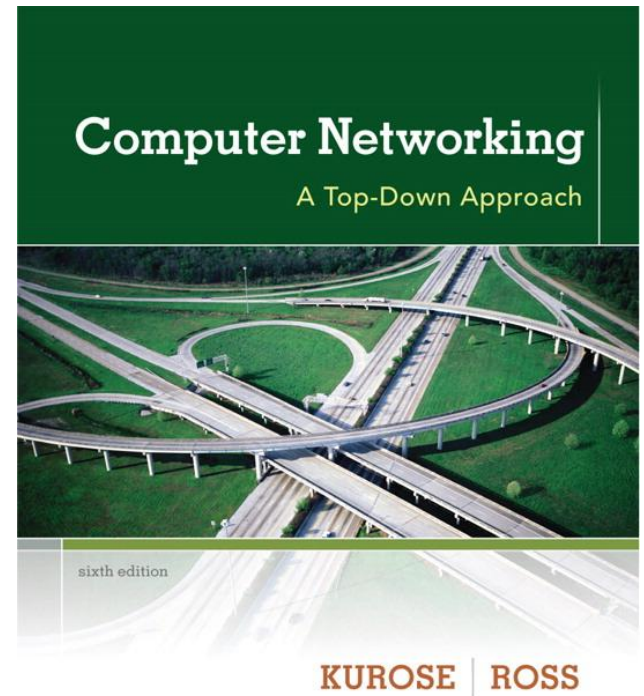# Chapter 4: network layer

chapter goals:

❖ 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

The Book

*Start Reading
Chapter 4 Now*

Computer Networking
A Top-Down Approach

sixth edition

KUROSE | ROSS

Chapter 4: outline

Network layer

❖ transports segments 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
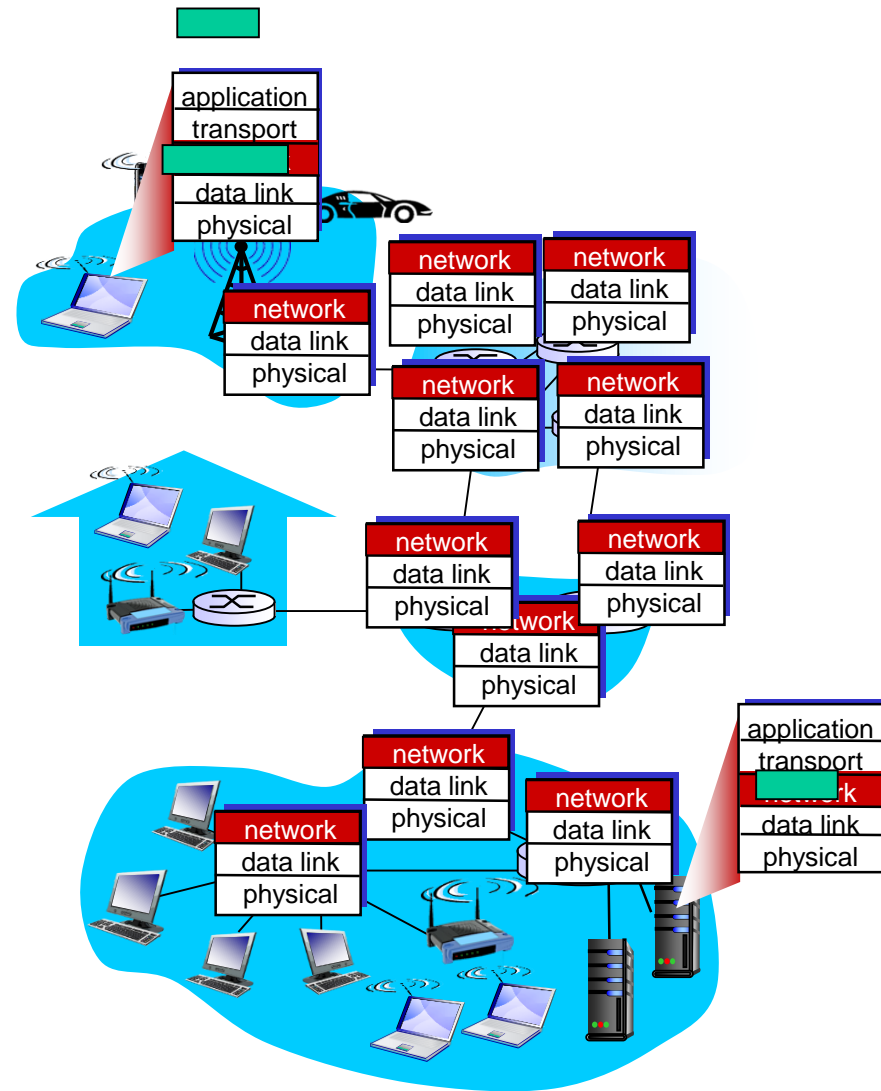
Two key network-layer functions

