

Network layer overview

- 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

Chapter 4: outline

4.1 introduction

4.2 virtual circuit and datagram networks

4.3 what's inside a router

4.4 IP: Internet Protocol

- datagram format
- IPv4 addressing
- ICMP
- IPv6

4.5 routing algorithms

- link state
- distance vector
- hierarchical routing

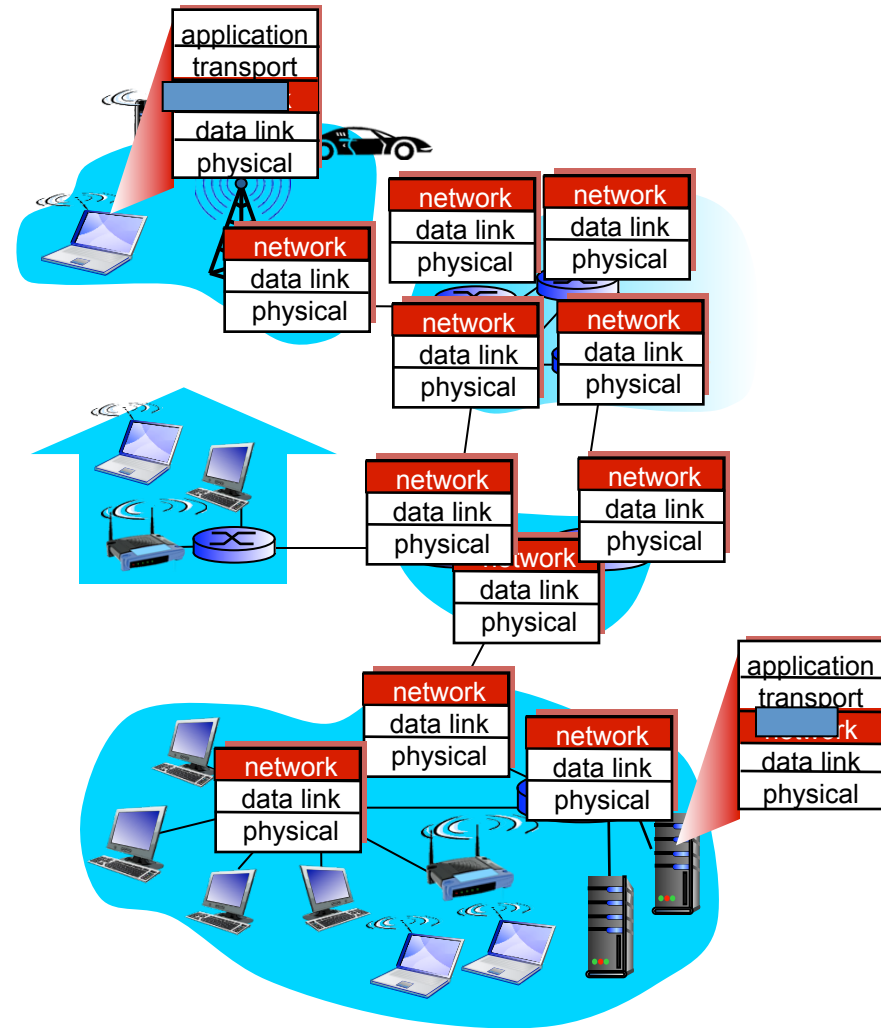
4.6 routing in the Internet

- RIP
- OSPF
- BGP

4.7 broadcast and multicast routing

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



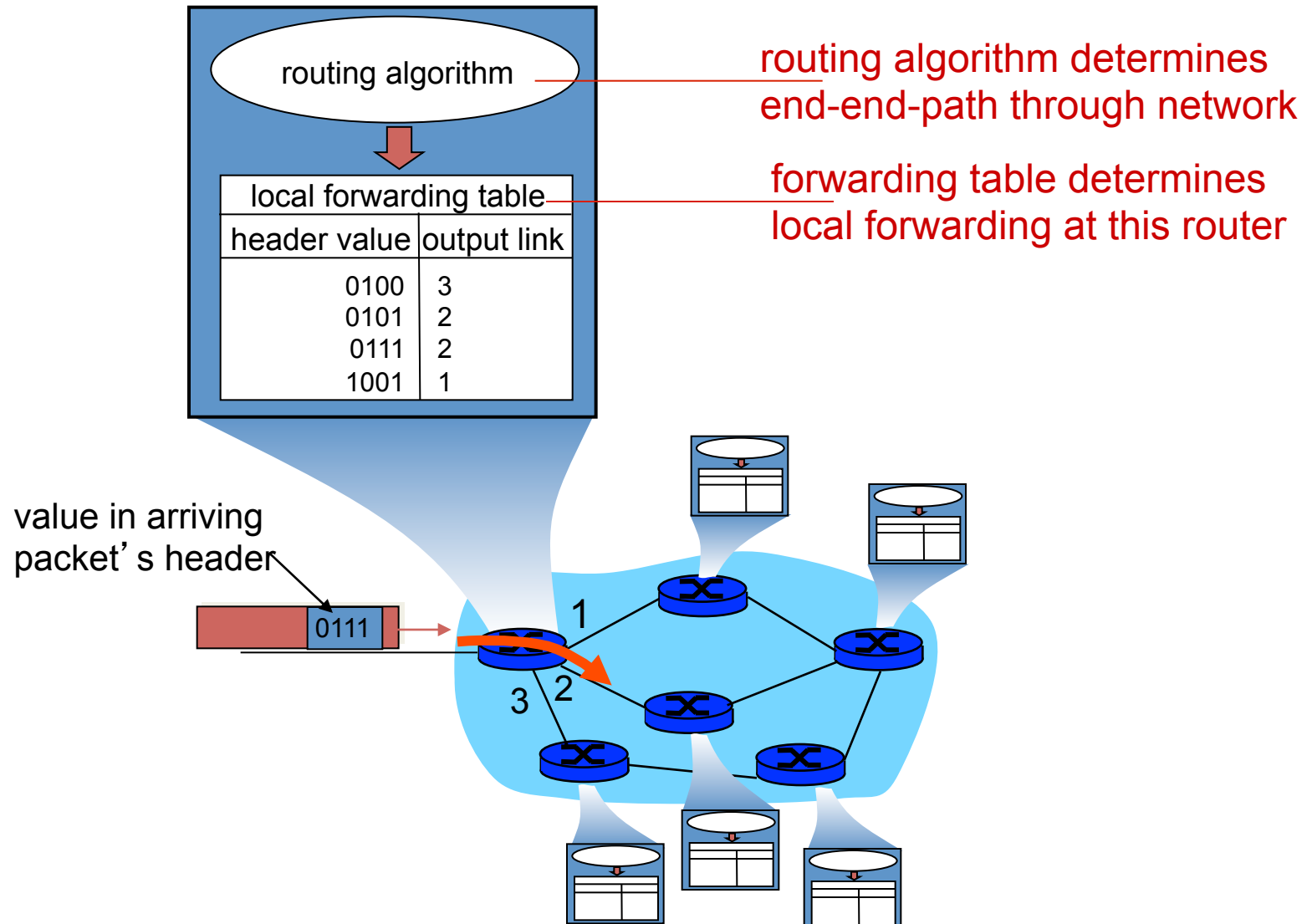
Two key network-layer functions

- *forwarding*: move packets from router's input to appropriate router output
- *routing*: determine route taken by packets from source to dest.
 - *routing algorithms*

analogy:

- ❖ *routing*: process of planning trip from source to dest
- ❖ *forwarding*: process of getting through single interchange

Interplay between routing and forwarding



Connection setup

- 3rd important function in *some* network architectures:
 - ATM, frame relay, X.25
- before datagrams flow, two end hosts *and* intervening routers establish virtual connection
 - routers get involved
- network vs transport layer connection service:
 - *network*: between two hosts (may also involve intervening routers in case of VCs)
 - *transport*: between two processes

Network service model

Q: What *service model* for “channel” transporting datagrams from sender to receiver?

example services for individual datagrams:

- guaranteed delivery
- guaranteed delivery with less than 40 msec delay

example services for a flow of datagrams:

- in-order datagram delivery
- guaranteed minimum bandwidth to flow
- restrictions on changes in inter-packet spacing

Chapter 4: outline

4.1 introduction

4.2 virtual circuit and datagram networks

4.3 what's inside a router

4.4 IP: Internet Protocol

- datagram format
- IPv4 addressing
- ICMP
- IPv6

4.5 routing algorithms

- link state
- distance vector
- hierarchical routing

4.6 routing in the Internet

- RIP
- OSPF
- BGP

4.7 broadcast and multicast routing

Connection, connection-less service

- *datagram* network provides network-layer *connectionless* service
- *virtual-circuit* network provides network-layer *connection* service
- analogous to TCP/UDP connection-oriented / connectionless transport-layer services, but:
 - *service*: host-to-host
 - *no choice*: network provides one or the other
 - *implementation*: in network core

Select a correct statement

- A. UDP is connection-oriented
- B. TCP is connectionless
- C. Datagram network is connectionless
- D. Virtual circuit network is connection-oriented
- E. C and D

