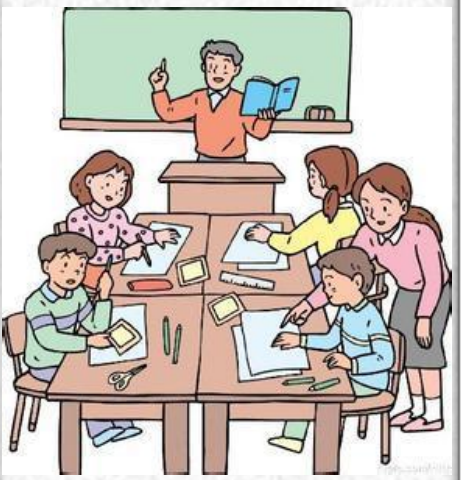
哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

立足航天，服务国防，面向国民经济主战场

《计算机网络》

第4章 网络层

主要内容



本章学习目标

- 理解网络层服务
- 理解虚电路网络与数据报网络
- 掌握路由器体系结构
- 掌握IP协议
  - IP数据报
  - IP地址与子网划分
  - CIDR与路由聚合
- 掌握DHCP、NAT、ICMP、ARP等协议
- 掌握典型路由算法和路由协议

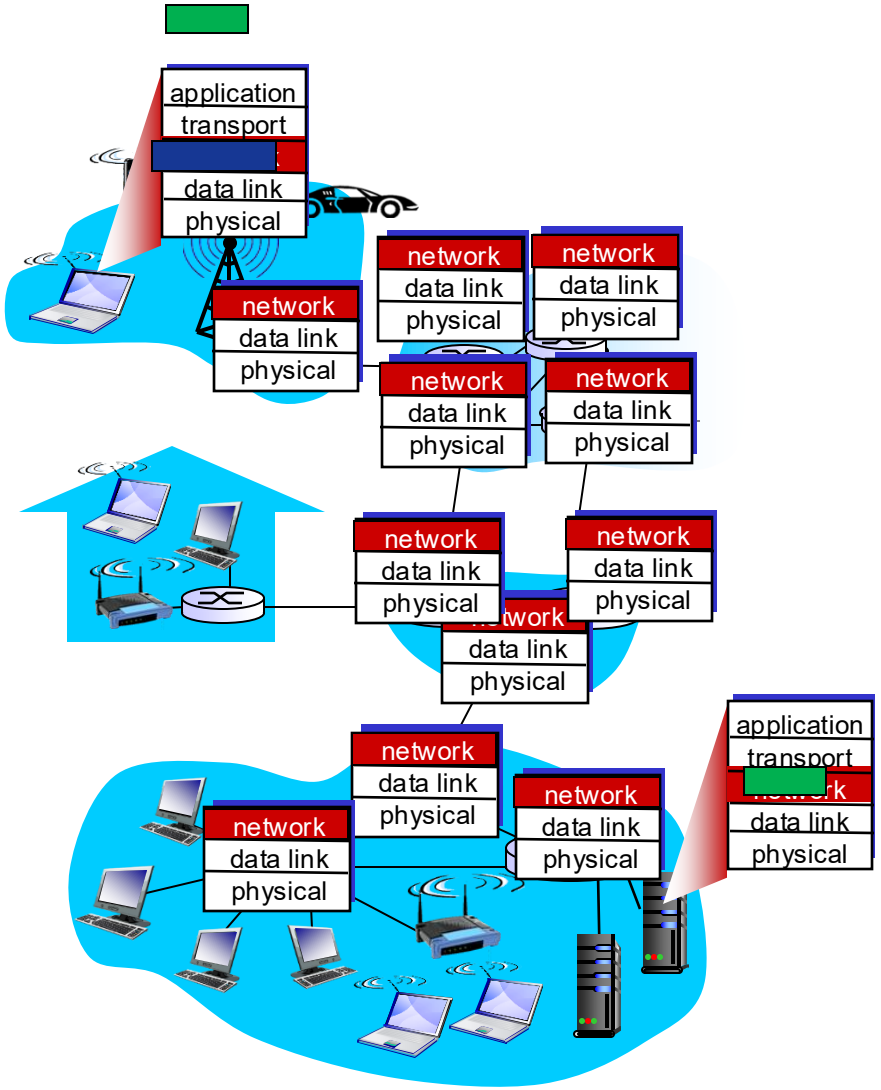
主要内容

哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY
立足航天，服务国防，面向国民经济主战场

4.1 网络层服务

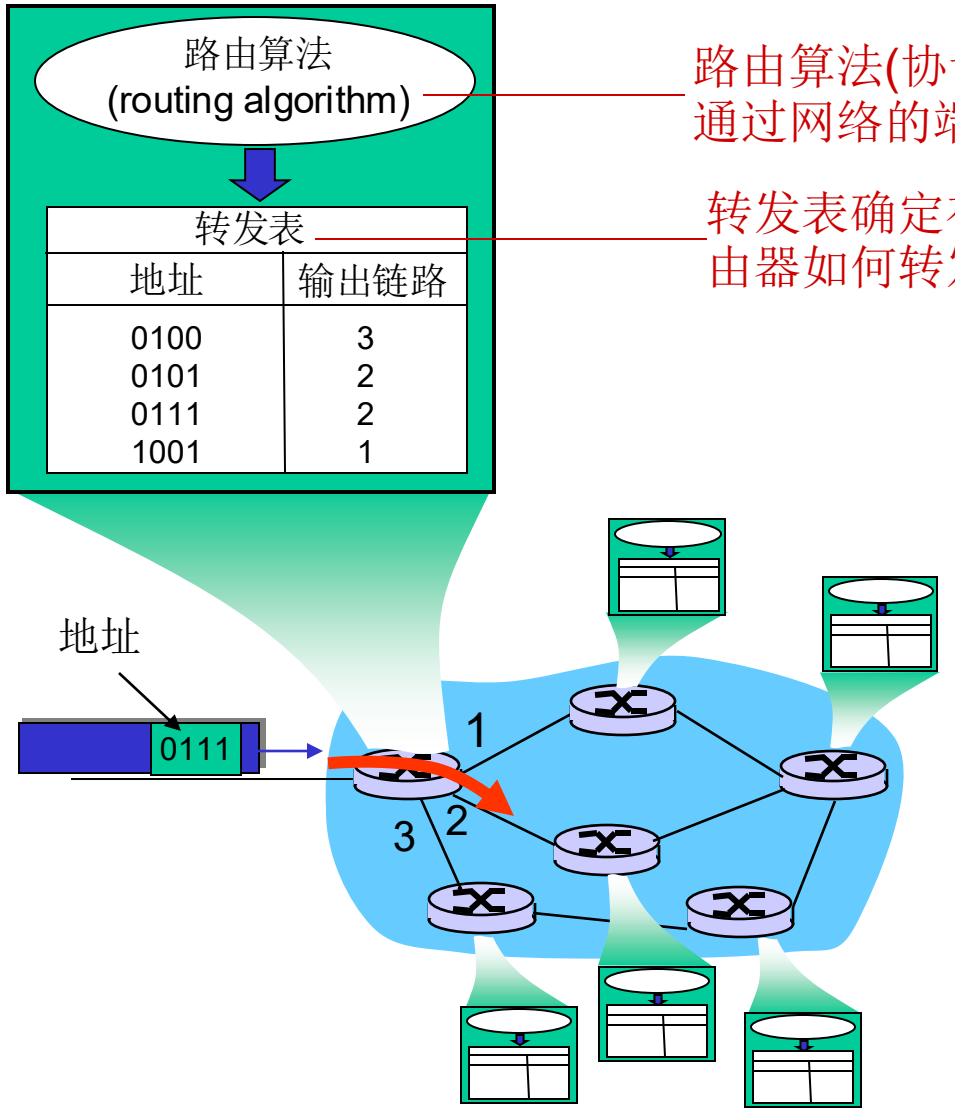
网络层

- 从发送主机向接收主机传送数据段（segment）
- 发送主机：将数据段封装到数据报（datagram）中
- 接收主机：向传输层交付数据段（segment）
- 每个主机和路由器都运行网络层协议
- 路由器检验所有穿越它的IP数据报的头部域
  - 决策如何处理IP数据报



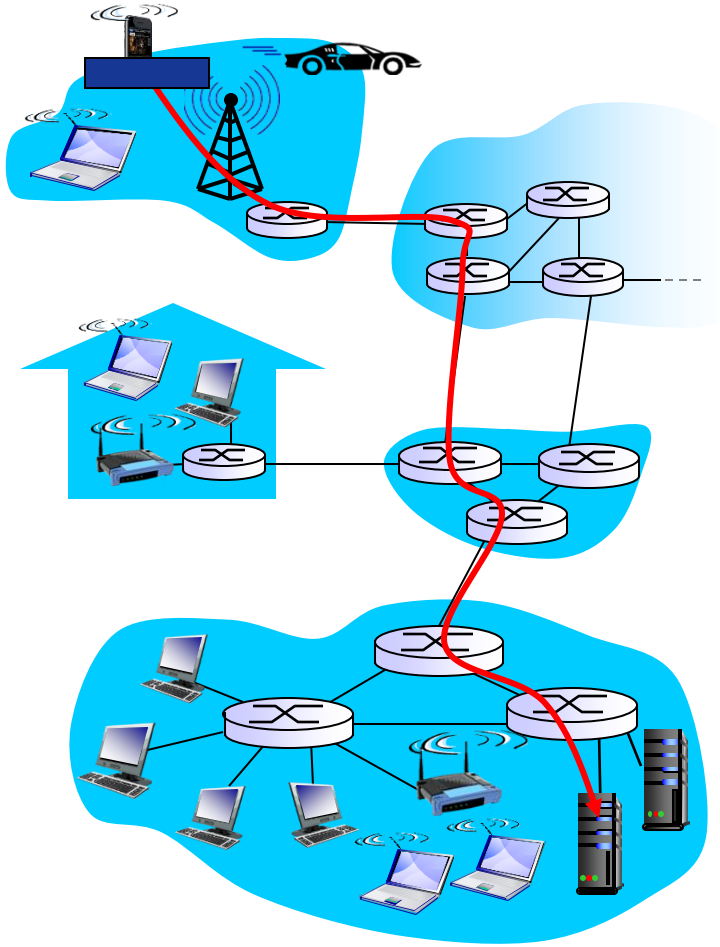
网络层核心功能-转发与路由

- 转发(forwarding): 将分组从路由器的输入端口转移到合适的输出端口
- 路由(routing): 确定分组从源到目的经过的路径
  - 路由算法（routing algorithms）



网络层核心功能-连接建立

- 某些网络的重要功能:
  - ATM, 帧中继, X.25
- 数据分组传输之前两端主机需要首先建立虚拟/逻辑连接
  - 网络设备（如路由器）参与连接的建立
- 网络层连接与传输层连接的对比:
  - 网络层连接: 两个主机之间 (路径上的路由器等网络设备参与其中)
  - 传输层连接: 两个应用进程之间（对中间网络设备透明）



网络层服务模型

Q: 网络层为发送端（主机）到接收端（主机）的数据报传送“通道(channel)”提供什么样的服务模型(service model)?

