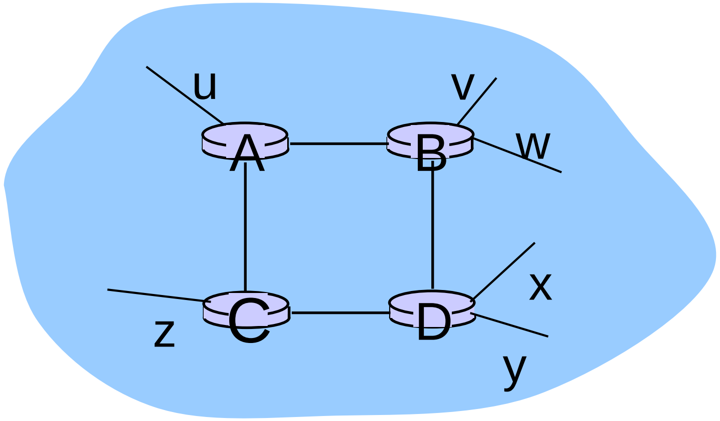
Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Intra-AS Routing

- Also known as interior gateway protocols (IGP)
- Most common intra-AS routing protocols:
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - IGRP: Interior Gateway Routing Protocol (Cisco proprietary)

RIP (Routing Information Protocol)

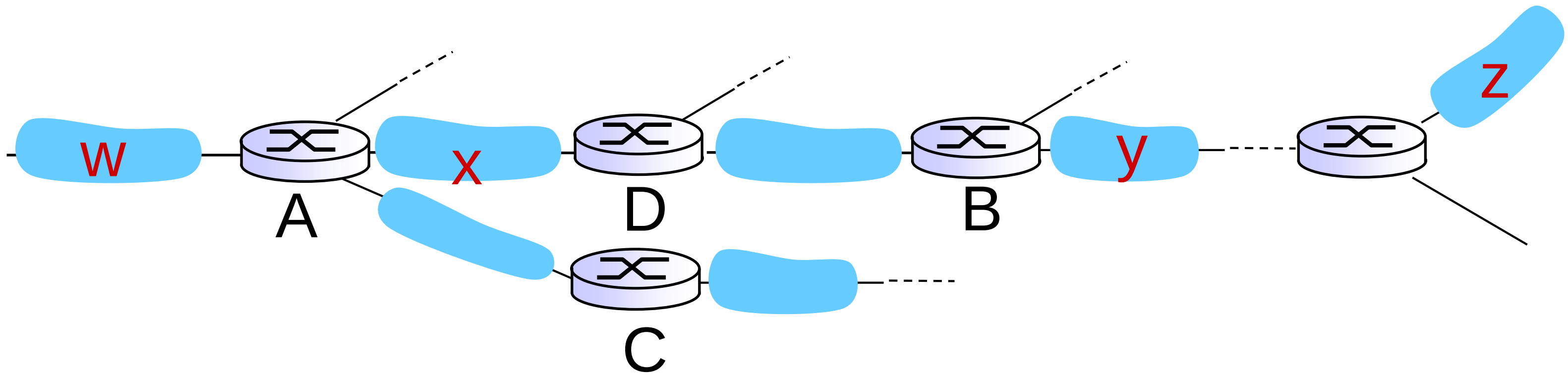
- Included in BSD-UNIX distribution in 1982
  - Distance vector algorithm
  - Distance metric: # hops (max = 15 hops), each link has cost 1
  - DVs exchanged with neighbors every 30 sec in response message (aka advertisement)
  - Each advertisement: list of up to 25 destination subnets (in IP addressing sense)



