Lecture 07: Routing

Acknowledgement: Materials presented in this lecture are predominantly based on slides from:

- *Computer Networking: A Top Down Approach,* J. Kurose, K. Ross, 7th ed., 2017, Addison-Wesley, Chapter 5
- *Business Data Communications and Networking,* J. Fitzgerald, A. Dennis, 10/11th ed., 2013, John Wiley & Sons, Chapter 5

Lecture 07: Routing Outline

- Routing fundamentals
- Classification of routing algorithms
  - A Link-State routing algorithm
  - Distance Vector routing algorithm
- Hierarchical routing on the Internet
- Autonomous Systems
- Intra-AS and Inter-AS routing
- Internet routing protocols:
  - RIP, OSPF, BGP

Routing Fundamentals

- Process of identifying what path a packet has to take through a network from sender to receiver
- **Routing Tables**

  At each routing computer,

  – for each final destination, it stores

    - The (Link layer address of) the next router to send packets to to reach a given destination

  – Kept by computers making routing decisions (routers)

- (Remember that in a subnet every computer is responsible for routing to a computer in the same subnet)

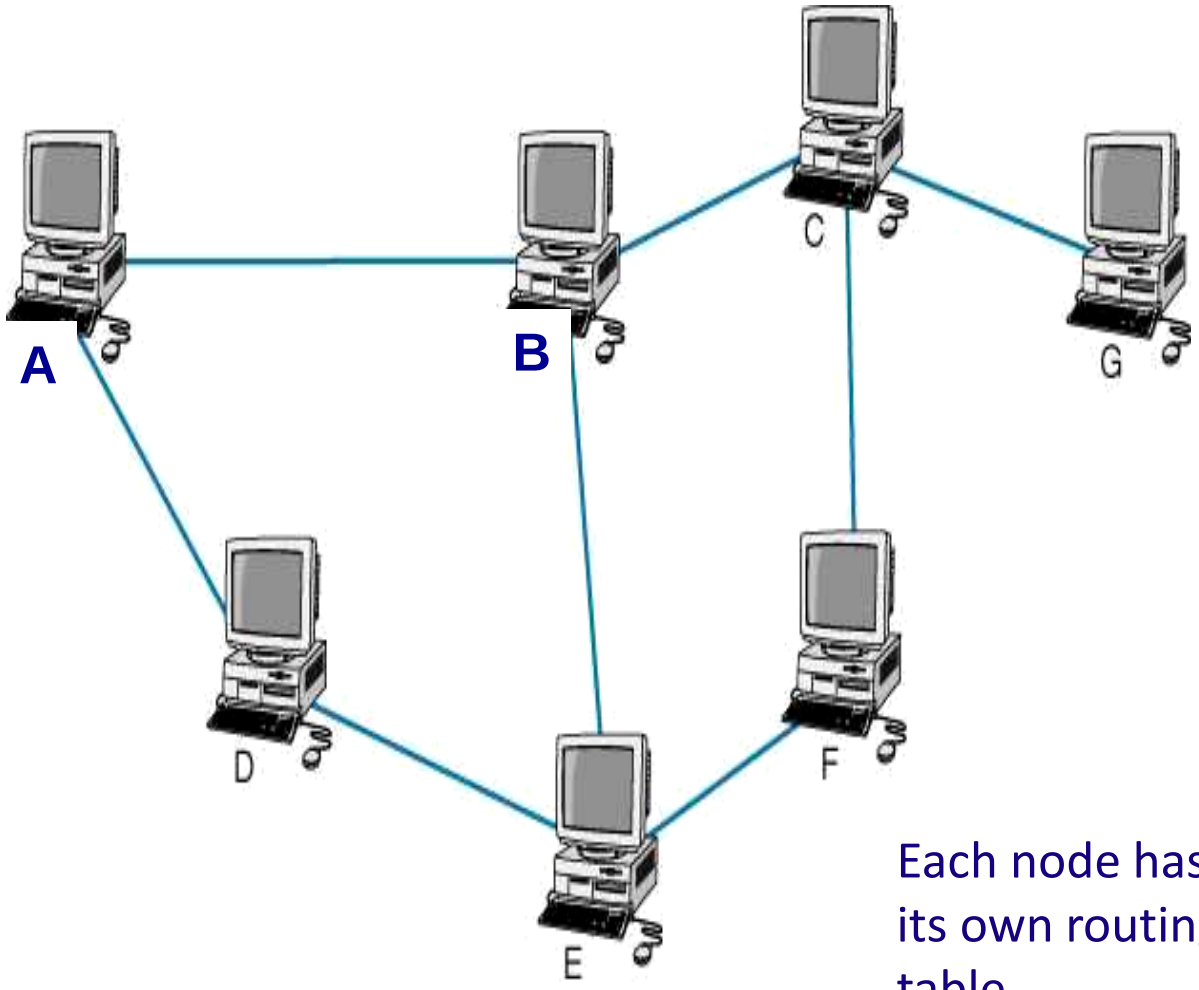
Dest.	Next
B	B
C	B
D	D
E	D
F	D
G	B

Try to draw a network given the routing table

Routing Example

Possible paths from A to G:

- ABCG
- ABEFCG
- ADEFCG
- ADEBCG

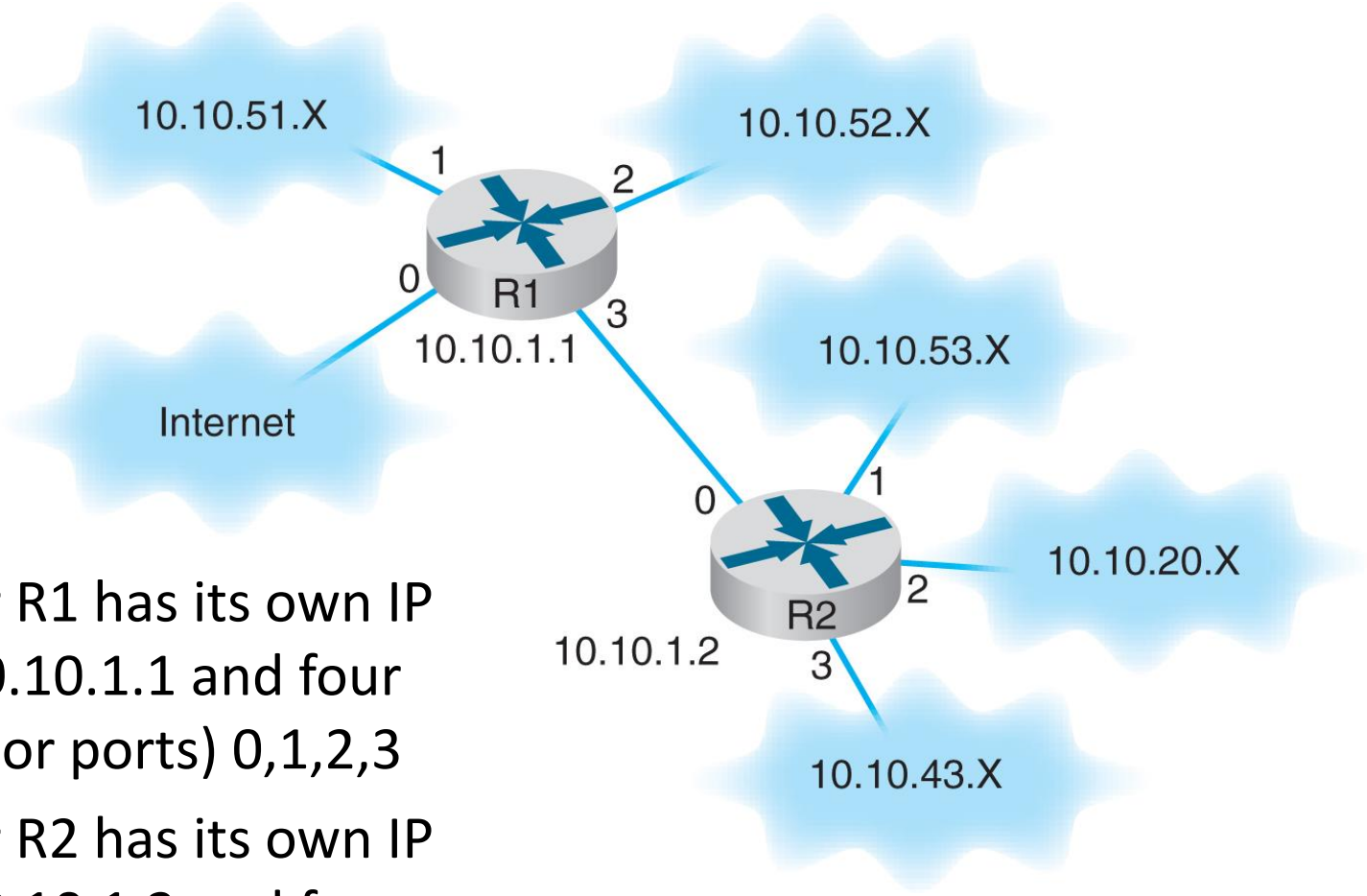


Routing Table for A

Dest.	Next
B	B
C	B
D	D
E	D
F	D
G	B

Each node has its own routing table

Two Routers Example



- The router R1 has its own IP address 10.10.1.1 and four interfaces (or ports) 0,1,2,3
- The router R2 has its own IP address 10.10.1.2 and four interfaces (or ports) 0,1,2,3

Two Routers Example. Routing Tables

Router R1's Routing Table

Network Address	Interface
10.10.51.0 to 10.10.51.255	1
10.10.52.0 to 10.10.52.255	2
10.10.53.0 to 10.10.53.255	3
10.10.20.0 to 10.10.20.255	3
10.10.43.0 to 10.10.43.255	3
10.10.1.2	3
All other addresses	0

Router R2's Routing Table

Network Address	Interface
10.10.1.1	0
10.10.53.0 to 10.10.53.255	1
10.10.20.0 to 10.10.20.255	2
10.10.43.0 to 10.10.43.255	3
All other addresses	0

- Selects appropriate interfaces for local network
- Selects interface 0 to get out on the Internet

Static and Dynamic Routing

Classification/taxonomy of routing methods:

- Static routing:
  - Uses **fixed routing tables** developed by network managers
    - Each node has its own routing table
    - Changes when computers added or removed
  - Used on relatively simple networks with few routing options that rarely change
- Dynamic routing aka Adaptive routing:
  - Uses routing tables at each node that are **updated dynamically** using Routing Protocols
  - Based on routing condition information exchanged between routing devices