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

网络层服务模型

- 无连接服务(connection-less service):
  - 不事先为系列分组的传输确定传输路径
  - 每个分组独立确定传输路径
  - 不同分组可能传输路径不同
  - 数据报网络(datagram network)
- 连接服务(connection service):
  - 首先为系列分组的传输确定从源到目的经过的路径(建立连接)
  - 然后沿该路径（连接）传输系列分组
  - 系列分组传输路径相同
  - 传输结束后拆除连接
  - 虚电路网络(virtual-circuit network)

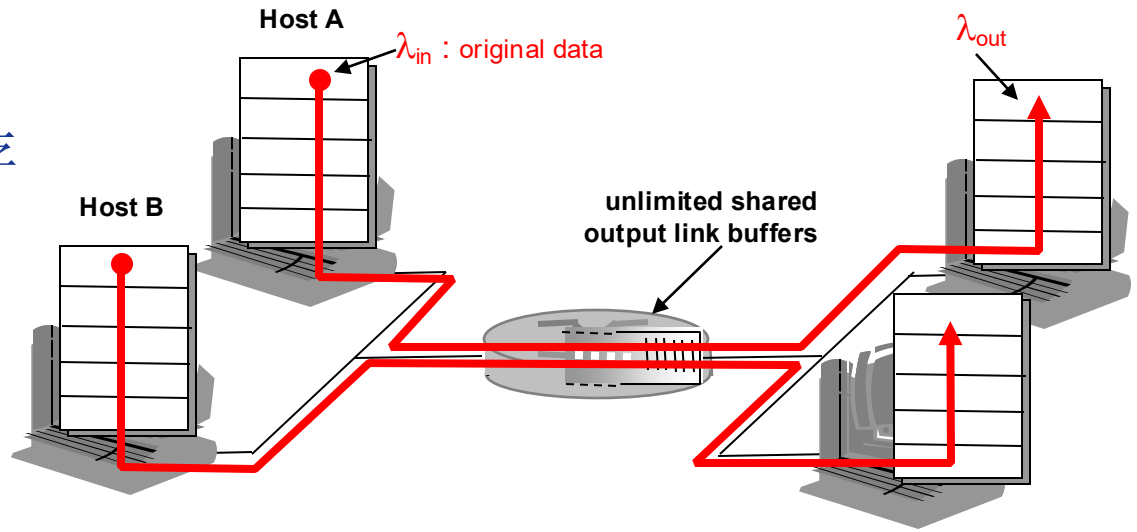
拥塞控制

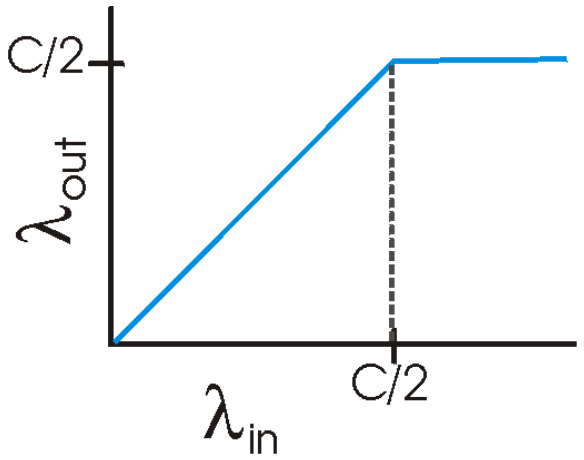
拥塞(Congestion)

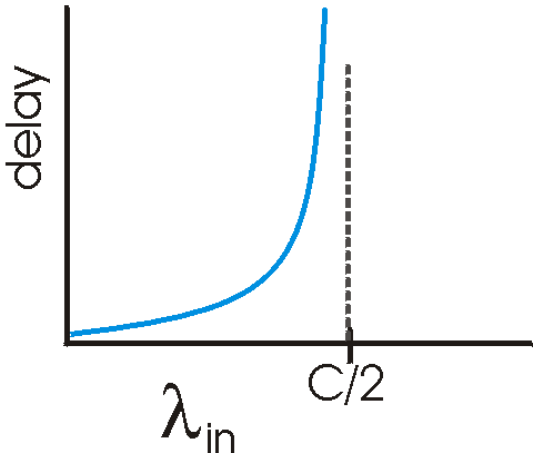
- 非正式定义：“太多发送主机发送了太多数据或者发送速度太快，以至于网络无法处理”
- 表现：
  - 分组丢失（路由器缓存溢出）
  - 分组延迟过大（在路由器缓存中排队）
- 拥塞控制 vs. 流量控制
- A top-10 problem.

拥塞的成因和代价：场景1

- 两个senders,两个receivers
- 一个路由器, 无限缓存
- 没有重传



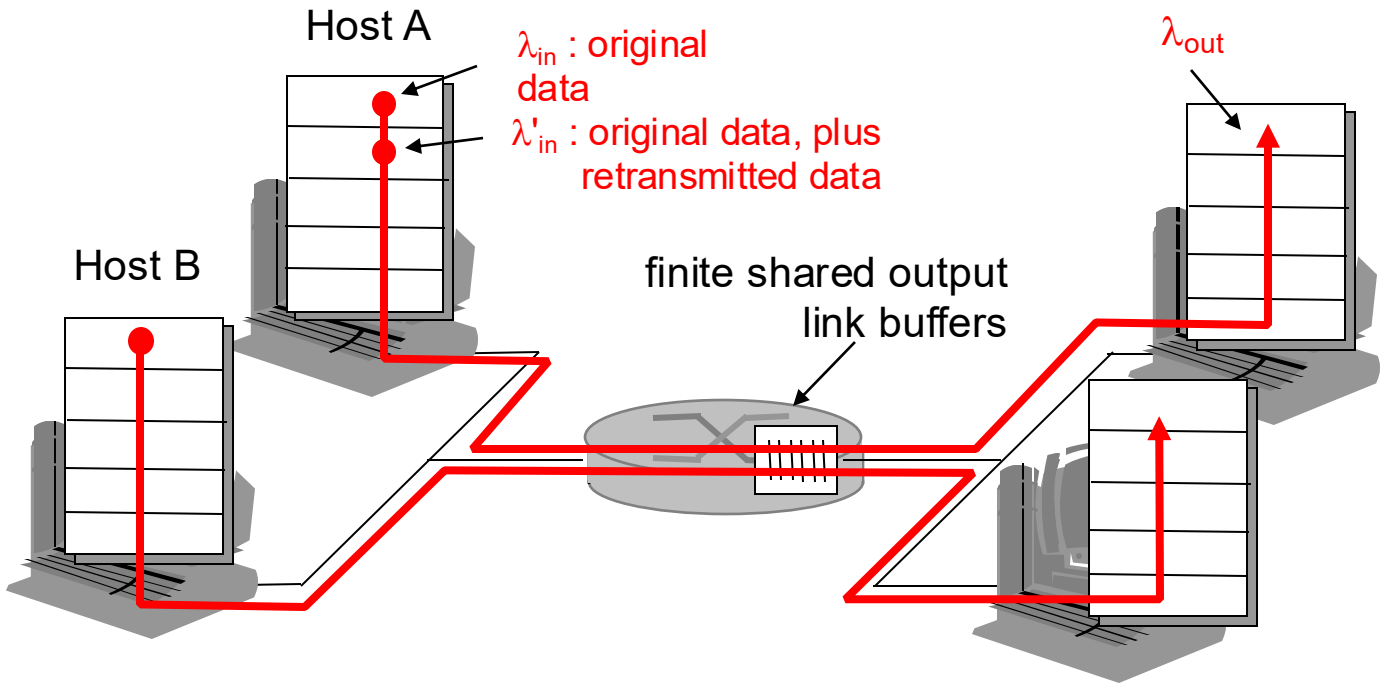




- 拥塞时分组延迟太大
- 达到最大 throughput

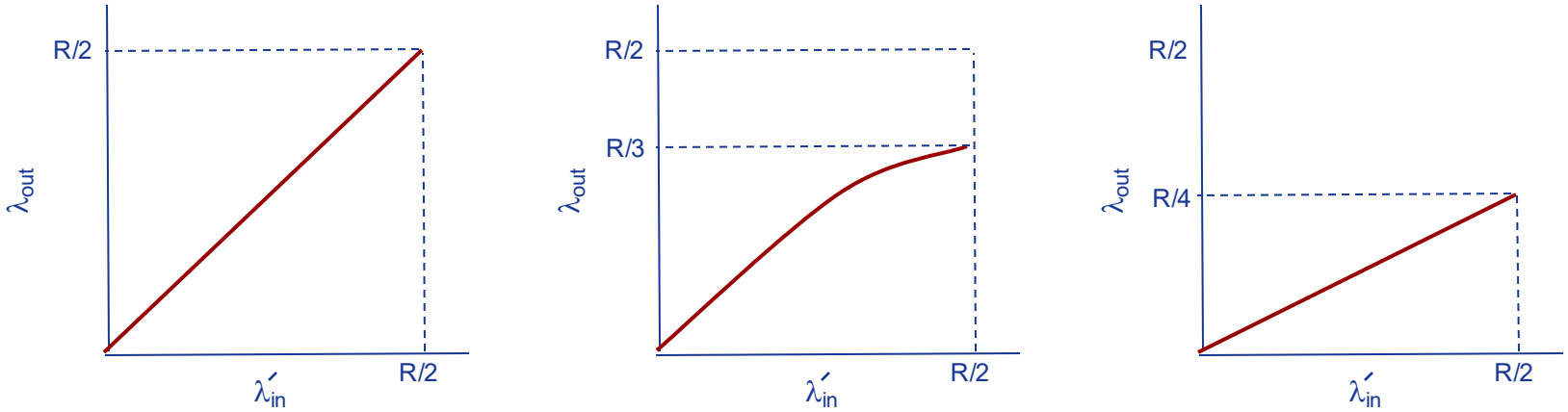
拥塞的成因和代价：场景2

- 一个路由器，有限buffers
- Sender重传分组



拥塞的成因和代价：场景2

- 情况a: Sender能够通过某种机制获知路由器buffer信息，有空间才发：$\lambda_{in} = \lambda_{out}$ (goodput)
- 情况b: 丢失后才重发: $\lambda'_{in} > \lambda_{out}$
- 情况c：分组丢失和定时器超时后都重发，λ'_{in} 变得更大



