TDTS06: Computer Networks

Instructor: Niklas Carlsson

Email: niklas.carlsson@liu.se

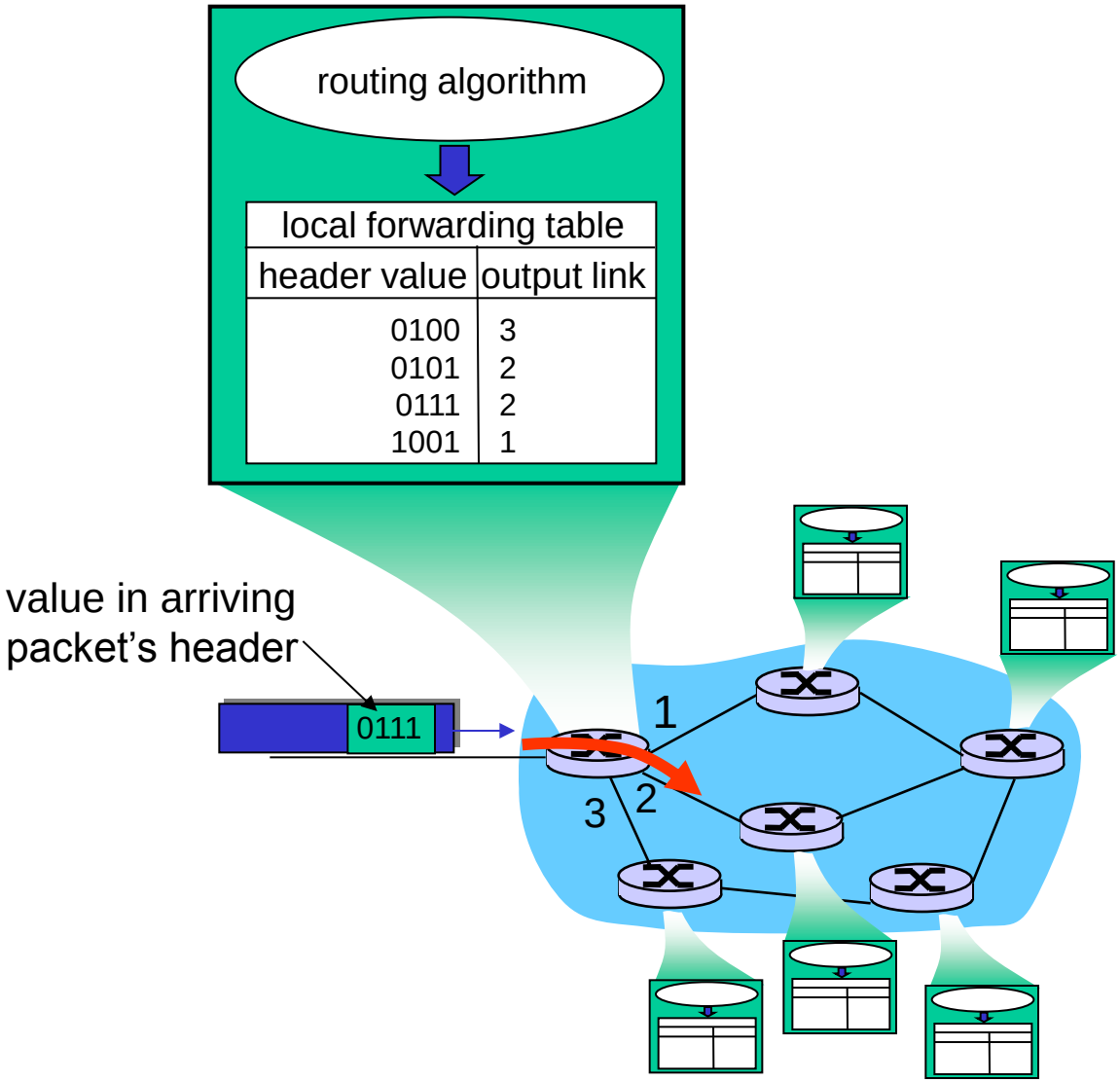
Notes derived from "*Computer Networking: A Top Down Approach*", by Jim Kurose and Keith Ross, Addison-Wesley.

The slides are adapted and modified based on slides from the book's companion Web site, as well as modified slides by Anirban Mahanti and Carey Williamson.

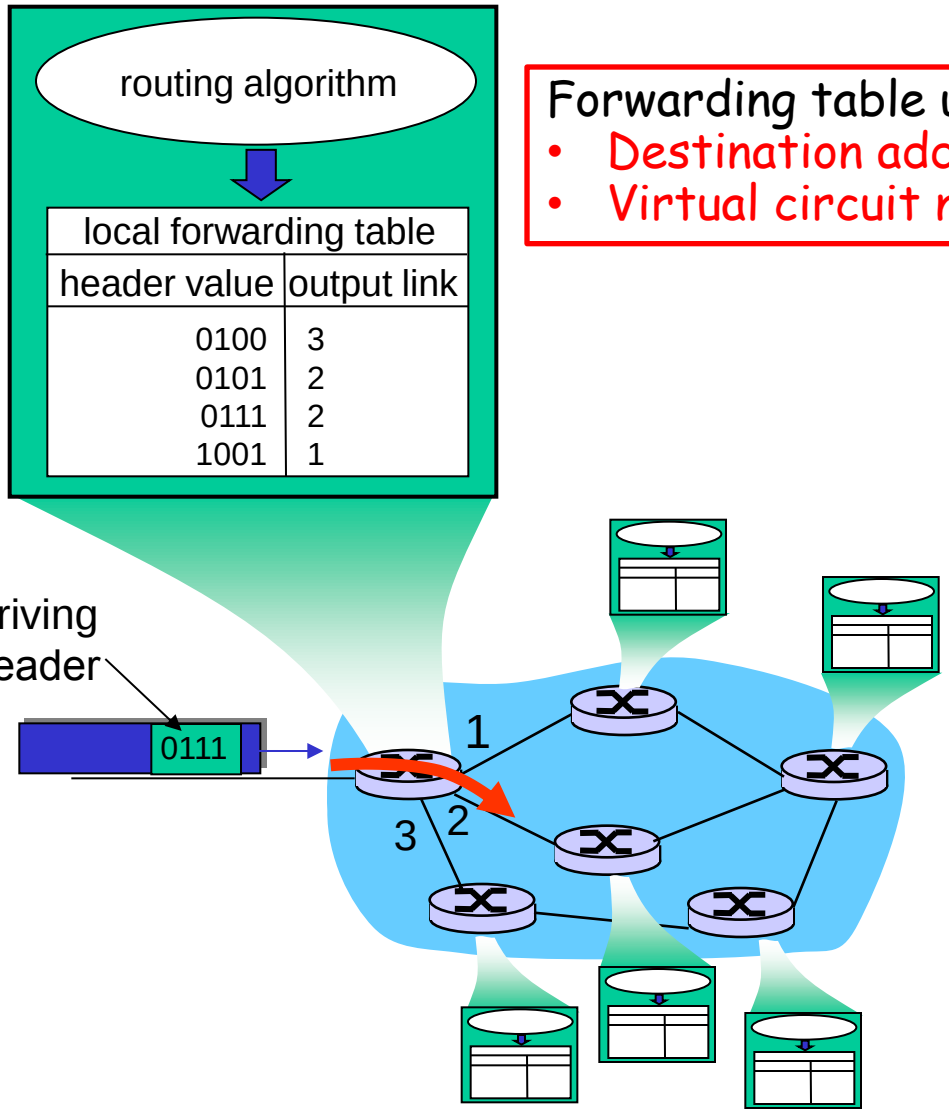
Key Network-Layer Functions

- *forwarding:* move packets from router's input to appropriate router output
- *routing:* determine the path taken by packets as they flow from a sender to a receiver