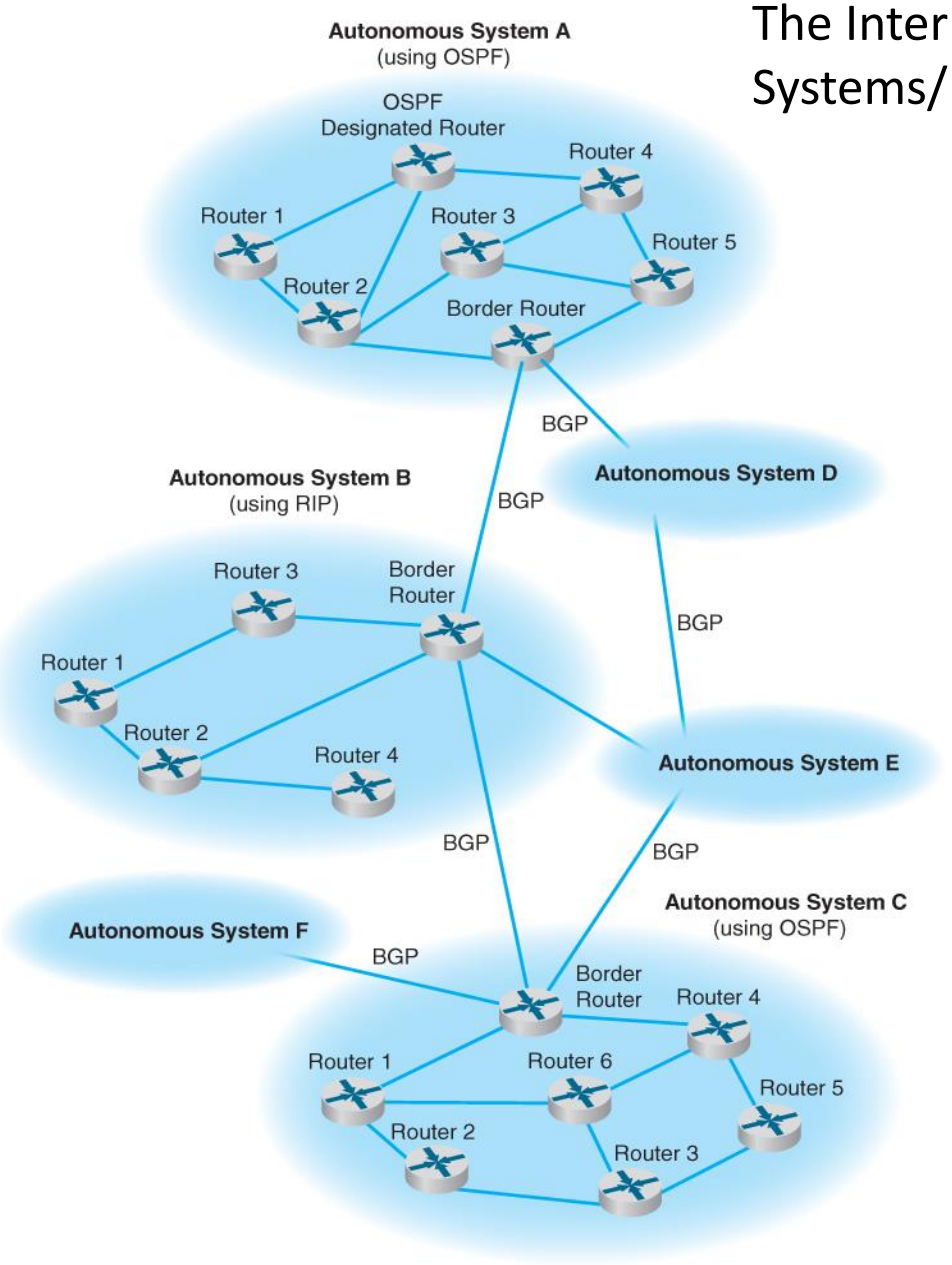
Hierarchical Routing on the Internet

- Having entry for each destination in a routing table is un-realistic in the Internet with ~800 millions of possible destinations
- Since the Internet is a network of networks, the routing follows its hierarchical structure.
- The basic building block of the Internet is called an Autonomous System which coincides with an **enterprise network** or an **Internet Service Provider** (ISP) network

There are two types of routing protocols

- Protocols that operate **within** an Autonomous System
  - **Intra**-AS or Intra-Domain protocols
- Protocols that connect (operate in between) the Autonomous Systems
  - **Inter**-AS or Inter-Domain protocols

The Internet as a connection of Autonomous Systems/Networks



- Intra-AS protocols are also called interior protocols
- Each Autonomous System uses its own interior protocol.
- Inter-AS protocols are also called exterior protocols
- Autonomous Systems are connected by the Border Routers that use the same exterior protocol

Taxonomy of Routing Protocols

Two **intra-domain** routing methods:

- **distance vector (DV)**
- **link state (LS)**

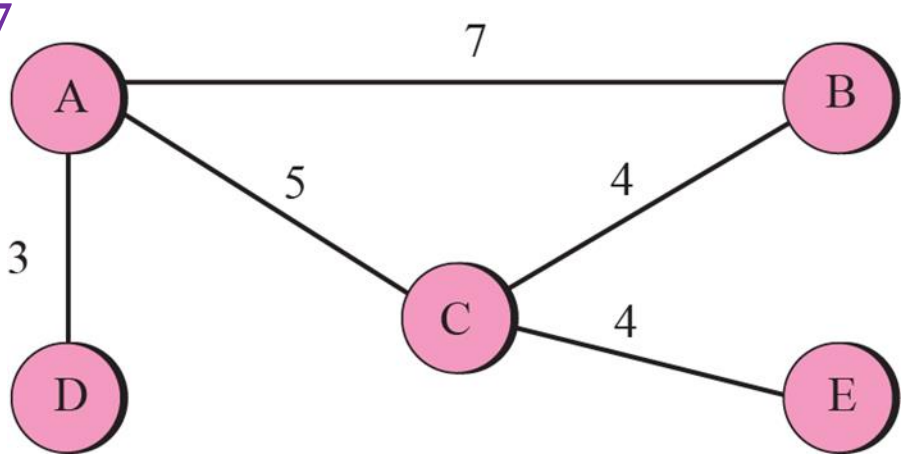
- Routing Information Protocol (RIP) is the implementation of the Distance Vector routing method.
- Open Shortest Path First (OSPF) is the implementation of the Link State routing method.
  - RIP and OSPF are **interior** routing protocols;

One **inter-domain** routing protocol: **path vector (PV)**

- Border Gateway Protocol (BGP) is the implementation of the Path Vector protocol.
  - BGP is an **exterior** routing protocol.

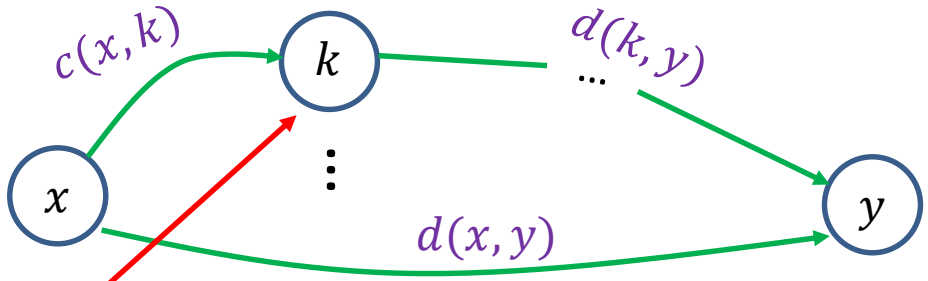
Distance Vector (DV) Routing

- *Distance Vector* routing protocols are based on the *Bellman-Ford* **dynamic programming** algorithm
- An **AS** (autonomous system) is seen as a *graph*, a set of nodes and lines (edges) connecting the nodes.
- A router is represented by a node, x, and a network by links connecting nodes. The final node, is denoted by y.
- Each path has a **cost** associated with it:
  $c(x, v)$ is the cost between the neighbouring nodes, x, and v, e.g., $c(A, B) = 7$
- How to find the best path between, say D and B, that **minimises the total cost** ("distance") $d(D, B)$



Bellman-Ford Algorithm

Consider a node x and all its neighbours k



The algorithm is based on the fact that:

- if all neighbours of node x know the shortest distance to node x, $c(x,k)$
- then the shortest distance between node x and y, $d(x,y)$
- can be found by adding the distance between node x and each neighbour, $c(x,k)$ to the neighbour's shortest distance to node y, $d(k,y)$ and then select the minimum over all neighbours

$$d(x,y) = \min_k(c(x,k) + d(k,y))$$

- It is an iterative process, since $d(k,y)$ is again calculated in the same way, until we reach the neighbour of the destination y.