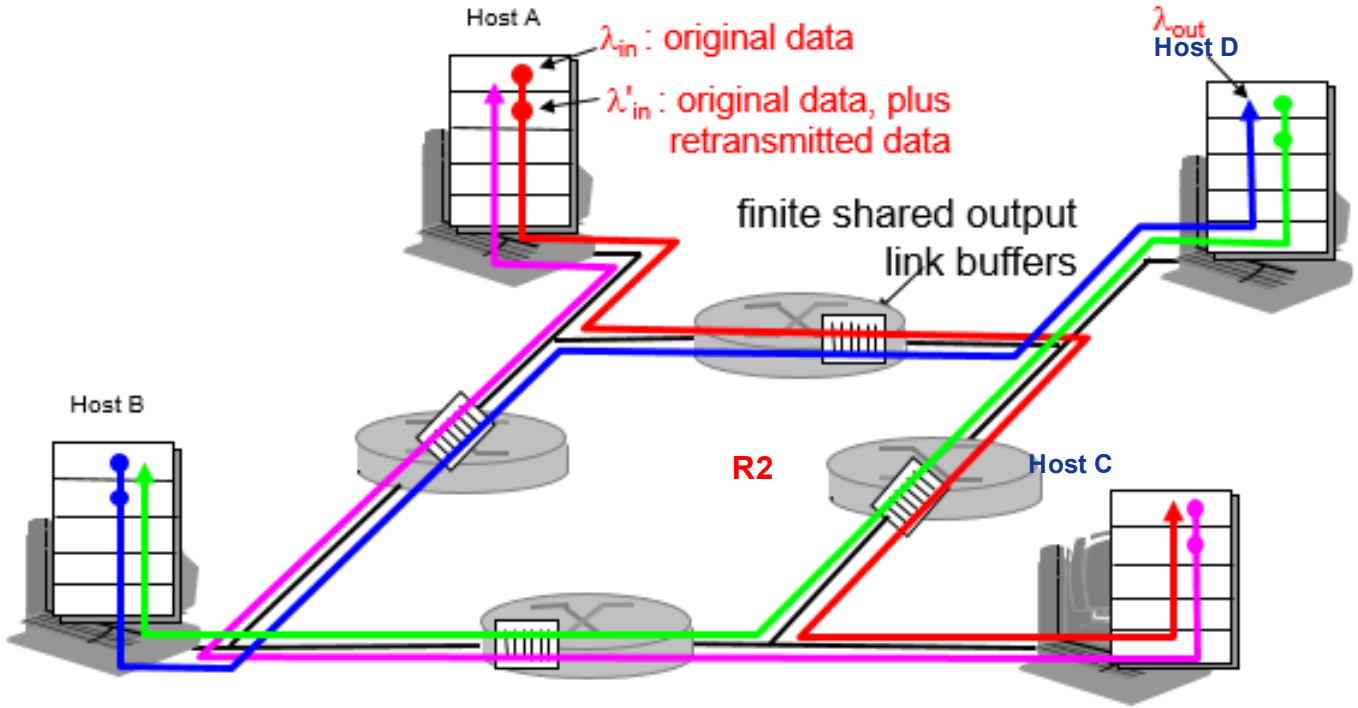
拥塞的代价:

- 对给定的"goodput"，要做更多的工作 (重传)
- 造成资源的浪费

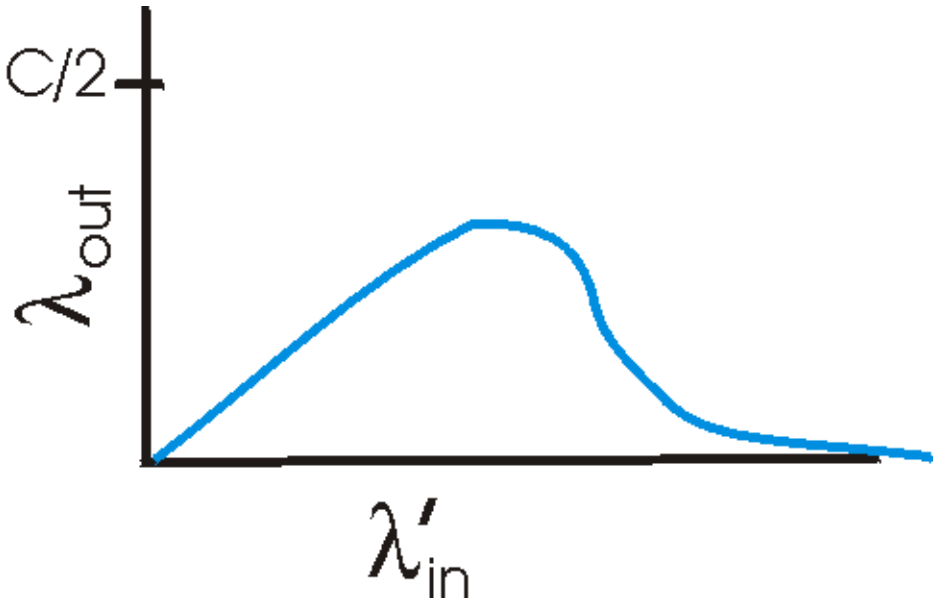
拥塞的成因和代价：场景3

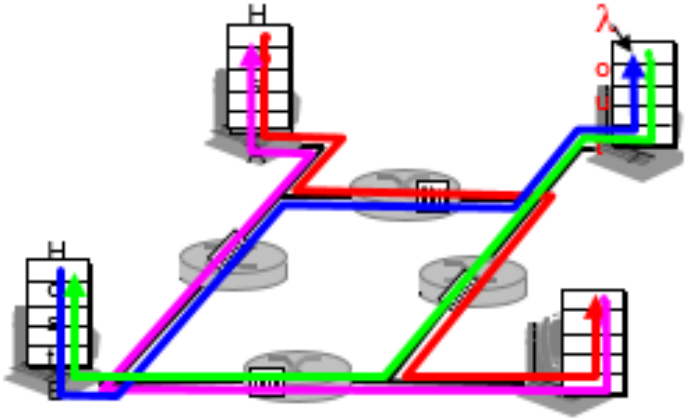
- 四个发送方
- 多跳
- 超时/重传

Q: 随着λ_{in}和λ'_{in}不断增加，会怎么样？



拥塞的成因和代价：场景3





拥塞的另一个代价：

- 当分组被丢弃时，任何用于该分组的“上游”传输能力全都被浪费掉

拥塞控制的方法

❖网络层（辅助）拥塞控制：

- 路由器向发送方显式地反馈网络拥塞信息
- 简单的拥塞指示(1bit)：SNA, DECbit, TCP/IP ECN, ATM)
- 指示发送方应该采取何种速率

❖传输层端到端拥塞控制：

- 网络层不需要显式的提供支持
- 端系统通过观察loss，delay等网络行为判断是否发生拥塞
- TCP采取这种方法

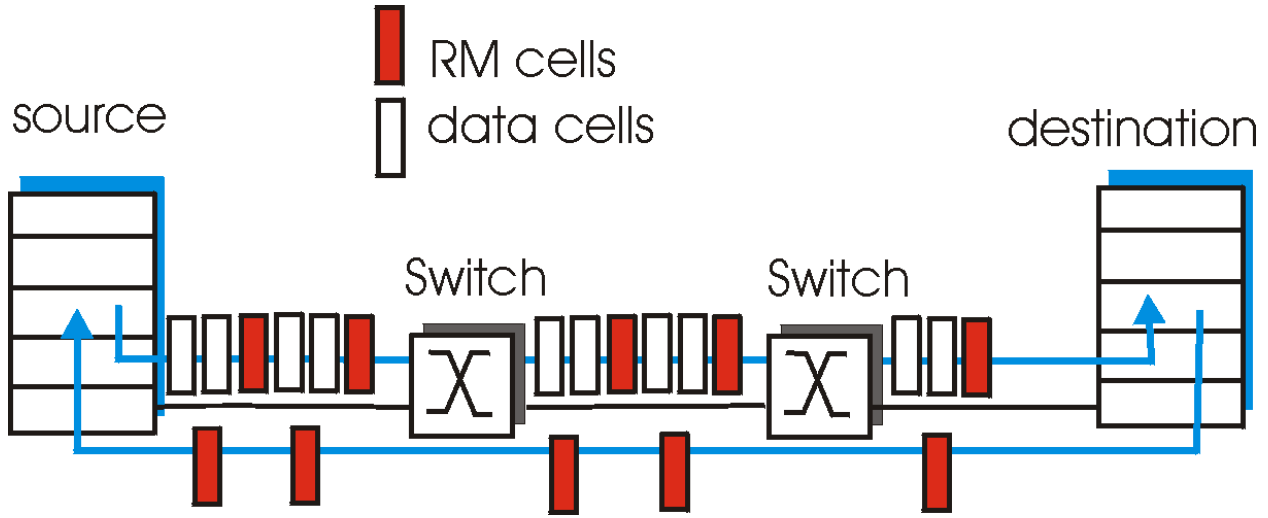
网络层拥塞控制策略

- 流量感知路由
- 准入控制
- 流量调节
  - 抑制分组
  - 背压
- 负载脱落

案例：ATM ABR拥塞控制

- ABR：available bit rate
  - “弹性服务”
  - 如果发送方路径“underloaded”
    - 使用可用带宽
  - 如果发送方路径拥塞
    - 将发送速率降到最低保障速率
- RM(resource management) cells
  - 发送方发送
  - 交换机设置RM cell位(网络辅助)
    - NI bit: 速率不许增长
    - CI bit: 拥塞指示
  - RM cell由接收方返回给发送方

案例： ATM ABR拥塞控制



- 在RM cell中有显式的速率(ER)字段：两个字节
  - 拥塞的交换机可以将ER置为更低的值
  - 发送方获知路径所能支持的最小速率
- 数据cell中的EFCI位: 拥塞的交换机将其设为1
  - 如果RM cell前面的data cell的EFCI位被设为1，那么发送方在返回的RM cell中置CI位

4.2 虚电路网络与数据报网络

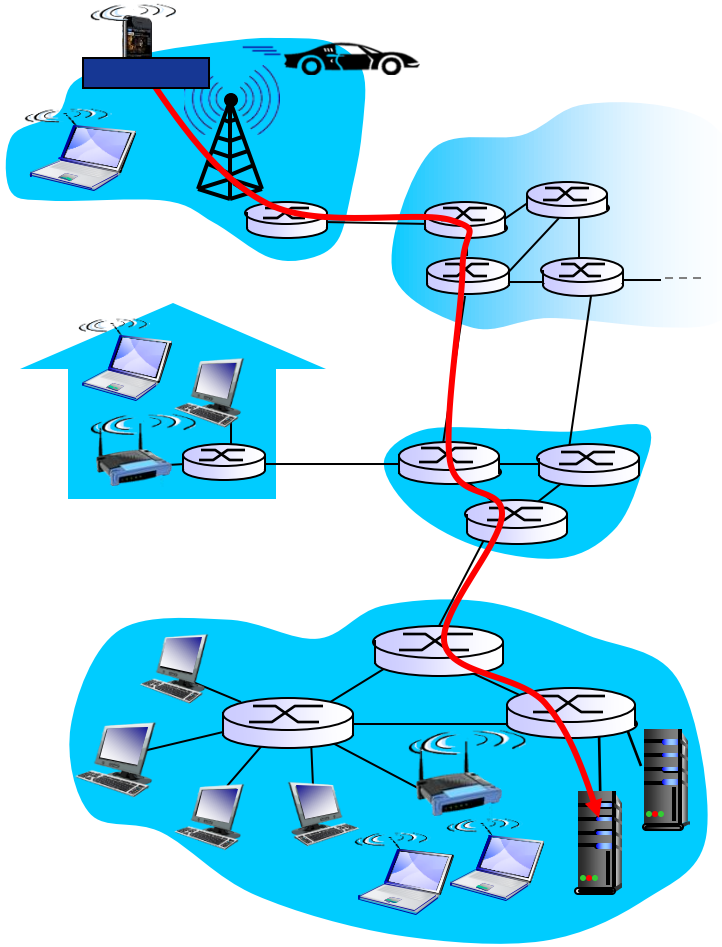
连接服务与无连接服务

- 数据报(datagram)网络与虚电路(virtual-circuit)网络是典型两类分组交换网络
- 数据报网络提供网络层无连接服务
- 虚电路网络提供网络层连接服务
- 类似于传输层的无连接服务（UDP）和面向连接服务（TCP），但是网络层服务：
  - 主机到主机服务
  - 网络核心实现

