DIPARTIMENTO DI INFORMATICA
Via Salaria 113 - 00198, Roma

SAPIENZA UNIVERSITÀ DI ROMA

Chapter 4
Network Layer

Reti di Elaboratori

Corso di Laurea in Informatica

Università degli Studi di Roma "La Sapienza"

Canale A-L

Prof.ssa Chiara Petrioli

Parte di queste slide sono state prese dal materiale associato al libro *Computer Networking: A Top Down Approach* , 5th edition.

Thanks also to Antonio Capone, Politecnico di Milano, Giuseppe Bianchi and Francesco LoPresti, Un. di Roma Tor Vergata

Chapter 4: Network Layer

Hierarchical Routing

Our routing study thus far - idealization
- all routers identical
- network "flat"

... *not* true in practice

scale: with 200 million destinations:
- can't store all dest's in routing tables!
- routing table exchange would swamp links!

administrative autonomy
- internet = network of networks
- each network admin may want to control routing in its own network

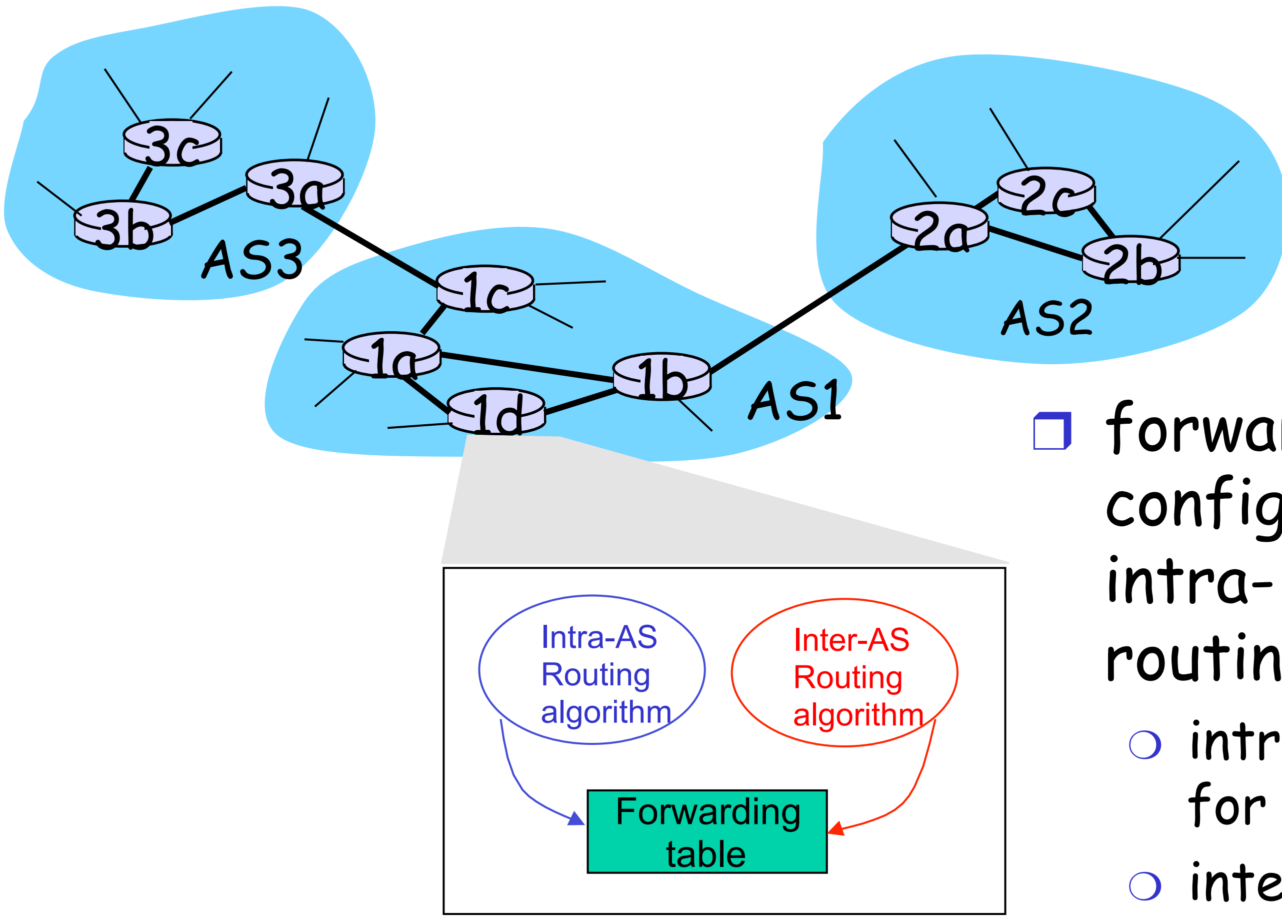
Hierarchical Routing

- aggregate routers into regions, “autonomous systems” (AS)
- routers in same AS run same routing protocol
  - “intra-AS” routing protocol
  - routers in different AS can run different intra-AS routing protocol

Gateway router

- Direct link to router in another AS

Interconnected ASes



- forwarding table configured by both intra- and inter-AS routing algorithm
  - intra-AS sets entries for internal dests
  - inter-AS & intra-As sets entries for external dests

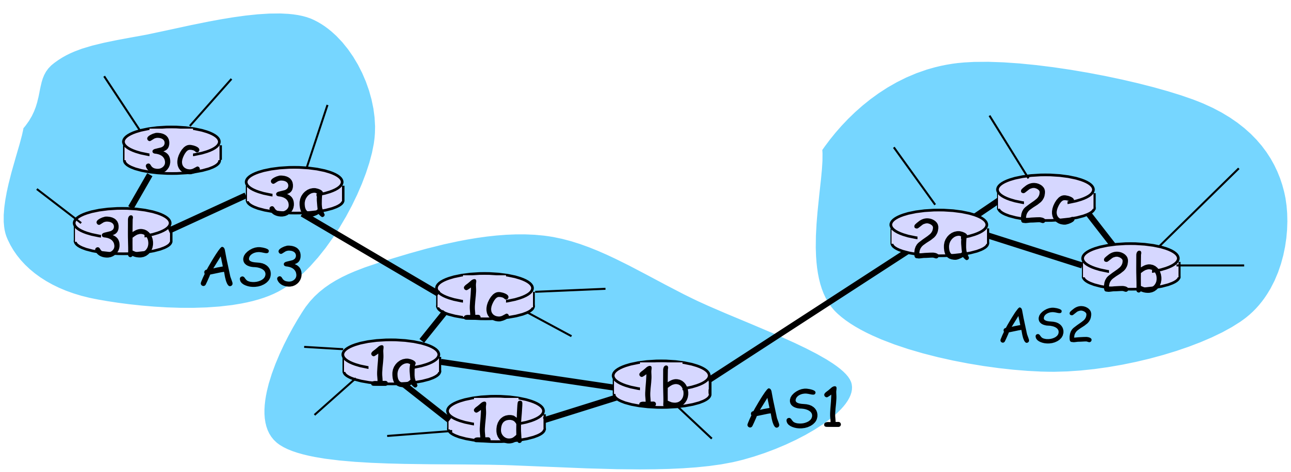
Inter-AS tasks

- suppose router in AS1 receives datagram destined outside of AS1:
  - router should forward packet to gateway router, but which one?

AS1 must:

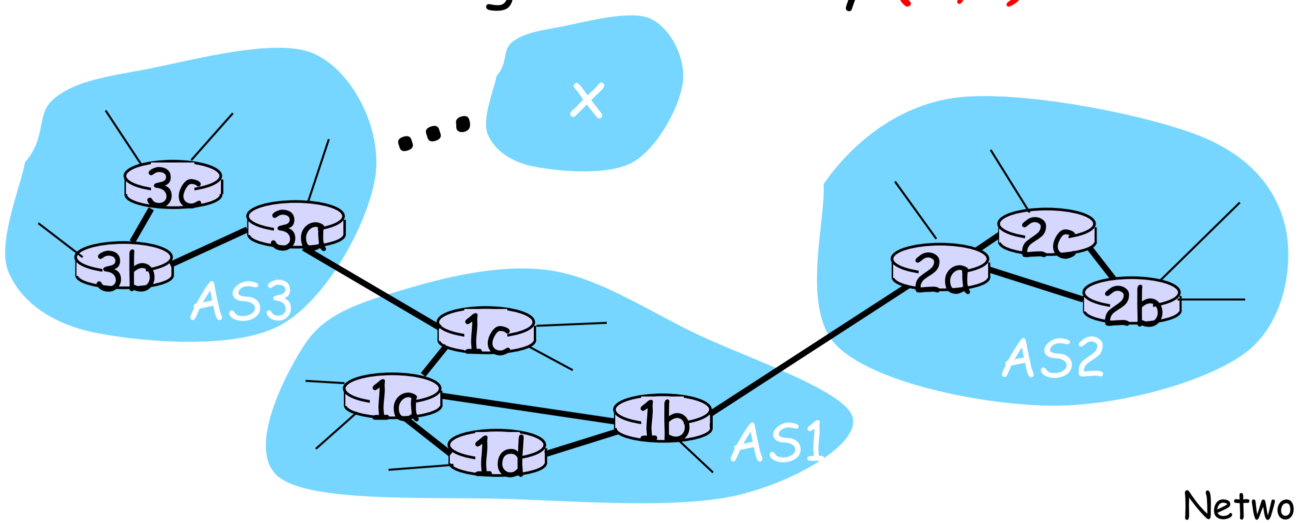
1. learn which dests are reachable through AS2, which through AS3
2. propagate this reachability info to all routers in AS1

Job of inter-AS routing!