Interplay between routing and forwarding



Interplay between routing and forwarding



Forwarding table uses:

- Destination address in Datagram networks
- Virtual circuit number in VC Networks

Connection-oriented vs connection-less

- 1) VC network provides network-layer connection-oriented service
- 2) Datagram network provides network-layer connectionless service

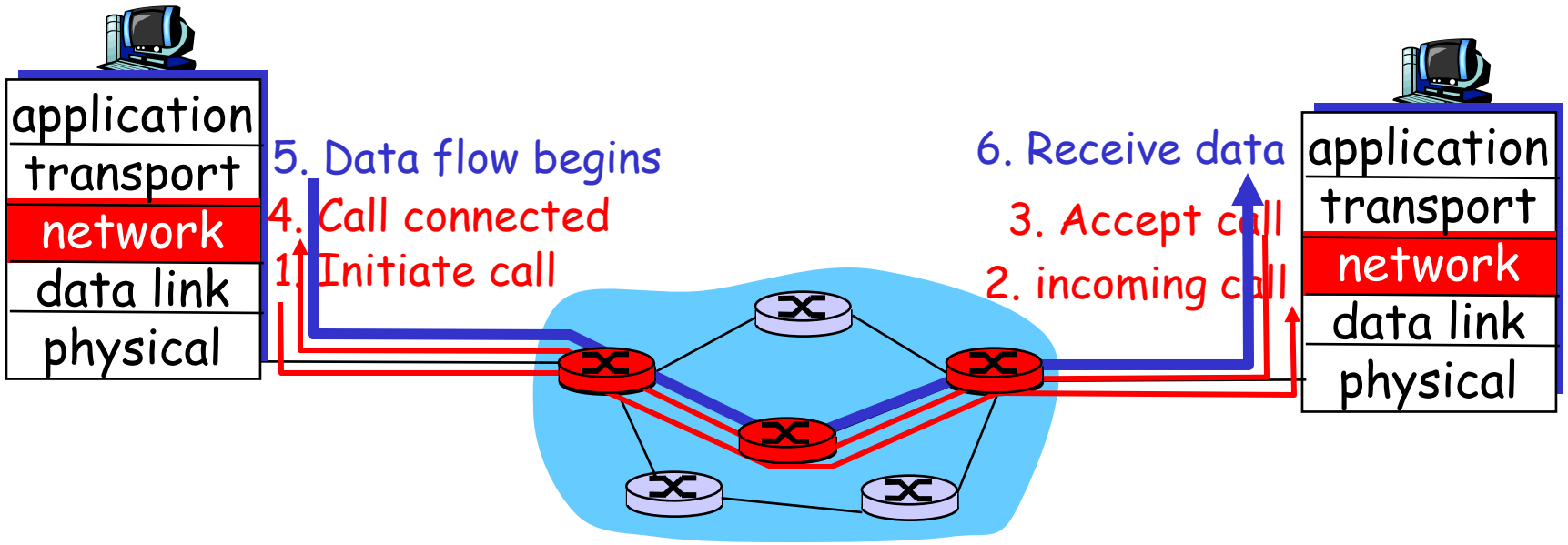
1) Virtual circuits (think phone networks)

"source-to-dest path behaves much like telephone circuit"

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

Virtual circuits: signaling protocols

- 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



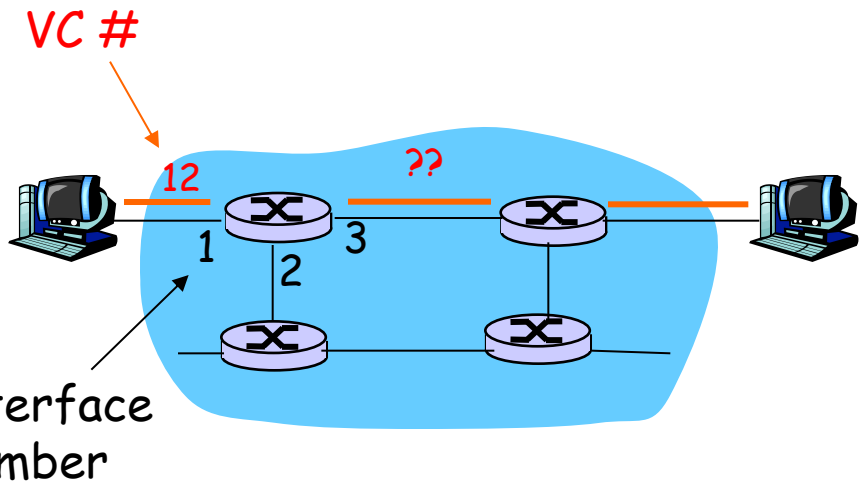
VC implementation

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 a VC number.
- VC number must be changed on each link.
  - New VC number comes from forwarding table

Forwarding table

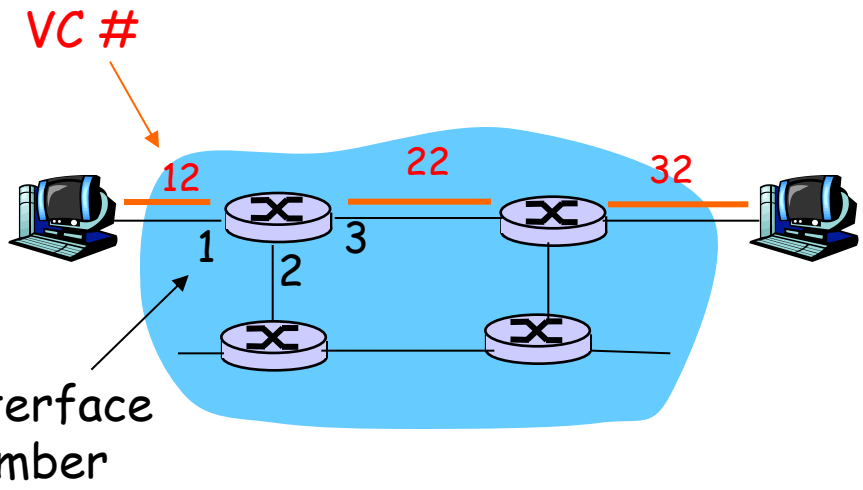


Forwarding table in Northwest router:

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

Routers maintain connection state information!

