CS 43: Computer Networks

21: Intra and Interdomain Routing

November 21, 2019

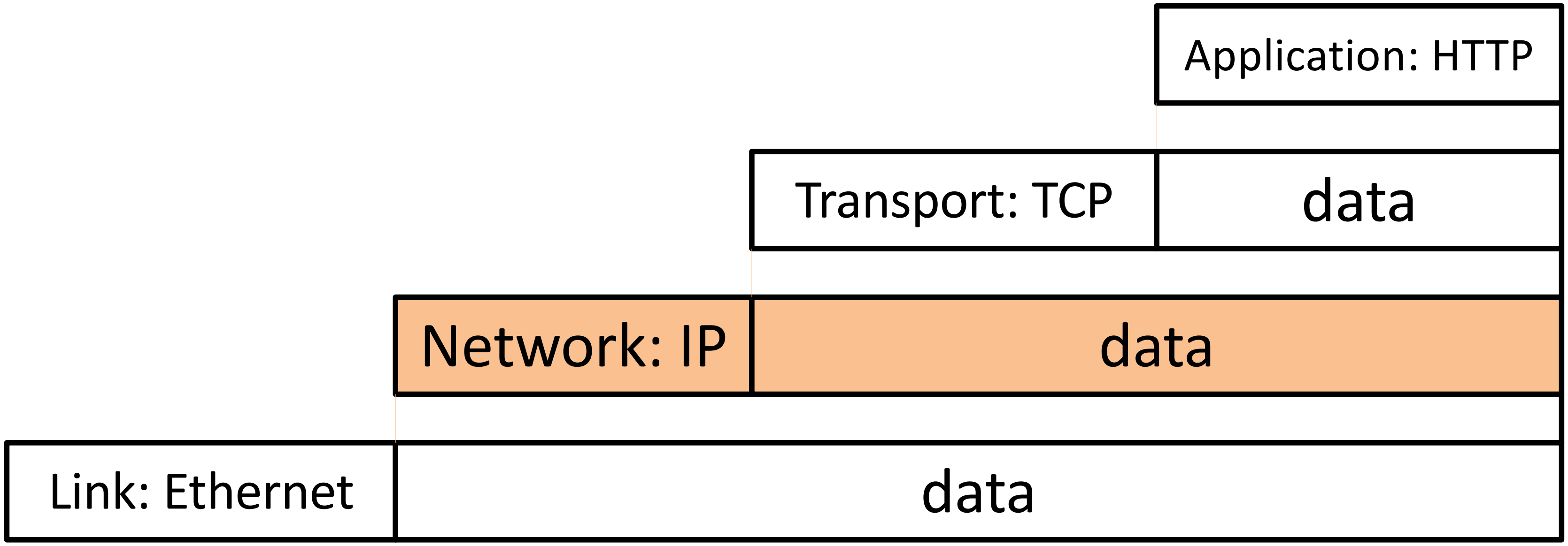
Adapted from Slides by: J.Kurose, D. Choffnes, K. Webb

SWARTHMORE COLLEGE

Reading Quiz

Network Layer

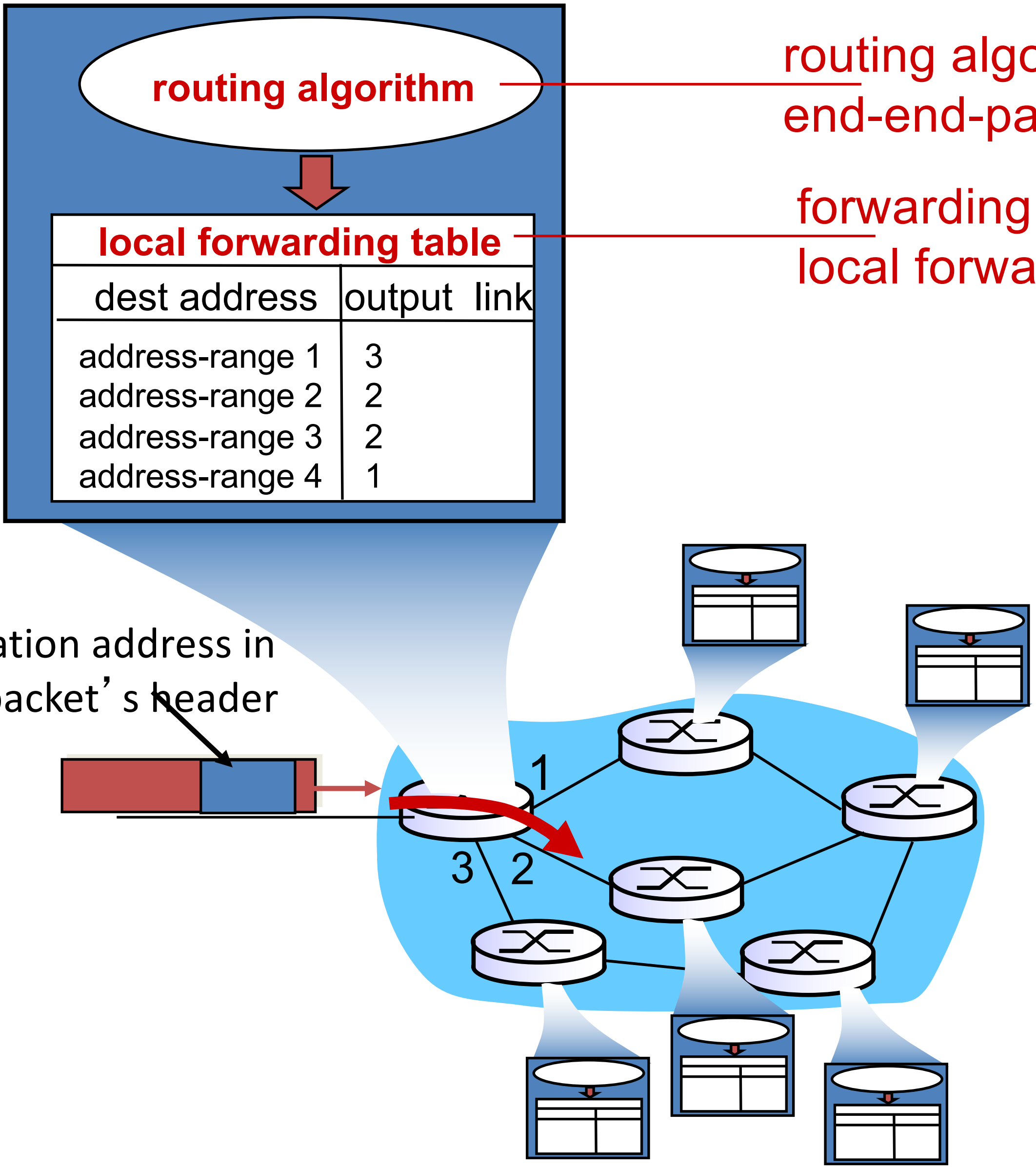
- Function: Route packets end-to-end on a network, through multiple hops



