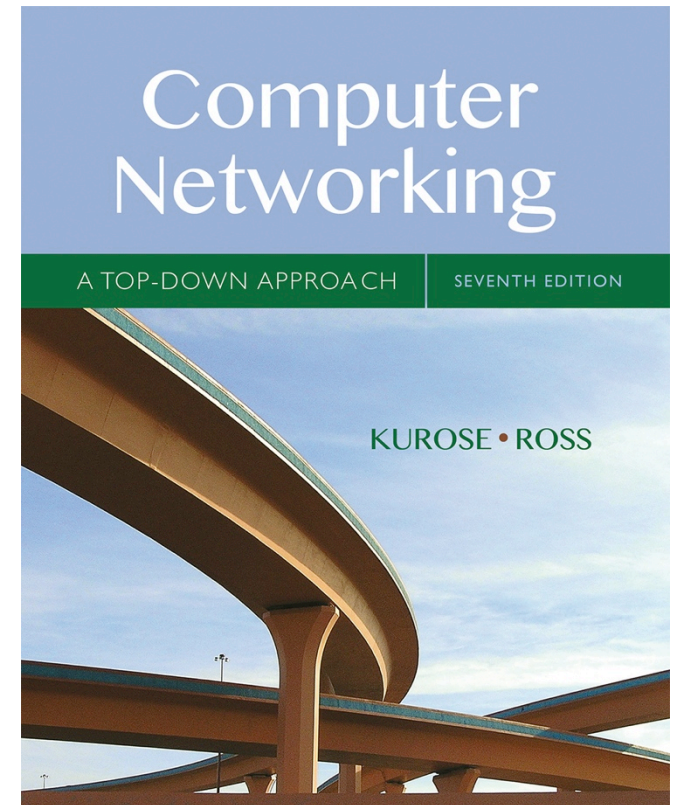# Chapter 4
Network Layer: The Data Plane

A note on the use of these Powerpoint slides:
We're making these slides freely available to all (faculty, students, readers).
They're in PowerPoint form so you see the animations; and can add, modify,
and delete slides (including this one) and slide content to suit your needs.
They obviously represent a *lot* of work on our part. In return for use, we only
ask the following:

- If you use these slides (e.g., in a class) that you mention their source
  (after all, we'd like people to use our book!)
- If you post any slides on a www site, that you note that they are adapted
  from (or perhaps identical to) our slides, and note our copyright of this
  material.

Thanks and enjoy! JFK/KWR

Computer Networking: A Top Down Approach

7th edition
Jim Kurose, Keith Ross
Pearson/Addison Wesley
April 2016

Chapter 4: outline

4.1 Overview of Network layer
- data plane
- control plane

4.2 What's inside a router

4.3 IP: Internet Protocol
- datagram format
- fragmentation
- IPv4 addressing
- network address translation
- IPv6

4.4 Generalized Forward and SDN
- match
- action
- OpenFlow examples of match-plus-action in action

Chapter 4: network layer

chapter goals:

- understand principles behind network layer services, focusing on data plane:
  - network layer service models
  - forwarding versus routing
  - how a router works
  - generalized forwarding
- instantiation, implementation in the Internet