Virtual circuits

“source-to-dest path behaves much like telephone circuit”

- performance-wise
- network actions along source-to-dest path

- call setup, teardown for each call *before* data can flow
- each packet carries VC identifier (not destination host address)
- every router on source-dest path maintains “state” for each passing connection
- link, router resources (bandwidth, buffers) may be *allocated* to VC (dedicated resources = predictable service)

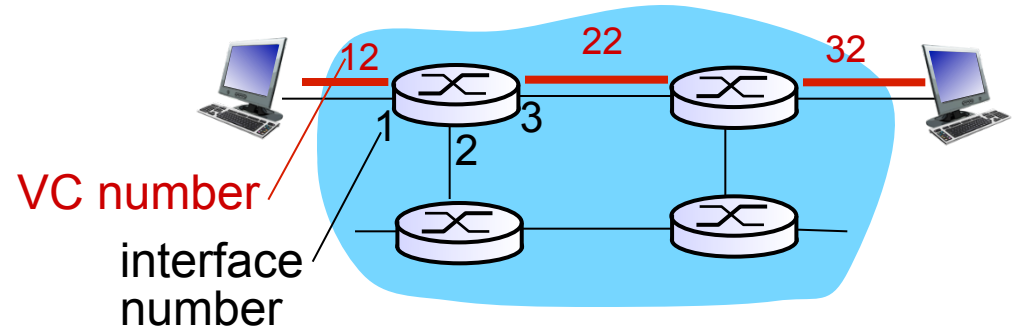
VC implementation

a VC consists of:

1. *path* from source to destination
 2. *VC numbers*, one number for each link along path
 3. *entries in forwarding tables* in routers along path
- packet belonging to VC carries VC number (rather than dest address)
 - VC number can be changed on each link.
 - new VC number comes from forwarding table

VC forwarding table

*forwarding table in
northwest router:*



Incoming interface	Incoming VC #	Outgoing interface	Outgoing VC #
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
...

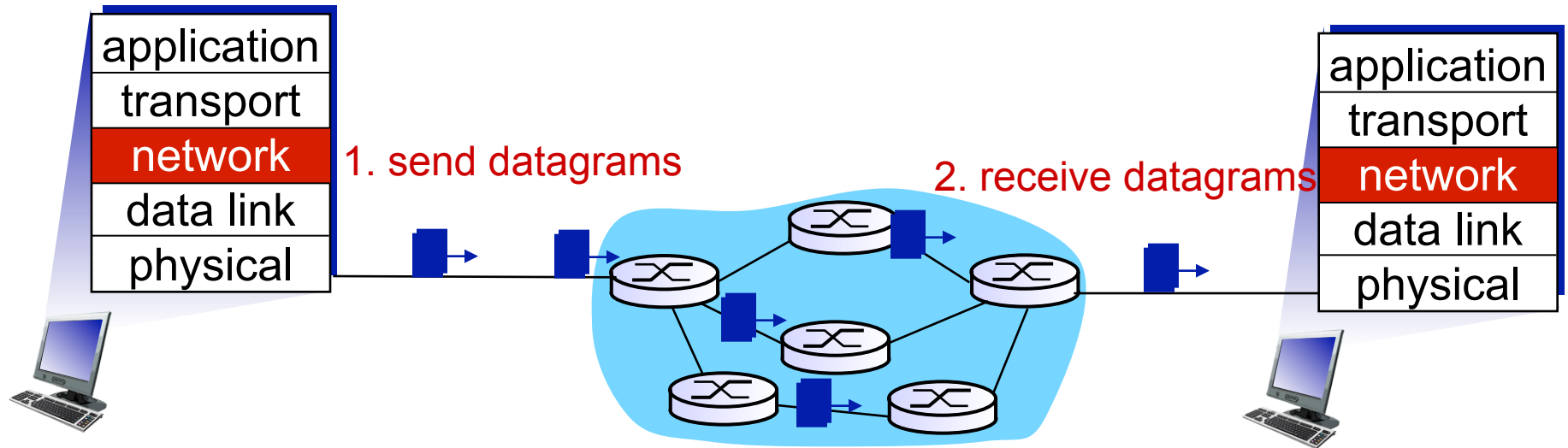
VC routers maintain connection state information!

Why doesn't the Internet use Virtual Circuit?

