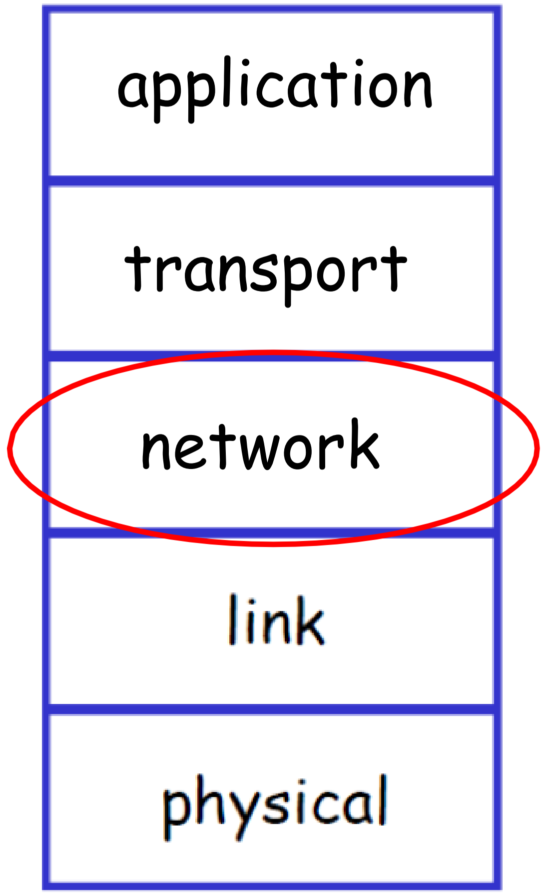
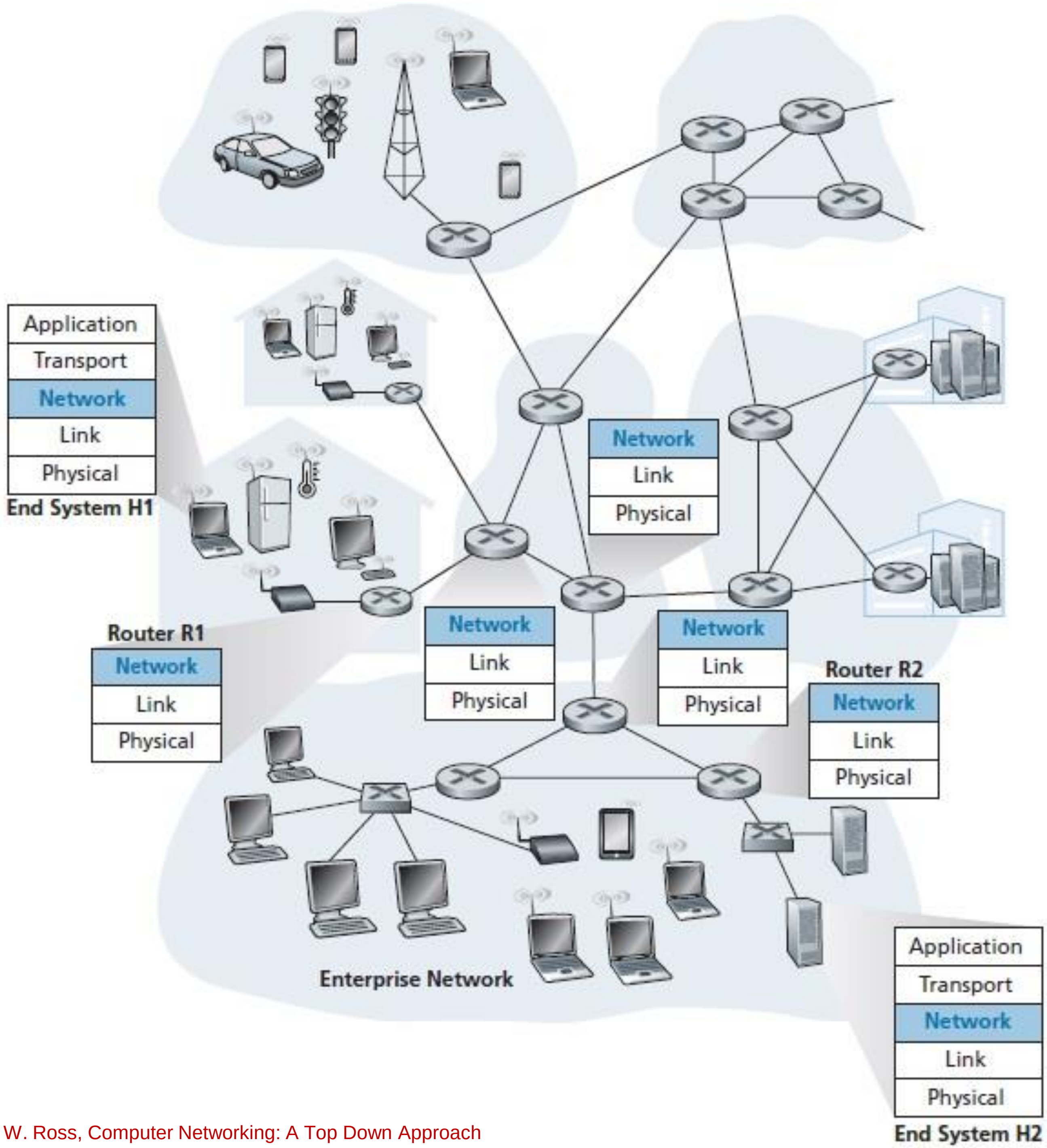
Network Layer





Source: J. F. Kurose and K. W. Ross, Computer Networking: A Top Down Approach

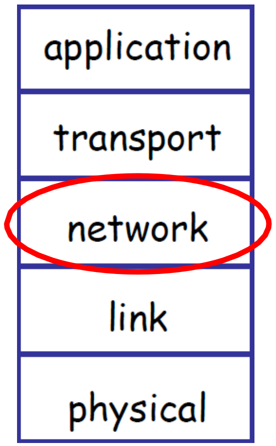
Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What’s inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

(Recall)

Internet protocol stack

Network Layer



- When transferring application messages, the transport layer provides process-to-process communication service whereas the network layer provides host-to-host communication service
- Transfers ***packets or datagrams*** from one host to another (passing through several routers!).
- The transport layer protocol (TCP or UDP) in a source host passes a segment and a destination address (got from the user inputs) to the network layer. The network layer then provides the service of delivering the segment to the transport layer in the destination host.
- The Internet's network layer provides "connectionless service" which is "best-effort connectionless packet transfer".
- The Internet‟s network layer includes the key protocol, IP (Internet Protocol).
  - IP defines the fields in the packet/datagram and how the end systems and routers act on these fields.
  - There is only one IP protocol and all Internet components which have a network layer must run the IP protocol.
  - The Internet‟s network layer also contains „routing protocols‟ that determine the routes/paths that packets/datagrams take between sources and destinations.
  - Although the network layer contains both the IP protocol and many routing protocols, it is often simply referred to as the IP layer (since **IP is the one which glues different networks in the Internet together!**)

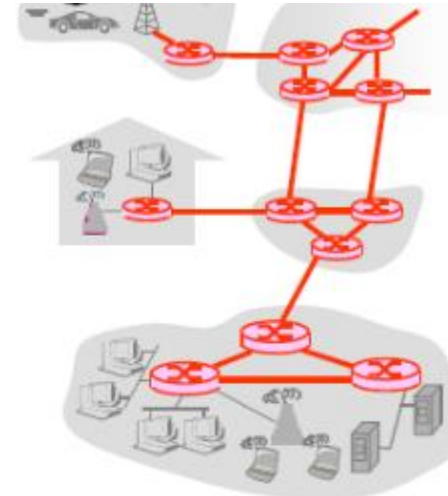
Note: The term "packets" might, sometimes, be used to refer to any data chunks.

Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What's inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

(Recall)

Network Core: Packet Switching



each end-end data stream divided into *packets*

- user A, B packets *share* network resources
- each packet uses full link bandwidth
- resources used *as needed*

~~Bandwidth division into "pieces"~~

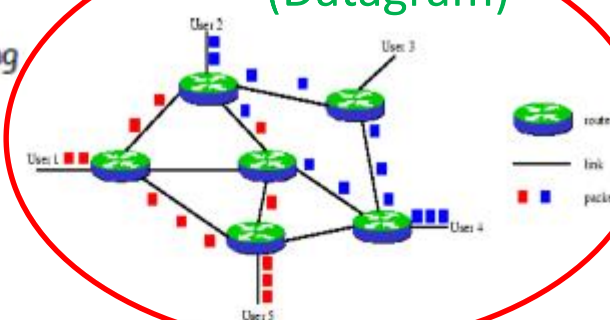
~~Dedicated allocation~~

~~Resource reservation~~

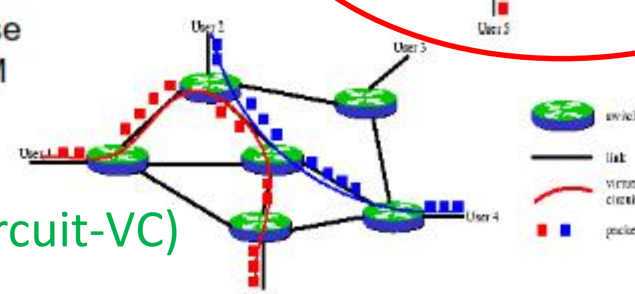
resource contention:

- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - Node receives complete packet before forwarding

(Datagram)



- Could be **connection-less service (IP networks)** (Different packets of the same message may traverse different routes) or connection oriented service (ATM networks)



(Virtual circuit-VC)

- The Internet is a datagram network

Network layer service models:

Network Architecture	Service Model	Guarantees ? Bandwidth	Loss	Order	Timing	Congestion feedback
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

* There have also been proposed service model extensions to the Internet architecture; for example, the Intserv architecture [RFC 1633] aims to provide end-end delay guarantees and congestion-free communication. Interestingly, in spite of these well-developed alternatives, the Internet's basic best-effort service model combined with adequate bandwidth provisioning and bandwidth-adaptive application-level protocols such as the DASH protocol have arguably proven to be more than "good enough" to enable an amazing range of applications, including multimedia.

Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What's inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

Two Key Network-Layer Functions

- forwarding: move packets from router's input to appropriate router output (router-local action)

  (switching)

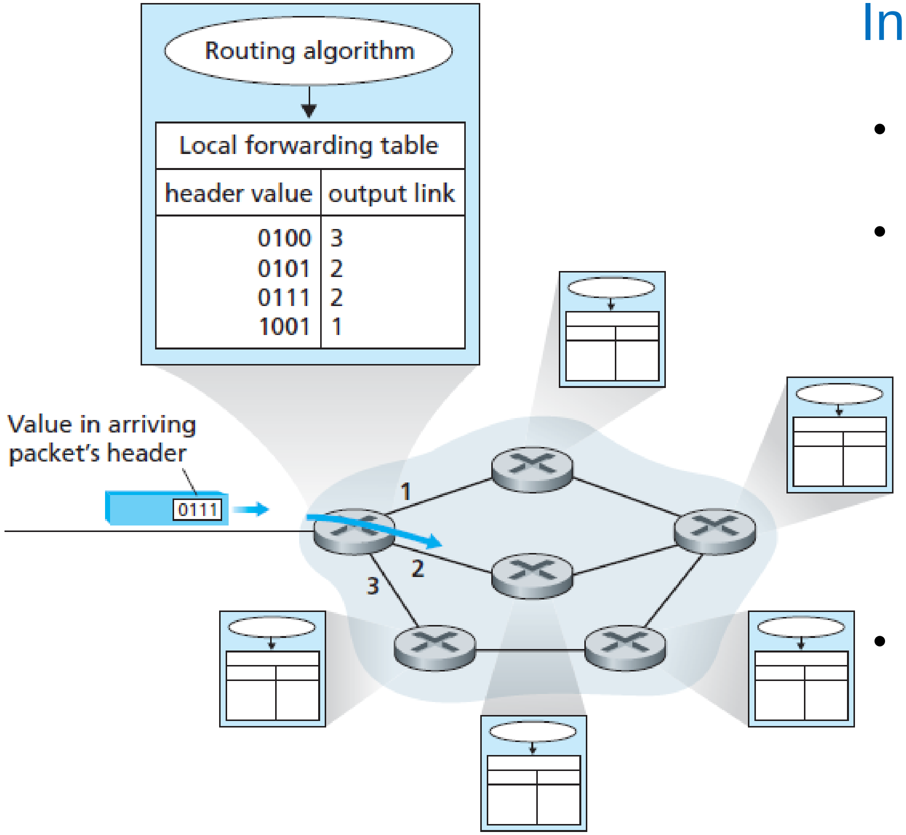
- routing: determine routes/paths taken by packets from source to dest. (network-wide process).

  determine the full path

  - 'routing algorithms' calculate these paths

(another important function in ‘virtual circuit networks’: ‘connection setup’)

Interplay between routing and forwarding



- Every router has a forwarding table
- 'Forwarding table' and 'routing table' are sometimes used interchangeably (there are some differences).
  - Routing tables are generally not used directly for packet forwarding in modern router architectures; instead, they are used to generate the information for a smaller 'forwarding table'.
- Traditionally, a router forwards a packet by examining the value of a field in its header (address/identifier), and then using this header value to index into the router's forwarding table.
- Internet's routing algorithms/protocols (distributed):
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - BGP: Border Gateway Protocol
- Recent development: the Software Defined Network (SDN) approach uses a logically centralized controller (server) to compute routes
- Routing algorithm determines the values that are inserted into the routers' forwarding tables.
- The routing algorithm may be
  - Centralized (e.g., with an algorithm executing on a central site and downloading routing information to each router) or
  - Decentralized (i.e., with a piece of distributed routing algorithm running in each router).

  easier, do not take lot of time

Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What’s inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

Network layer: Data plane, Control plane

Data plane

- All activities involving as well as resulting from data packets sent by the end user, e.g.,
  - Forwarding
  - Fragmentation and reassembly
  - Replication for multicasting
- Local, per-router function

user packets are get forwarded to the data plane.

- per router data plane is always happen. because if we consider a centralized data plane, all the routers are connected to the Centralized router. Data plane of centralized router get damaged, the all the data of the network can be lost.

Control plane

- All activities that are necessary to perform data plane activities but do not involve end-user data packets
  - Routing (Making routing tables)
  - Setting packet handling policies (e.g., security)
  - Base station beacons announcing availability of services
- Network-wide logic

Two approaches to structuring network control plane:

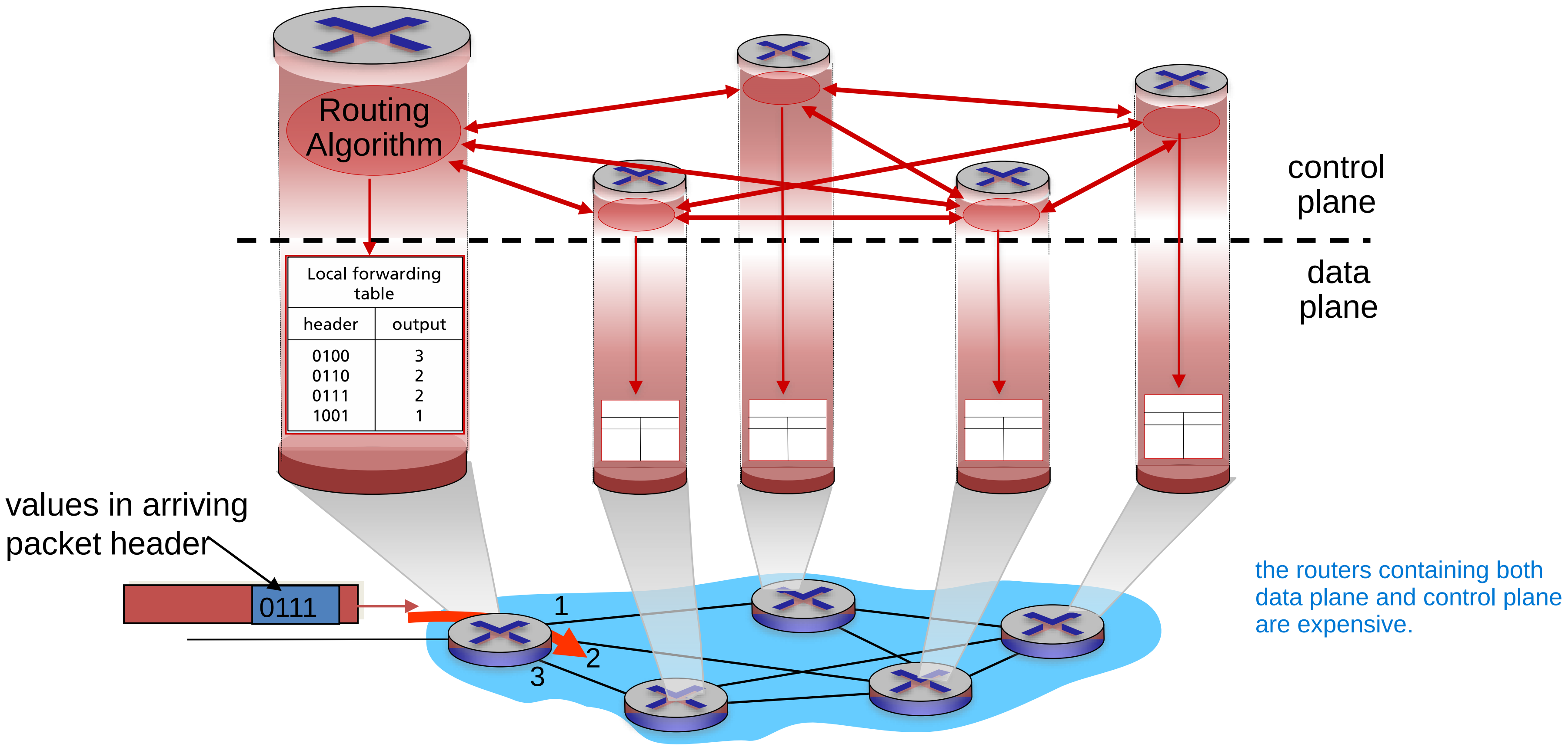
- Per-router control (traditional)- implemented in routers
- Logically centralized control (software defined networking (SDN) approach)- implemented in (remote) server(s)

Other 'optional' planes:

- Management (provisioning and monitoring of the networks- Fault, Configuration, Accounting, Performance and Security (**FCAPS**)).
- Services (Middlebox services to improve performance or security, e.g., Load Balancers, Proxy Service, Intrusion Detection, Firewalls, SSL Off-loaders)

Per-router control plane

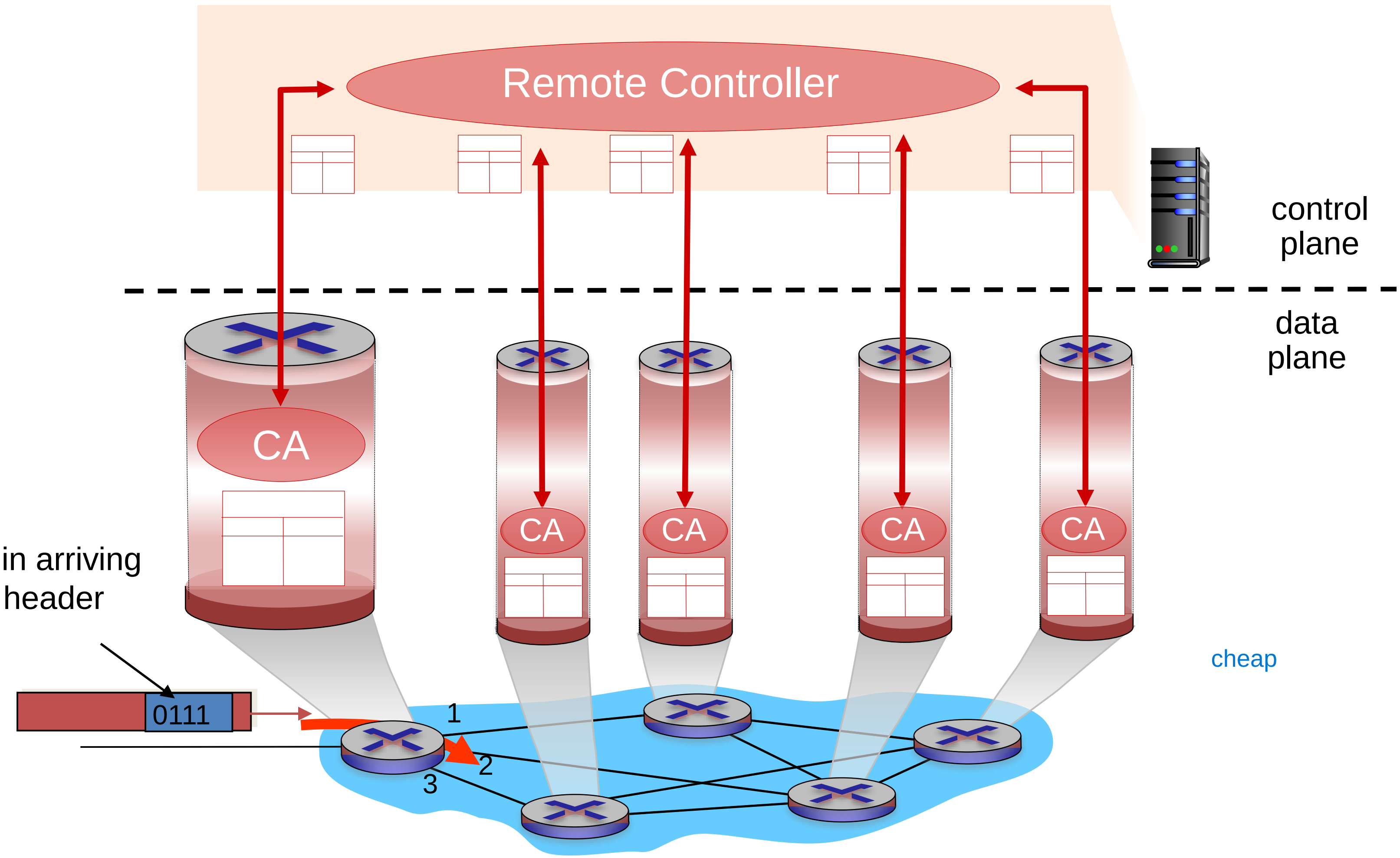
Individual routing algorithm components *in each router* interact in the control plane, and it determines values in forward tables.



- In the network shown above, a routing algorithm runs in each and every router and both forwarding and routing functions are contained within a router.