Forwarding table



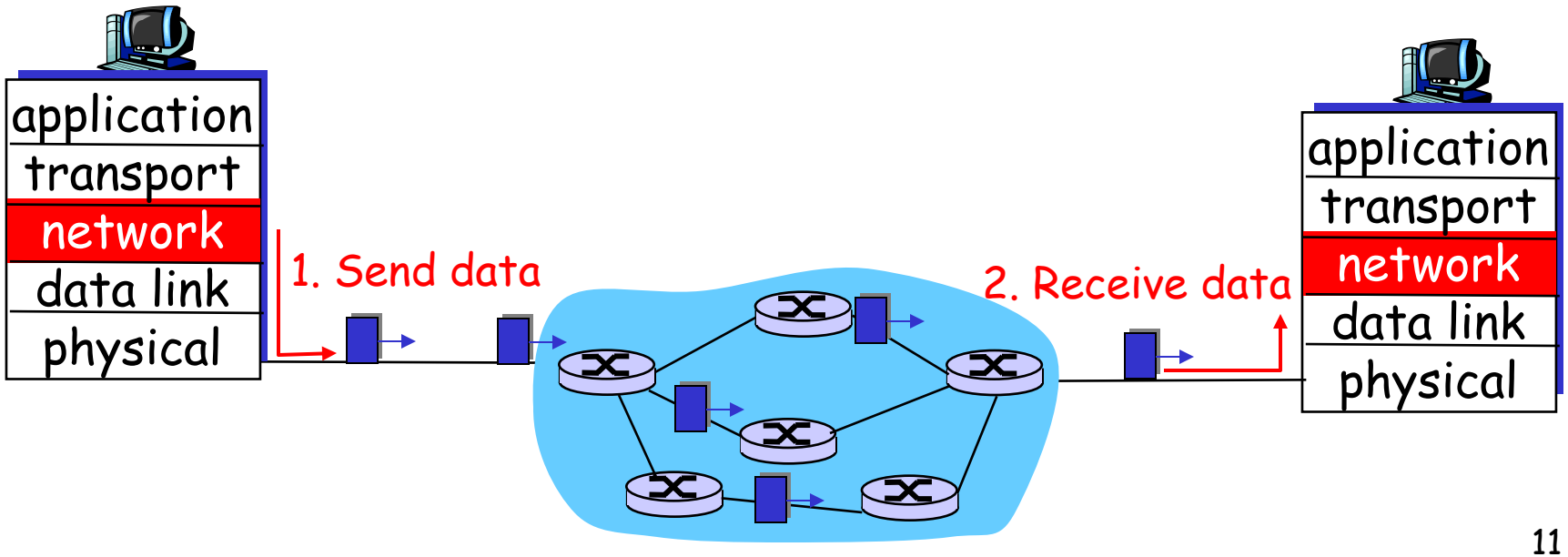
Forwarding table in Northwest router:

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

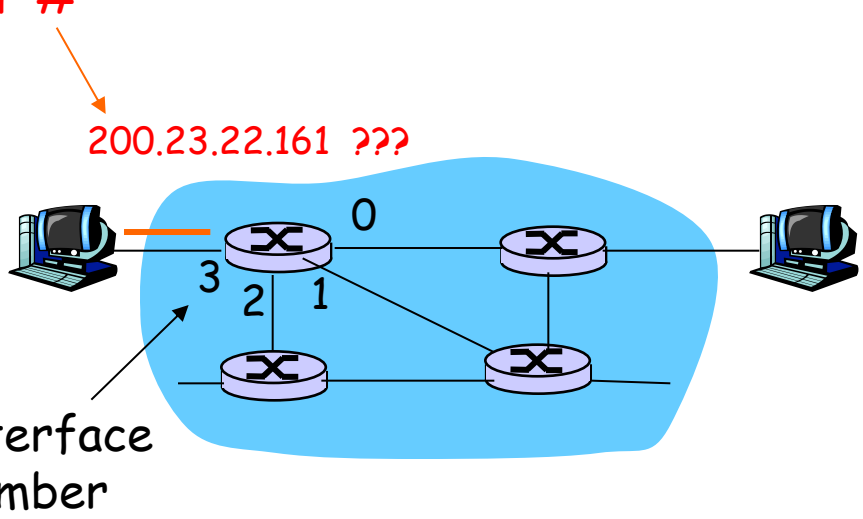
Routers maintain connection state information!

2) Datagram networks (used on the Internet)

- 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
  - packets between same source-dest pair may take different paths



Forwarding table



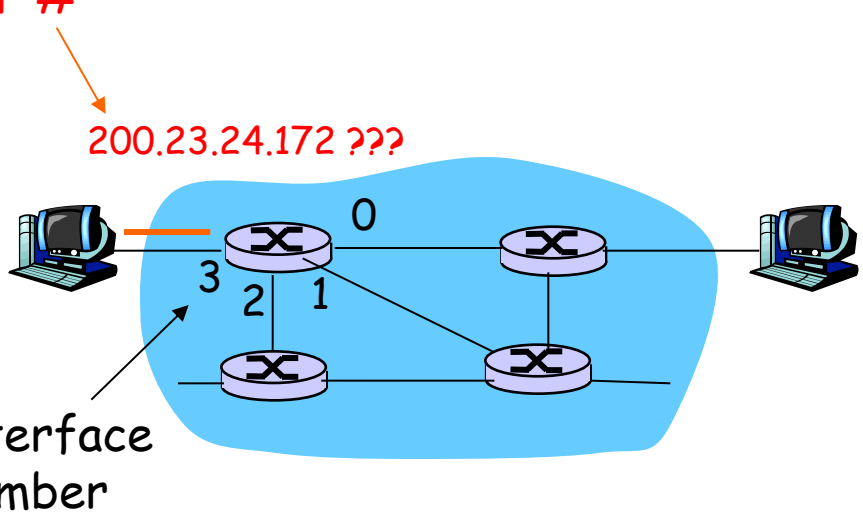
Forwarding table in Northwest router:

Prefix	Outgoing interface
200.23.16.0/21	0
200.23.24.0/24	1
200.23.24.0/21	2
…	…

See example from last lecture (part 1) for details

Routers aggregation and longest prefix matching

Forwarding table



Forwarding table in Northwest router:

Prefix	Outgoing interface
200.23.16.0/21	0
200.23.24.0/24	1
200.23.24.0/21	2
…	…

See example from last lecture (part 1) for details

Routers aggregation and longest prefix matching

Datagram or VC network: why?