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
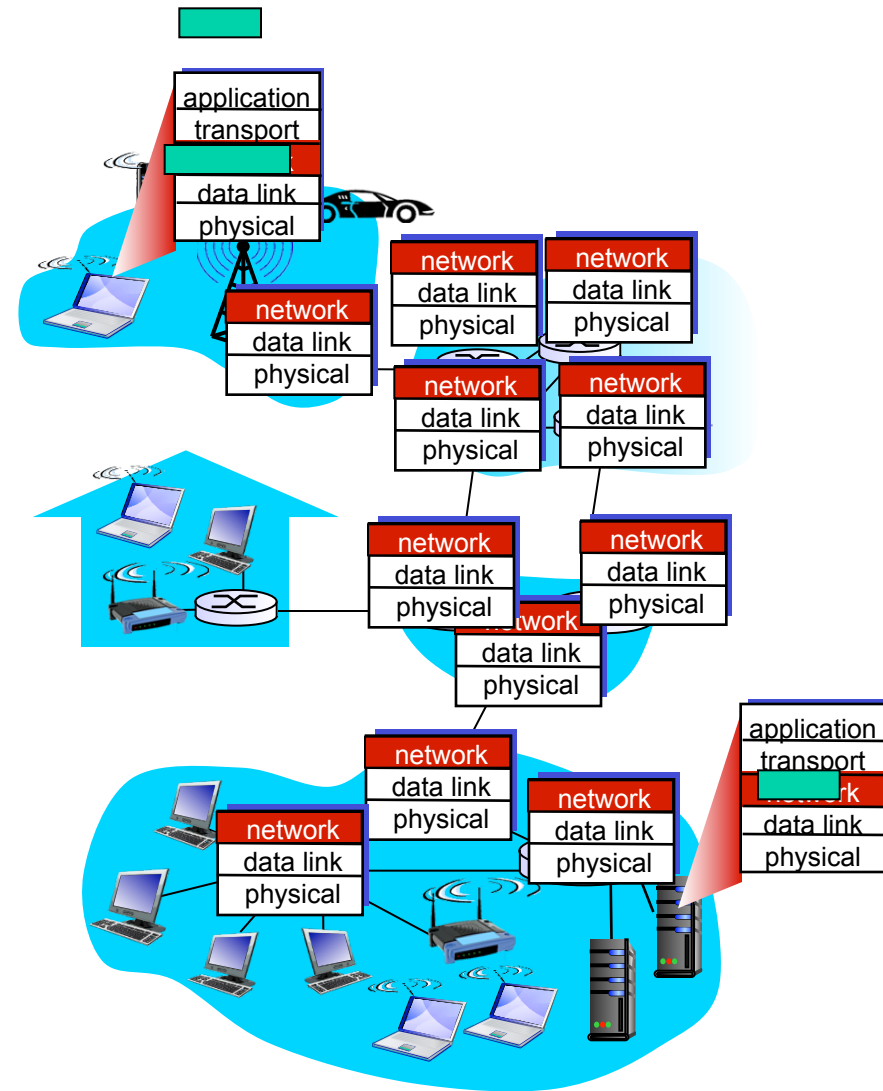
Two key network-layer functions

network-layer functions:

- *forwarding:* move packets from router's input to appropriate router output

- *routing:* determine route taken by packets from source to destination
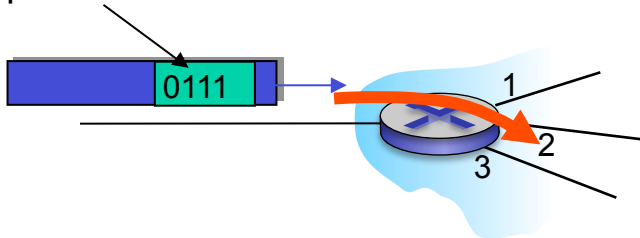 - *routing algorithms*