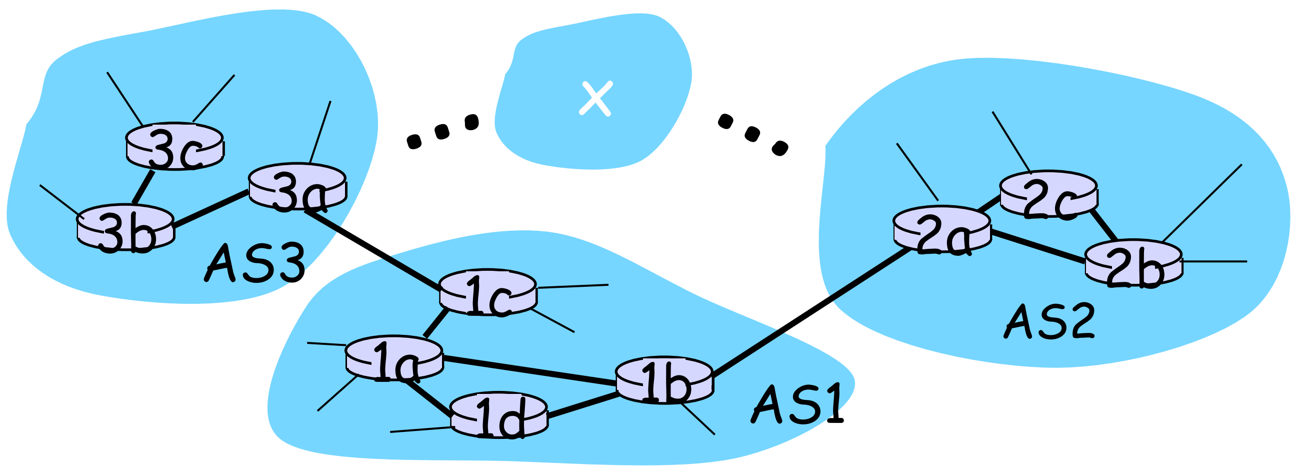
Example: Setting forwarding table in router 1d

- suppose AS1 learns (via inter-AS protocol) that subnet *x* is reachable via AS3 (gateway 1c) but not via AS2.
- inter-AS protocol propagates reachability info to all internal routers.
- router 1d determines from intra-AS routing info that its interface *I* is on the least cost path to 1c.
  - installs forwarding table entry *(x,I)*



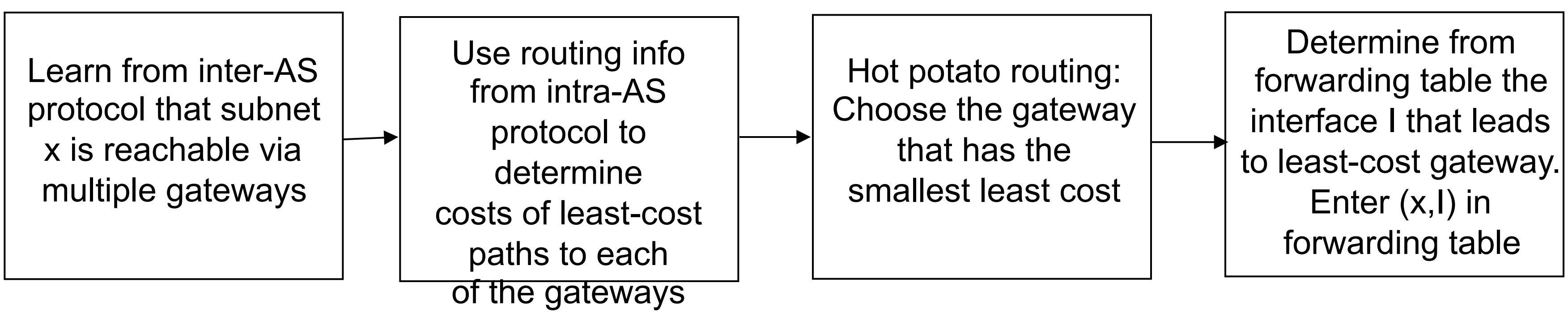
Example: Choosing among multiple ASes

- now suppose AS1 learns from inter-AS protocol that subnet x is reachable from AS3 *and* from AS2.
- to configure forwarding table, router 1d must determine towards which gateway it should forward packets for dest x.
    - this is also job of inter-AS routing protocol!



Example: Choosing among multiple ASes

- now suppose AS1 learns from inter-AS protocol that subnet *x* is reachable from AS3 *and* from AS2.
- to configure forwarding table, router 1d must determine towards which gateway it should forward packets for dest x.
  - this is also job of inter-AS routing protocol!
- hot potato routing: send packet towards closest of two routers.



Chapter 4: Network Layer

- 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

Intra-AS Routing

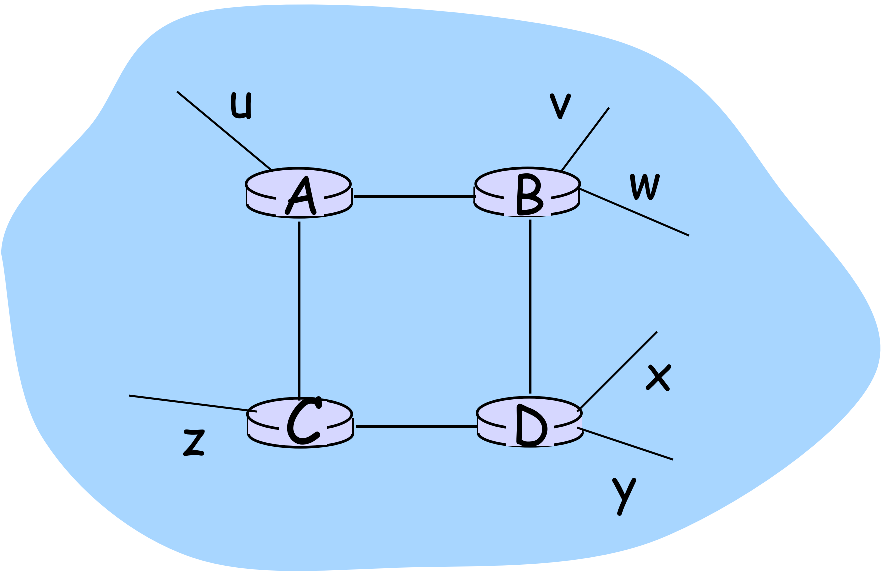
- also known as Interior Gateway Protocols (IGP)
- most common Intra-AS routing protocols:
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - IGRP: Interior Gateway Routing Protocol (Cisco proprietary)

Chapter 4: Network Layer

- 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

RIP (Routing Information Protocol)

- distance vector algorithm
- included in BSD-UNIX Distribution in 1982
- distance metric: # of hops (max = 15 hops)



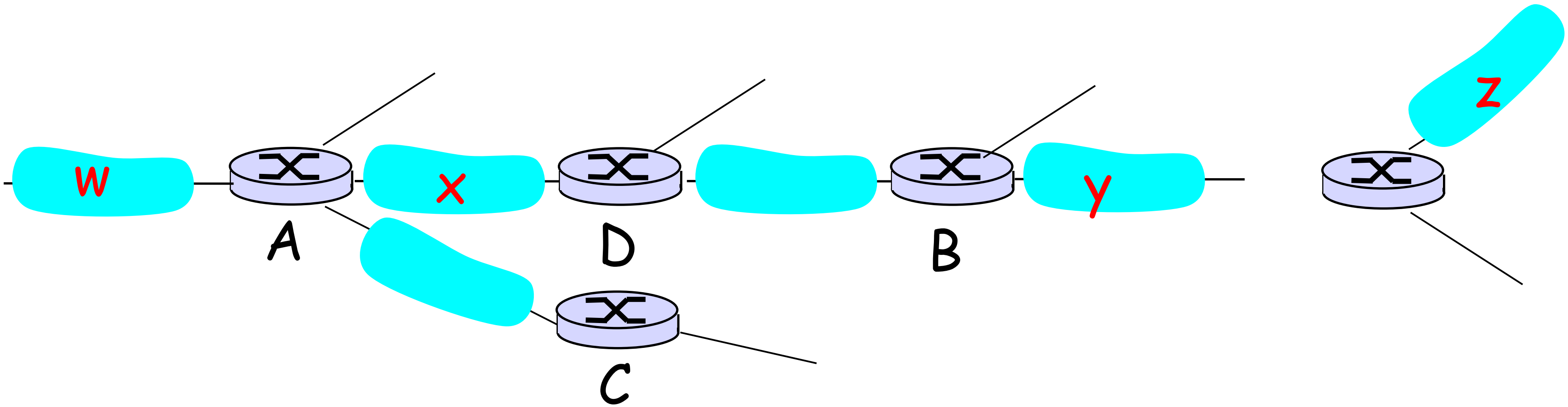
From router A to subnets:

destination	hops
u	1
v	2
w	2
x	3
y	3
z	2

RIP advertisements

- *distance vectors:* exchanged among neighbors every 30 sec via Response Message (also called advertisement)
- each advertisement: list of up to 25 destination subnets within AS