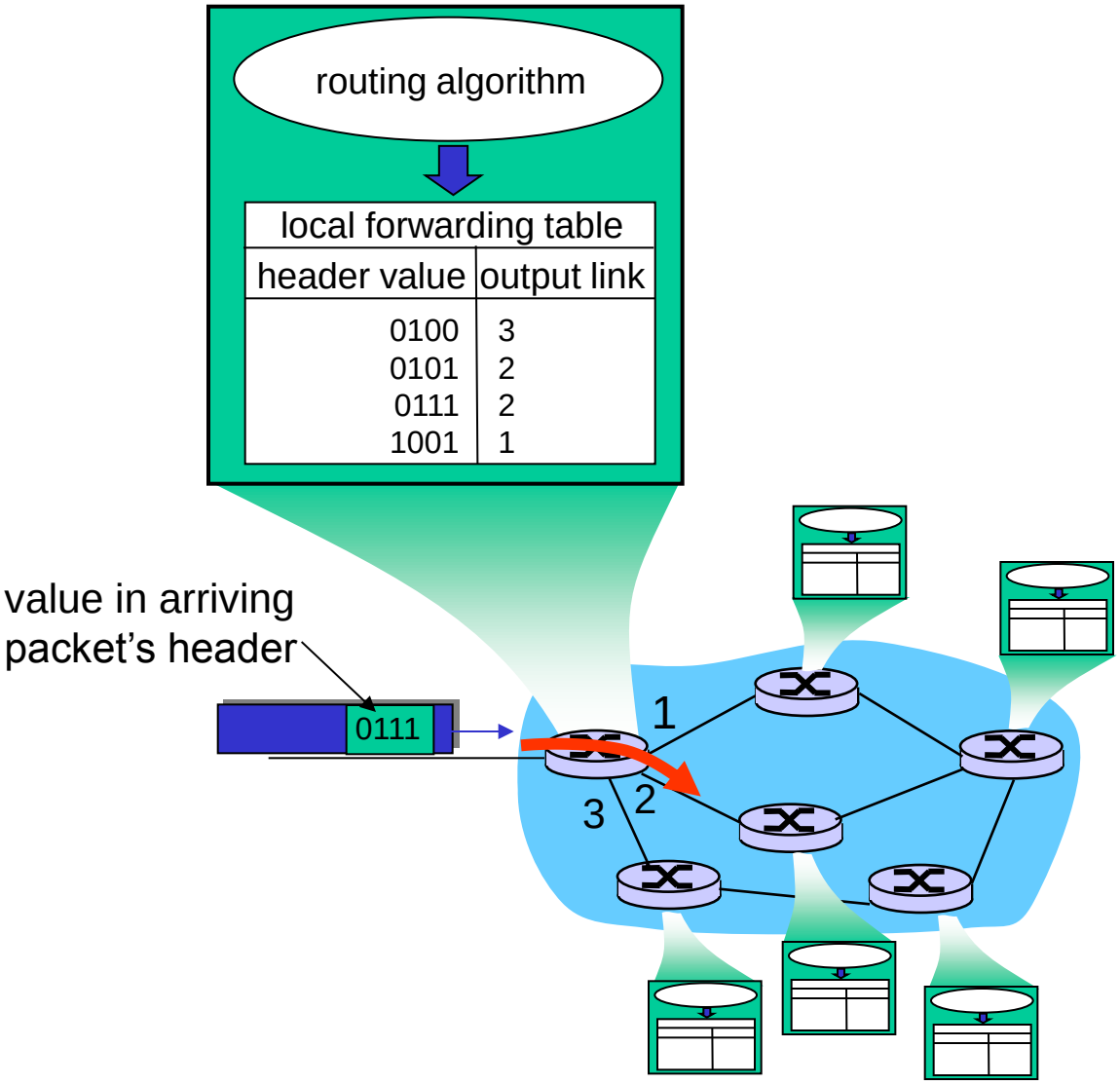
Internet

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

ATM

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

Interplay between routing and forwarding



Inside a Router

Router Architecture Overview

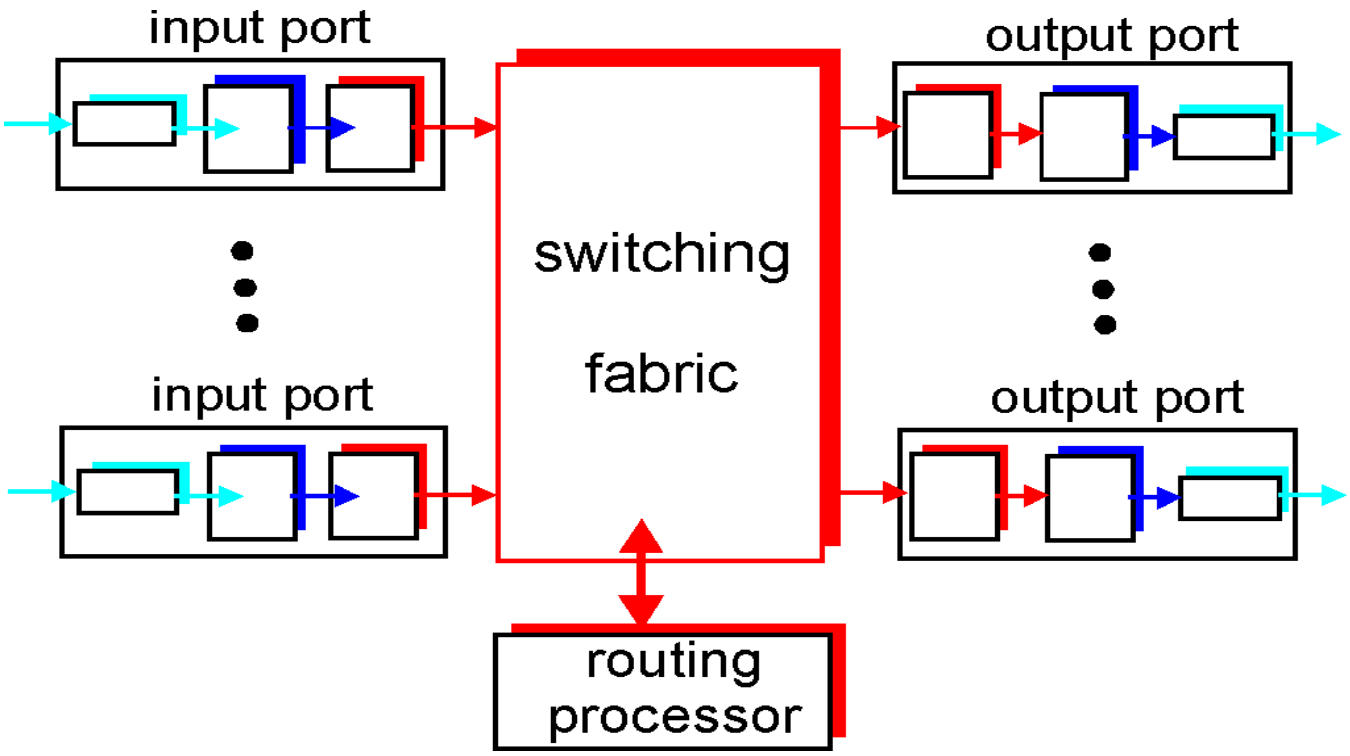
Two key router functions:

- run routing algorithms/protocol (RIP, OSPF, BGP)
- *forwarding* datagrams from incoming to outgoing link