Network Layer Functions

- Forwarding: move packets from router's input to appropriate router output
  - Look up in a table
- Routing: determine route taken by packets from source to destination.
  - Populating the table

Interplay between routing, forwarding



routing algorithm

local forwarding table

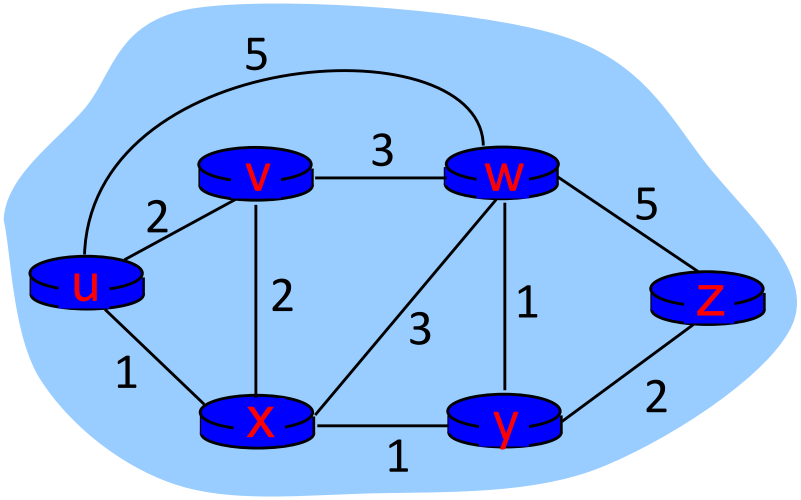
dest address	output link
address-range 1	3
address-range 2	2
address-range 3	2
address-range 4	1

IP destination address in arriving packet’s header

routing algorithm determines end-end-path through network

forwarding table determines local forwarding at this router

Graph Abstraction



c(x,x') = cost of link (x,x')

e.g., c(w,z) = 5

Cost of path $(x_1, x_2, x_3, \ldots, x_p) = c(x_1,x_2) + c(x_2,x_3) + \ldots + c(x_{p-1},x_p)$

Key question: what is the least-cost path between u and z ?

Routing algorithm: algorithm that finds that least cost path

Routing Algorithm Classes

Link State (Global)

- Routers maintain cost of each link in the network.

Distance Vector (Decentralized)

- Routers maintain next hop & cost of each destination.

Dijkstra’s Algorithm

```
Initialization:
  N' = {u}
  for all nodes v
    if v adjacent to u
        then $D(v) = c(u,v)$
    else $D(v) = \infty$

Loop
  find w not in N' such that D(w) is a minimum
  add w to N'
  update D(v) for all v adjacent to w and not in N' :
      $D(v) = \min( D(v), D(w) + c(w,v) )$
  /* new cost to v is either old cost to v or known
   shortest path cost to w plus cost from w to v */
until all nodes in N'
```

Pick the node (w) that isn’t already in N’ with the shortest distance (least cost path) and add it to N’.

Check all possible destinations from w. If going through w gives a lower cost to destination v, update D(v).

Dijkstra's Algorithm – Complexity

- With N nodes and E edges…

- As previously described it's O(N^2)
  - At each step, there are N nodes to choose next
  - Total of N steps (each node must be chosen)

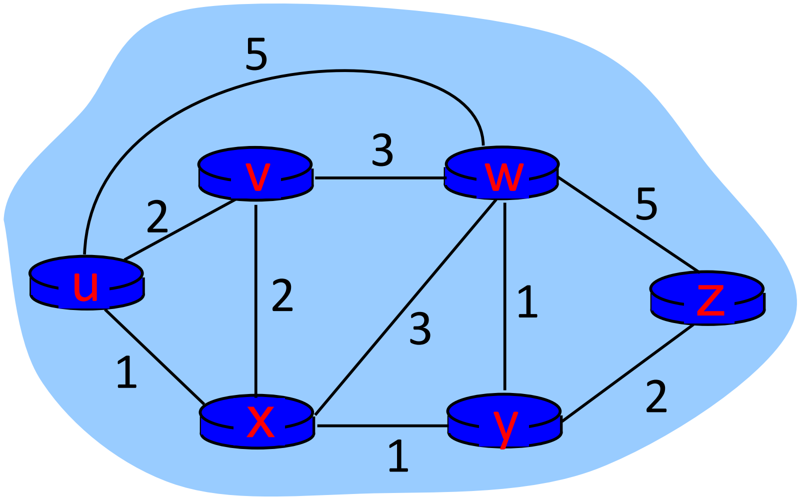
- Fastest known is O(N log N + E)
  - Uses a min-heap