Cont …

- The principle of the Bellman-Ford algorithm is simple and intuitive, but the algorithm itself looks circular.
- How have the neighbours k calculated the shortest path to the destination $d(k, y)$?
- It is an iterative process repeating the same consideration as before
- We create a shortest distance table (vector) for each node x with entries $d(x, y)$ using the following steps:

1. The shortest distance and the cost between a node and itself is initialized to 0: $d(x, x) = 0$
2. The shortest distance between a node and any other node is initialized to infinity: $d(x, y) = \infty$
3. The cost between a node and its neighbours is given $c(x, k)$,

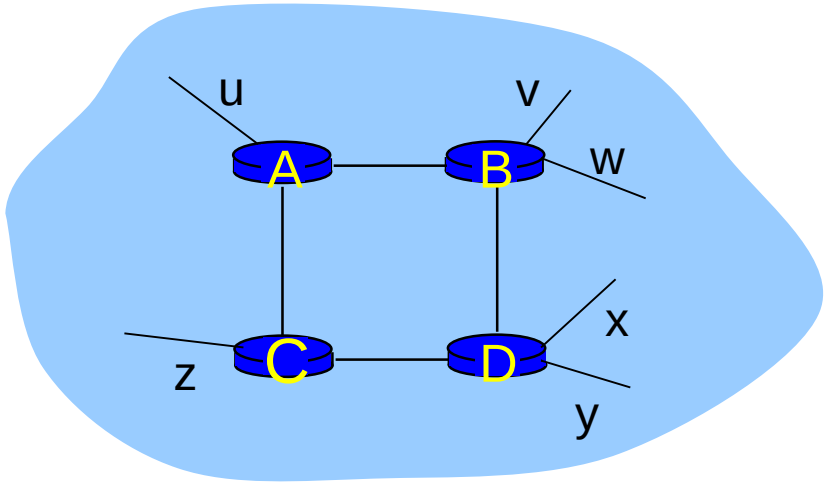
   (can be infinity if the nodes are not connected).

Distance Vector Routing Algorithm

- The B-F algorithm is designed to create the result *synchronously*, all nodes at the same time which is impractical.
- In distance vector routing, the cost $d(x, y)$ is normally the *hop counts* so the cost between any two neighbours is set to 1: $c(x, k) = 1$.
- Each router needs to update its routing table *asynchronously,* whenever it has received some information from its neighbours.
- Each router executes its part of the whole algorithm in the Bellman-Ford algorithm. Processing is *distributive*.
- After a router has updated its routing table, it should send the result to its neighbours so that they can also update their routing table.
- Each router should keep at least three pieces of information for each route: destination network, the cost, and the next hop.
- Information about each route received from a neighbour, has only two pieces of information: destination and cost.

RIP (Routing Information Protocol)

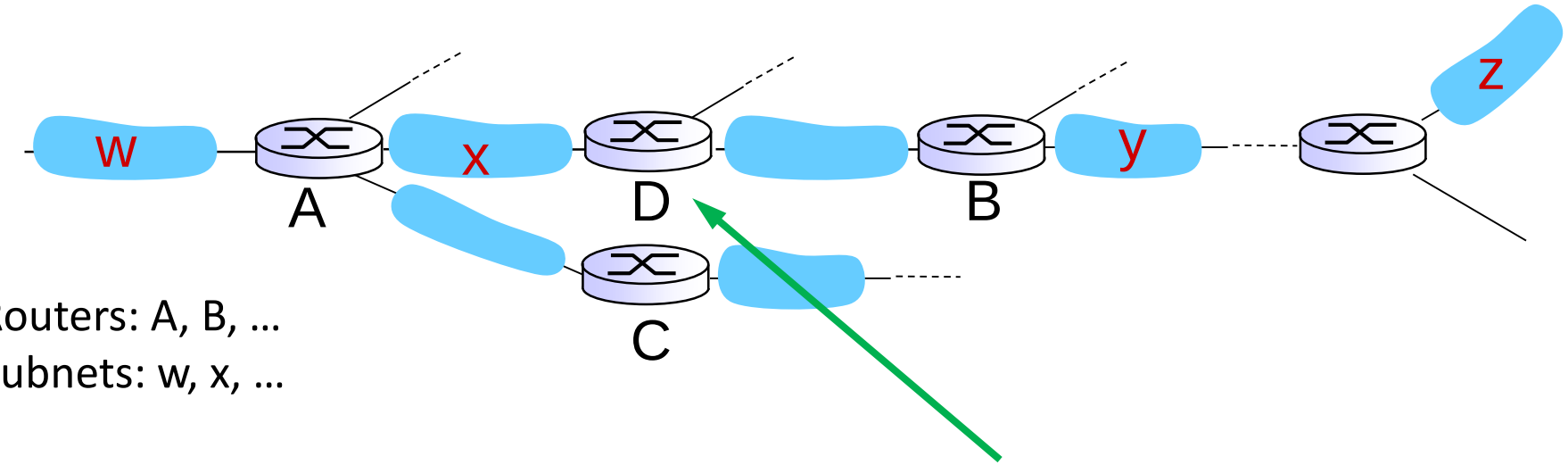
- Included in BSD-UNIX distribution in 1982 ! Version 2 in RFC2453
- RIP is based on the Distance Vector algorithm
  - distance metric/cost: number of hops (max = 15 hops),
  - each link has cost 1
  - neighbors exchange their DVs every 30 sec in a response message (aka advertisement)
  - each advertisement: list of up to 25 destination *subnets (in IP addressing sense)*