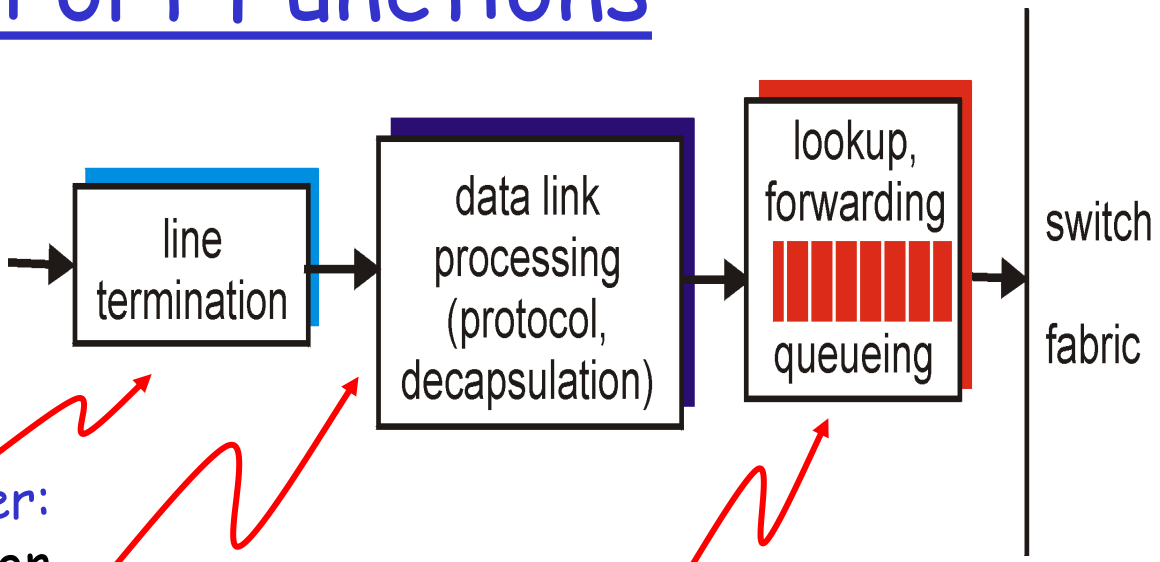
Router Architecture Overview

Two key router functions:

- run routing algorithms/protocol (RIP, OSPF, BGP)
- *forwarding* datagrams from incoming to outgoing link



Input Port Functions



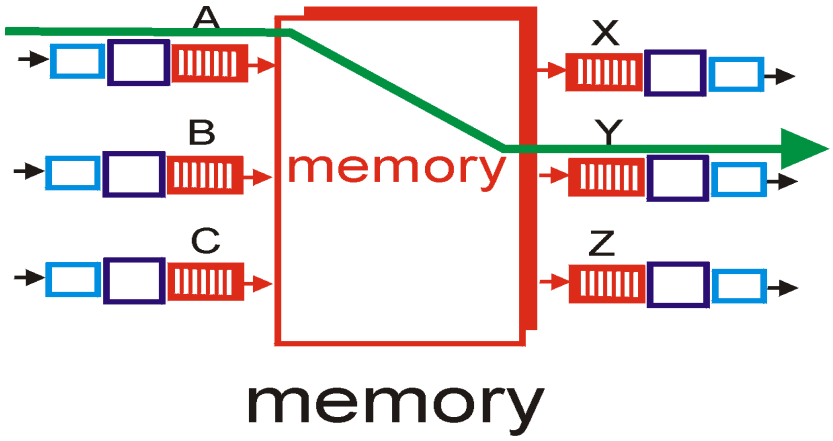
Physical layer:
bit-level reception

Data link layer:
e.g., Ethernet

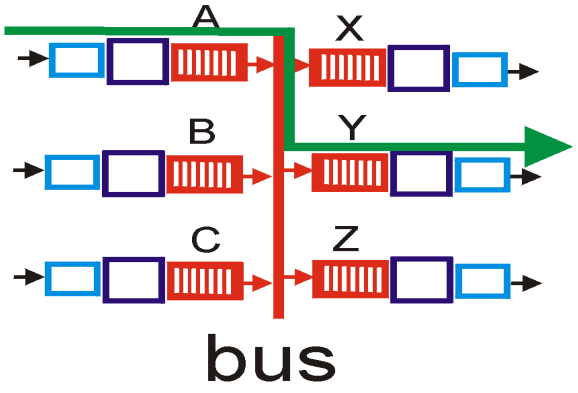
Decentralized switching:

- given datagram dest., lookup output port using forwarding table in input port memory (caching of entries)
- goal: complete input port processing at 'line speed'
- queuing: if datagrams arrive faster than forwarding rate into switch fabric