RIP: Example



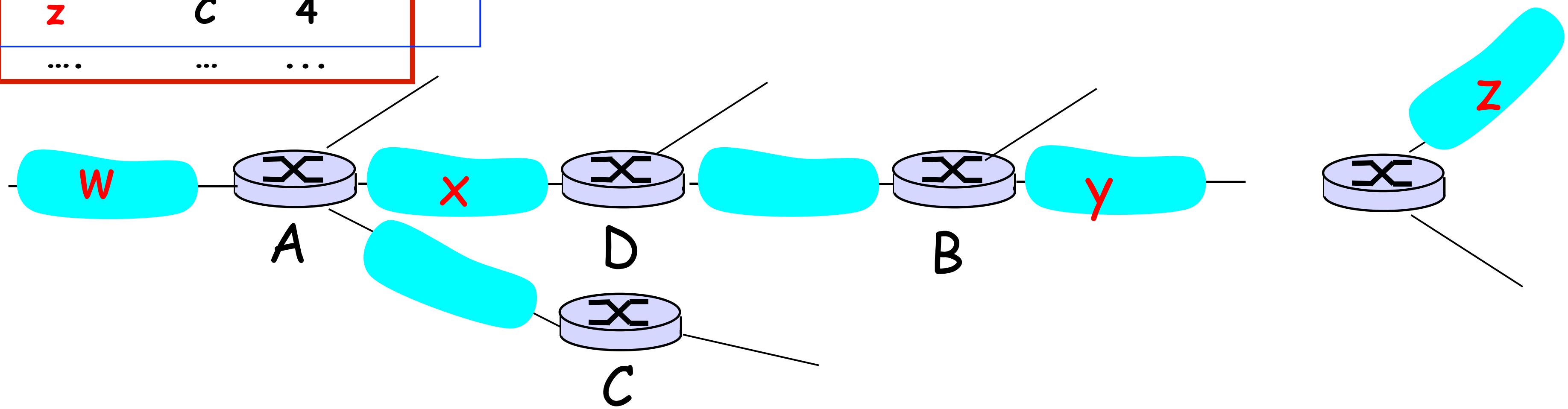
Destination Network	Next Router	Num. of hops to dest.
w	A	2
y	B	2
z	B	7
x	--	1
….	….	….

Routing/Forwarding table in D

RIP: Example

Dest	Next	hops
w	-	1
x	-	1
z	C	4
….	…	. . .

Advertisement from A to D



Destination Network	Next Router	Num. of hops to dest.
w	A	2
y	B	2
z	~~B~~ A	~~7~~ 5
x	--	1
….	….	. . . .

Routing/Forwarding table in D

Differences wrt Bellman Ford

- Since count to infinity not solved upper bound on the network size
- Info on the cost of going through destination X through neighbor Z is maintained ONLY IF the path through Z is the current "best" (min cost) path
  - Different way of updating costs
    - Suppose current route to dest has cost D and goes through G
    - if an update arrives from X!=G then updates route ONLY IF cost is <D
    - if an update arrives from G always update route cost
    - PROBLEM: what if the router we go through crashes?
      - Route cost aging MUST be adopted
- Cost is maintained for each subnetwork (rather than node)
- Periodic exchange of messages

The RIP algorithm (from RFC)

- Keep a table with an entry for every possible destination in the system. The entry contains the distance D to the destination, and the first router G on the route to that network. Conceptually, there should be an entry for the entity itself, with metric 0, but this is not actually included.

- Periodically, send a routing update to every neighbor. The update is a set of messages that contain all of the information from the routing table. It contains an entry for each destination, with the distance shown to that destination.

- When a routing update arrives from a neighbor G', add the cost associated with the network that is shared with G'. (This should be the network over which the update arrived.) Call the resulting distance D'. Compare the resulting distances with the current routing table entries. If the new distance D' for N is smaller than the existing value D, adopt the new route. That is, change the table entry for N to have metric D' and router G'. If G' is the router from which the existing route came, i.e., G' = G, then use the new metric even if it is larger than the old one.

RIP: Link Failure and Recovery

If no advertisement heard after 180 sec --> neighbor/ link declared dead

- routes via neighbor invalidated
- new advertisements sent to neighbors
- neighbors in turn send out new advertisements (if tables changed)
- link failure info quickly (?) propagates to entire net
- *poison reverse* used to prevent ping-pong loops (infinite distance = 16 hops)

Differences with Bellman Ford

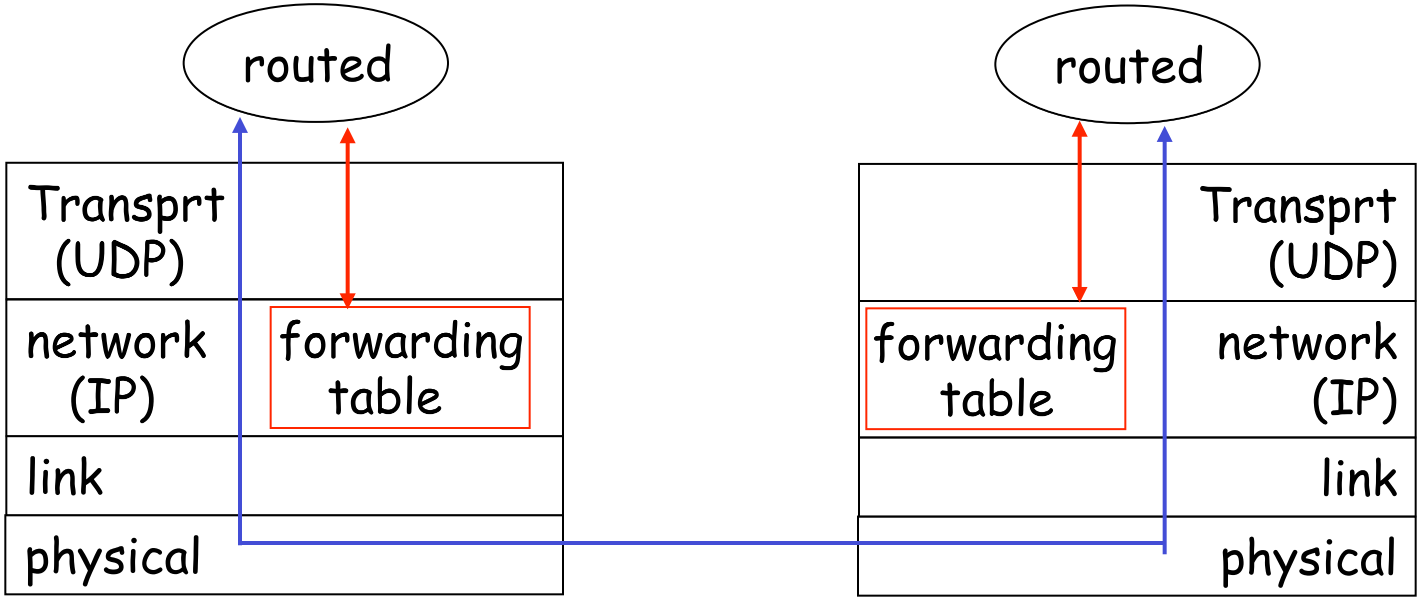
- Split Horizon
  - with Poison Reverse
  - Simple split horizon (omits cost of reaching destination when advertising through the router it goes through)
- Clear implementation with point to point links. But consider the possibility that A and C are connected by a broadcast network such as an Ethernet, and there are other routers on that network. Is it a problem?
  - If A has a route through C, it should indicate that D is unreachable when talking to any other router on that network. The other routers on the network can get to C themselves. They would never need to get to C via A.
  - If A's best route is really through C, no other router on that network needs to know that A can reach D. This is fortunate, because it means that the same update message that is used for C can be used for all other routers on the same network. Thus, update messages can be sent by broadcast.