虚电路(Virtual circuits)

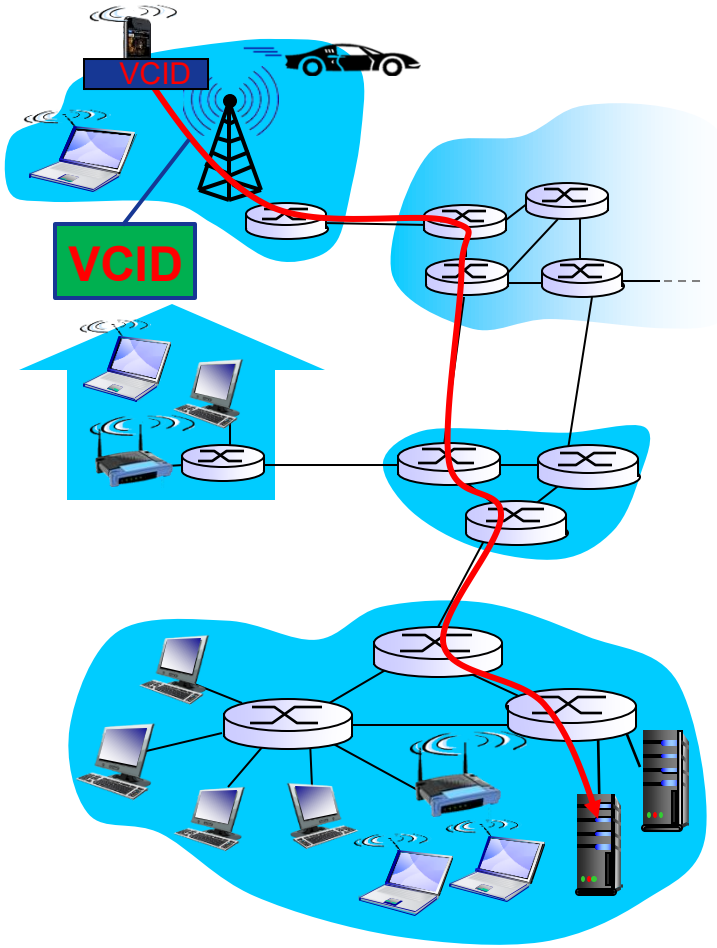
虚电路：一条从源主机到目的主机，类似于电路的路径(逻辑连接)

- 分组交换
- 每个分组的传输利用链路的全部带宽
- 源到目的路径经过的网络层设备共同完成虚电路功能



虚电路(Virtual circuits)

- 通信过程：
  - 呼叫建立(call setup)→数据传输→拆除呼叫
- 每个分组携带虚电路标识(VC ID)，而不是目的主机地址
- 虚电路经过的每个网络设备（如路由器），维护每条经过它的虚电路连接状态
- 链路、网络设备资源(如带宽、缓存等)可以面向VC进行预分配
  - 预分配资源=可预期服务性能
  - 如ATM的电路仿真（CBR）



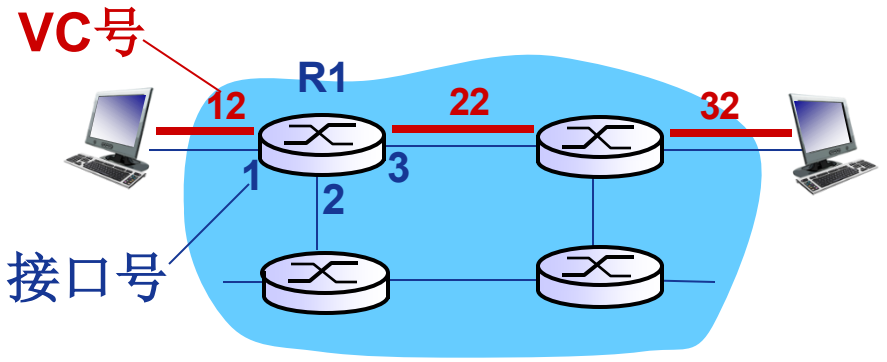
VC的具体实现

每条虚电路包括:

1. 从源主机到目的主机的一条路径
2. 虚电路号（VCID）， 沿路每段链路一个编号
3. 沿路每个网络层设备（如路由器），利用转发表记录经过的每条虚电路

- 沿某条虚电路传输的分组，携带对应虚电路的VCID，而不是目的地址
- 同一条VC，在每段链路上的VCID通常不同
  - 路由器转发分组时依据转发表改写/替换虚电路号

VC转发表



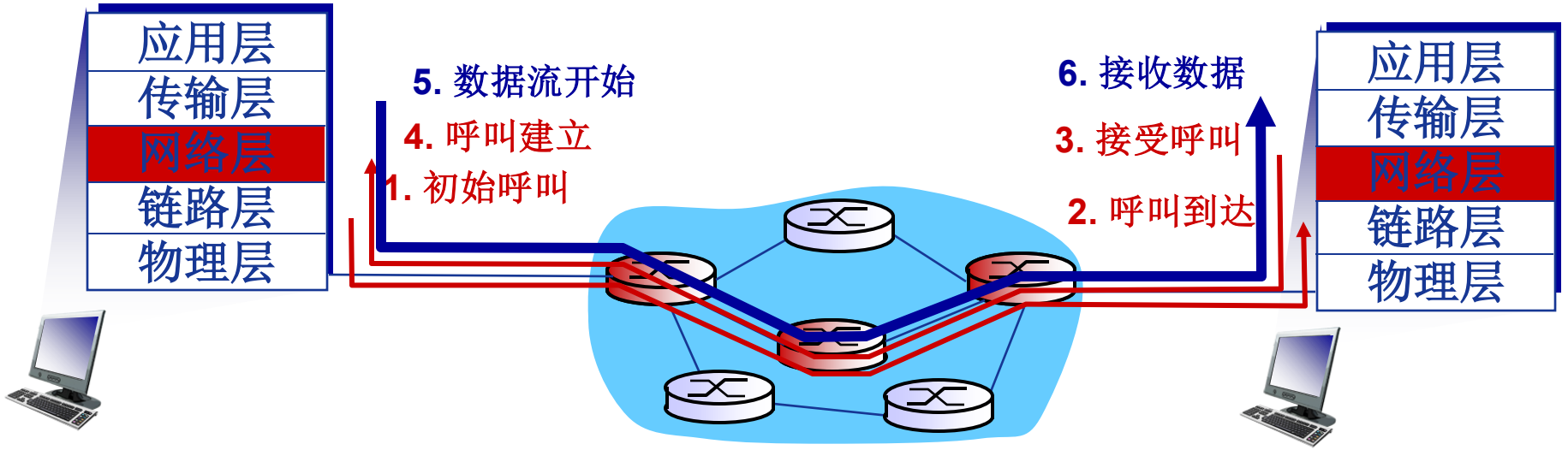
路由器R1的VC转发表:

输入接口	输入VC #	输出接口	输出VC #
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
…	…	…	…

VC路径上每个路由器都需要维护VC连接的状态信息！

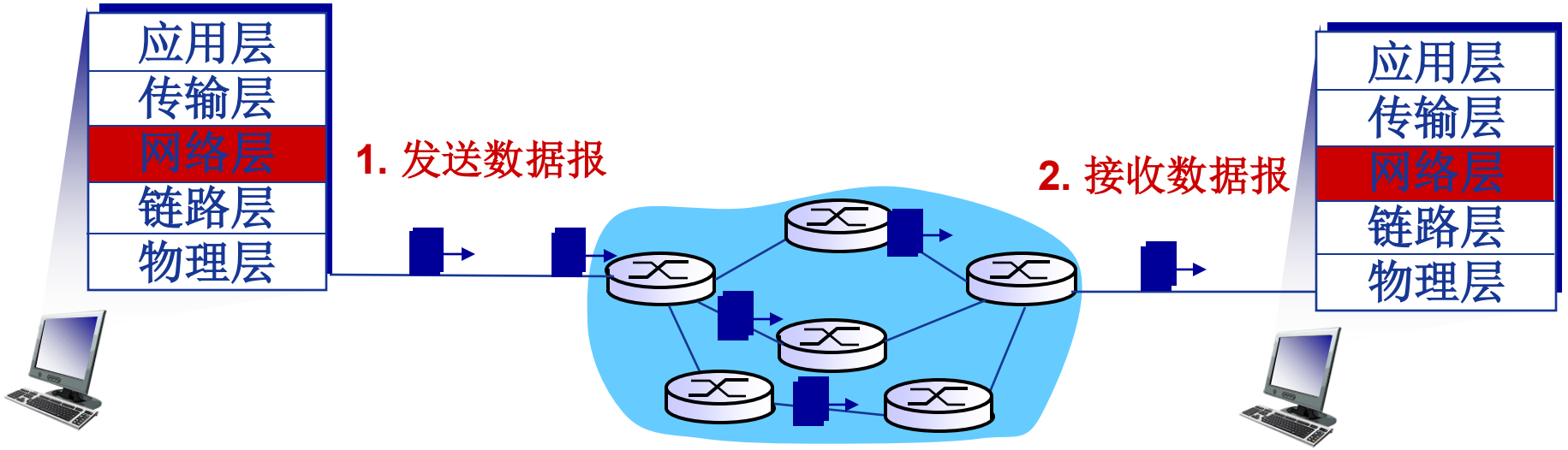
虚电路信令协议(signaling protocols)

- 用于VC的建立、维护与拆除
  - 路径选择
- 应用于虚电路网络
  - 如ATM、帧中继(frame-relay)网络等
- 目前的Internet不采用