❖ *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

routing algorithm

local forwarding table

header value	output link
0100	3
0101	2
0111	2
1001	1

routing algorithm determines
end-end-path through network

forwarding table determines
local forwarding at this router

value in arriving
packet's header

0111

1

3 2
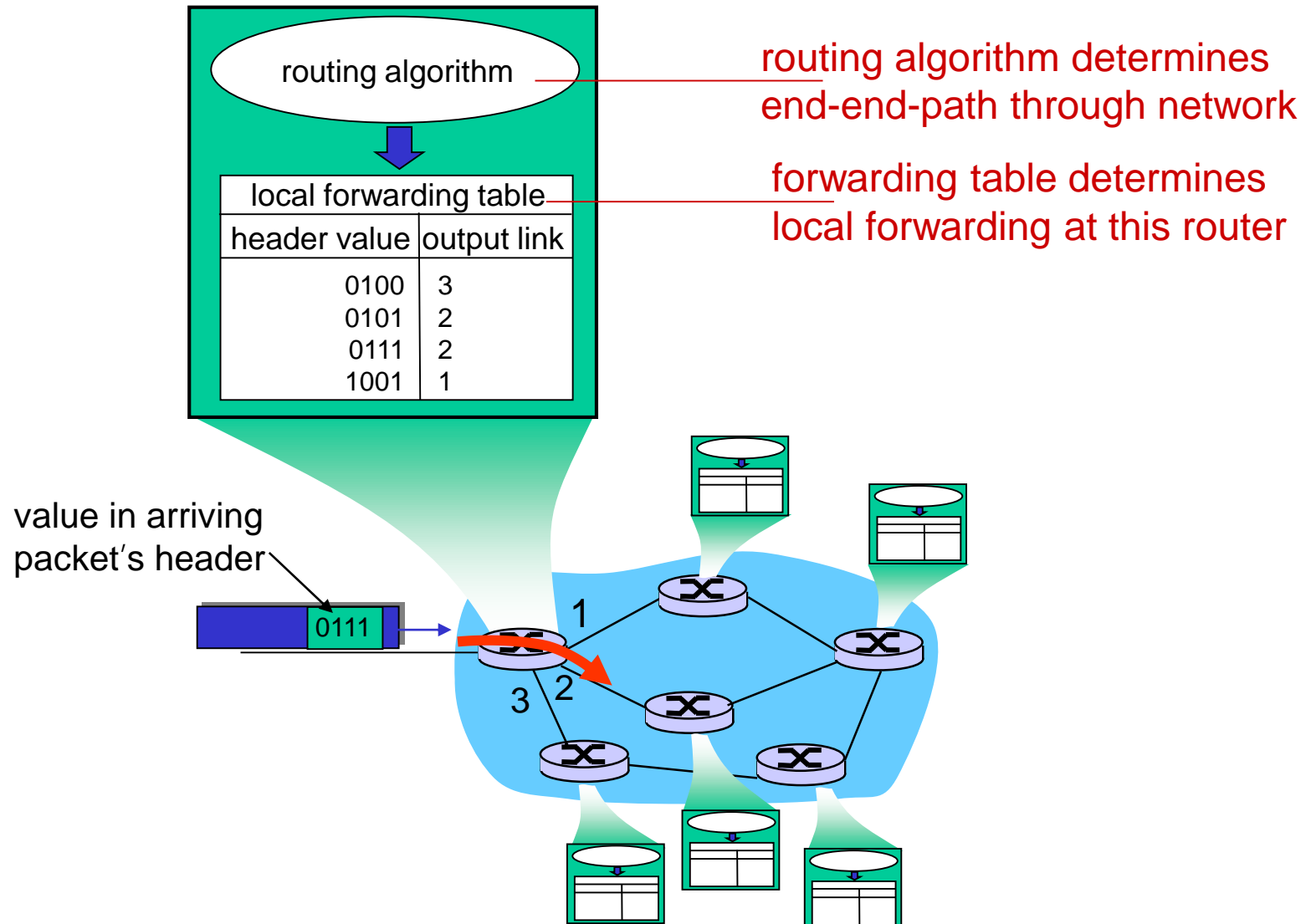
Connection setup

❖ 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 a flow
❖ restrictions on changes in inter-packet spacing (controls jitter)

Network layer service models:

Network Architecture	Service Model	Guarantees ?				Congestion feedback
		Bandwidth	Loss	Order	Timing	
Internet	best effort	none	no	no	no	no (inferred via loss)