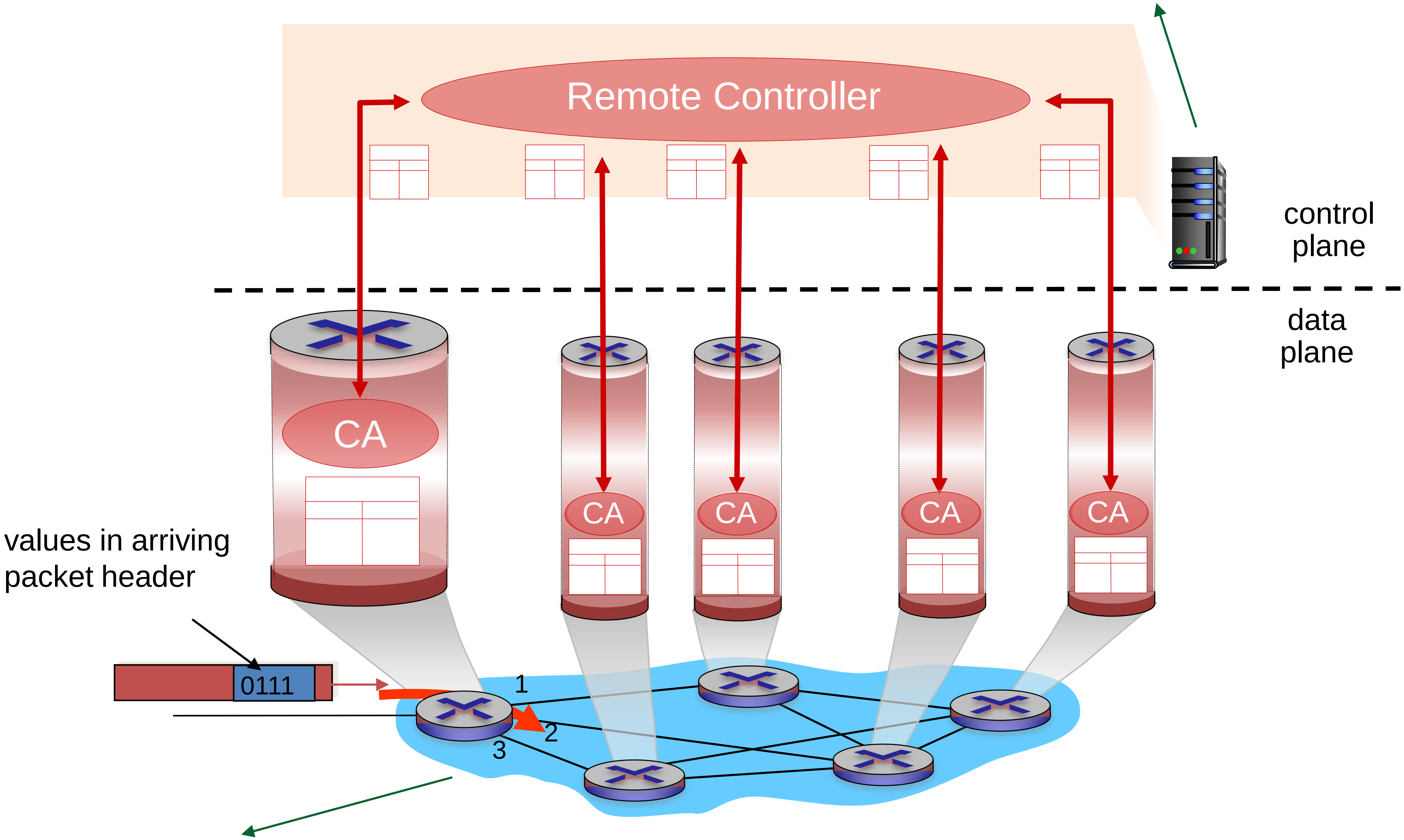
Logically centralized control plane

A distinct (typically remote) logically centralized controller (interacts with local control agents (CAs) and) determines and distributes values in forwarding tables.



- Figure shows a physically separate, remote controller computing and distributing the forwarding tables to be used by each and every router.

The remote controller might be implemented in a remote data center with high reliability and redundancy, and might be managed by the ISP or some third party.



- The network is “software-defined” because the controller that computes forwarding tables and interacts with routers is implemented in software => SDN
- Increasingly, these software implementations are also open, that is, similar to Linux OS code, the code is publically available => Innovation!

Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What's inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What's inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

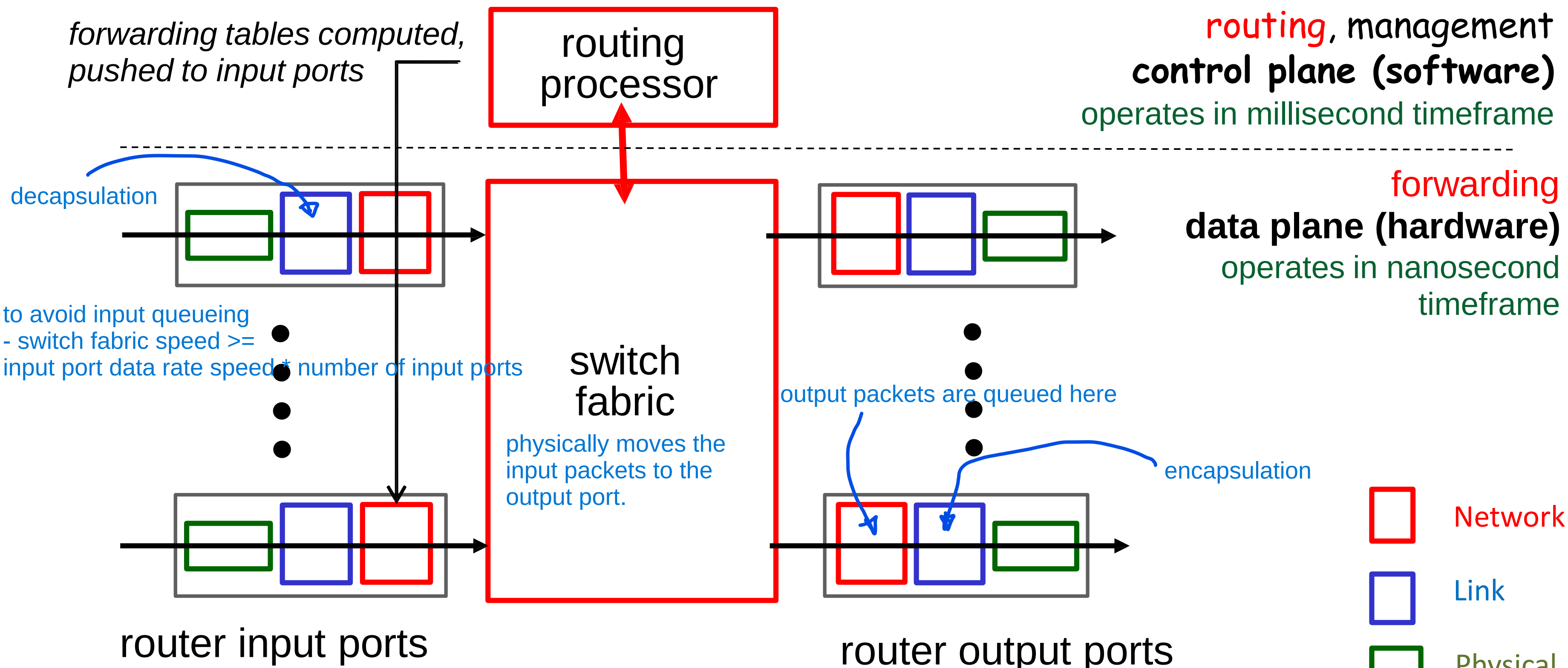
Router architecture overview

there can be control routing packets can be arrived from the input ports. Then they are forwarded to the routing processor.