Number of hops from **router A** to destination *subnets:*

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

RIP: example



Routers: A, B, …
Subnets: w, x, …

routing table in router D

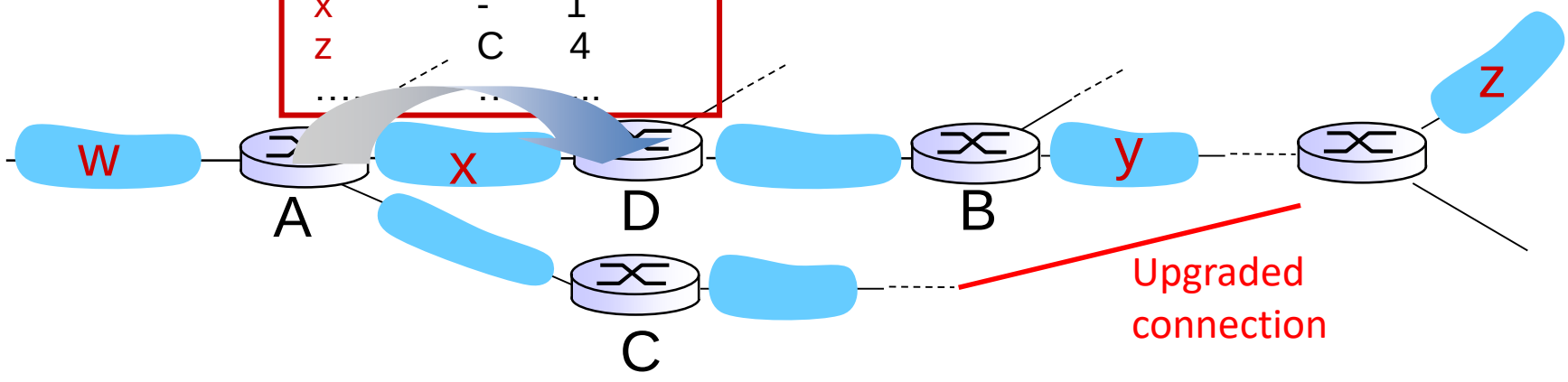
From **D** to **z** through **B**

destination subnet	next router	# hops to dest
w	A	2
y	B	2
z	B	7
x	--	1
….	….	….

RIP: example

A-to-D advertisement

dest	next	hops
w	-	1
x	-	1
z	C	4
....	....	....



Updated routing table in router D

destination subnet	next router	# hops to dest
w	A	2
y	B	2
z	B → A	7 → 5
x	--	1
....	....	....

RIP: link failure, 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)

RIP timers (1)

RIP uses three timers to support its operation.

- Periodic timer controls the sending of messages,
  - A random number between 25 and 35 s.
  - This is to prevent any possible synchronization (all messages at the same time) and therefore overload on an internet if routers update simultaneously.
  - It counts down; when zero is reached, the update message is sent, and the timer is randomly set once again

RIP timers (2)

- Expiration timer governs the validity of a route
  - If there is a problem with a network connection, and no update is received within the allotted 180 sec,
  - the route is considered expired and the hop count of the route is set to 16, which means the destination is unreachable
- Garbage collection timer advertises the failure of a route.
  - A router does not immediately purge a route from its table; advertise the route with a metric value of 16 and timer is set to 120 sec for that route.
  - When the count reaches zero, the route is purged from the table.

END of RIP

A Link-State Routing Algorithm

- In Link State routing, each node in the domain knows the entire topology of the domain discovered via "link state broadcast"
- Each node creates:
  - the list of nodes and links,
  - how they are connected
  - their types, cost (metric),
  - the condition of the links (up or down)
- A node uses the **Dijkstra algorithm** to build a **routing table**.
- Each node uses the same topology to create a routing table, but the routing table for each node is unique because the calculations are based on different interpretations of the topology.
- This is analogous to a city map: Two persons in two different cities may have the same map, but each needs to take a different route to reach his destination.

OSPF

- **Open Shortest Path First** (**OSPF**) protocol uses a Link State routing algorithm
- It is an **interior**, or Intra-AS routing protocols (operating within a single autonomous system).
- OSPF Version 2 for IPv4 is specified in RFC 2328 (1998)
- OSPF Version 3 for IPv6 is specified in RFC 5340 (2008)
- OSPF is widely used interior gateway protocol (IGP) in large enterprise networks.
- Intermediate System to Intermediate System (IS-IS) is another link-state dynamic routing protocol commonly used in large service provider networks.