Link State - Summary

- \+ Fast convergence (reacts to events quickly)
- \+ Small window of inconsistency

- – Large number of messages sent on events
- – Large routing tables as network size grows

Intradomain / Intra-AS Routing



Routing algorithm to find the least-cost path between routers within an Autonomous System

Intra-AS Routing

- Also known as *interior gateway protocols (IGP)*

Goal:
Get traffic that is already in an AS to a destination inside that same AS.

OSPF and IS-IS are deployed most commonly today

Real Protocols: OSPF vs. IS-IS

☐ Two different implementations of link-state routing

- OSPF

- Favored by companies, datacenters
- More optional features
- Built on top of IPv4
  - LSAs are sent via IPv4
  - OSPFv3 needed for IPv6

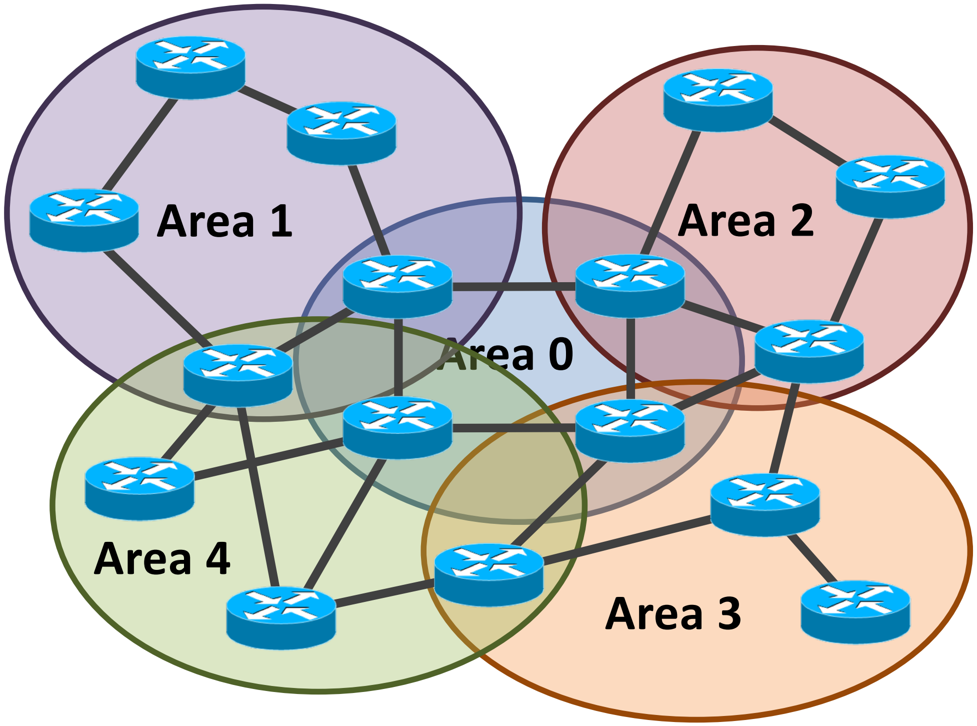
- IS-IS

- Favored by ISPs
- Less “chatty”
  - Less network overhead
  - Supports more devices
- Not tied to IP
  - Works with IPv4 or IPv6