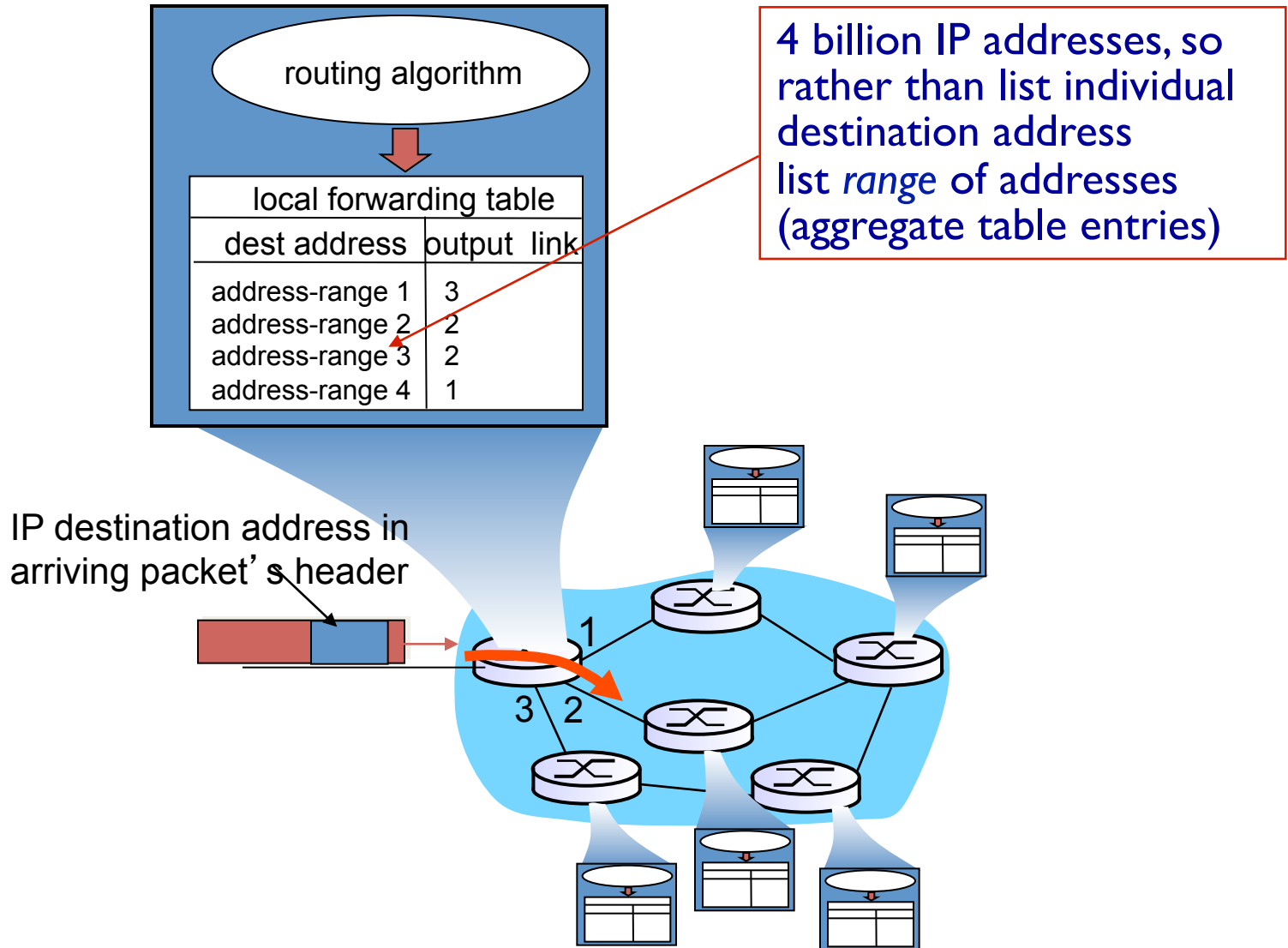
- A. It's too complicated to implement
- B. It consumes too much resources
- C. A and B
- D. Other reasons

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



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?

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?

Select Link interface for incoming datagram address of 1110 1000 1010

Destination Address Range	Link interface
1*** **	A
11** **	B
111* **	C
otherwise	D

E: A, B, and C

Datagram or VC network: why?

Internet (datagram)

- data exchange among computers
 - “elastic” service, no strict timing req.
- many link types
 - different characteristics
 - uniform service difficult
- “smart” end systems (computers)
 - can adapt, perform control, error recovery
 - **simple inside network, complexity at “edge”**

ATM (VC)

- evolved from telephony
- human conversation:
 - strict timing, reliability requirements
 - need for guaranteed service
- “dumb” end systems
 - telephones
 - **complexity inside network**

Next class: Midterm exam

Friday, March 6

- Next lecture Monday March 9
 - Routers and IP
 - Readings 4.3-4.4.2