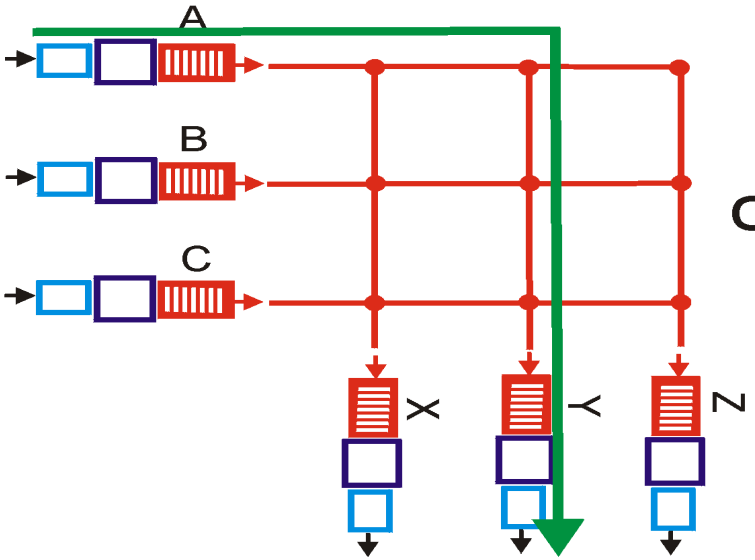
Three types of switching fabrics



memory

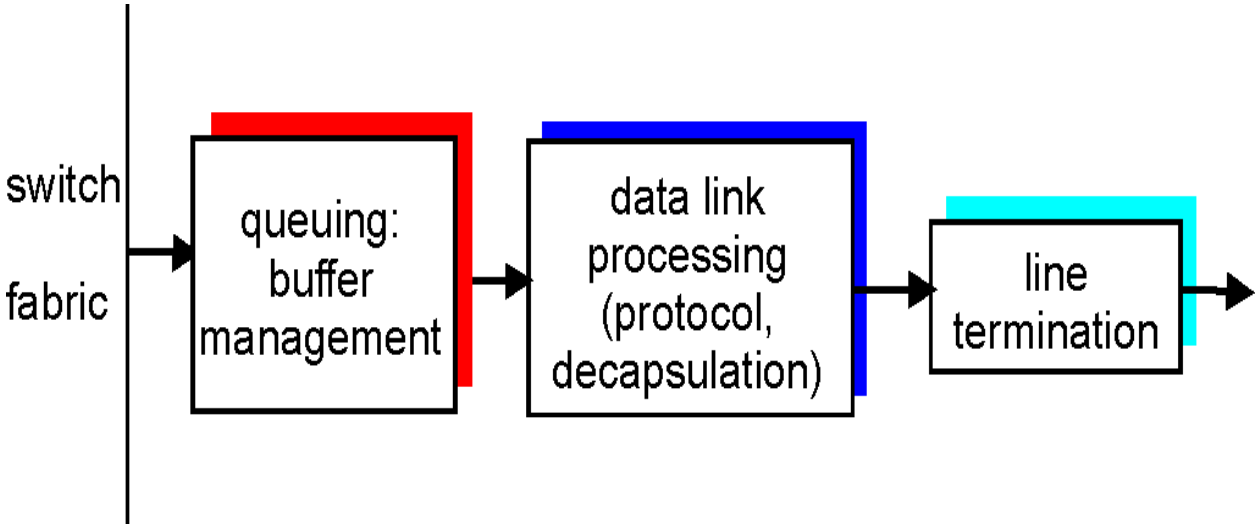


bus



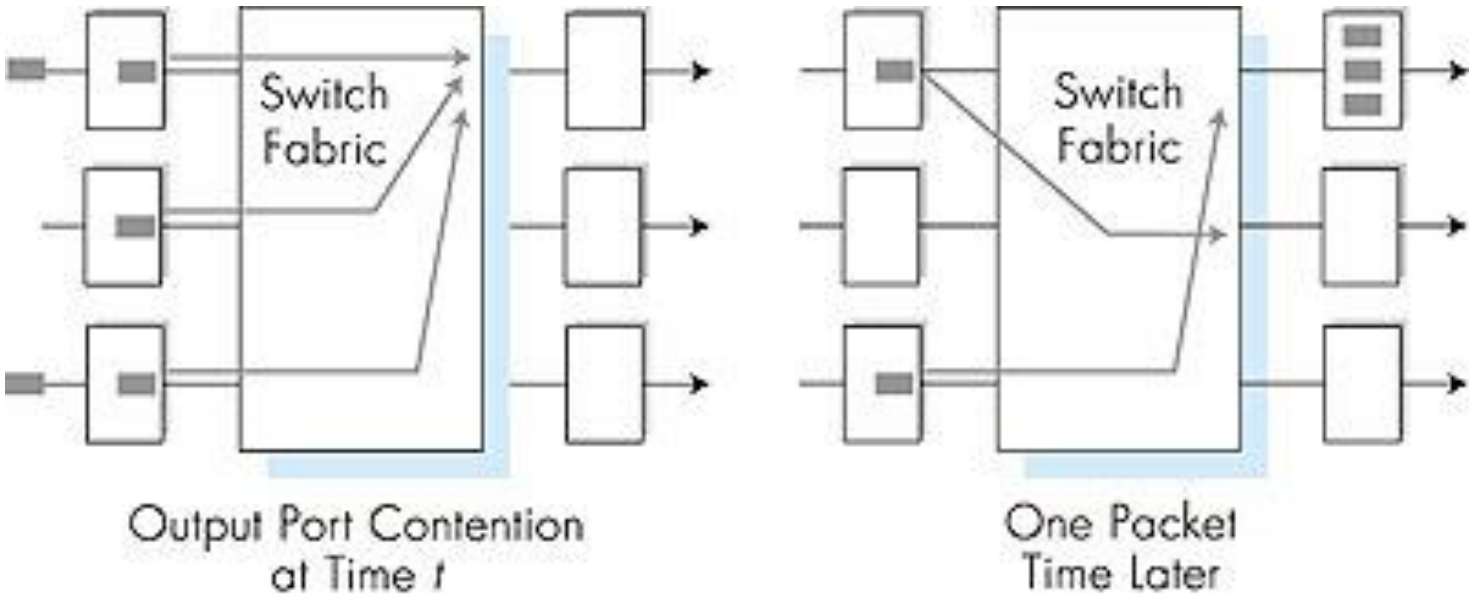
crossbar

Output Ports



- *Buffering* required when datagrams arrive from fabric faster than the transmission rate
  - Need queue management policy (Drop-Tail, AQM)
  - Also need packet scheduling policy (FCFS, WFQ)

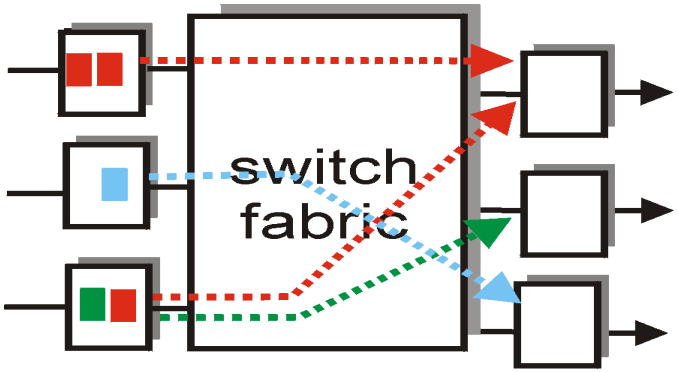
Output port queueing



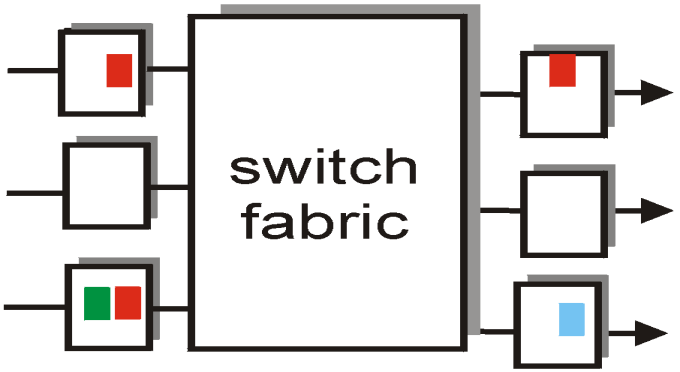
- buffering when arrival rate via switch exceeds output line speed
- *queueing (delay) and loss due to output port buffer overflow!*

Input Port Queuing?

- Fabric slower than input ports combined -> queueing may occur at input queues
- Head-of-the-Line (HOL) blocking: queued datagram at front of queue prevents others in queue from moving forward (even though o/p port is free for the other datagram)