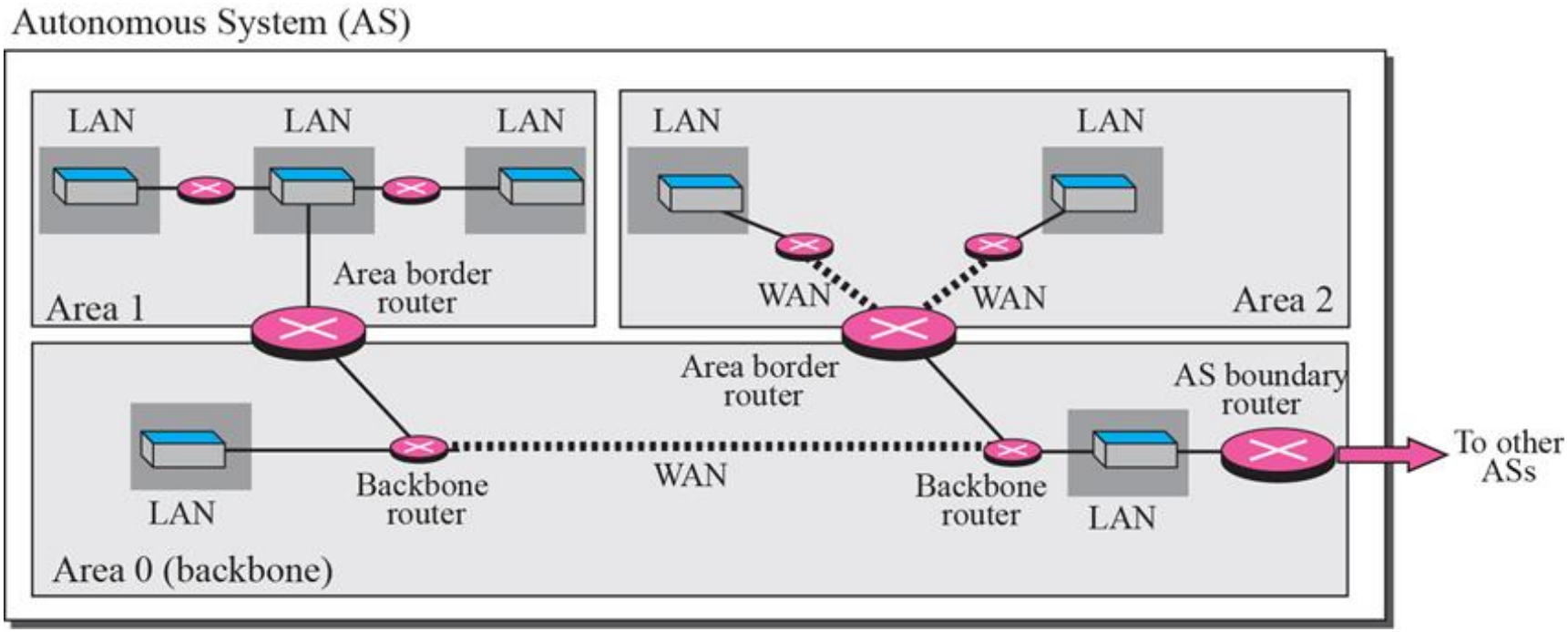
OSPF (2)

- OSPF gathers link state information from available routers and constructs a topology map of the network presented to the IP layer as a routing table.
- OSPF was designed to support variable-length subnet masking aka Classless Inter-Domain Routing (CIDR) addressing models.
- OSPF detects changes in the topology, such as link failures, and converges on a new loop-free routing structure within seconds.
- It computes the shortest path tree for each route using a method based on Dijkstra's algorithm, a shortest path first algorithm.
- The OSPF routing policies for constructing a route table are governed by link **cost factors** (*external metrics*) associated with each routing interface.
- **Cost factors** expressed as simple unit-less numbers may be:
  - the distance of a router (round-trip time),
  - data throughput of a link,
  - link availability and reliability

OSPF Network Model

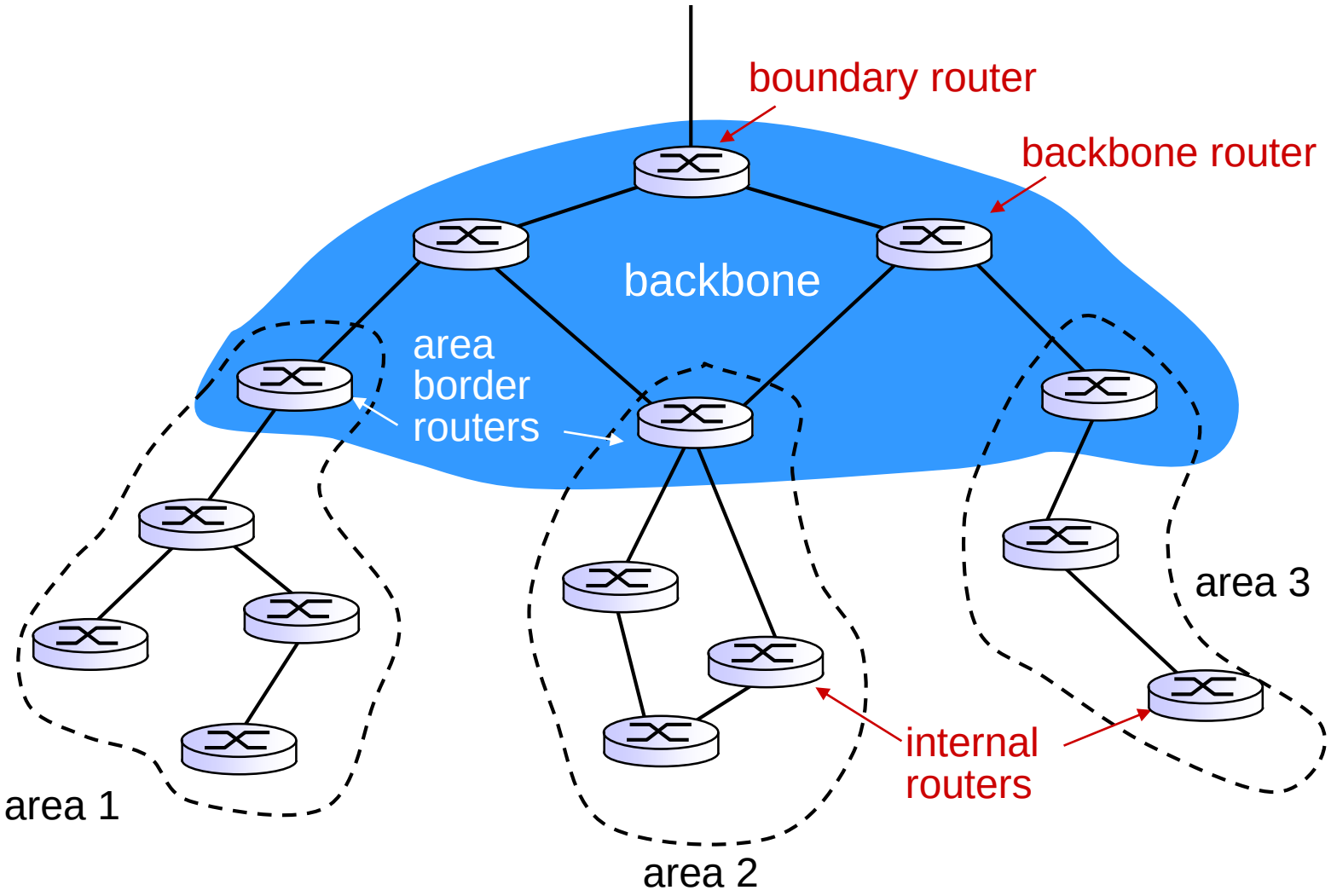
- An OSPF network may be structured, or subdivided, into routing *areas* to simplify administration and optimize traffic and resource utilization.
- Areas are identified by 32-bit numbers as in the IPv4 address notation.
- By convention, area 0 (zero), or 0.0.0.0, represents the core or *backbone* area of an OSPF network.
- The identifications of other areas may be chosen at will.
- Often, administrators select the IP address of a main router in an area as area identification.
- Each additional area must have a direct or virtual connection to the OSPF backbone area.
- Such connections are maintained by an interconnecting router, known as *area border router* (ABR).
- An ABR maintains separate link state databases for each area it serves and maintains summarized routes for all areas in the network.