analogy: taking a trip

- *forwarding:* process of getting through single interchange

- *routing:* process of planning trip from source to destination

Network layer: data plane, control plane

Data plane

- local, per-router function
- determines how datagram arriving on router input port is forwarded to router output port
- forwarding function
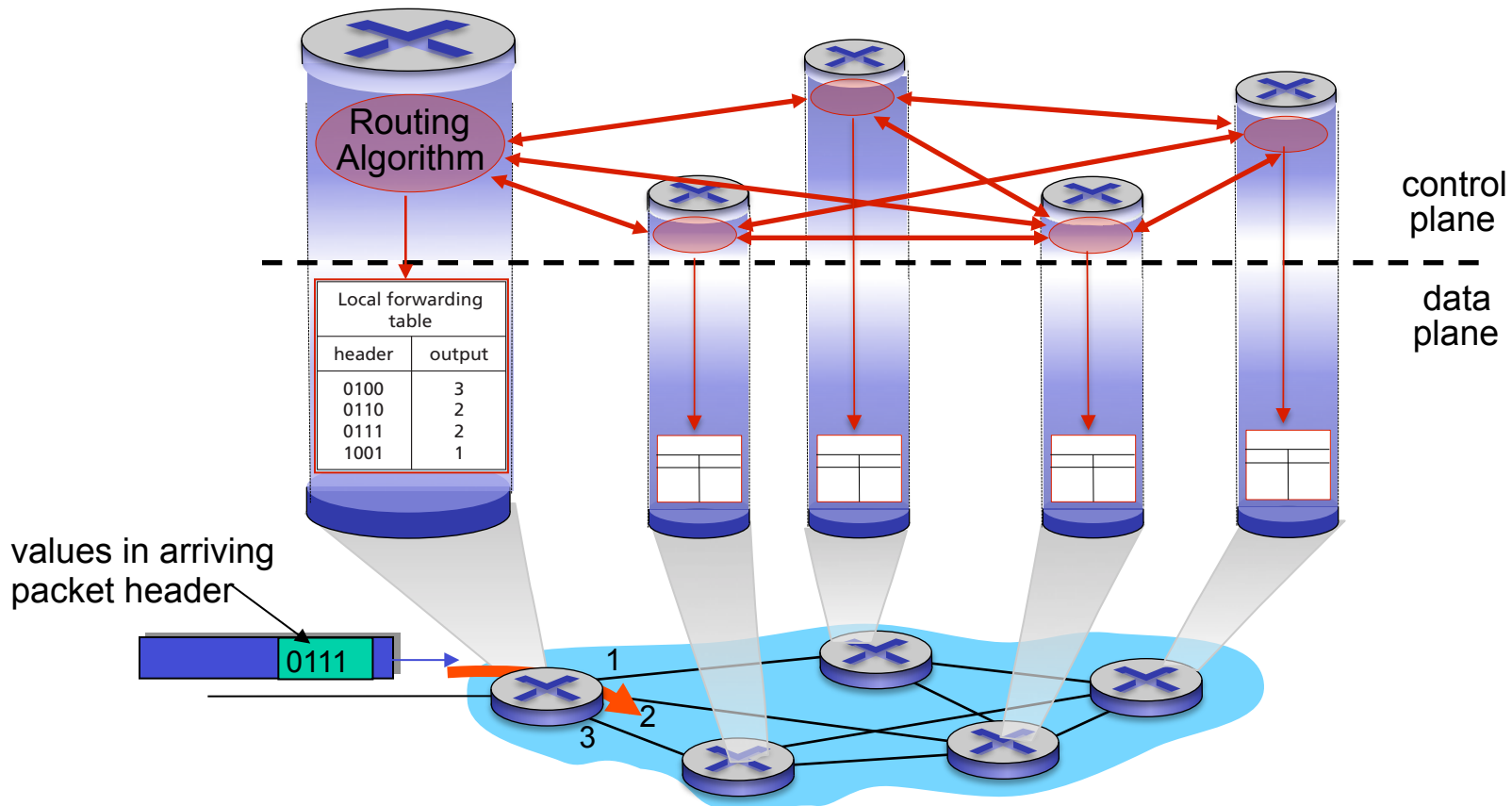
values in arriving packet header

0111

1
2
3

Control plane

- network-wide logic
- determines how datagram is routed among routers along end-end path from source host to destination host
- two control-plane approaches:
  - *traditional routing algorithms:* implemented in routers
 - *software-defined networking (SDN)*: implemented in (remote) servers