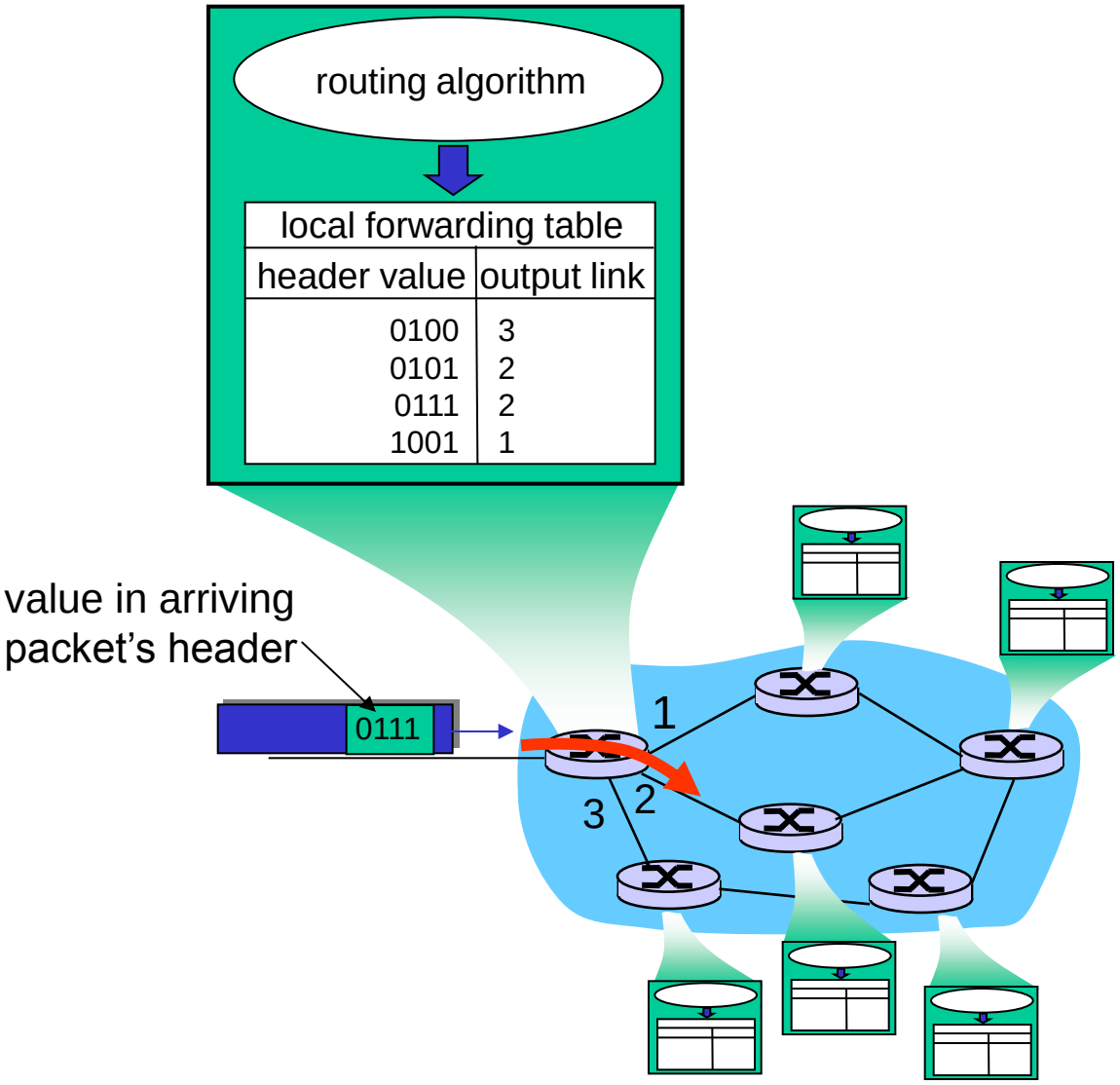
output port contention
at time t - only one red
packet can be transferred



green packet
experiences HOL blocking

More slides …

Interplay between routing and forwarding



Network service model

Q: What *service model* for the "channel" transporting datagrams from sender to receiver?

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

Network layer connection-oriented and connection-less service

- Datagram network provides network-layer connectionless service
- VC network provides network-layer connection-oriented service
- Analogous to the transport-layer services, but:
  - Service: host-to-host
  - No choice: network provides one or the other
  - Implementation: in the 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

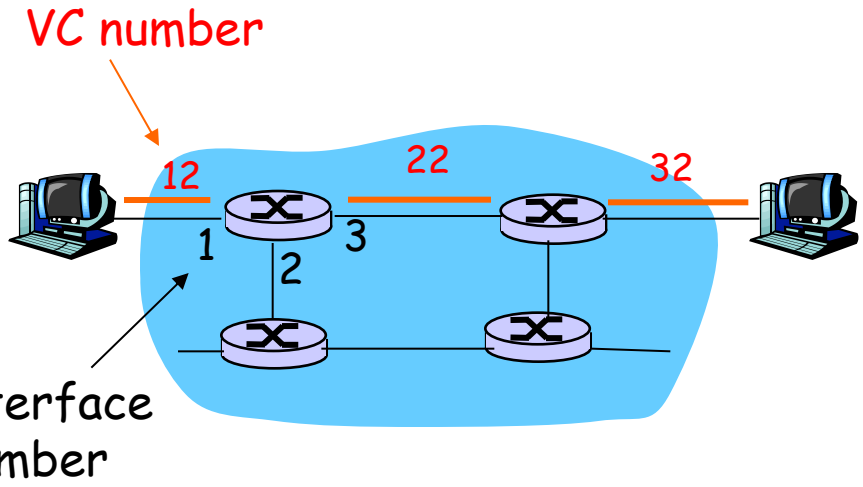
VC implementation

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 a VC number.
- VC number must be changed on each link.
  - New VC number comes from forwarding table

Forwarding table



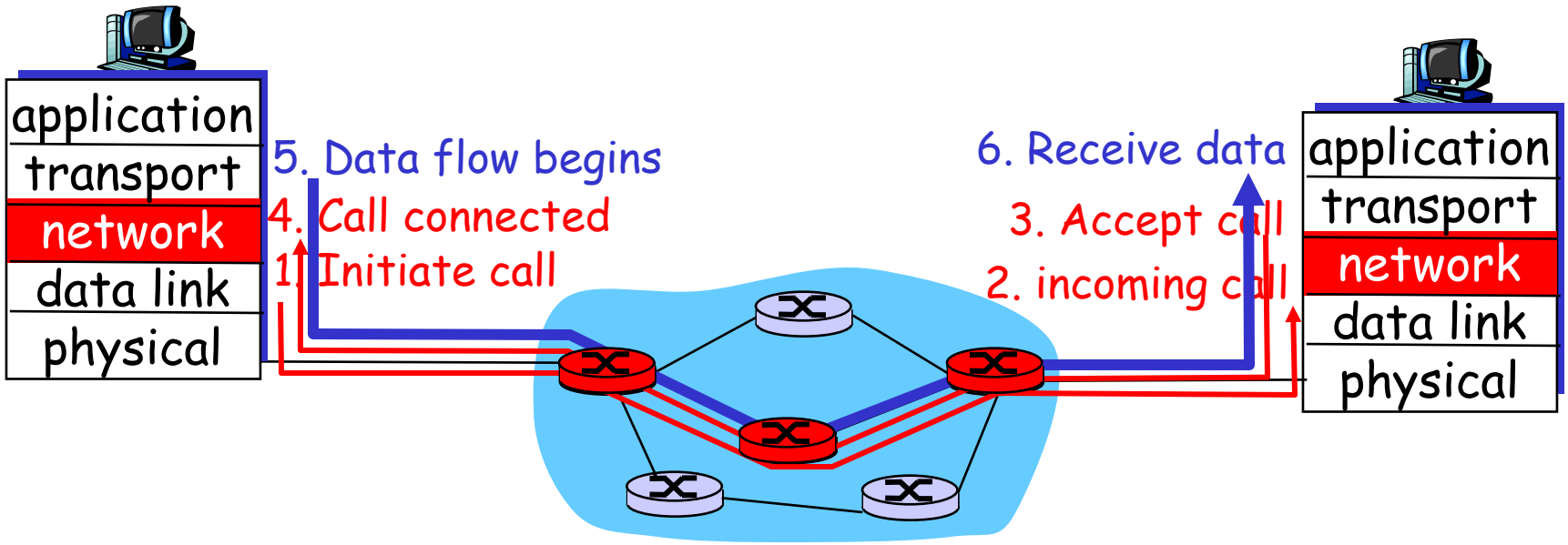
Forwarding table in Northwest router:

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

Routers maintain connection state information!