Two key router functions:

- run routing algorithms/protocol (RIP, OSPF, BGP) @ routing processor
- forward datagrams from incoming to outgoing link

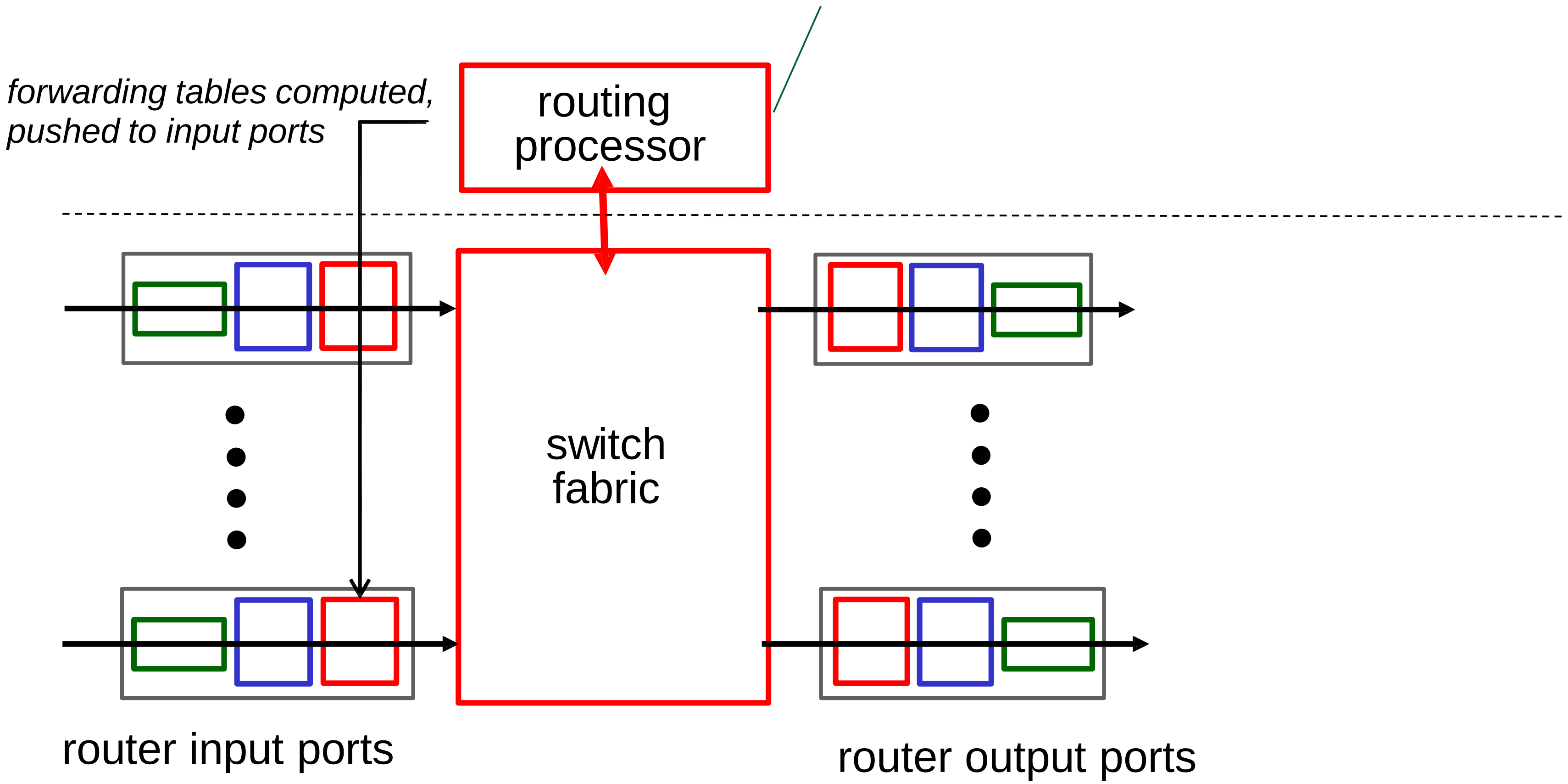
routing processor makes the forwarding table and store a copy of that in each input port.



we cannot avoid the queueing in output port.
Queue in a output port has a finite length. Therefore, If the packets are arrived when the queue is full, there can be packet lost are happen.

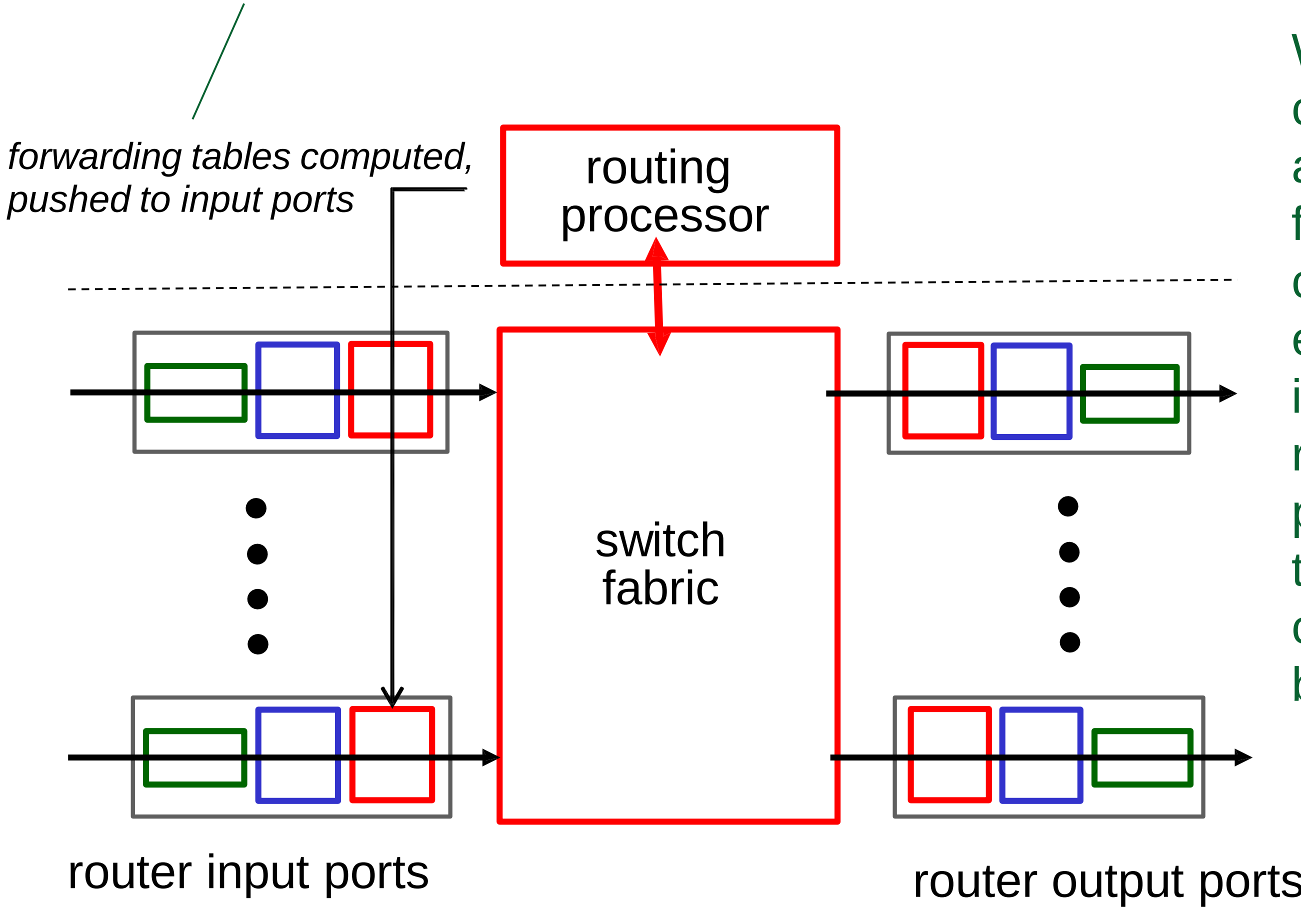
Link layer functions - Error detection, MAC address assigning

- The routing processor (typically a traditional CPU) performs control-plane functions.
- Traditional routers: it executes the routing protocols, maintains routing tables and link state information, and computes the forwarding table for the router.
- SDN routers: the routing processor is responsible for communicating with the remote controller in order to (among other activities) receive forwarding table entries computed by the remote controller, and install these entries in the router’s input ports.
- The routing processor also performs the network management functions