from router A to destination *subnets:*

subnet	hops
u	1
v	2
w	2
x	3
y	3
z	2

RIP: example



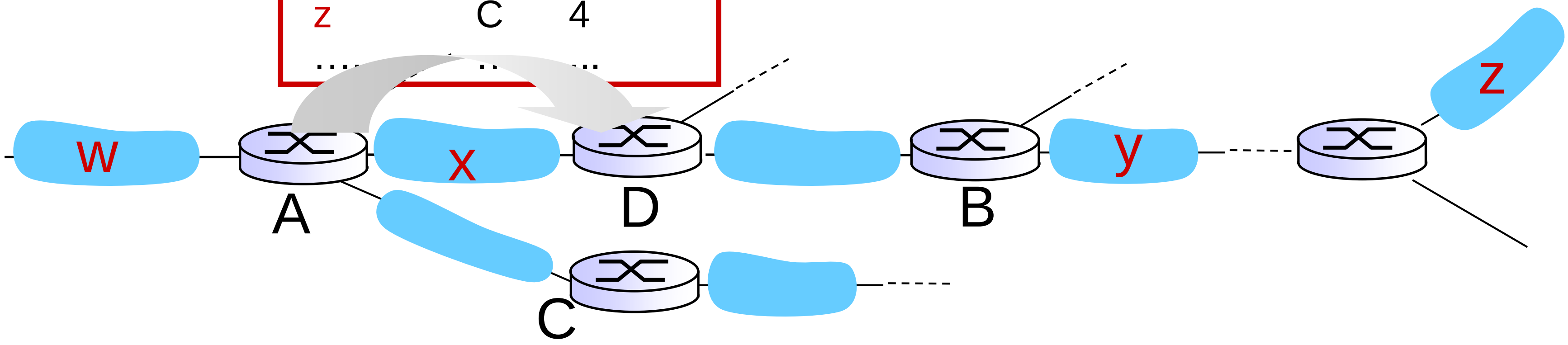
routing table in router D

destination subnet	next router	# hops to dest
w	A	2
y	B	2
z	B	7
x	--	1
….	….	….

RIP: example

A-to-D advertisement

dest	next	hops
w	-	1
x	-	1
z	C	4
….	…	...



routing table in router D

destination subnet	next router	# hops to dest
w	A	2
y	B	2
z	~~B~~ A	~~7~~ 5
x	--	1
….	….	….