An additional way to speed up convergence

- Triggered updates
  - Whenever a router changes the metric for a route it is required to send update messages almost immediately
    - must be implemented for deleted routes

RIP Table processing

- RIP routing tables managed by **application-level** process called route-d (daemon)
  - port number 520
- advertisements sent in UDP packets, periodically repeated



Packet format

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  command (1)  |  version (1)  |       must be zero (2)        |
+---------------+---------------+-------------------------------+
|                                                               |
~                         RIP Entry (20)                        ~
|                                                               |
+---------------+---------------+---------------+---------------+
```

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| address family identifier (2) |      must be zero (2)         |
+-------------------------------+-------------------------------+
|                        IPv4 address (4)                       |
+---------------------------------------------------------------+
|                        must be zero (4)                       |
+---------------------------------------------------------------+
|                        must be zero (4)                       |
+---------------------------------------------------------------+
|                           metric (4)                          |
+---------------------------------------------------------------+
```

Chapter 4: Network Layer

- 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

OSPF (Open Shortest Path First)

- "open": publicly available
- uses Link State algorithm
  - LS packet dissemination
  - topology map at each node
  - route computation using Dijkstra's algorithm

- OSPF advertisement carries one entry per neighbor router
  - Each node disseminates its local view of the topology
  - i.e., the router usable interfaces and reachable neighbors

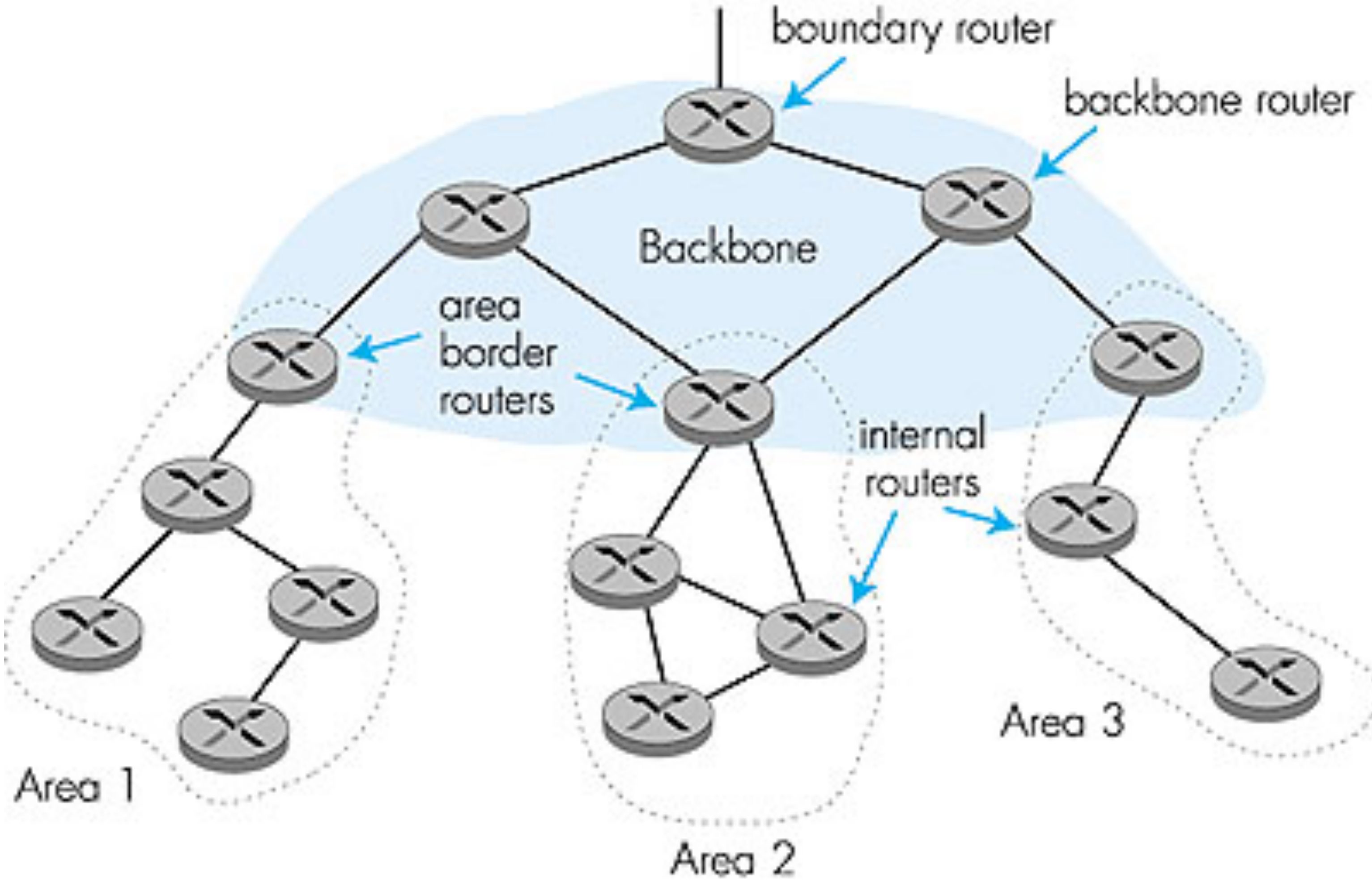
- advertisements disseminated to entire AS (via flooding)
  - carried in OSPF messages directly over IP (using protocol nuner 89)

OSPF "advanced" features (not in RIP)

- security: all OSPF messages authenticated (to prevent malicious intrusion)
- multiple same-cost paths allowed (only one path in RIP)
- For each link, multiple cost metrics for different TOS (e.g., satellite link cost set "low" for best effort; high for real time)
  - Periodic updates (30 min) or event driven (link cost change)
- integrated uni- and multicast support:
  - Multicast OSPF (MOSPF) uses same topology data base as OSPF
- hierarchical OSPF in large domains.

Externally derived routing data is advertised throughout the Autonomous System unaltered

Hierarchical OSPF



Splitting the AD into areas

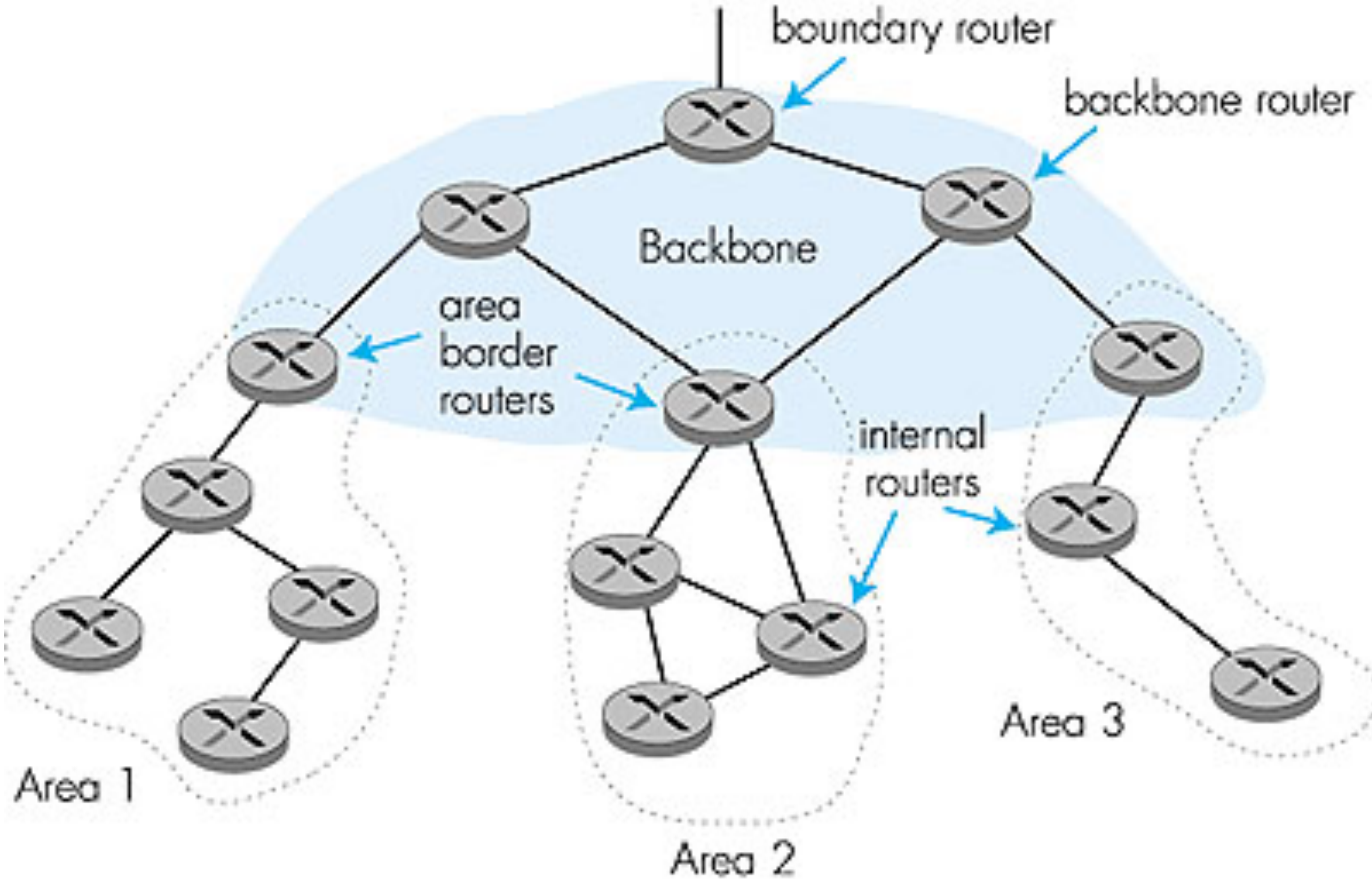
- OSPF allows collections of contigous networks and hosts to be grouped together
  - Such a group together with the routers with interfaces to any of the included networks is called an area
  - Each area runs a separate copy of the basic link-state routing algorirthm
    - has its own link state database
  - The topology of an area is invisble from the outside
  - Routers internal to a given area know nothing of the detailed topology external to the area