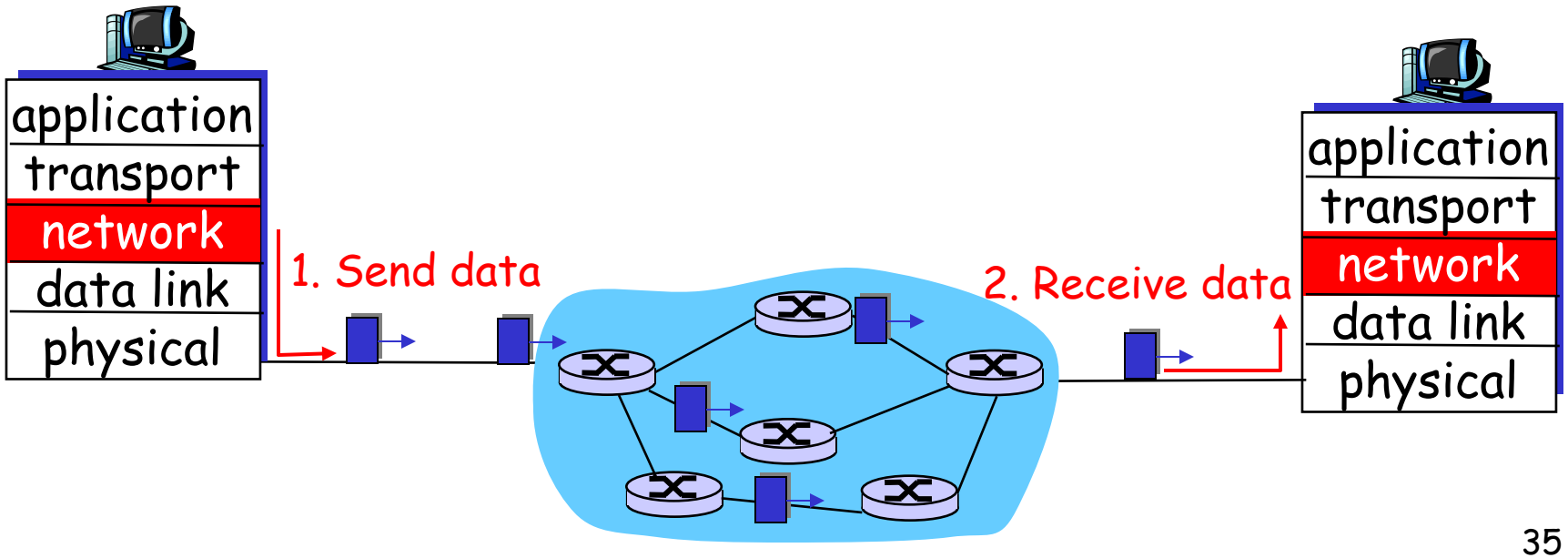
Virtual circuits: signaling protocols

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



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
  - packets between same source-dest pair may take different paths



Datagram or VC network: why?

Internet

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

ATM

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

Next Topic

- Routing Algorithms
- Routing in the Internet
- Readings: Chapter 4 in the K&R book

More slides …

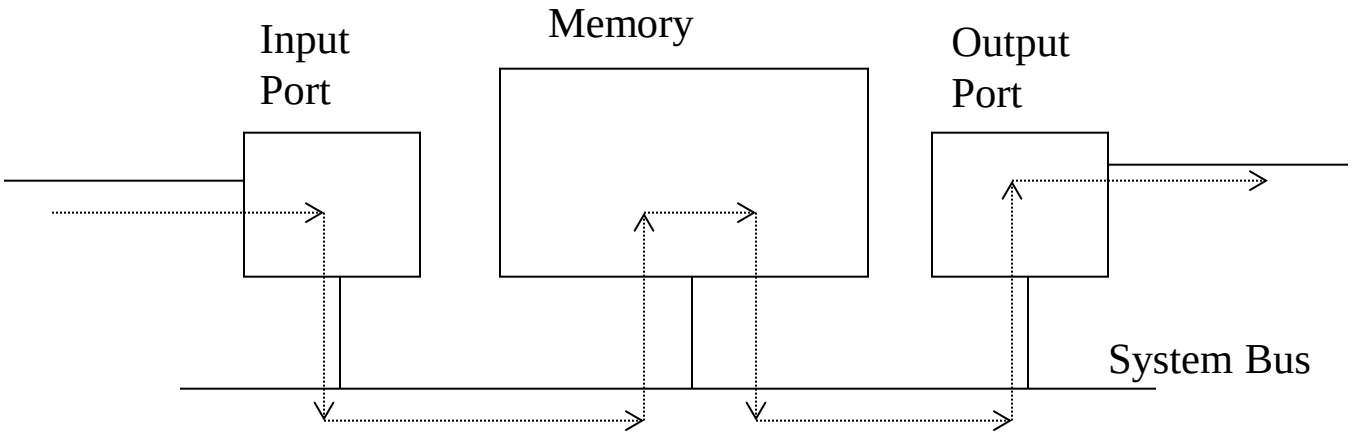
VC Networks: Connection setup

- 3rd important function in *some* network architectures:
  - ATM, frame relay, X.25
- Before datagrams flow, two hosts and intervening routers establish virtual connection
  - Routers get involved
- Network and transport layer conn service:
  - Network: between two hosts
  - Transport: between two processes

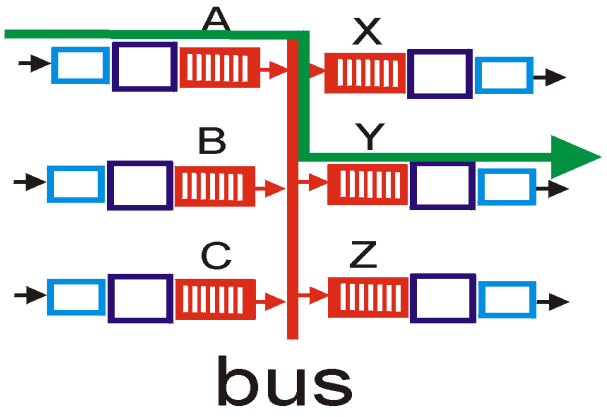
Switching via Memory

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
- 1 Gbps bus, Cisco 1900: sufficient speed for access and enterprise routers (not regional or backbone)

Switching via an Interconnection Network

- Overcomes bus bandwidth limitations
- Banyan networks, 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 Gbps through the interconnection network