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

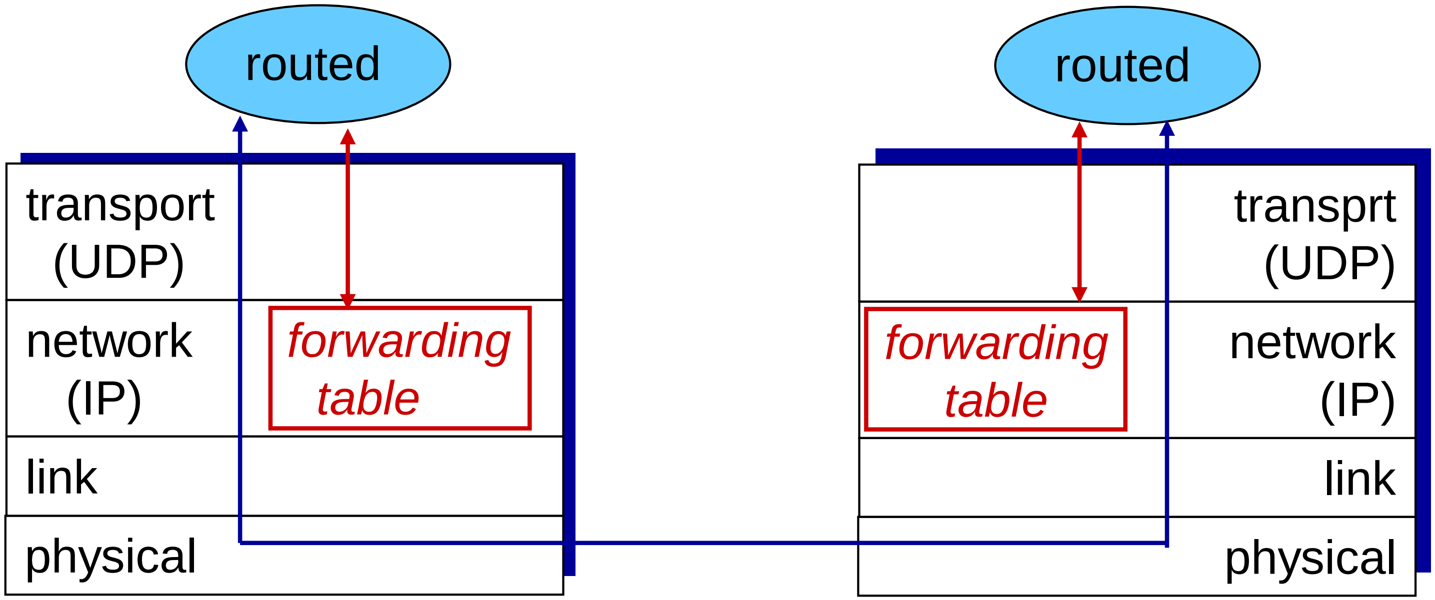
RIP: Link failure, recovery

- If no advertisement heard after 180 sec --> neighbor/link declared dead
  - Routes via neighbor invalidated
  - New advertisements sent to neighbors
  - Neighbors in turn send out new advertisements (if tables changed)
  - Link failure info quickly (?) propagates to entire net
  - Poison reverse used to prevent ping-pong loops (infinite distance = 16 hops)

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

RIP table processing

- RIP routing tables managed by application-level process called route-d (daemon)
- Advertisements sent in UDP packets, periodically repeated



OSPF (Open Shortest Path First)

- “open”: publicly available
- Uses link state algorithm
  - LS packet dissemination
  - Topology map at each node
- Route computation using Dijkstra’s algorithm
- OSPF advertisement carries one entry per neighbor
- Advertisements flooded to entire AS
  - Carried in OSPF messages directly over IP (rather than TCP or UDP

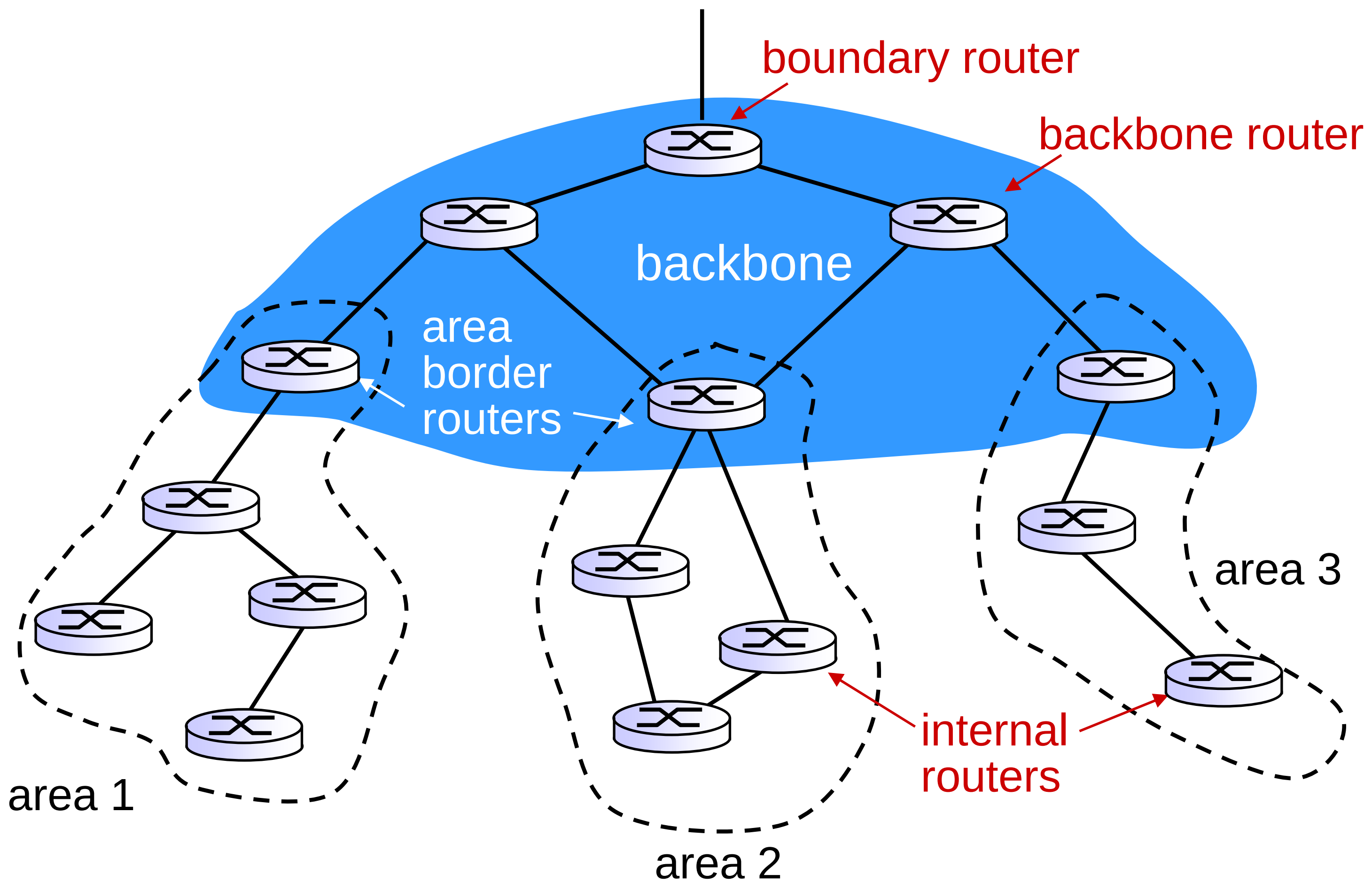
Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