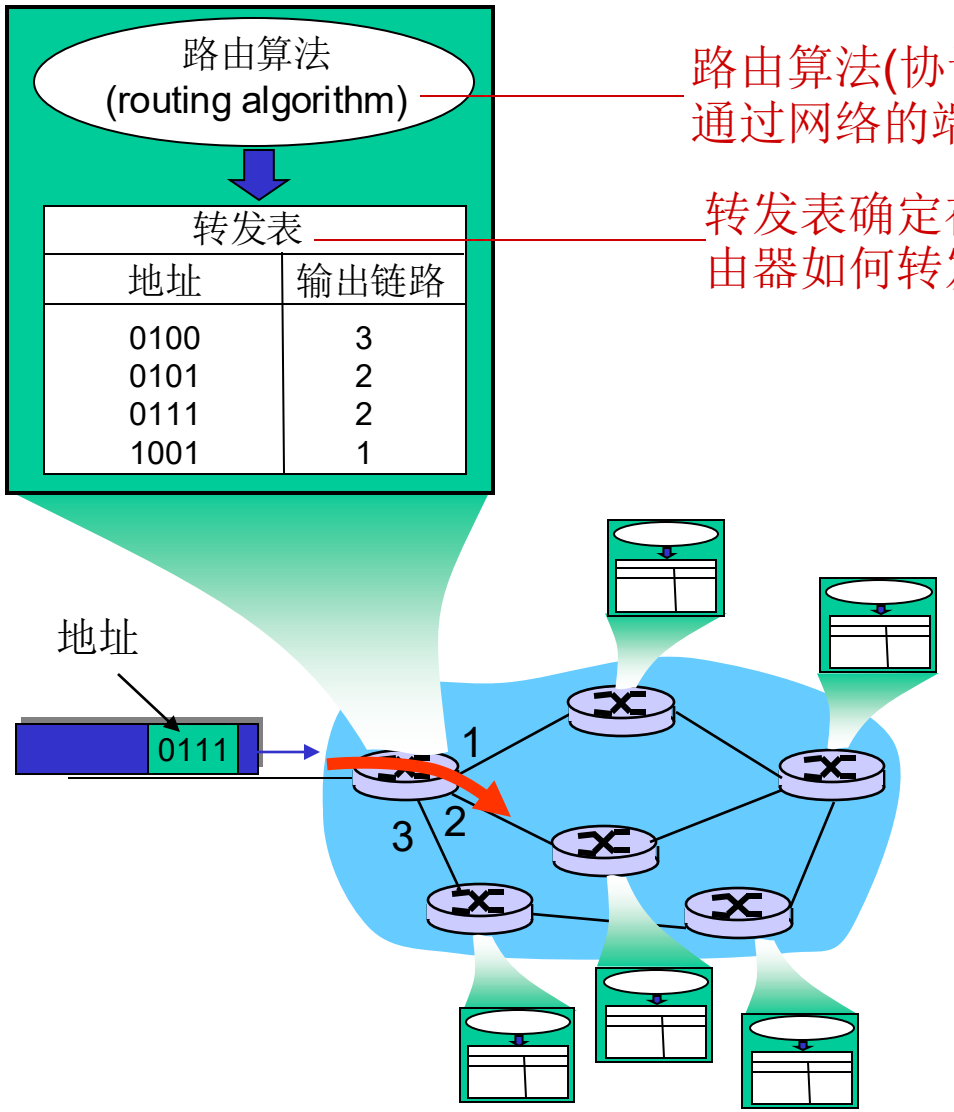


数据报网络

- 网络层无连接
- 每个分组携带目的地址
- 路由器根据分组的目的地址转发分组
  - 基于路由协议/算法构建转发表
  - 检索转发表
  - 每个分组独立选路