ATM	CBR	constant rate	yes	yes	yes	no congestion
ATM	VBR	guaranteed rate	yes	yes	yes	no congestion
ATM	ABR	guaranteed minimum	no	yes	no	yes
ATM	UBR	none	no	yes	no	no

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 connecton-oriented / connectionless transport-layer services, but:

   ▪ *service:* host-to-host

   ▪ *no choice:* network provides one or the other

   ▪ *implementation:* in network core

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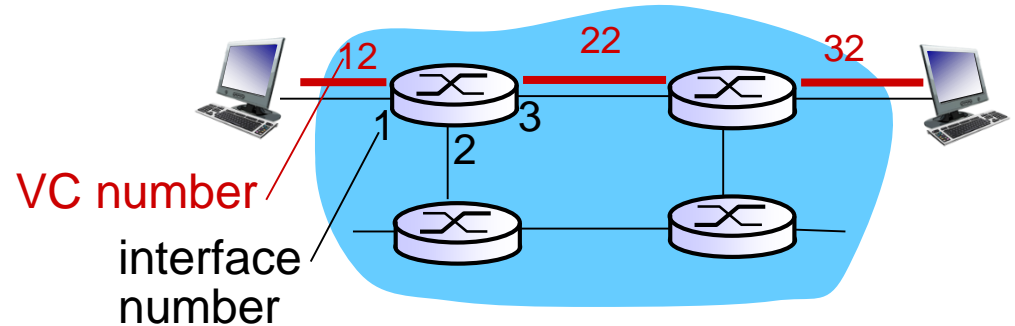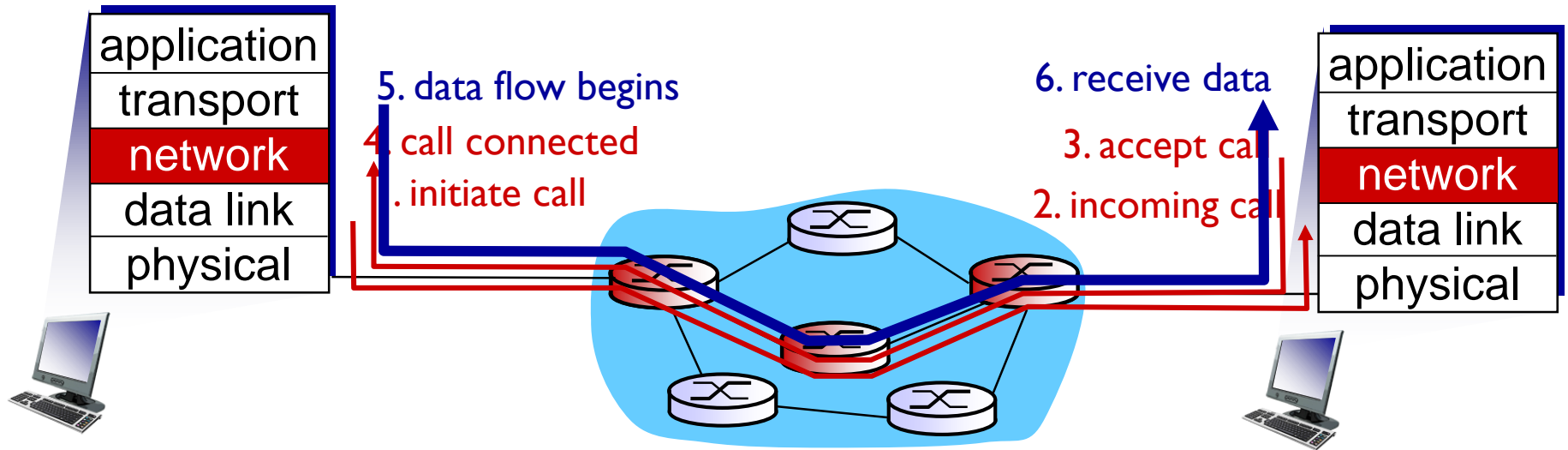
Incoming interface	Incoming VC #	Outgoing interface	Outgoing VC #
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
…	…	…	…