An Autonomous System as seen by the OSPF protocol



- The routers inside the backbone are called the *backbone routers*. Note that a backbone router can also be an area border router.
- If, because of some problem, the connectivity between a backbone and an area is broken, a virtual link between routers must be created by the administration to allow continuity of the functions of the backbone as the primary area.

Another view of an in OSPF



Messages in OSPF

Five types of messages

- **Hello:** is used to discover other adjacent routers on its local links and networks.
- **Database Description:** contains descriptions of the topology of the AS or area.
  - These messages convey the contents of **the link-state database** (LSDB) for the autonomous system or area from one router to another.
- **Link State Request:** is used by one router to request updated information about a portion of the LSDB from another router.
- **Link State Update:** contains updated information about the state of certain links on the LSDB in response to the LS request
- **Link State Acknowledgment:** provides reliability to the link-state exchange process, by explicitly acknowledging receipt of a Link State Update message.

OSPF packet formats

- OSPF does not use UDP or TCP protocol, but encapsulates its data in IP datagrams with the port number 89.
- OSPF implements its own error detection and correction functions.
- The main OSPF packet header (RFC 2328 Appendix A.3.1) is the same for all five types of packets:

The Main OSPF Packet Header

Octet	Bit	0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15	16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
0	0	Version	Type	Packet Length
4	32	Router ID		
8	64	Area ID		
12	96	Checksum	Instance ID	0

OSPF "advanced" features (not in RIP)

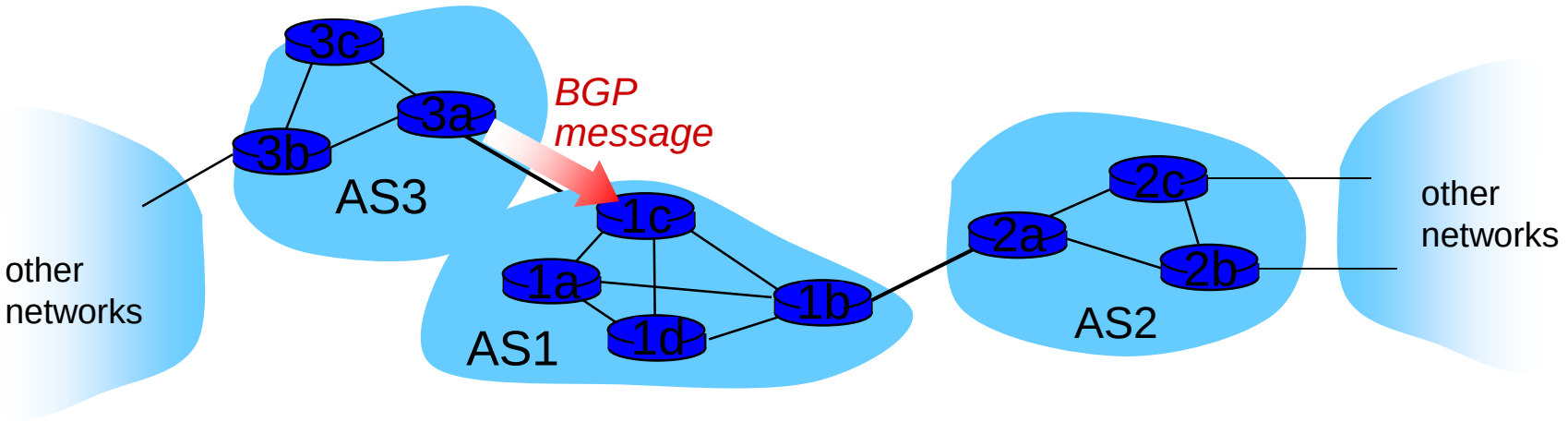
- *security:* all OSPF messages are authenticated (to prevent malicious intrusion)
- multiple same-cost paths allowed (only one path in RIP)
- for each link, multiple cost metrics for different QoS (e.g., satellite link cost set "low" for best effort QoS; high for real time QoS)
- integrated uni- and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

The Internet inter-AS routing: BGP

- BGP (Border Gateway Protocol): Version 4 in RFC 4271
- *The* de facto standard **inter-domain** routing protocol
  - "glue that holds the Internet together"
- BGP provides each AS a means to:
  - eBGP: obtain subnet reachability information from neighboring ASs.
  - iBGP: propagate reachability information to all AS-internal routers.
  - determine "good" routes to other networks based on reachability information and policy.
- allows subnet to advertise its existence to rest of the Internet: *"I am here"*