Different Organizational Structure

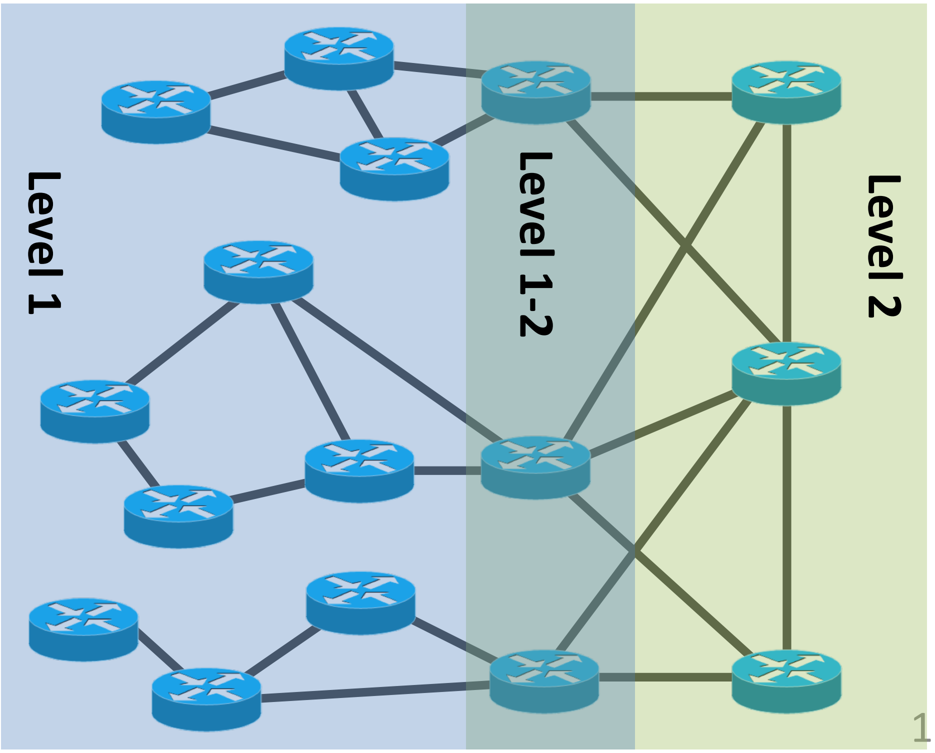
OSPF

- Organized around overlapping areas
- Area 0 is the core network



IS-IS

- Organized as a 2-level hierarchy
- Level 2 is the backbone



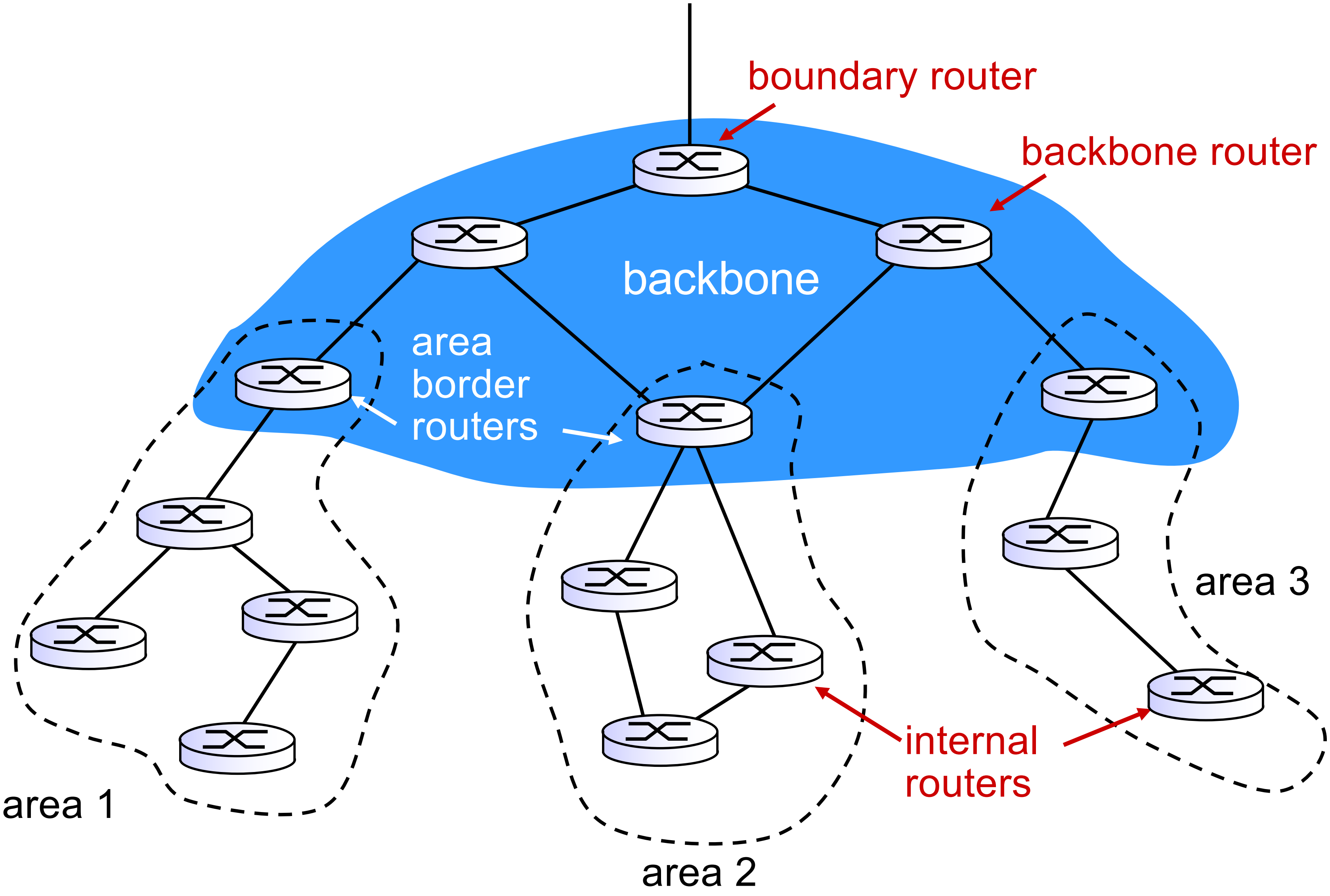
OSPF (Open Shortest Path First)

- Link state protocol (reliable flooding of LSAs)
- “Open”: standardized, publicly available implementations
- Multiple equal-cost paths allowed (load balancing)
- Additional features:
  - OSPF messages authenticated (to prevent malicious intrusion)
  - Hierarchical OSPF for large autonomous systems.

Hierarchical OSPF

- Two-level hierarchy*:* local area, backbone.
  - link-state advertisements only in area
  - each nodes has detailed area topology; only know direction (shortest path) to nets in other areas.
- Area border routers: "summarize" distances to nets in own area, advertise to other Area Border routers.
- Backbone routers*:* route between local areas
- Boundary routers: connect to other AS's.

Hierarchical OSPF



Routing Algorithm Classes

Link State (Global)

- Routers maintain cost of each link in the network.
- Connectivity/cost changes flooded to all routers.
- Converges quickly (less inconsistency, looping, etc.).
- Limited network sizes.