Hierarchical OSPF

Isolamento info di area funzionali a ridurre il traffico di routing

- two-level hierarchy: local area, backbone.
  - Link-state advertisements only in area
  - each node has detailed area topology; only known direction (shortest path) to nets in other areas.
- *area border routers:* "summarize" distances to nets in own area, advertise to other Area Border routers.

Hierarchical OSPF



Hierarchical OSPF

- The OSPF backbone is the special OSPF Area 0
- The OSPF backbone always contains backbone routers
- *Backbone routers:* run OSPF routing limited to backbone.
- The backbone is responsible for distributing routing information between non backbone areas
  - Every area border router hears the area summaries from all other area border routers
  - adding backbone distance+distance in summaries each router knows distance to different destinations
  - These distances are then advertised internally to their areas
- The backbone must be contiguous but not physically contiguous
  - Backbone connectivity can be established/maintained through the configuration of virtual links (part of the backbone with actual way to route between end piint of the virtual link based on intra_AS routing)
- *Boundary routers:* connect to other AS's.
- AS external LSAs are advertised in the AS WITH THE EXCEPTION OF stub areas
  - Stub areas use a default routing

Types of networks

- Transit networks are capable of carrying data traffic which is neither locally originated nor locally destined
- A stub network only carries traffic it either generates or addressed to it

LSA (Link State Advertisement)

- Periodic advertisement
- Link state is also advertised when a router state changes
  - Hello packets used to discover and maintain neighbor relationships
- Disseminated via flooding
- Flooding algorithm is reliable ensuring that all routers in the area have the same link state database

Chapter 4: Network Layer

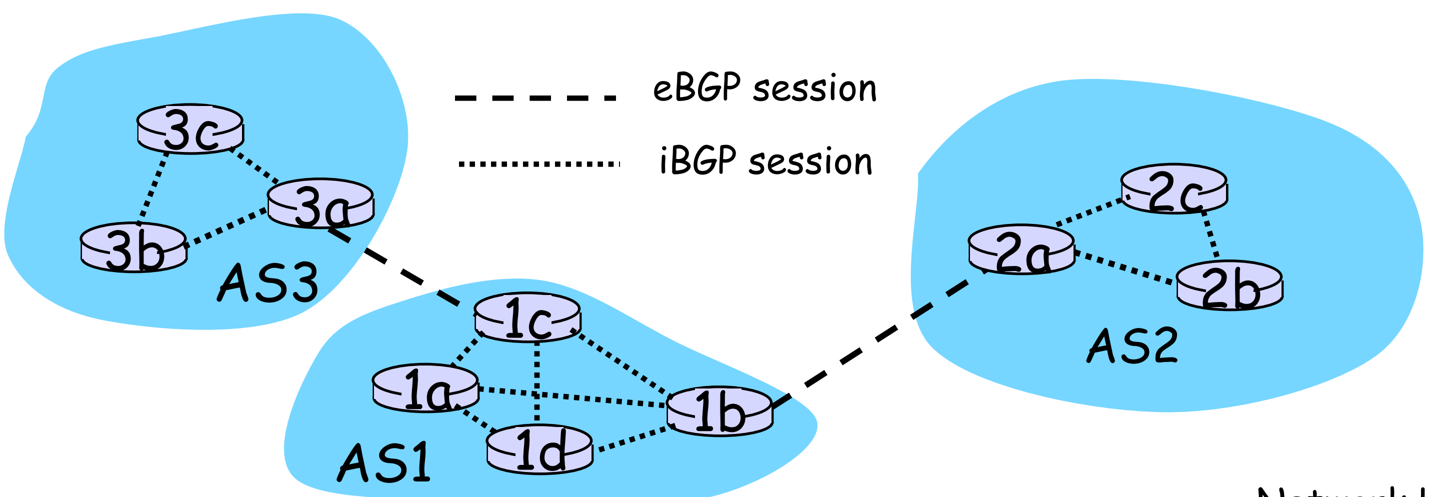
- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Internet inter-AS routing: BGP

- BGP (Border Gateway Protocol): *the* de facto standard
- BGP provides each AS a means to:
  1. Obtain subnet reachability information from neighboring ASs.
  2. Propagate reachability information to all AS-internal routers.
  3. Determine "good" routes to subnets based on reachability information and policy.
- allows subnet to advertise its existence to rest of Internet: *"I am here"*

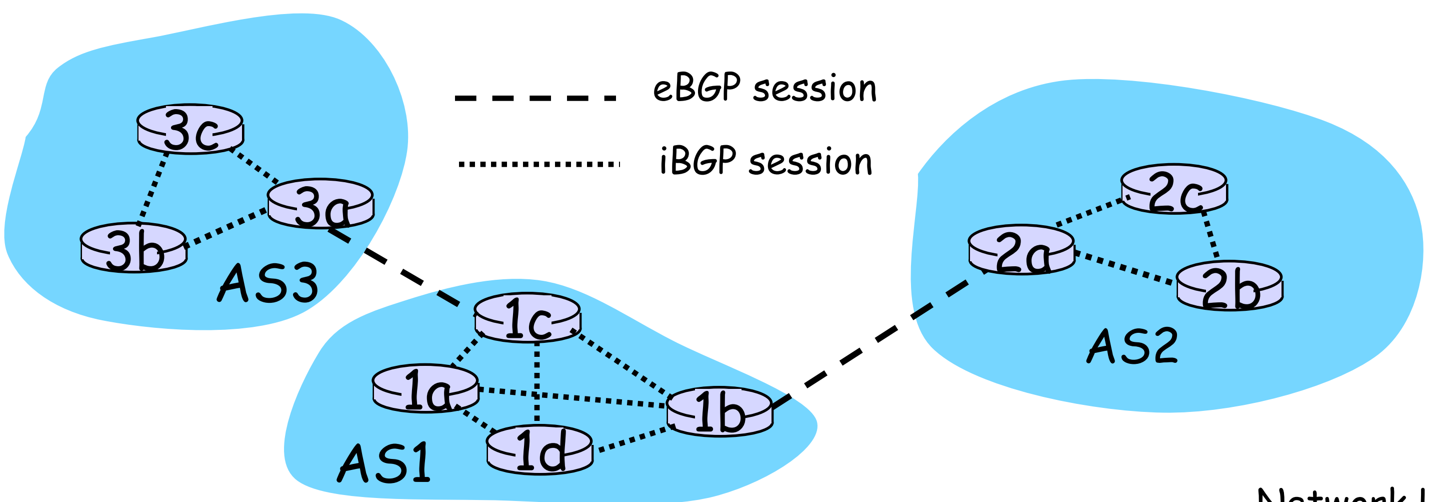
BGP basics

- pairs of routers (BGP peers) exchange routing info over semi-permanent TCP connections: BGP sessions
  - BGP sessions need not correspond to physical links.
- when AS2 advertises a prefix to AS1:
  - AS2 *promises* it will forward datagrams towards that prefix.
  - AS2 can aggregate prefixes in its advertisement



Distributing reachability info

- using eBGP session between 3a and 1c, AS3 sends prefix reachability info to AS1.
  - 1c can then use iBGP do distribute new prefix info to all routers in AS1
  - 1b can then re-advertise new reachability info to AS2 over 1b-to-2a eBGP session
- when router learns of new prefix, it creates entry for prefix in its forwarding table.



Path attributes & BGP routes

- advertised prefix includes BGP attributes.
  - prefix + attributes = "route"
- two important attributes:
  - AS-PATH: contains ASs through which prefix advertisement has passed: e.g, AS 67, AS 17
  - NEXT-HOP: indicates specific internal-AS router to next-hop AS. (may be multiple links from current AS to next-hop-AS)
- when gateway router receives route advertisement, uses import policy to accept/ decline.