Per-router control plane

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

Routing Algorithm

Local forwarding table	
header	output
0100	3
0110	2
0111	2
1001	1

control plane

data plane

values in arriving packet header

0111

1
2
3

Logically centralized control plane

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

Remote Controller

control plane

data plane

CA

CA

CA

CA

CA

values in arriving packet header

0111

1

2

3
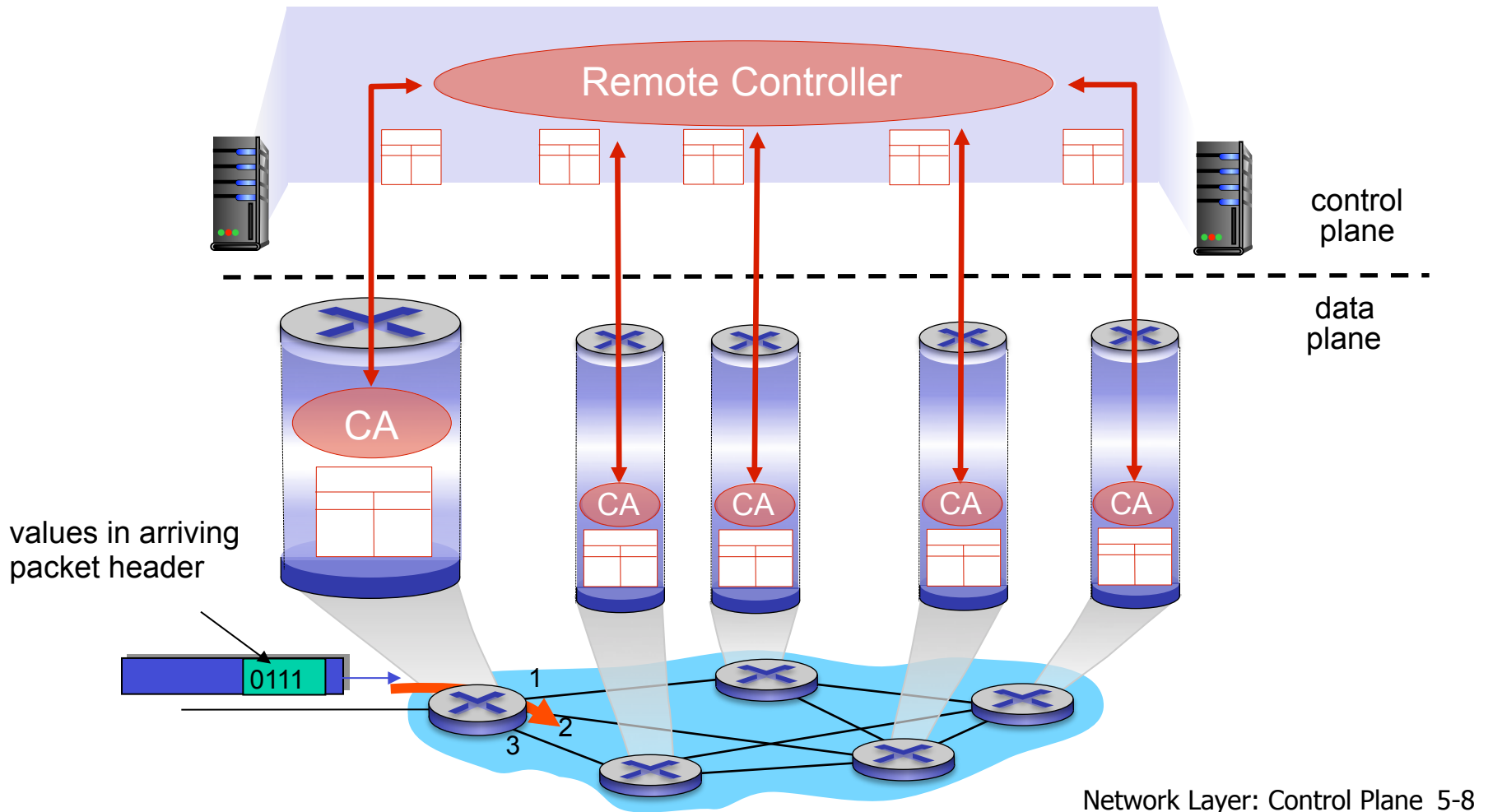
Network service model

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

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 Overview of Network layer
- data plane
- control plane
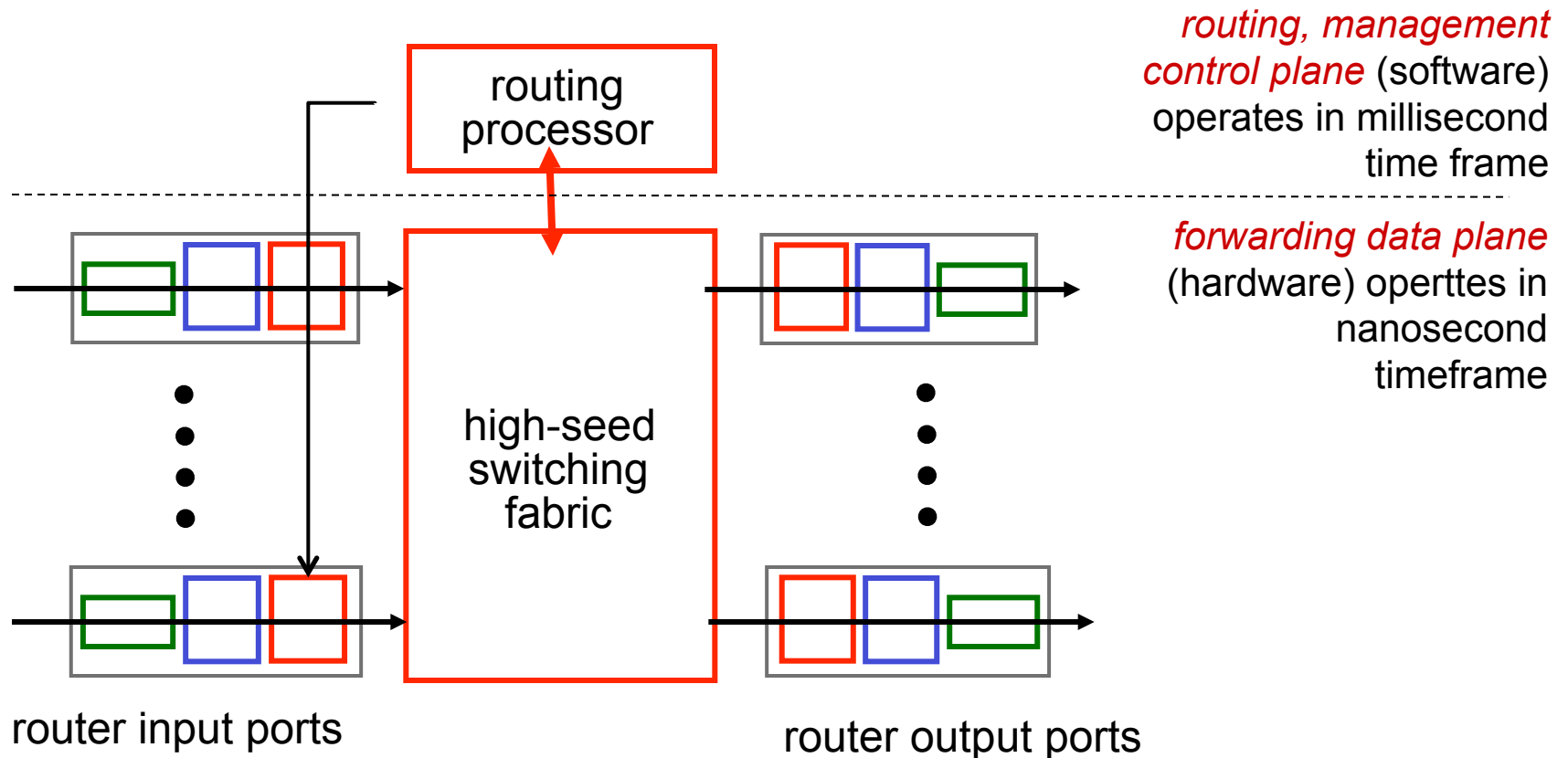
4.2 What's inside a router

4.3 IP: Internet Protocol
- datagram format
- fragmentation
- IPv4 addressing
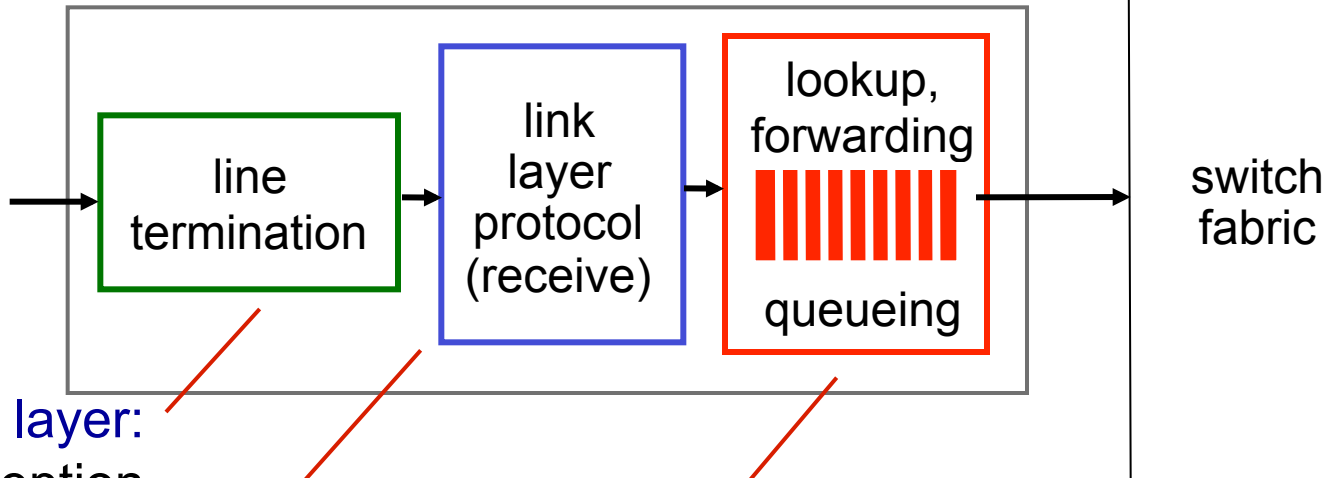- network address translation
- IPv6