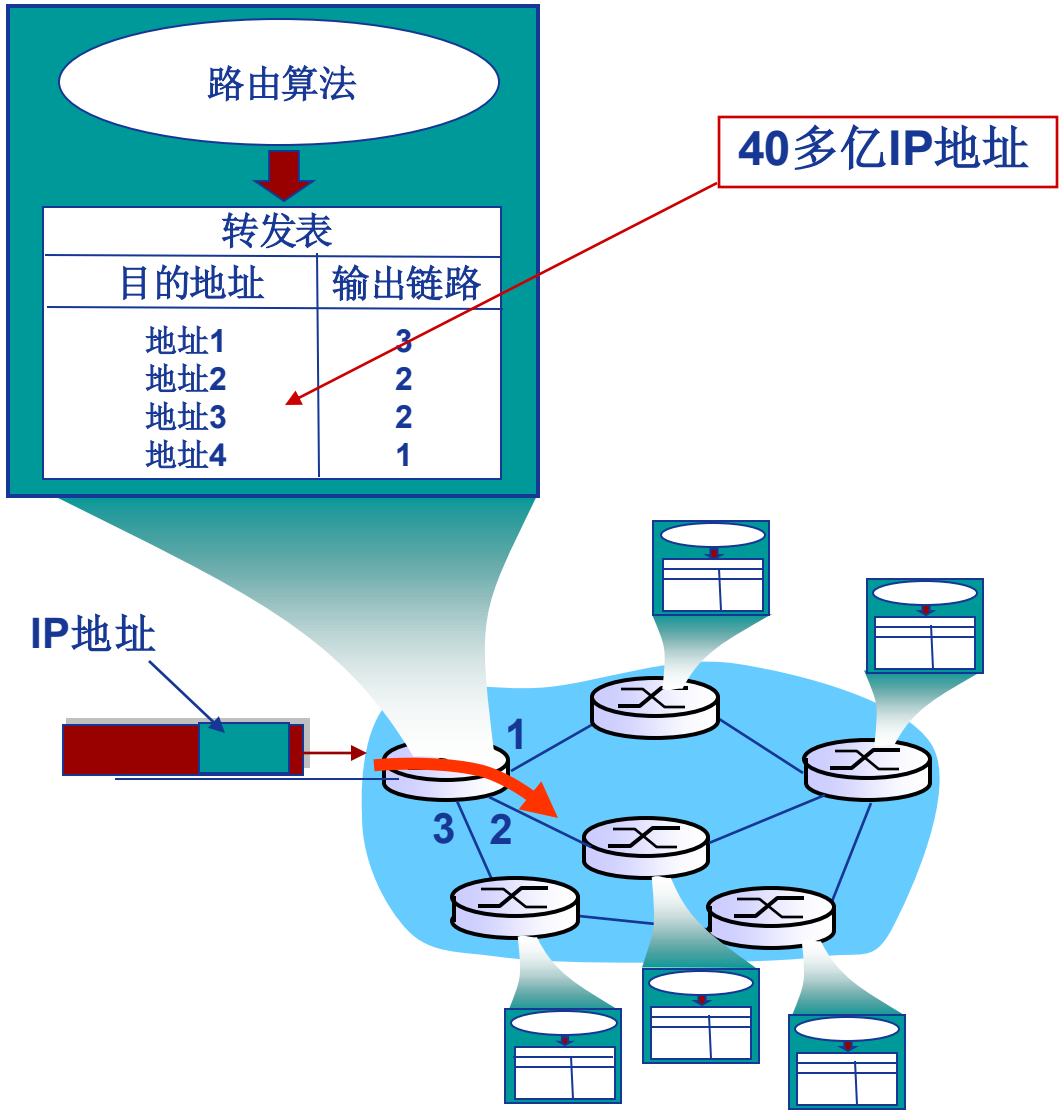
数据报转发表



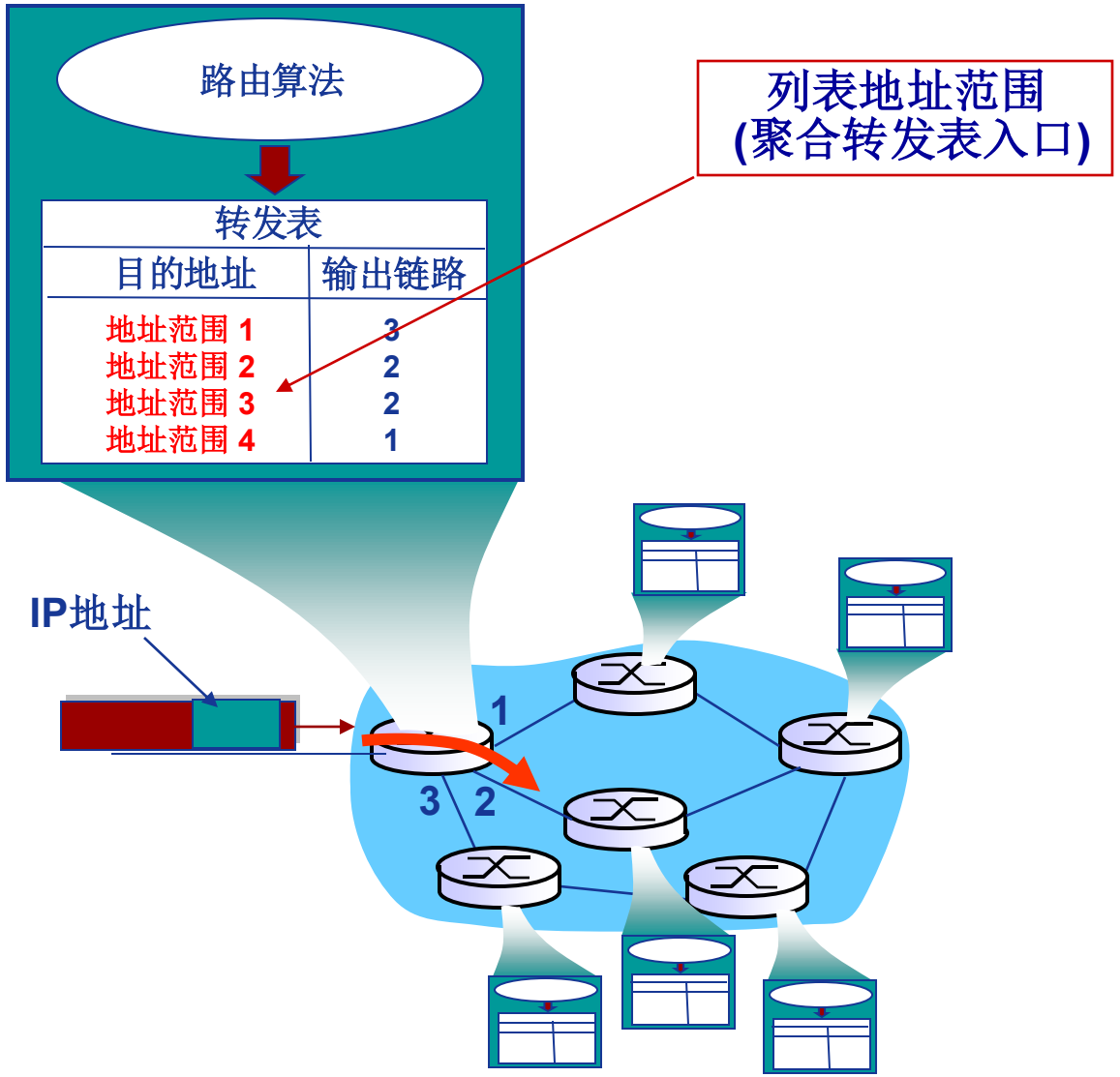
路由算法(协议)确定
通过网络的端到端路径

转发表确定在本路
由器如何转发分组

数据报转发表



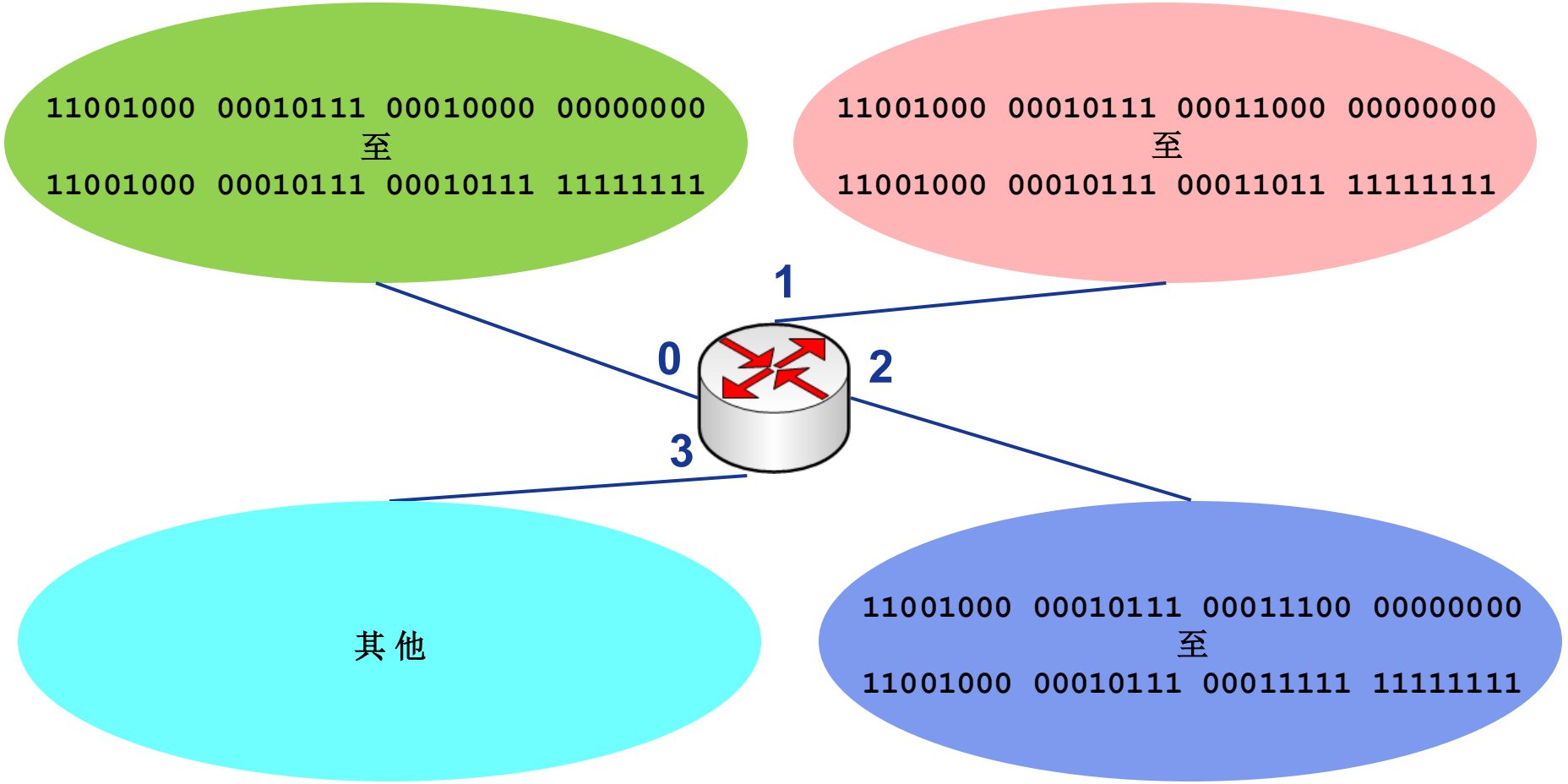
数据报转发表



数据报转发表

目的地址范围	链路接口
11001000 00010111 00010000 00000000 至 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 至 11001000 00010111 00011011 11111111	1
11001000 00010111 00011100 00000000 至 11001000 00010111 00011111 11111111	2
其他	3

数据报转发表



Q: 如果地址范围划分的不是这么“完美”会怎么样？

最长前缀匹配优先

例如：

目的地址范围	链路接口
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
其他	3

DA: 11001000 00010111 00010110 10100001　　从哪个接口转发？ A:0

DA: 11001000 00010111 00011000 10101010　　从哪个接口转发？ A:1

最长前缀匹配优先

在检索转发表时，优先选择与分组目的地址匹配前缀最长的入口（entry）。

数据报网络 or VC网络?

Internet (数据报网络)

- 计算机之间的数据交换
  - “弹性”服务，没有严格时间需求
- 链路类型众多
  - 特点、性能各异
  - 统一服务困难
- “智能”端系统 (计算机)
  - 可以自适应、性能控制、差错恢复
- ***简化网络，复杂“边缘”***

ATM (VC网络)

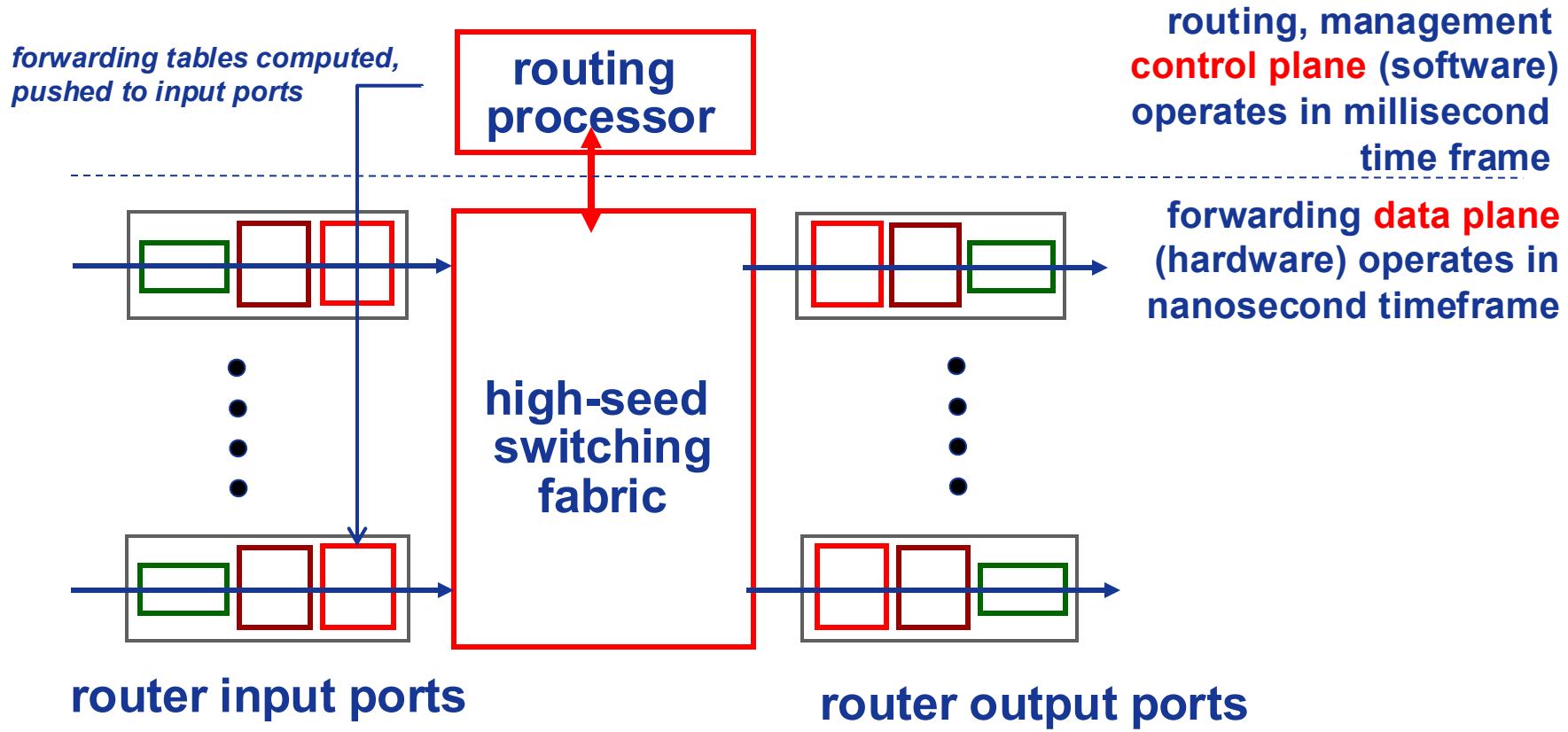
- 电话网络演化而来
- 核心业务是实时对话：
  - 严格的时间、可靠性需求
  - 需要有保障的服务
- “哑(dumb)” 端系统（非智能）
  - 电话机
  - 传真机
- ***简化“边缘”，复杂网络***

4.3 路由器体系结构

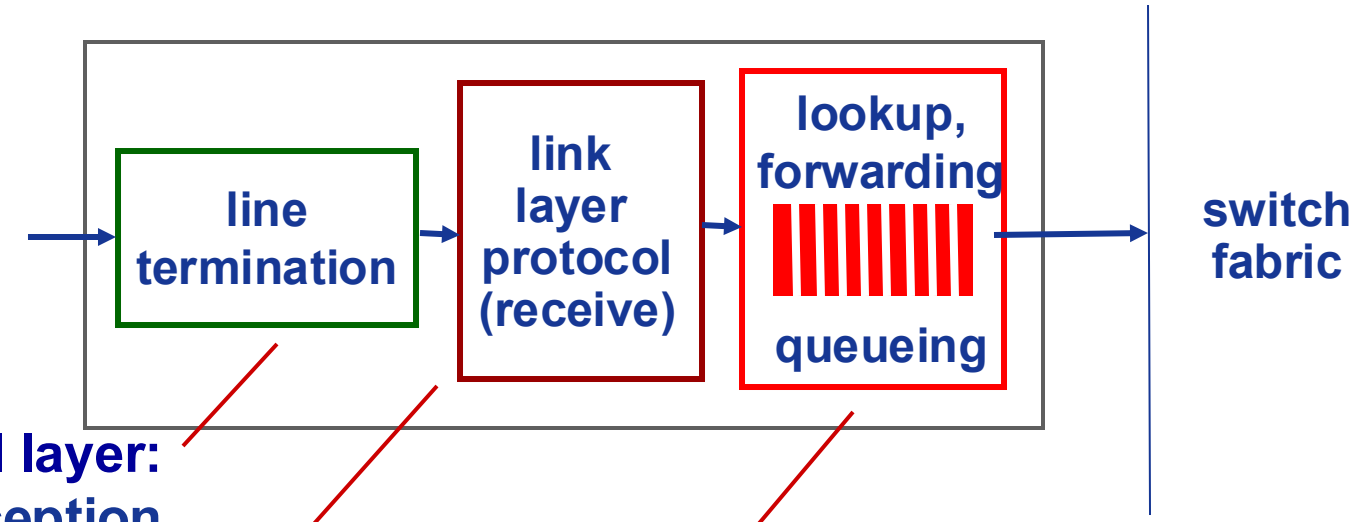
Router architecture overview

two key router functions:

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



Input port functions



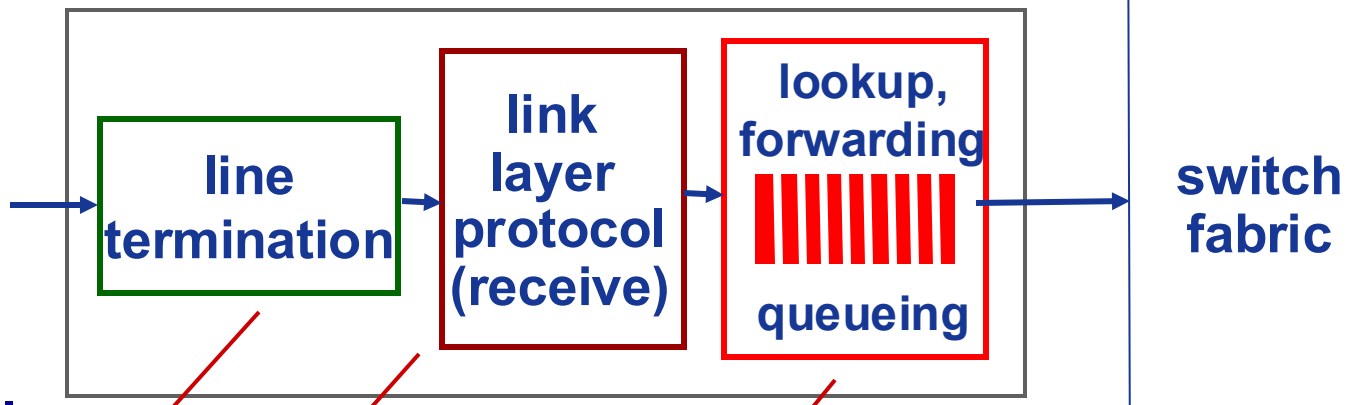
physical layer:
bit-level reception

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

decentralized switching:

- given datagram dest., 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

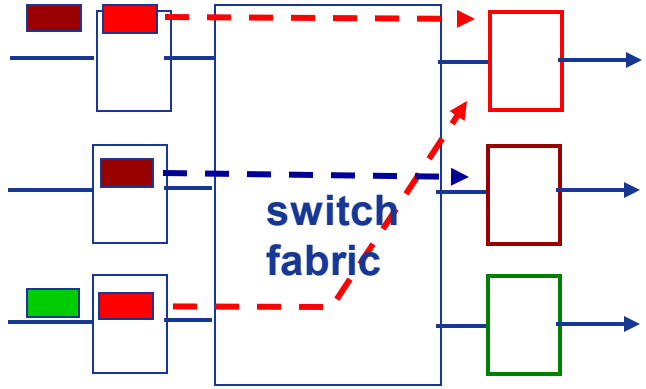
data link layer:
e.g., Ethernet
see chapter 5

decentralized switching*:*

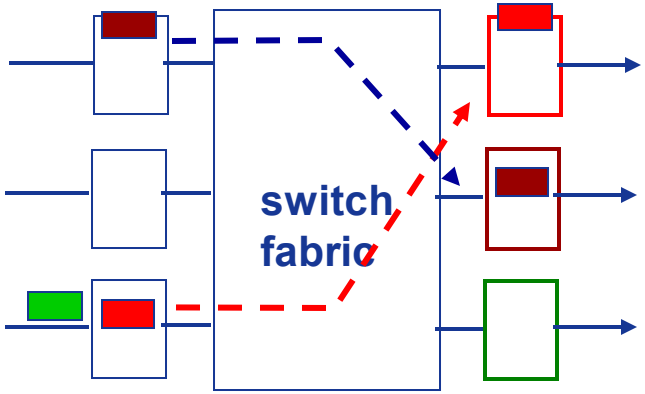
- using header field values, lookup output port using forwarding table in input port memory *("match plus action")*
- *destination-based forwarding:* forward based only on destination IP address (traditional)
- *generalized forwarding:* forward based on any set of header field values

Input port queuing

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



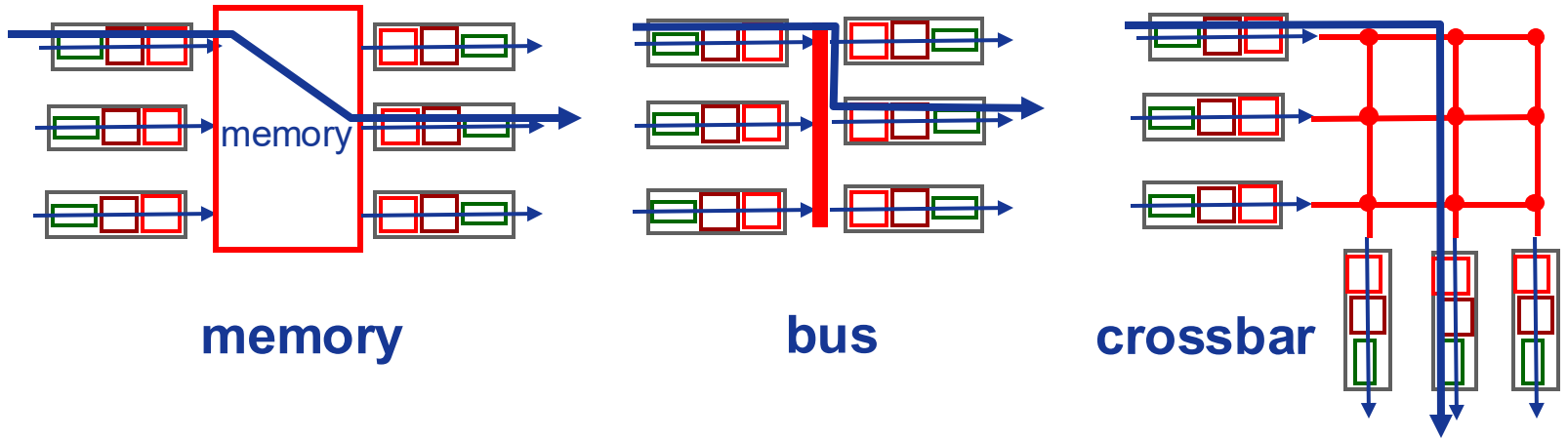
output port contention:
only one red datagram can be transferred.
lower red packet is blocked



one packet time later: green packet experiences HOL blocking

Switching fabrics

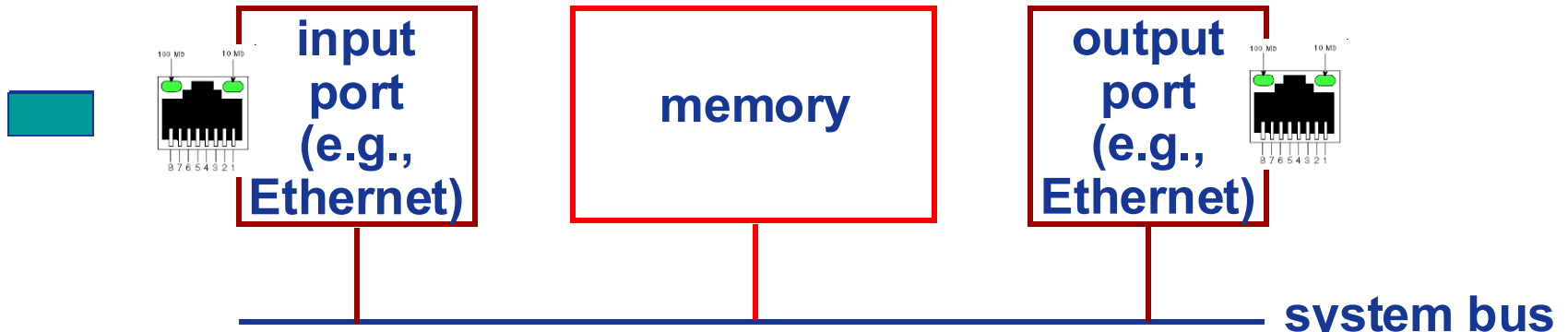
- transfer packet from input buffer to appropriate output buffer
- switching rate: rate at which packets can be transfer from inputs to outputs
  - often measured as multiple of input/output line rate
  - N inputs: switching rate N times line rate desirable
- three types of switching fabrics



Switching via memory

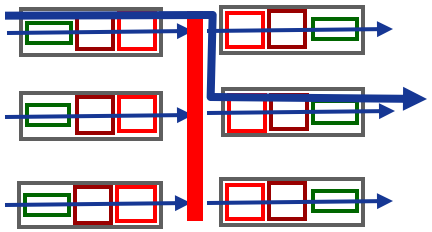
first generation routers:

- 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

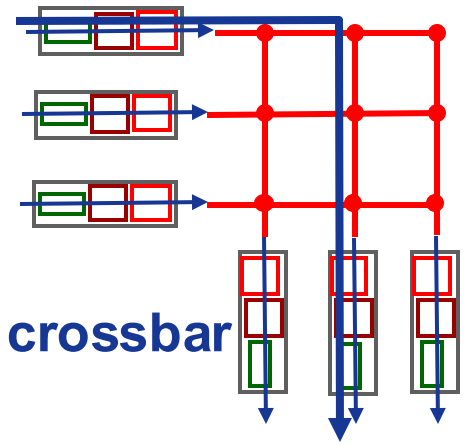
- datagram from input port memory to output port memory via a shared bus
- *bus contention:* switching speed limited by bus bandwidth
- 32 Gbps bus, Cisco 5600: sufficient speed for access and enterprise routers