BGP route selection

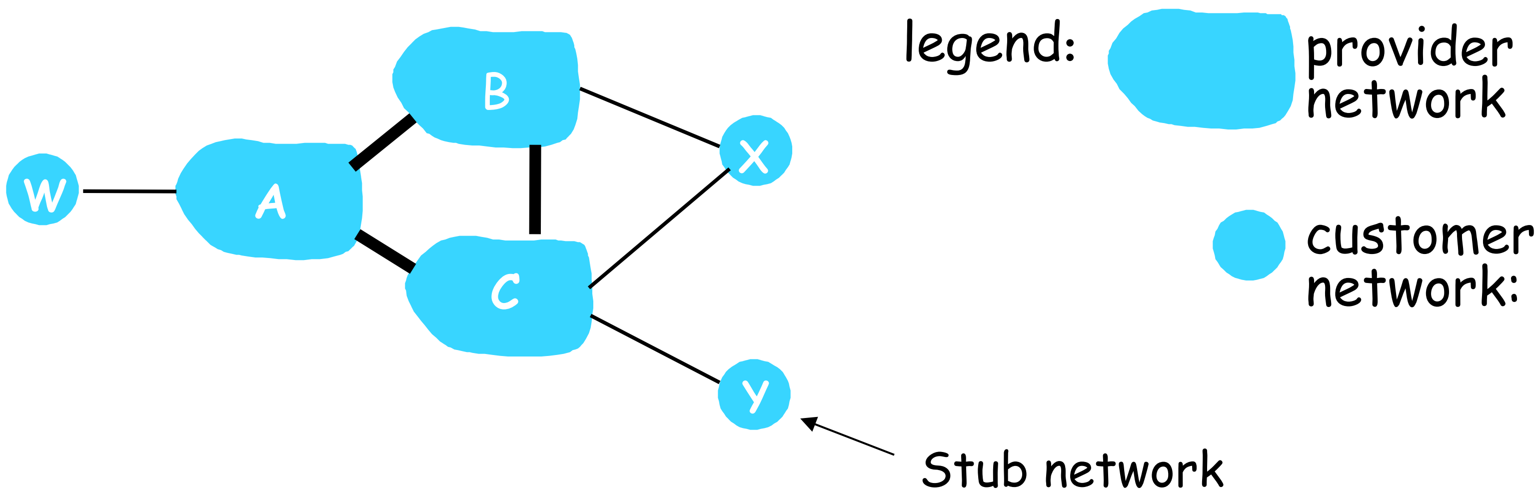
- router may learn about more than 1 route to some prefix. Router must select route.
- elimination rules (in priority order):
  1. local preference value attribute: policy decision
  2. (in case of same preference) shortest AS-PATH
  3. (in case of same preferehce and AS-PATH length) closest NEXT-HOP router: hot potato routing
  4. additional criteria to break the tie

BGP messages

- BGP messages exchanged using TCP.
- BGP messages:
  - OPEN: opens TCP connection to peer and authenticates sender
  - UPDATE: advertises new path (or withdraws old)
  - KEEPALIVE keeps connection alive in absence of UPDATES; also ACKs OPEN request
  - NOTIFICATION: reports errors in previous msg; also used to close connection

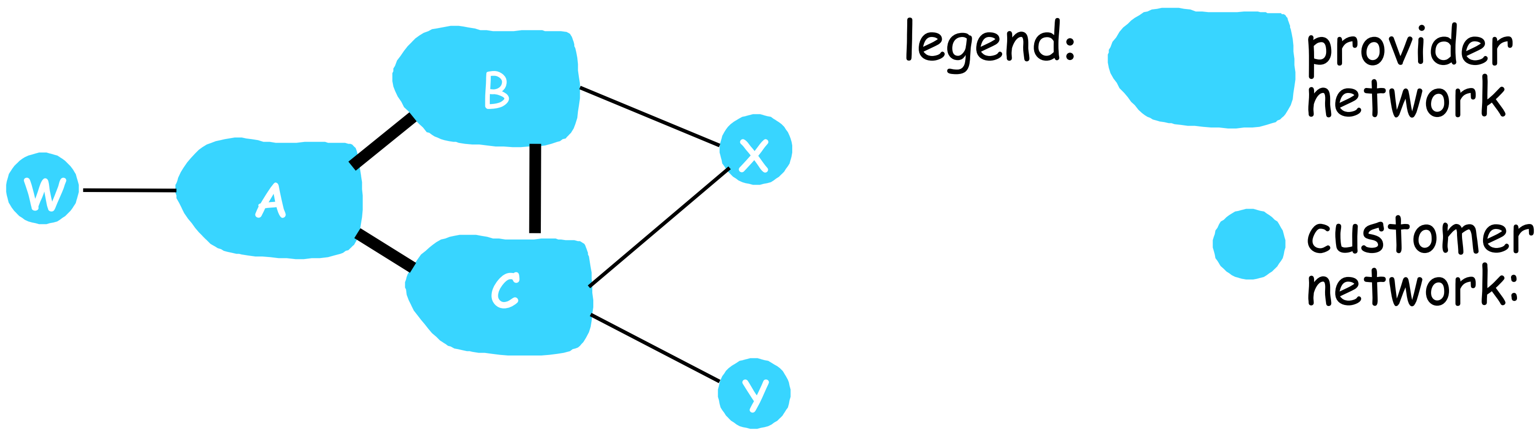
Sent periodically or in case of selected routes changes

BGP routing policy



- A,B,C are provider networks
- X,W,Y are customer (of provider networks)
- X is dual-homed: attached to two networks
  - X does not want to route from B via X to C
  - .. so X will not advertise to B a route to C

BGP routing policy (2)



- A advertises path AW to B
- B advertises path BAW to X (who is its client)
- Should B advertise path BAW to C?
  - No way! B gets no "revenue" for routing CBAW since neither W nor C are B's customers
  - B wants to force C to route to w via A
  - B wants to route *only* to/from its customers!
  - Peering agreements amongs pairs of ISP possible to solve this problem

Decision process

- The decision process selects routes for subsequent advertisement applying the policies in the local Policy Information Base (PIB) to the routes stored in its Adj-RIBs_In (Incoming Routing Information Base)
- A function takes as argument the attributes of a give route and returns a) either a non negative integer identifying the degree of preference for the route or b) a value indicating the route is inelegible

Why different Intra- and Inter-AS routing ?

Policy:

- Inter-AS: admin wants control over how its traffic routed, who routes through its net.
- Intra-AS: single admin, so no policy decisions needed

Scale:

- hierarchical routing saves table size, reduced update traffic

Performance:

- Intra-AS: can focus on performance
- Inter-AS: policy may dominate over performance