4.4 Generalized Forward and SDN
- match
- action
- OpenFlow examples of match-plus-action in action

Router architecture overview

- high-level view of generic router architecture:

routing, management control plane (software) operates in millisecond time frame

forwarding data plane (hardware) operttes in nanosecond timeframe

routing processor

high-seed switching fabric

router input ports

router output ports
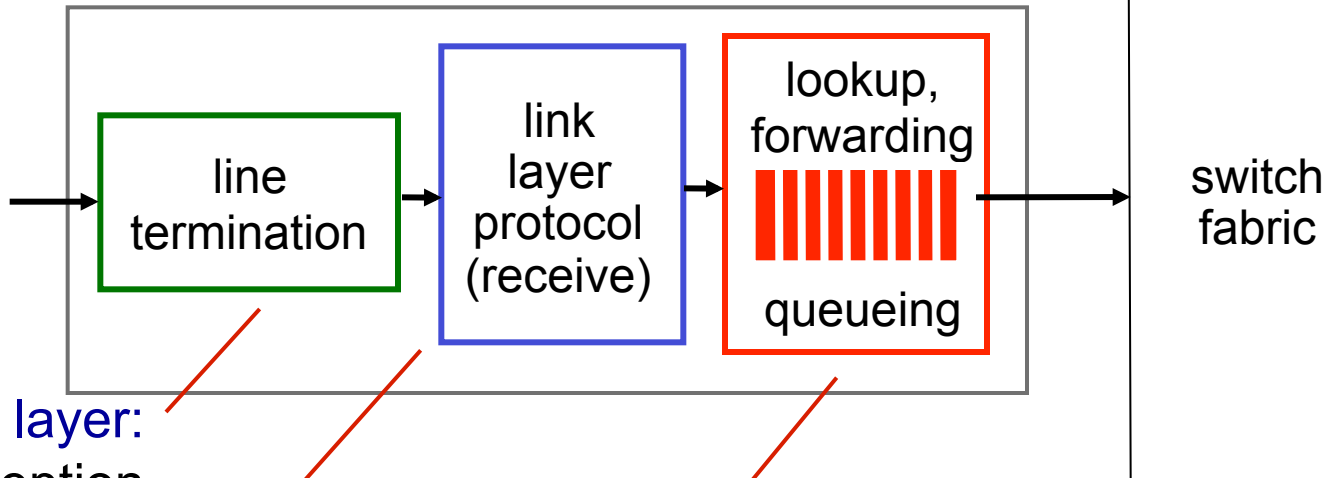
Input port functions

physical layer:
bit-level reception

data link layer:
  e.g., Ethernet
  see chapter 5

decentralized switching:

- using header field values, 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

Input port functions

physical layer:
bit-level reception

data link layer:
e.g., Ethernet
see chapter 5

decentralized switching:

- using header field values, lookup output port using forwarding table in input port memory ("*match plus action*")

- *destination-based forwarding:* forward based only on destination IP address (traditional)

- *generalized forwarding:* forward based on any set of header field values

Destination-based forwarding

Destination Address Range	Link Interface
forwarding table	
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?

Longest prefix matching

- we'll see *why* longest prefix matching is used shortly, when we study addressing

- longest prefix matching: often performed using ternary content addressable memories (TCAMs)
  - *content addressable:* present address to TCAM: retrieve address in one clock cycle, regardless of table size
  - Cisco Catalyst: can up ~1M routing table entries in TCAM