Here, we learn more about ‘forwarding’ or ‘switching’

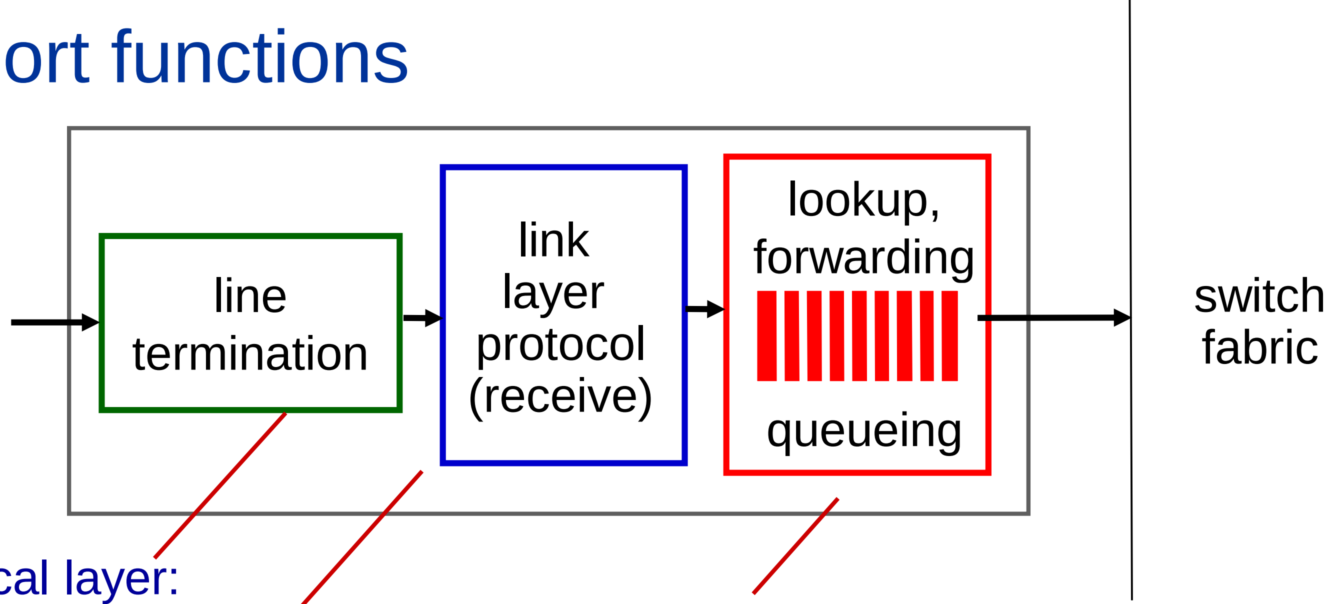
- The forwarding table is
  - either computed and updated by the routing processor (using a routing protocol to interact with the routing processors in other network routers) or
  - is received from a remote SDN controller.
- The forwarding table is copied from the routing processor to the line cards over a separate bus (e.g., a PCI bus)



With such a shadow copy of forwarding table at each line card, forwarding decisions can be made locally, at each input port, without invoking the centralized routing processor on a per-packet basis and thus avoiding a centralized processing bottleneck.

Here, we learn more about 'forwarding' or 'switching'

Input port functions



physical layer:
bit-level reception

data link layer:
e.g., Ethernet

decentralized switching/forwarding:

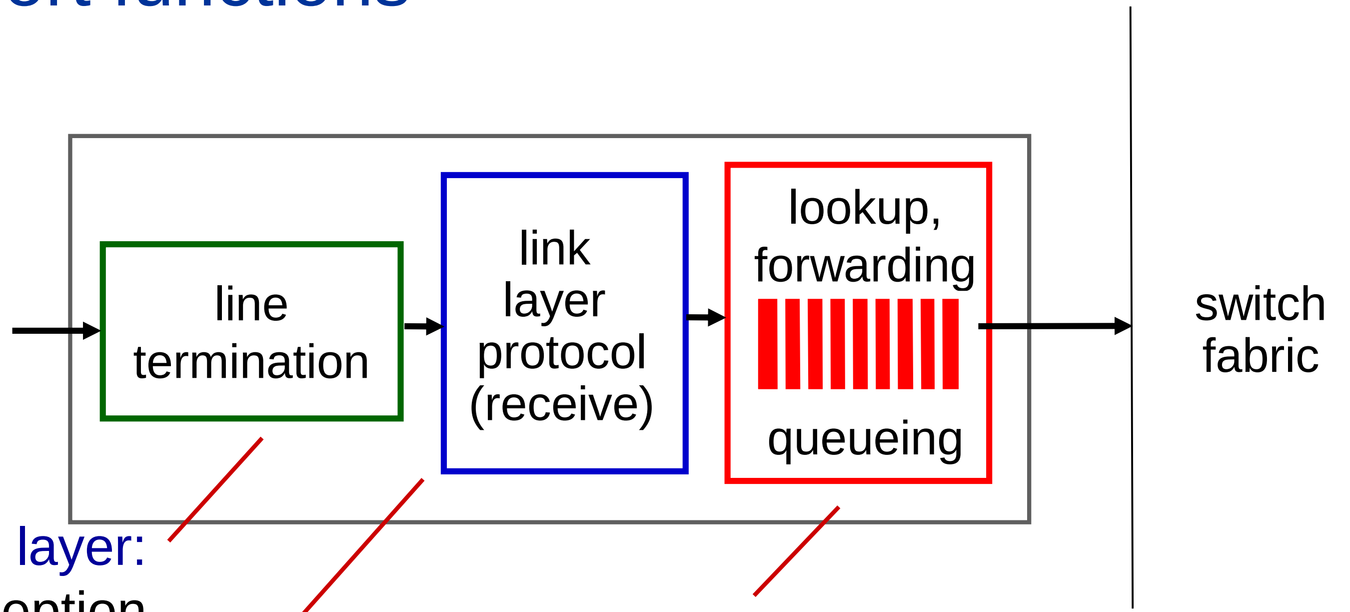
- For data packets: using header field values, lookup output port using forwarding table in input port memory *("match plus action")*
- goal: complete input port processing at 'line speed'
- queuing: if datagrams arrive faster than forwarding rate into switch fabric
- Other functions: checking the packet's version number, checksum (rewrite) and time-to-live field (rewrite). Update counters used for network management

Control packets (for example, packets carrying routing protocol information) are forwarded from an input port to the routing processor.

The number of ports @ a router:

- a relatively small number in enterprise routers.
- hundreds of 10/100 Gbps ports @ ISP's edge router. (e.g., Juniper MX2020 edge router - supports up to 800 100 Gbps Ethernet ports, with overall router system capacity of 800 Tbps)

Input port functions



physical layer:
bit-level reception

data link layer:
e.g., Ethernet

decentralized switching/forwarding*:*

- 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

consider not only destination IP address.

Outline

- Context/overview
- Basic approaches to operating a packet-switching network:
  - Datagrams and
  - Virtual circuits
- Network layer functions: Routing and forwarding
- Overview of Network layer: data plane and control plane
- Network layer: The Data Plane
  - What's inside a router
  - Destination-based forwarding
  - The Internet Protocol (IPv4, DHCP, NAT, IPv6)
  - Generalized forwarding and SDN
- Network layer: The Control Plane
  - Routing in packet networks
  - The SDN control plane
  - ICMP: The Internet Control Message Protocol

Destination-based forwarding

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

network ID

Longest prefix matching

Prefix	Link Interface
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
otherwise	3

A packet arrives at a router with the destination address (DA):

DA: 11001000 00010111 00010110 10100001 Which interface?

DA: 11001000 00010111 00011000 10101010 Which interface?