Distance Vector (Decentralized)

- Routers maintain next hop & cost of each destination.
- Connectivity/cost changes iteratively propagate from neighbor to neighbor.
- Requires multiple rounds to converge.
- Scales to large networks.

Bellman-Ford Equation

let

$d_x(y)$:= cost of least-cost path from x to y

then

$$d_x(y) = \mathit{min}_v \{c(x,v) + d_v(y) \}$$

cost from neighbor v to destination y

cost to neighbor v

min taken over all neighbors v of x

Distance Vectors

- Let $D_x(y)$ = vector of least cost from x to y

- Node x:
  - Knows cost to each neighbor v: c(x,v)
  - Maintains its neighbors’ distance vectors.
    For each neighbor v, x maintains:
    $\mathbf{D}_v = [D_v(y): y \in N]$

- As opposed to link state:
  - Only keeps state for yourself and direct neighbors

Distance Vector Algorithm

- Periodically, each node sends its own distance vector to neighbors
- Upon receiving new DV from neighbor, update its local DV using B-F equation:

$$D_x(y) \leftarrow \min_v\{c(x,v) + D_v(y)\} \text{ for each node } y \in N$$

- Under typical conditions, $D_x(y)$ will converge to the least cost $d_x(y)$

Distance Vector Algorithm

Iterative, asynchronous:

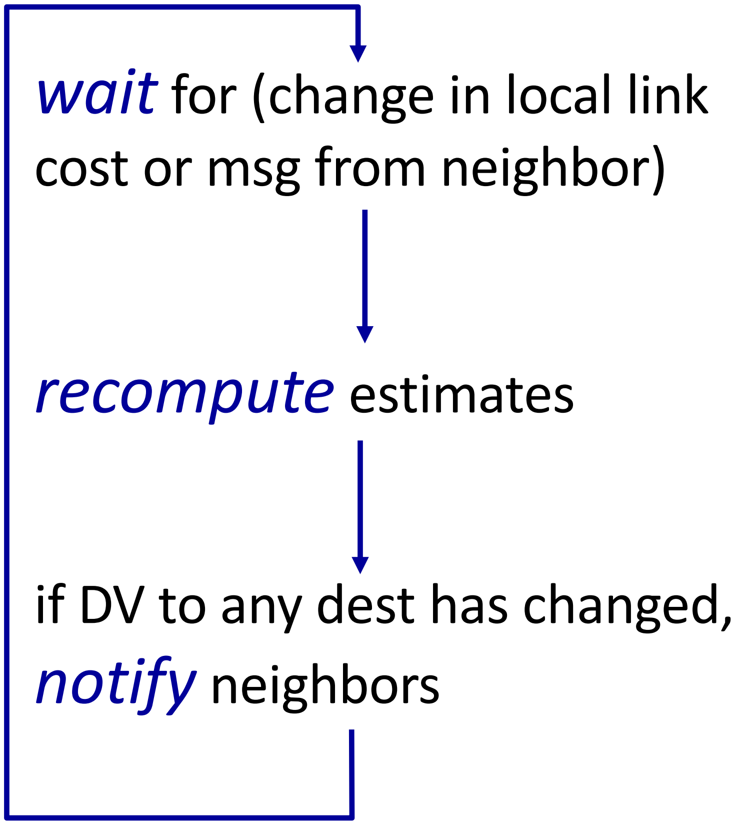
Iteration when:

- Local link cost change
- DV update from neighbor
- Periodic timer

Distributed:

- Each node knows only a portion of global link info

each node:

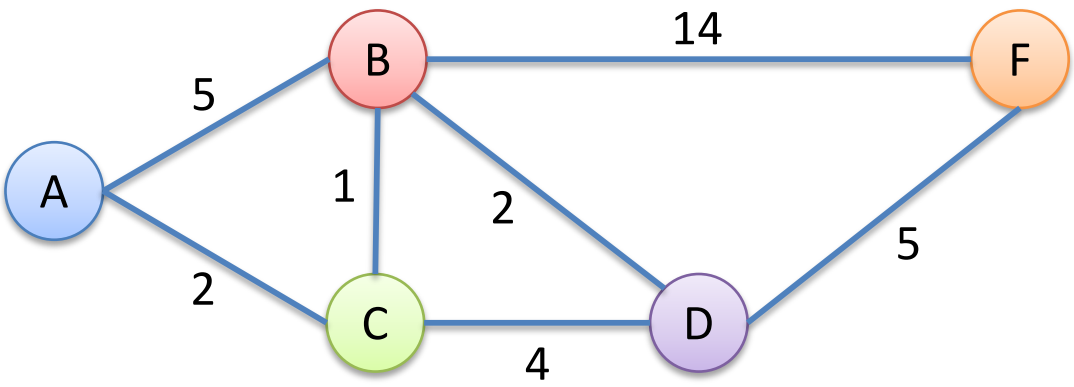


wait for (change in local link cost or msg from neighbor)