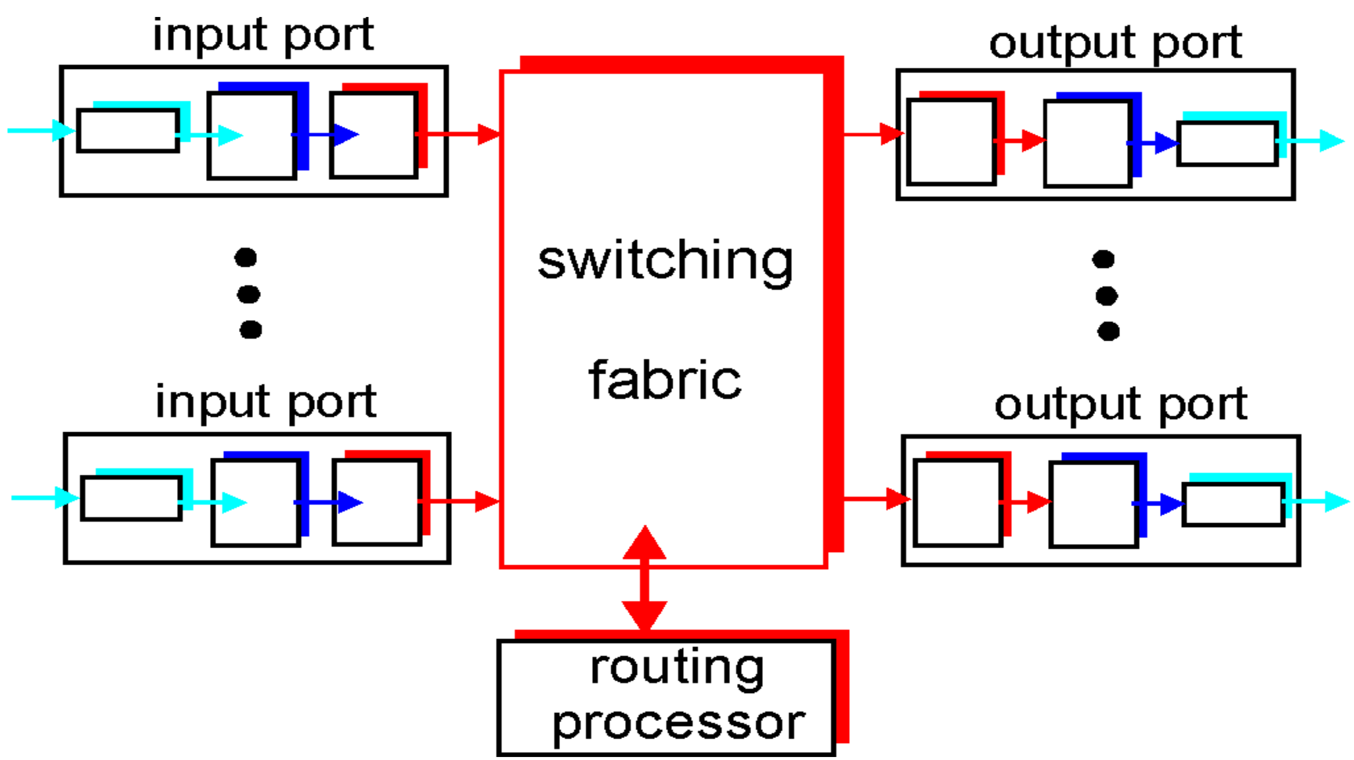
Chapter 4: Network Layer

- 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

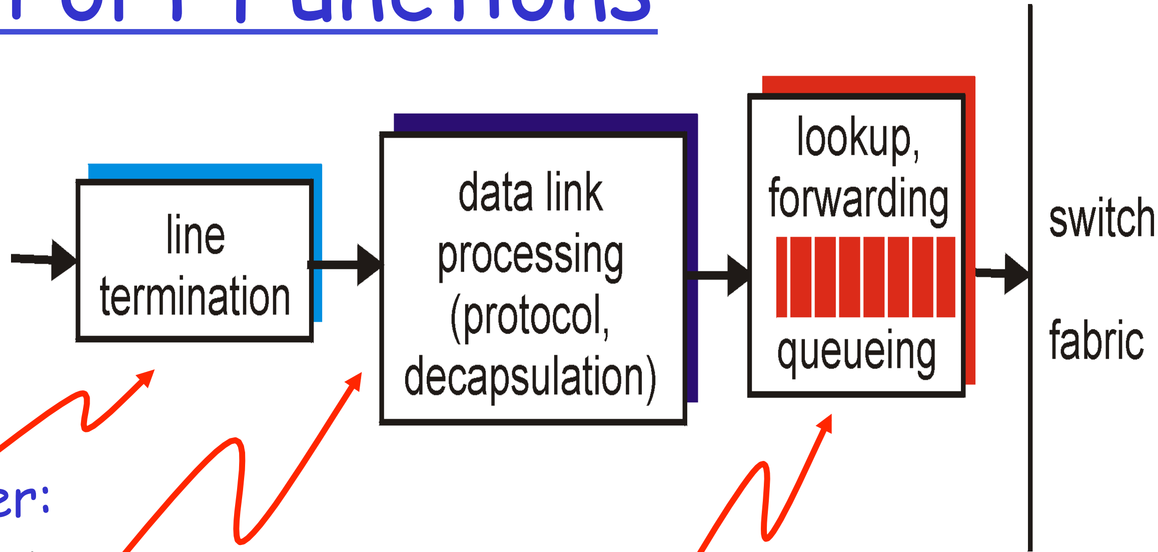
Router Architecture Overview

Two key router functions:

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



Input Port Functions



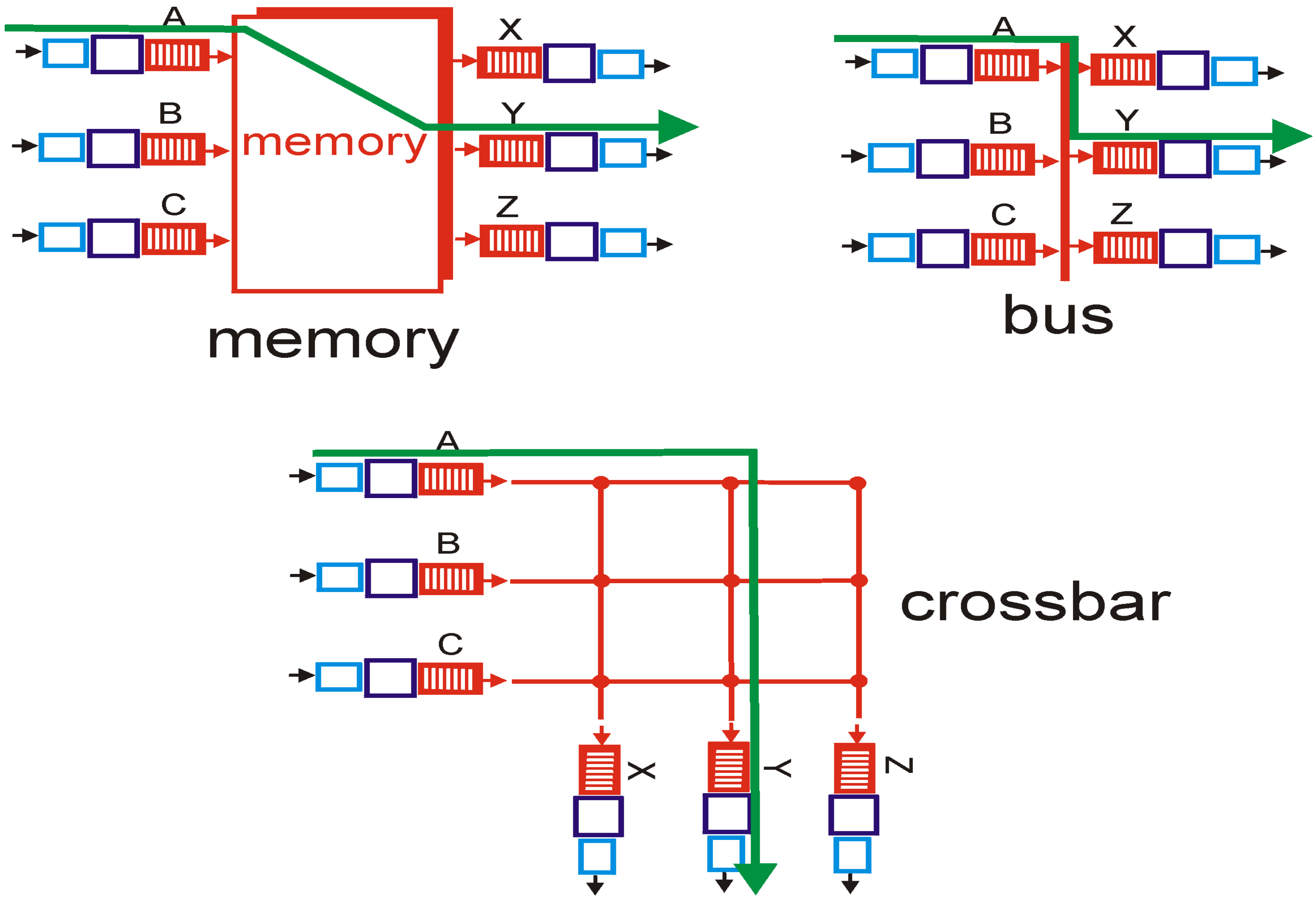
Physical layer:
bit-level reception

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

Decentralized switching:

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

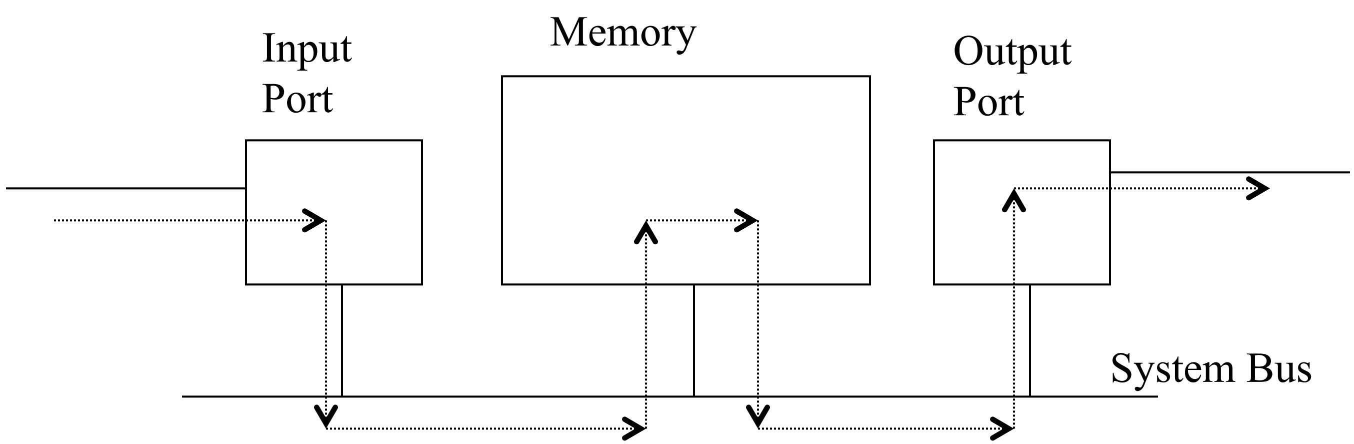
Three types of switching fabrics



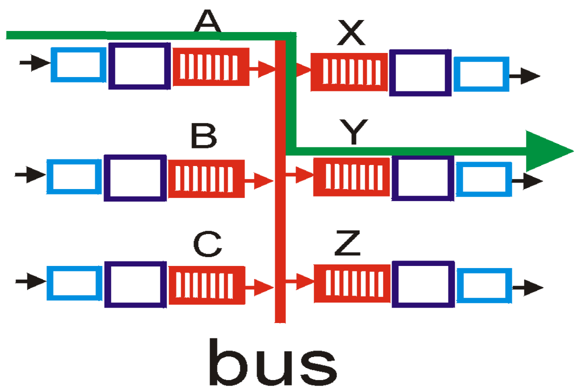
Switching Via Memory

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

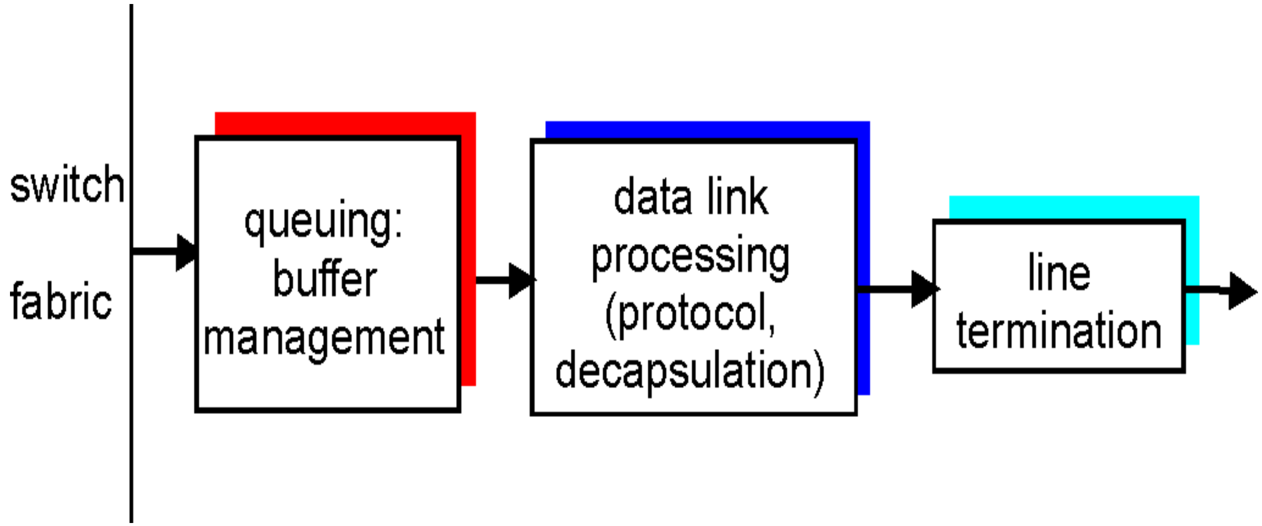


- datagram from input port memory to output port memory via a shared bus
- bus contention: switching speed limited by bus bandwidth
- 32 Gbps bus, Cisco 5600: sufficient speed for access and enterprise routers

Switching Via An Interconnection Network

- overcome 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 60 Gbps through the interconnection network

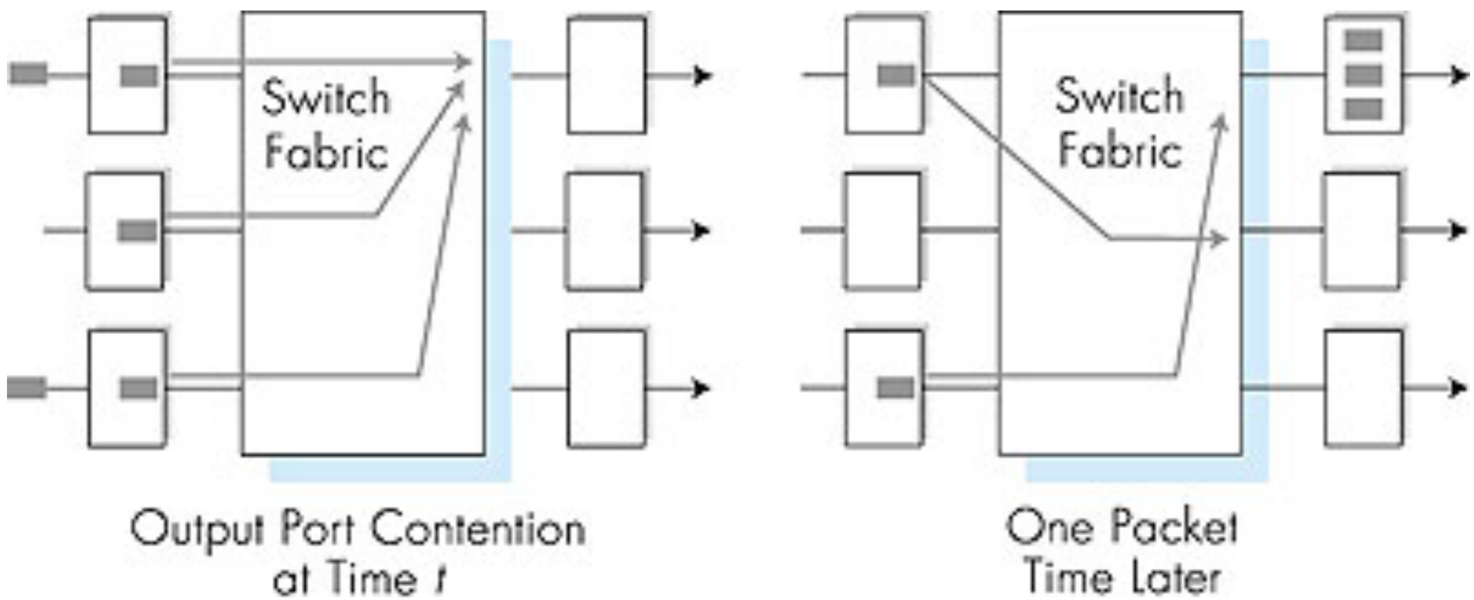
Output Ports



- *Buffering* required when datagrams arrive from fabric faster than the transmission rate
  - What if the queue builds up?
    - Drop Tail
    - Random Early Discard
- *Scheduling discipline* chooses among queued datagrams for transmission
  - First Come First Served
  - Weighted Fair Queueing

$$\frac{Rw_i}{(w_1 + w_2 + ... + w_N)}$$

Output port queueing



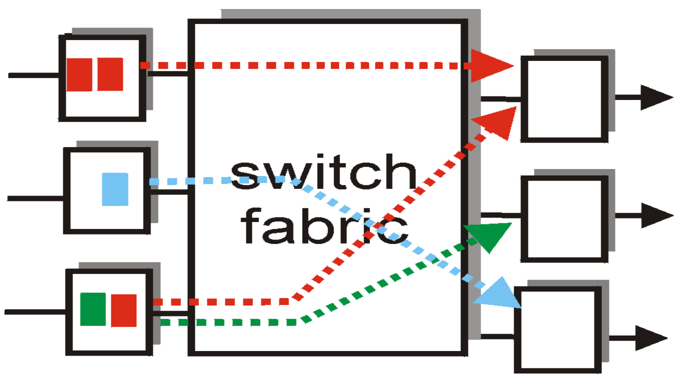
- buffering when arrival rate via switch exceeds output line speed
- *queueing (delay) and loss due to output port buffer overflow!*

How much buffering?

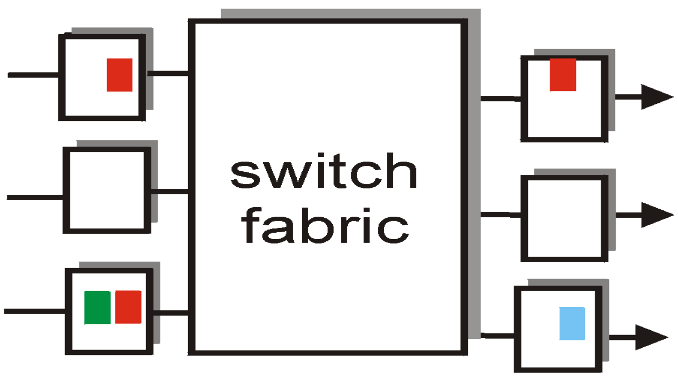
- RFC 3439 rule of thumb: average buffering equal to "typical" RTT (say 250 msec) times link capacity *C*
  - e.g., *C* = 10 Gps link: 2.5 Gbit buffer
- Recent recommendation: with *N* flows, buffering equal to $\frac{RTT \cdot C}{\sqrt{N}}$

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
- *queueing delay and loss due to input buffer overflow!*



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



green packet experiences HOL blocking