Switching fabrics

- transfer packet from input buffer to appropriate output buffer

- switching rate: rate at which packets can be transfer 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

memory   bus   crossbar
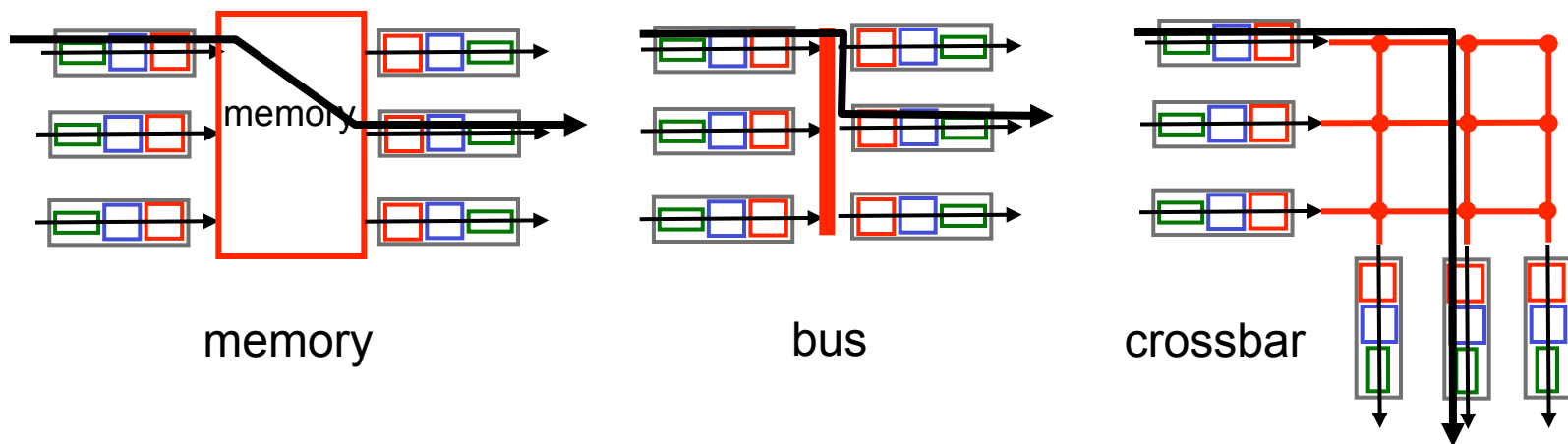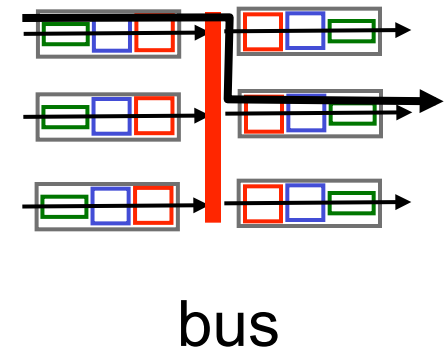
Switching via memory

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)

| input port (e.g., Ethernet) | memory | output port (e.g., Ethernet) |

system bus
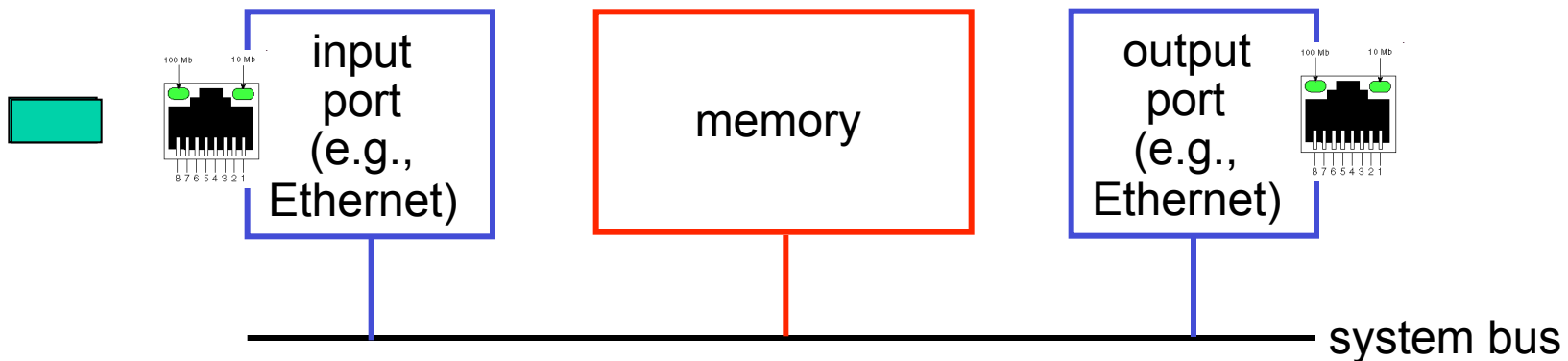
Switching via a bus

- 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

bus

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

crossbar