recompute estimates

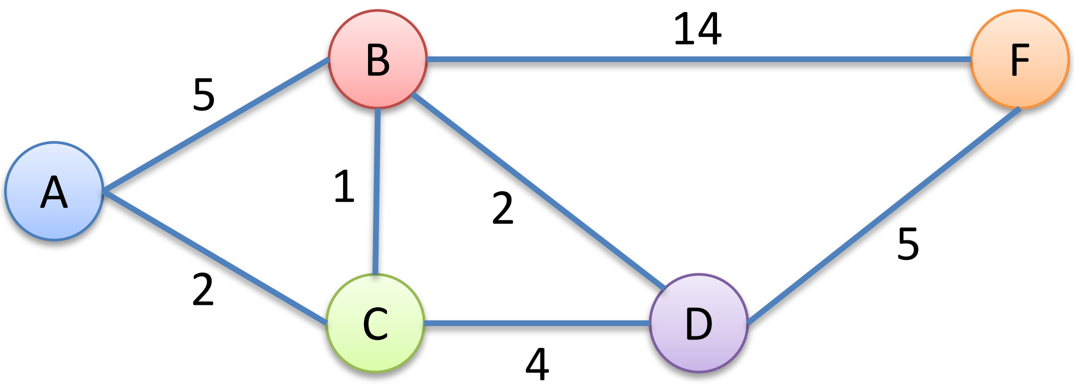
if DV to any dest has changed, *notify* neighbors

Distance Vector Example



- Same network as Dijkstra’s example, without node E.
- What I’ll show you next is routing table (of distance vectors) at each router.

Distance Vector – Round 0



Routers populate their forwarding table by taking the row minimum.

Router F

Via→ ↓ To	B	D
A		
B	14	
C		
D		5

Router A

Via→ ↓ To	B	C
B	5	
C		2
D		
F		

Router B

Via→ ↓ To	A	C	D	F
A	5			
C		1		
D			2	
F				14

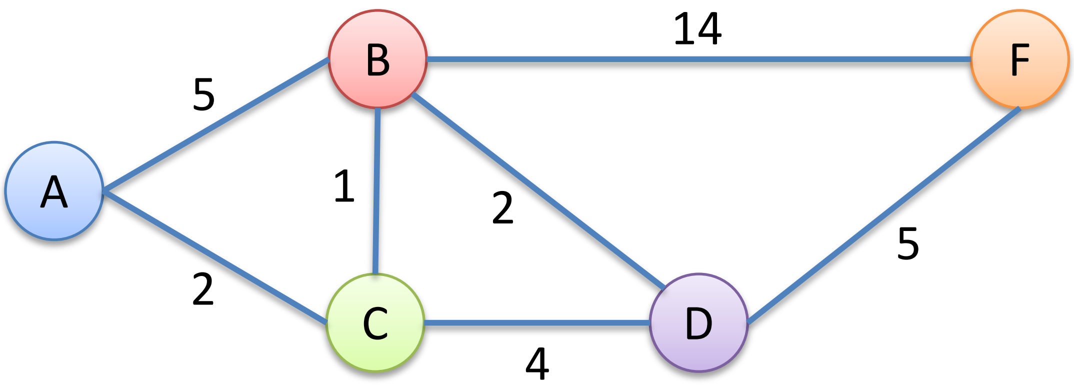
Router C

Via→ ↓ To	A	B	D
A	2		
B		1	
D			4
F			

Router D

Via→ ↓ To	B	C	F
A			
B	2		
C		4	
F			5

Distance Vector – Round 0



Router F

Via→ ↓ To	B	D
A		
B	14	
C		
D		5

Router exchange their local vectors with direct neighbors.
We'll assume they all exchange at once (synchronous). (Not realistic)

Router A

Via→ ↓ To	B	C
B	5	
C		2
D		
F		

Router B

Via→ ↓ To	A	C	D	F
A	5			
C		1		
D			2	
F				14

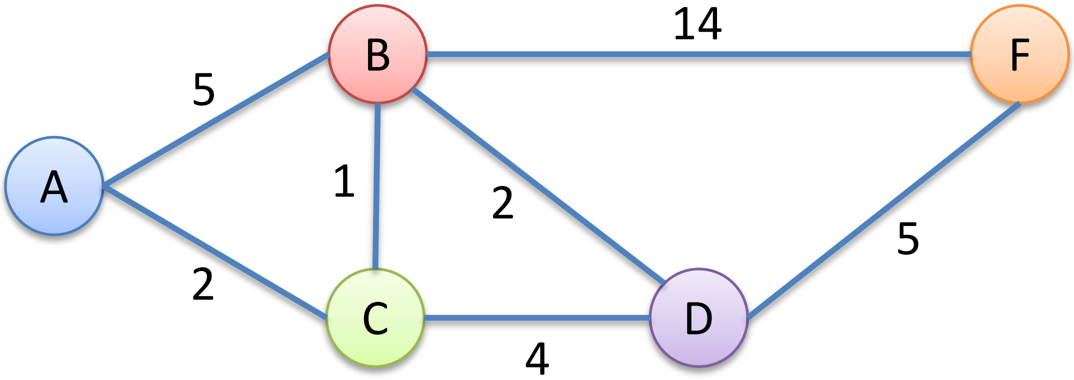
Router C

Via→ ↓ To	A	B	D
A	2		
B		1	
D			4
F			

Router D

Via→ ↓ To	B	C	F
A			
B	2		
C		4	
F			5

Distance Vector – Round 1



Router F

Via→ ↓ To	B	D
A		
B	14	
C		
D		5

A will send to neighbors (B & C):
I can get to B in 5 and C in 2.