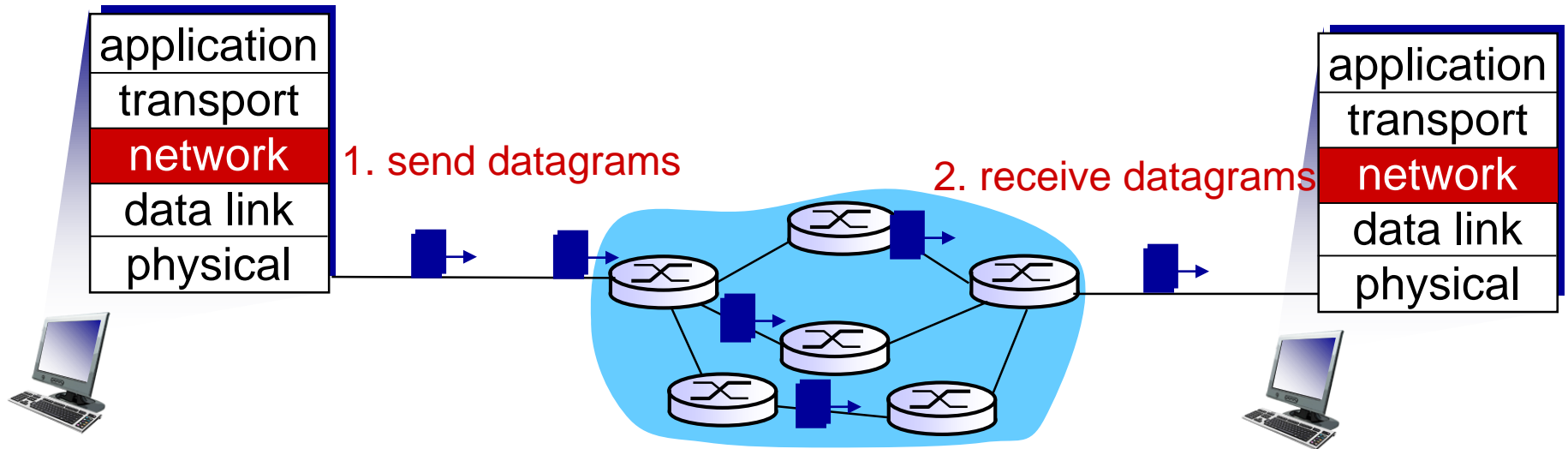*VC routers maintain connection state information!*

Virtual circuits: signaling protocols

❖ used to setup, maintain teardown VC
❖ used in ATM, frame-relay, X.25
❖ not used in today's Internet