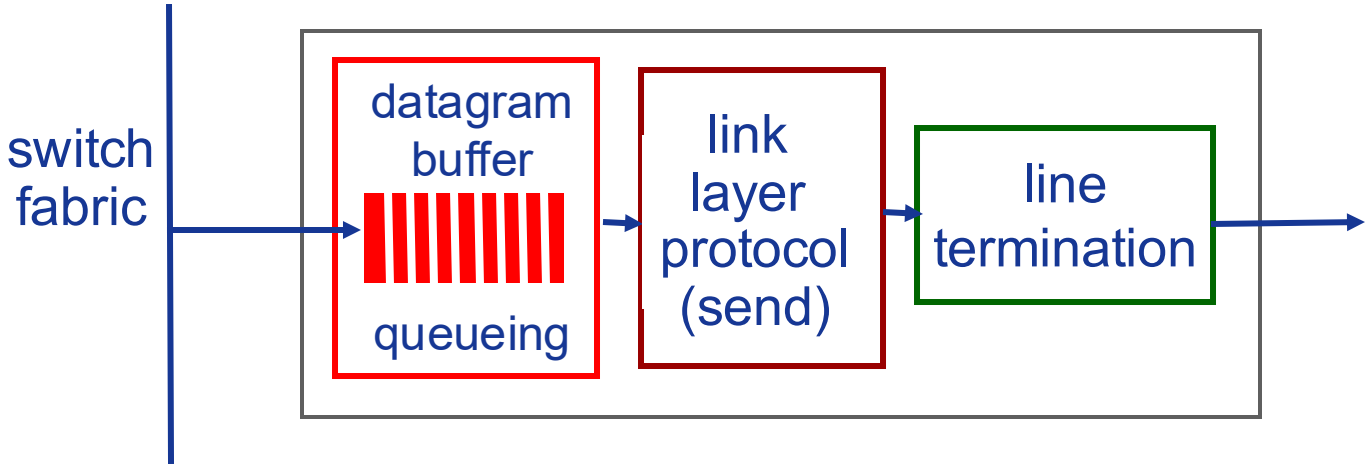
bus

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

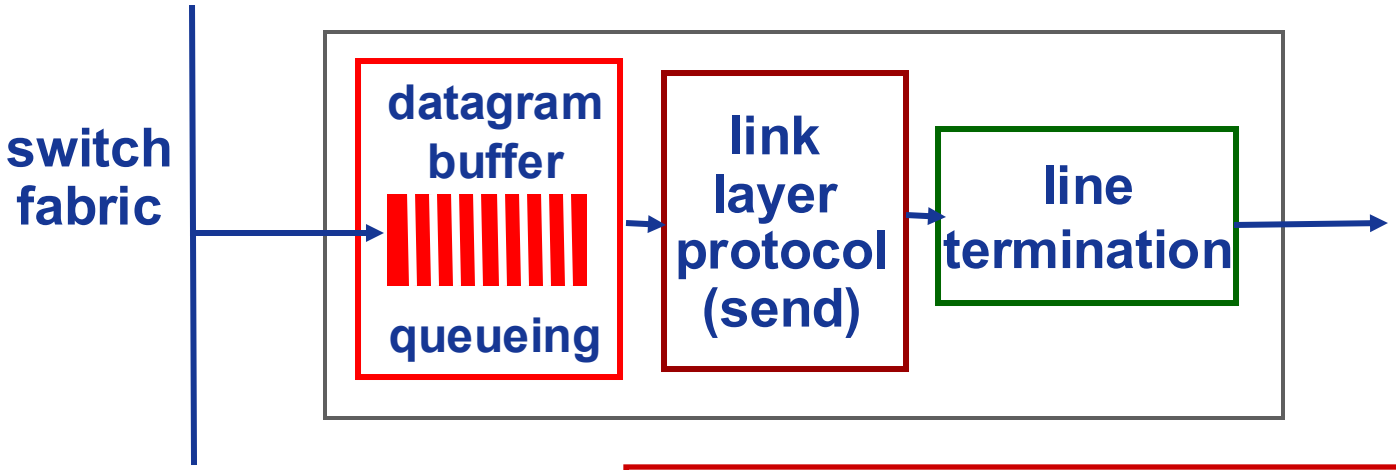


Output ports



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

Output ports

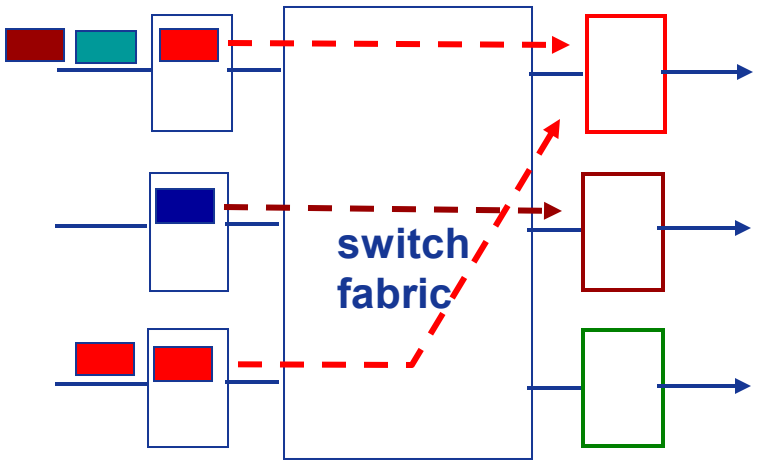


Datagram (packets) can be lost due to congestion, lack of buffers

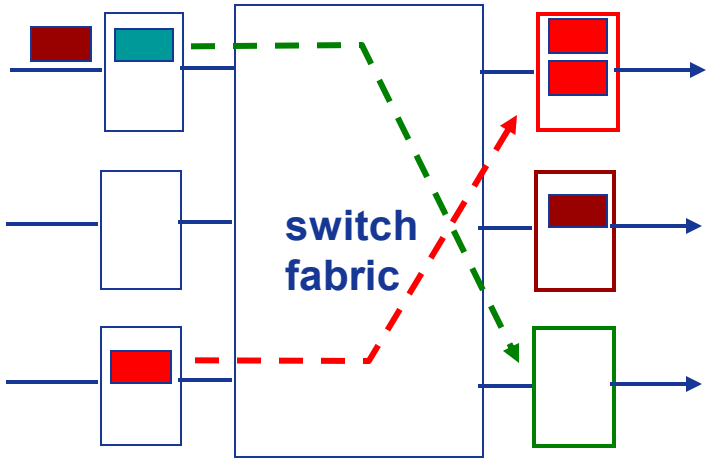
- *buffering* required when datagrams arrive from fabric faster than the transmission rate
- *scheduling discipline* chooses among queued datagrams for transmission

Priority scheduling – who gets best performance, network neutrality

Output port queueing



at *t*, packets more from input to output



one packet time later

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

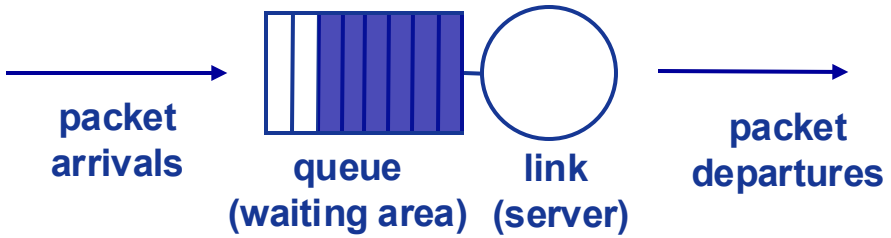
How much buffering?

- RFC 3439 rule of thumb: average buffering equal to “typical” RTT (say 250 msec) times link capacity C
  - e.g., C = 10 Gbps link: 2.5 Gbit buffer
- recent recommendation: with *N* flows, buffering equal to

$$\frac{RTT \cdot C}{\sqrt{N}}$$

Scheduling mechanisms

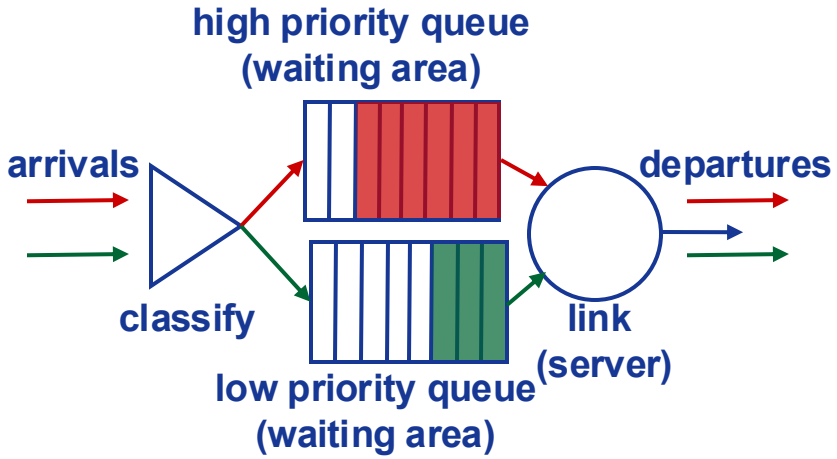
- *scheduling:* choose next packet to send on link
- *FIFO (first in first out) scheduling:* send in order of arrival to queue
  - real-world example?
  - *discard policy:* if packet arrives to full queue: who to discard?
    - *tail drop:* drop arriving packet
    - *priority:* drop/remove on priority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

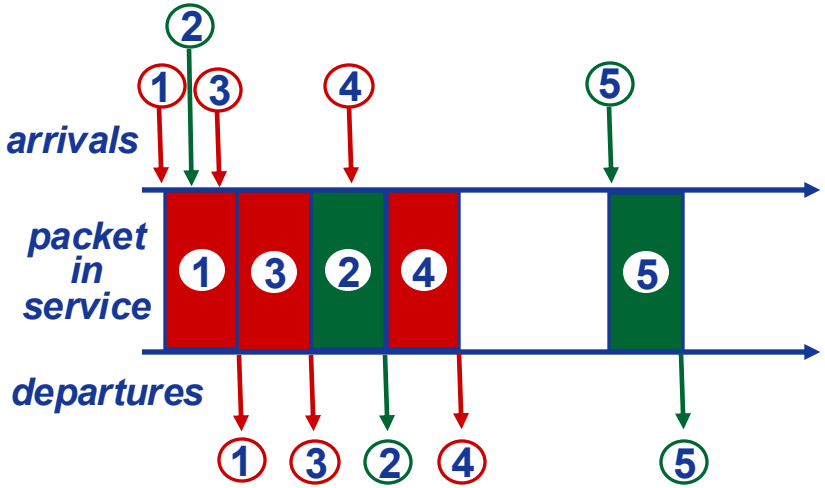


Scheduling policies: priority

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.
  - real world example?

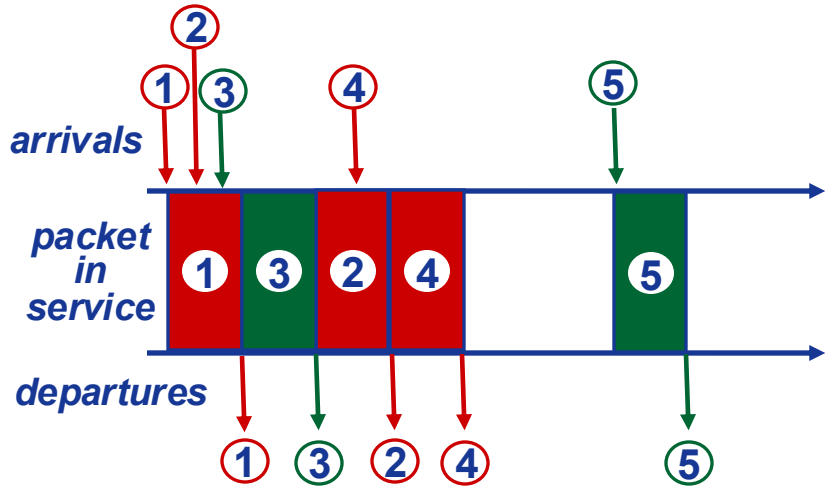




Scheduling policies: still more

Round Robin (RR) scheduling:

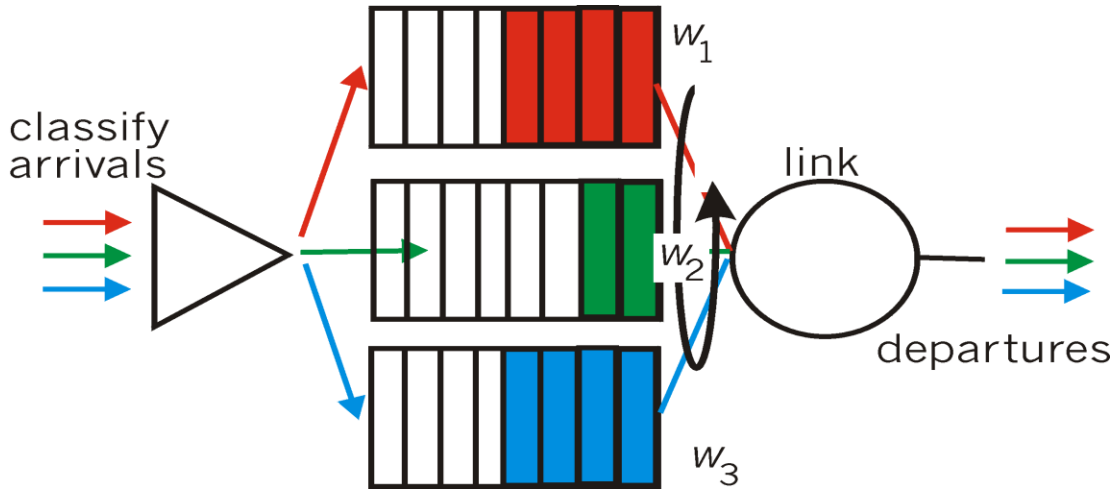
- multiple classes
- cyclically scan class queues, sending one complete packet from each class (if available)
- real world example?



Scheduling policies: still more

Weighted Fair Queuing (WFQ):

- generalized Round Robin
- each class gets weighted amount of service in each cycle
- real-world example?



网络设备对比

	集线器 (hub)	交换机 (switche)	网桥 (bridge)	路由器 (router)
层次	1	2	2	3
流量(冲突域)隔离	no	yes	yes	yes
广播域隔离	no	no	no	yes
即插即用	yes	yes	yes	no
优化路由	no	no	no	yes
直通传输 (Cut through)	yes	yes	yes	no