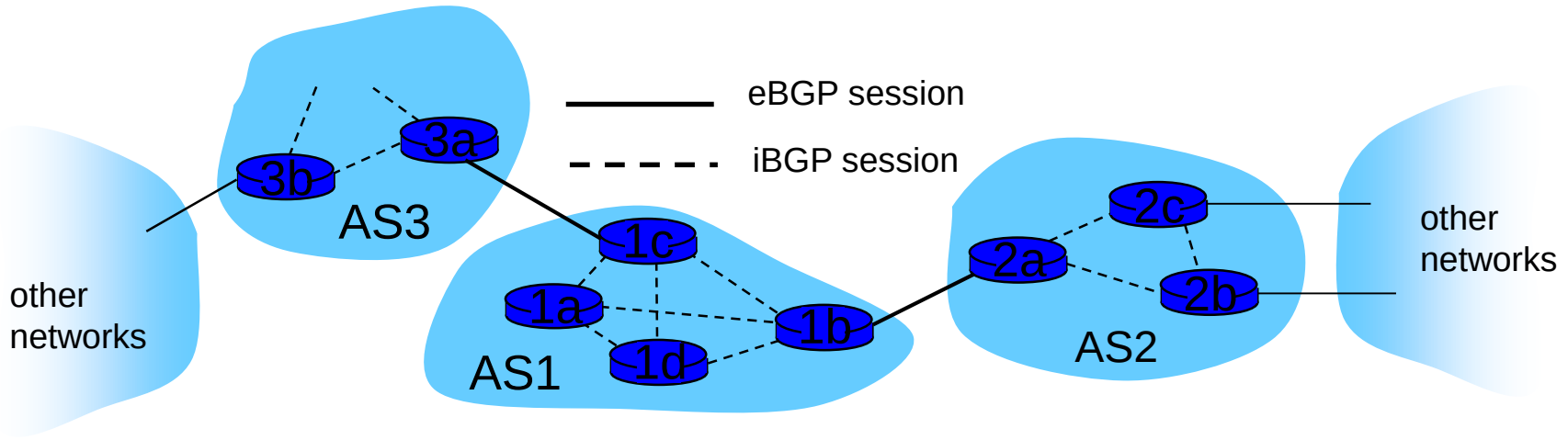
BGP basics

- BGP session: two BGP routers ("peers") exchange BGP messages:
  - advertising *paths* to different destination network **prefixes** ("path vector" protocol)
  - exchanged over semi-permanent TCP connections
- when AS3 advertises a prefix to AS1:
  - AS3 *promises* it will forward datagrams towards that prefix
  - AS3 can aggregate prefixes in its advertisement



BGP basics: distributing path information

- 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 a router learns of new prefix, it creates entry for prefix in its forwarding table.



Path attributes and 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)
- gateway router receiving route advertisement uses import policy to accept/decline
  - e.g., never route through AS x
  - *policy-based* routing

BGP route selection

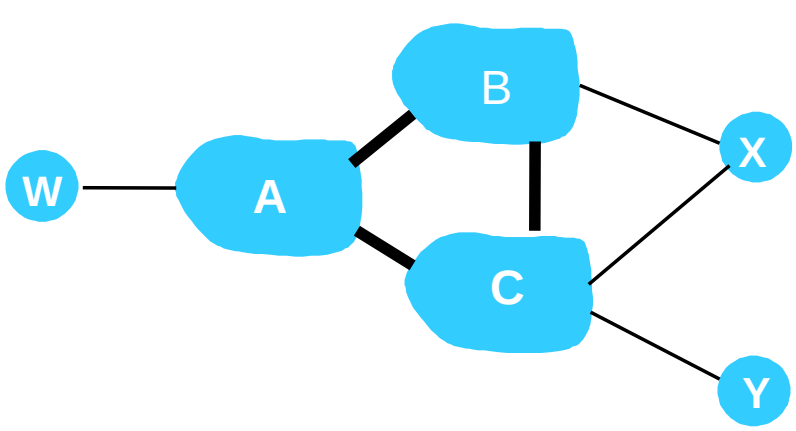
router may learn about more than 1 route to destination AS, selects route based on:

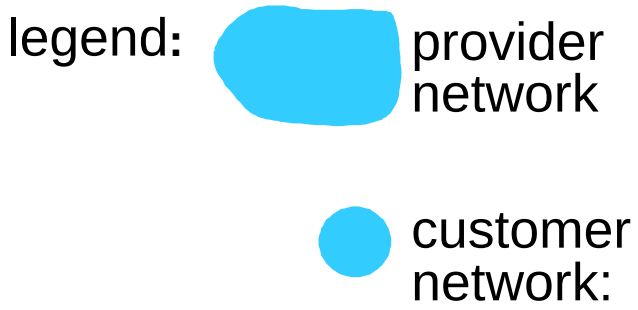
1. local preference value attribute: policy decision
2. shortest AS-PATH
3. closest NEXT-HOP router: hot potato routing
4. additional criteria

BGP messages

- BGP messages exchanged between peers over TCP connection
- 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

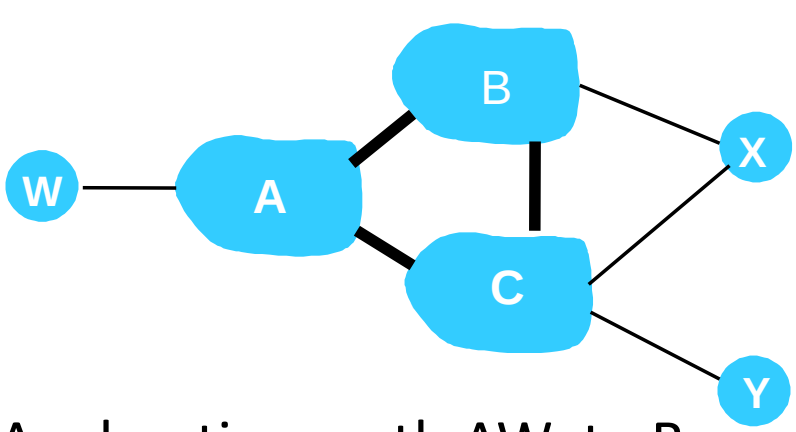
BGP routing policy

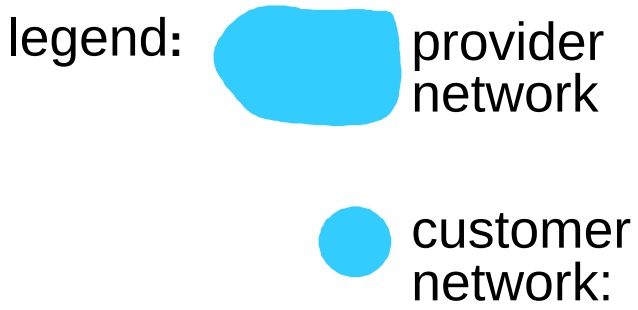




- A, B, C are *provider networks*
- X, W, Y are customers (of provider networks)
- X is *dual-homed:* attached to two networks
  - X does not want to route from B via X to C
  - .. so X will not advertise to B a route to C

BGP routing policy (2)





- A advertises path AW to B
- B advertises path BAW to X
- 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!