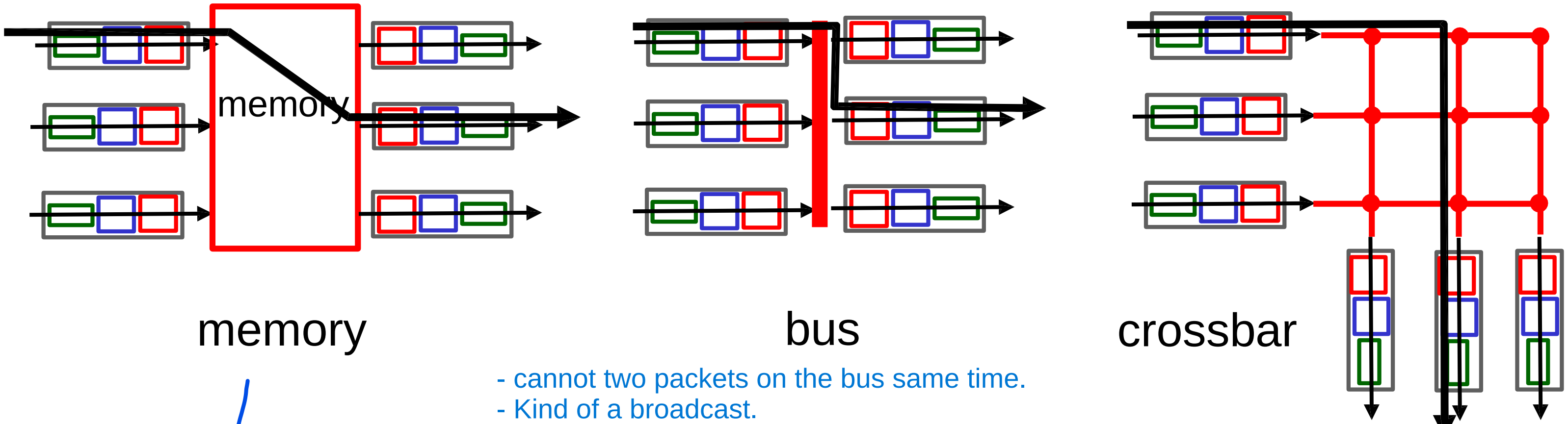
When there are multiple matches, the router uses the **longest prefix matching rule;** it finds the longest matching entry in the table and forwards the packet to the link interface associated with the longest prefix match

A forwarding table in an Internet core router might have more than 400,000 IP prefixes

- 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 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



using enterprise networks.

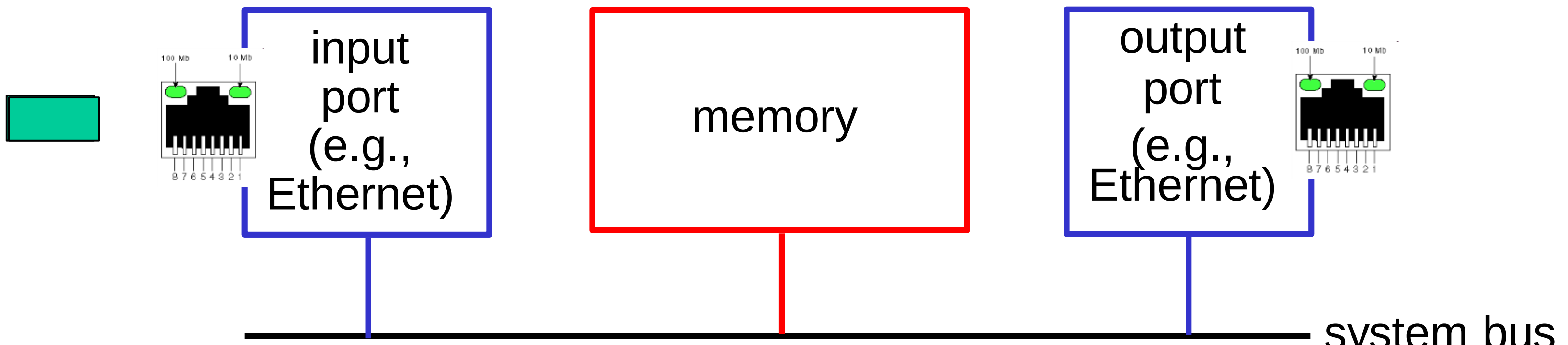
- cannot two packets on the bus same time.
- Kind of a broadcast.
- There are lot of limitations. Speed is low.

- parallel transition can be done, until output ports are become different.

Switching via memory

first generation routers:

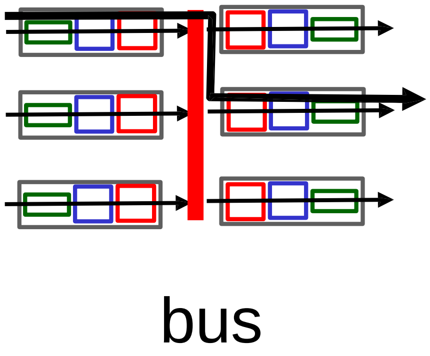
- traditional computers with switching under direct control of CPU
- packet copied to system's memory
- speed limited by memory bandwidth (2 bus crossings per datagram)



Cisco's Catalyst 8500 series switches internally switches packets via a shared memory

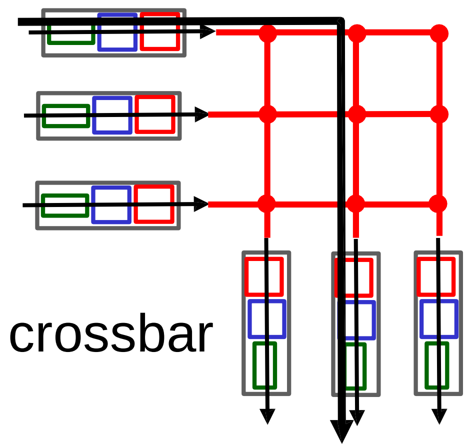
Switching via a bus

- 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



The Cisco 6500 router internally switches packets over a 32-Gbps-backplane bus

Switching via interconnection network



- overcome bus bandwidth limitations
- crossbar, banyan networks, other interconnection nets initially developed to connect processors in multiprocessor

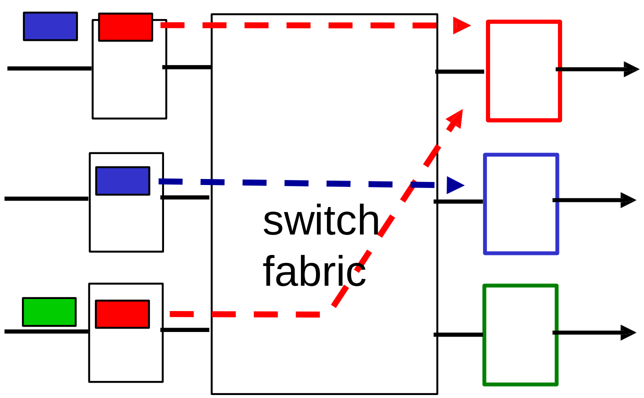
- A crossbar switch is **non-blocking**—a packet being forwarded to an output port will not be blocked from reaching that output port as long as no other packet is currently being forwarded to that output port. However, if two packets from two different input ports are destined to that same output port, then one will have to wait at the input, since only one packet can be sent over any given bus at a time.

- Cisco 12000 series switches use a crossbar switching network. The Cisco 7600 series can be configured to use either a bus or crossbar switch.