Router A

Via→ ↓ To	B	C
B	5	
C		2
D		
F		

Router B

Via→ ↓ To	A	C	D	F
A	5			
C	7	1		
D			2	
F				14

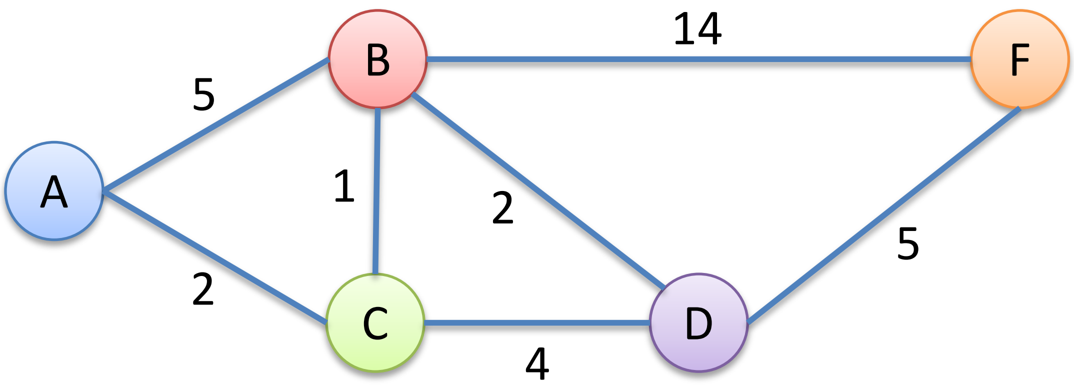
Router C

Via→ ↓ To	A	B	D
A	2		
B	7	1	
D			4
F			

Router D

Via→ ↓ To	B	C	F
A			
B	2		
C		4	
F			5

Distance Vector – Round 1



Router F

Via→ ↓ To	B	D
A	19	
B	14	
C	15	
D	16	5

B will send to neighbors (A, C, D, F):
I can get to A in 5, C in 1, D in 2, and F in 14.

Router A

Via→ ↓ To	B	C
B	5	
C	6	2
D	7	
F	19	

Router B

Via→ ↓ To	A	C	D	F
A	5			
C	7	1		
D			2	
F				14

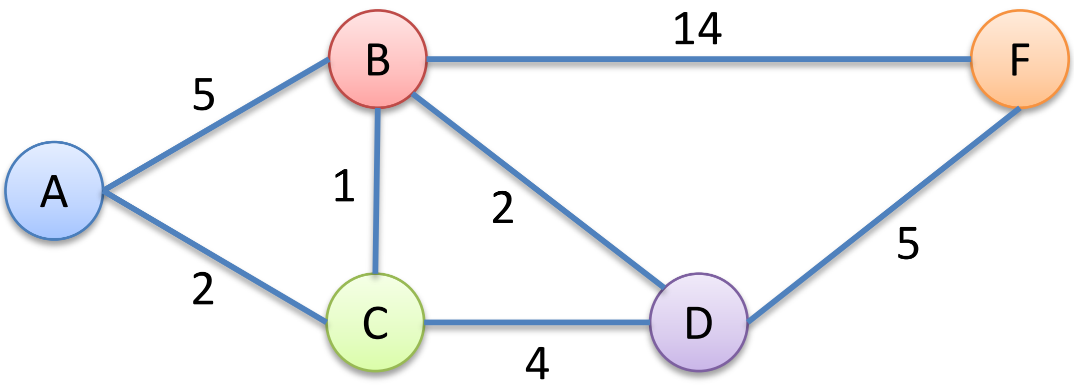
Router C

Via→ ↓ To	A	B	D
A	2	6	
B	7	1	
D		3	4
F		15	

Router D

Via→ ↓ To	B	C	F
A	7		
B	2		
C	3	4	
F	16		5

Distance Vector – Round 1



C will send to neighbors (A, B, D):
I can get to A in 2, B in 1, and D in 4.

Router F

Via→ ↓ To	B	D
A	19	
B	14	
C	15	
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	19	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1		
D		5	2	
F				14

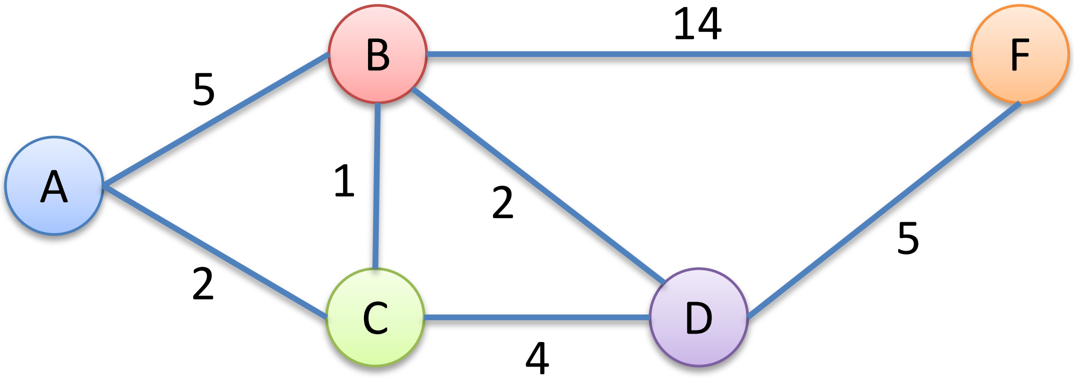
Router C

Via→ ↓ To	A	B	D
A	2	6	
B	7	1	
D		3	4
F		15	

Router D

Via→ ↓ To	B	C	F
A	7	6	
B	2	5	
C	3	4	
F	16		5

Distance Vector – Round 1



Router F

Via→ ↓ To	B	D
A	19	
B	14	7
C	15	9
D	16	5

D will send to neighbors (B, C, F):
I can get to B in 2, C in 4, and F in 5.