5. data flow begins
4. call connected
1. initiate call

6. receive data
3. accept call
2. incoming call

application
transport
network
data link
physical

application
transport
network
data link
physical

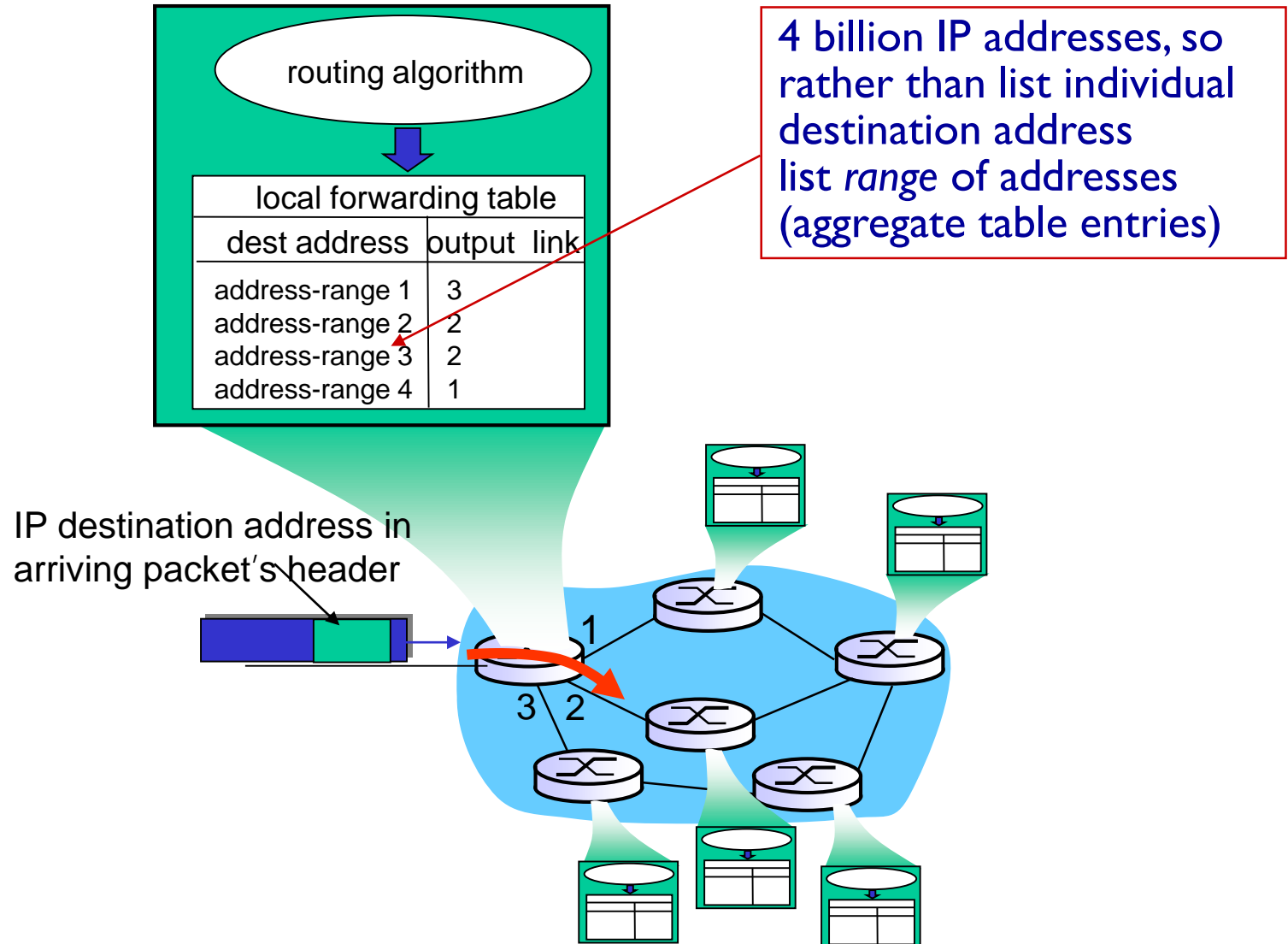# Datagram networks

❖ 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

application
transport
network
data link
physical

1. send datagrams

2. receive datagrams

application
transport
network
data link
physical

Datagram forwarding table

routing algorithm

local forwarding table

dest address	output link
address-range 1	3
address-range 2	2
address-range 3	2
address-range 4	1

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

IP destination address in
arriving packet's header

1

3   2

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, because it's more specific.

Destination Address Range Prefix	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?

Datagram vs VC network: origins

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*

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