OSPF “advanced” features (not in RIP)

- Security: all OSPF messages authenticated (to prevent malicious intrusion)
- Multiple same-cost paths allowed (only one path in RIP)
- For each link, multiple cost metrics for different TOS (e.g., satellite link cost set “low” for best effort ToS; high for real time ToS)
- Integrated uni- and multicast support:
- Multicast OSPF (MOSPF) uses same topology data base as OSPF
- Hierarchical OSPF in large domains.

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de

Hierarchical OSPF



Hierarchical OSPF

- Two-level hierarchy: local area, backbone.
  - Link-state advertisements only in area
  - Each nodes has detailed area topology; only know direction (shortest path) to nets in other areas.
- Area border routers: “summarize” distances to nets in own area, advertise to other Area Border routers.
- Backbone routers: run OSPF routing limited to backbone.
- Boundary routers: connect to other AS’s.

Internet inter-AS routing: BGP

- BGP (Border Gateway Protocol): the de facto inter-domain routing protocol
  - "glue that holds the Internet together"
- BGP provides each AS a means to:
  - eBGP: obtain subnet reachability information from neighboring ASs.
  - iBGP: propagate reachability information to all AS-internal routers.
  - Determine "good" routes to other networks based on reachability information and policy.
- Allows subnet to advertise its existence to rest of Internet: "I am here"

Why different Intra-, Inter-AS routing ?

- Policy:
  - Inter-AS: admin wants control over how its traffic routed, who routes through its net.
  - Intra-AS: single admin, so no policy decisions needed
- Scale:
  - Hierarchical routing saves table size, reduced update traffic
- Performance:
  - Intra-AS: can focus on performance
  - Inter-AS: policy may dominate over performance

Chapter 4: Done!

- Introduction
- Virtual circuit and datagram networks
- What's inside a router
- IP: Internet Protocol
  - Datagram format, IPv4 addressing, ICMP, IPv6

- Routing algorithms
- Link state, distance vector, hierarchical routing
- Routing in the Internet
  - RIP, OSPF, BGP

- Understand principles behind network layer services:
  - Network layer service models, forwarding versus routing how a router works, routing (path selection), broadcast, multicast
- Instantiation, implementation in the Internet

Prof. Dr. rer. nat. Andreas Berl – www.andreas-berl.de