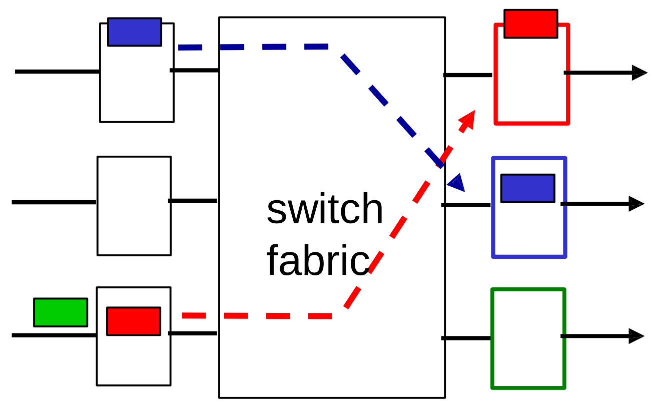
- More sophisticated interconnection networks use multiple stages of switching elements to allow packets from different input ports to proceed towards the same output port at the same time through the multi-stage switching fabric. E.g., The Cisco CRS employs a three-stage non-blocking switching strategy. A router's switching capacity can also be scaled by running multiple switching fabrics in parallel. Input ports and output ports are connected to *N* switching fabrics that operate in parallel. An input port breaks a packet into *K* smaller chunks, and sends ("sprays") the chunks through *K* of these *N* switching fabrics to the selected output port, which reassembles the *K* chunks back into the original packet.

Input port queuing

- fabric slower than input ports combined -> queuing may occur at input queues
  - queuing 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



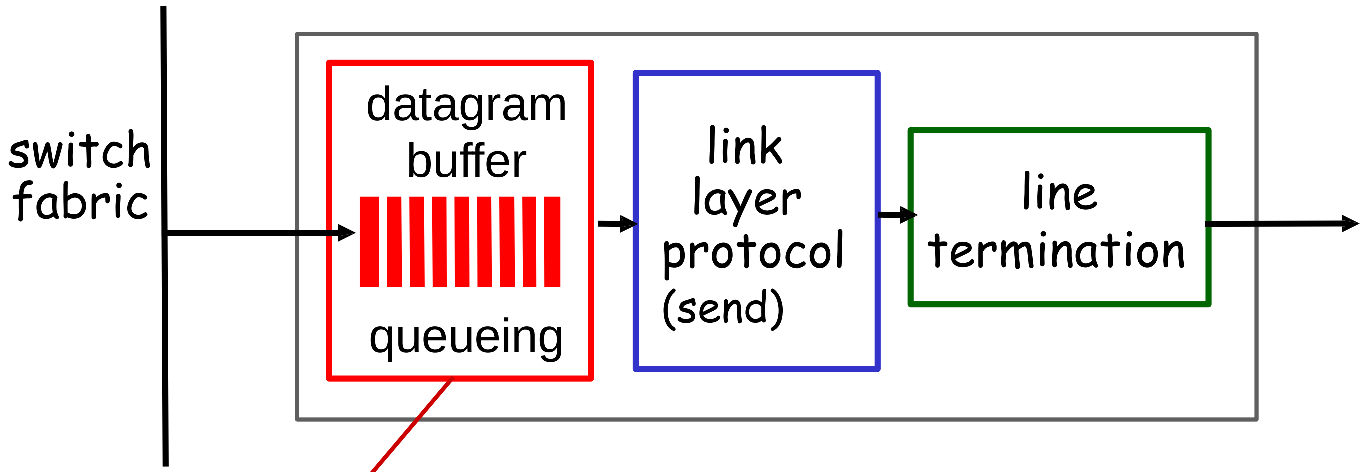
output port contention:
only one red datagram can be transferred.
lower red packet is blocked



one packet time later:
green packet experiences HOL blocking

Made two assumptions:
- Switch fabric is cross bar architecture.
- fetched the destination address of the of the packet as "First come - First fetched" method.

Output ports

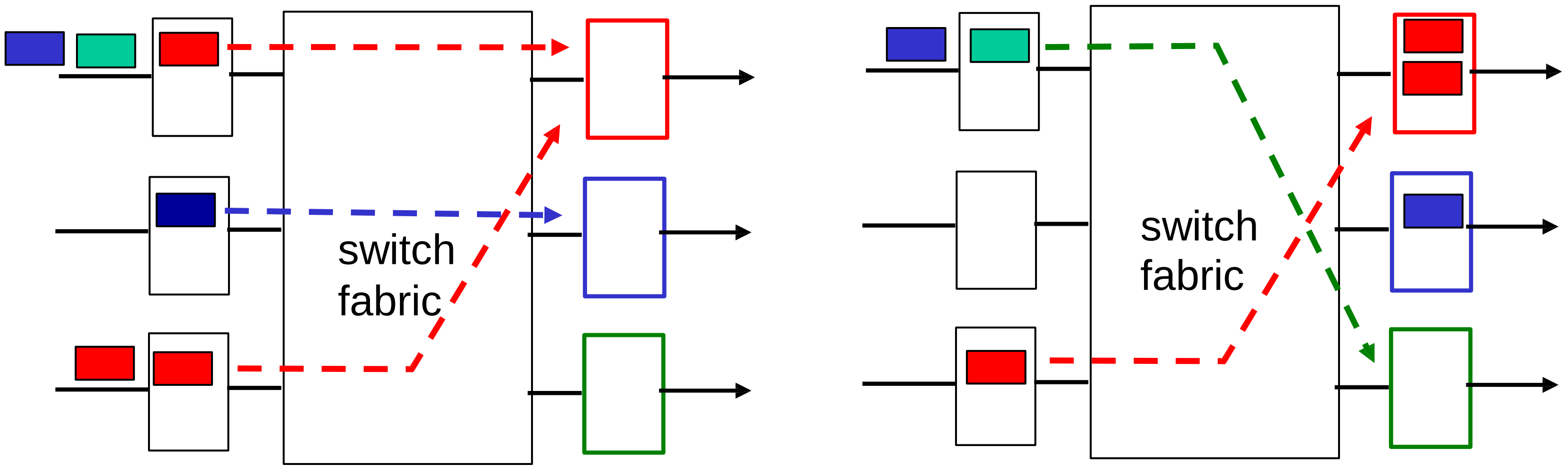


- buffering required when datagrams arrive from fabric faster than the transmission rate
- scheduling discipline chooses among queued datagrams for transmission

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

Priority scheduling – who gets best performance, network neutrality

Output port queueing



at *t,* packets move from input to output

one packet time later

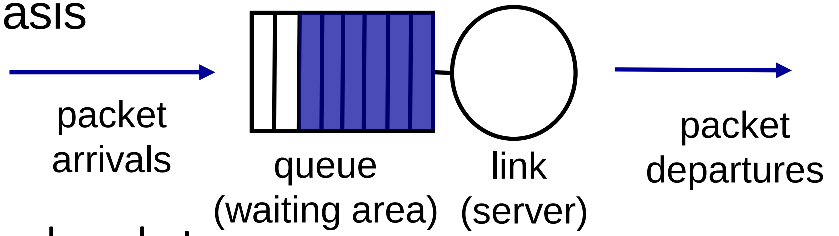
- buffering when arrival rate via switch exceeds output line speed
- queuing (delay) and loss due to output port buffer overflow!

How much buffering?

- RFC 3439 rule of thumb: average buffering equal to “typical” RTT (say 250 msec) times link capacity C
  - e.g., C = 10 Gpbs link: 2.5 Gbit buffer
- recent recommendation: with N flows, buffering equal to $\frac{RTT \cdot C}{\sqrt{N}}$

Scheduling mechanisms

- *Scheduling:* choose the next packet to send on the link (how?)
- *FIFO (first in first out) scheduling:* send in order of arrival to queue
  - *discard policy:* if packet arrives to full queue: which one to discard?
    - *tail drop:* drop arriving packet
    - *priority:* drop/remove on priority basis
    - *random:* drop/remove randomly



- *priority scheduling:* send highest priority queued packet
  - multiple *classes*, with different priorities. Class may depend on marking or other header info, e.g. IP source/dest, port numbers, etc.
- *Round Robin (RR) scheduling:*
  - multiple classes
  - Cyclically scan class queues, sending one complete packet from each class (if available)
- *Weighted Fair Queuing (WFQ):*
  - generalized Round Robin
  - each class gets weighted amount of service in each cycle

Active Queue Management (Buffer Management): RED (Random Early Detection)