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	19	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	
F			7	14

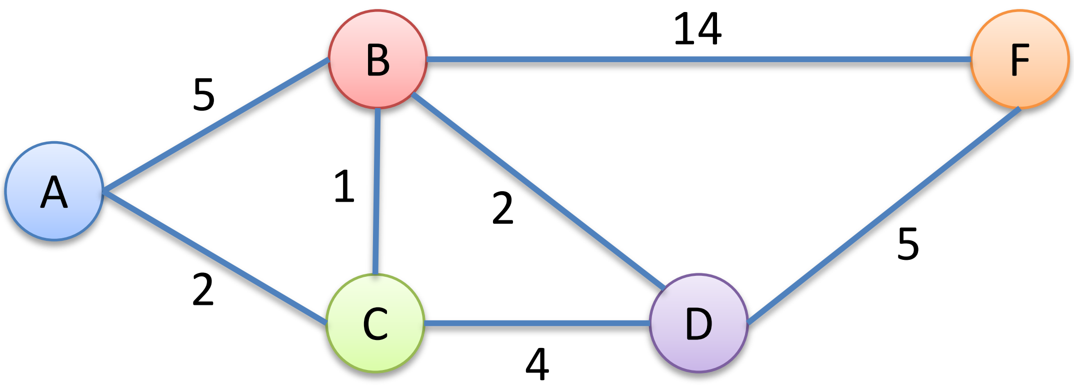
Router C

Via→ ↓ To	A	B	D
A	2	6	
B	7	1	6
D		3	4
F		15	9

Router D

Via→ ↓ To	B	C	F
A	7	6	
B	2	5	
C	3	4	
F	16		5

Distance Vector – Round 1



F will send to neighbors (B, D):
I can get to B in 14, D in 5.

Router F

Via→ ↓ To	B	D
A	19	
B	14	7
C	15	9
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	19	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	19
F			7	14

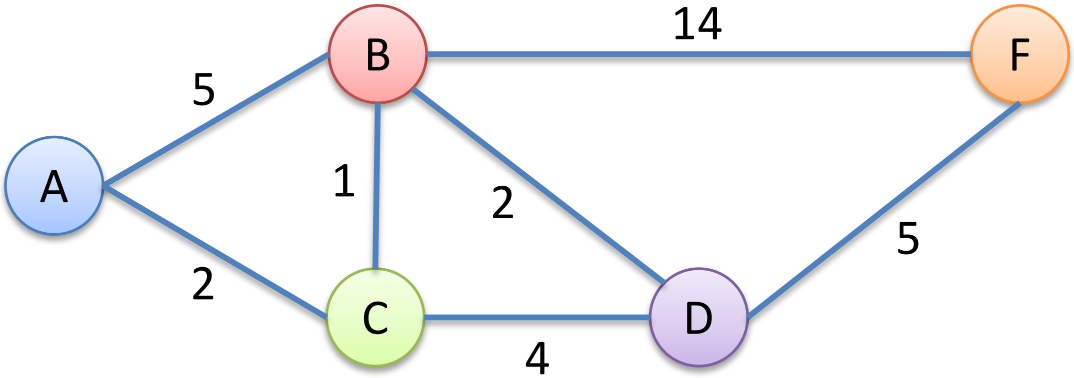
Router C

Via→ ↓ To	A	B	D
A	2	6	
B	7	1	6
D		3	4
F		15	9

Router D

Via→ ↓ To	B	C	F
A	7	6	
B	2	5	19
C	3	4	
F	16		5

At the end of round 1, how many routers need to update their forwarding tables?



A – 1, B – 2, C – 3, D – 4, E – 5

Router F

Via→ ↓ To	B	D
A	19	
B	14	7
C	15	9
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	19	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	19
F			7	14

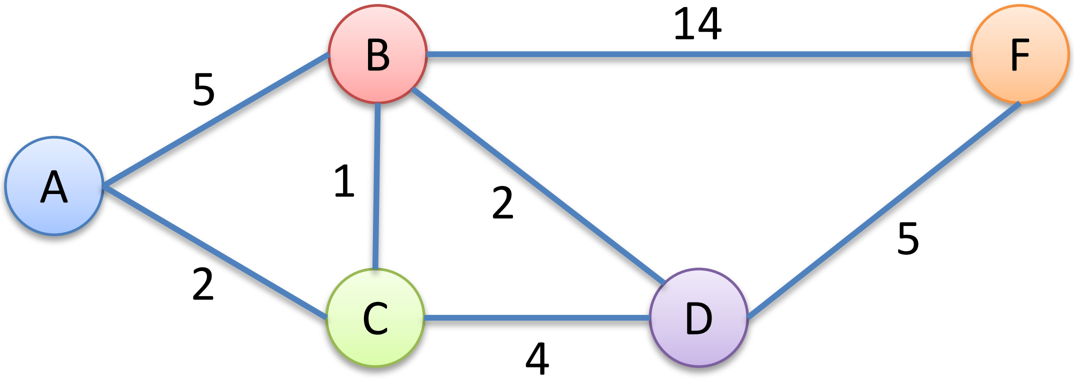
Router C

Via→ ↓ To	A	B	D
A	2	6	
B	7	1	6
D		3	4
F		15	9

Router D

Via→ ↓ To	B	C	F
A	7	6	
B	2	5	19
C	3	4	
F	16		5

Distance Vector – Round 2



Router F

Via→ ↓ To	B	D
A	19	
B	14	7
C	15	9
D	16	5

Each router advertises the best cost it has to each destination.
Nothing new to learn from A or F, so we'll skip their announcements.

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	19	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	19
F			7	14

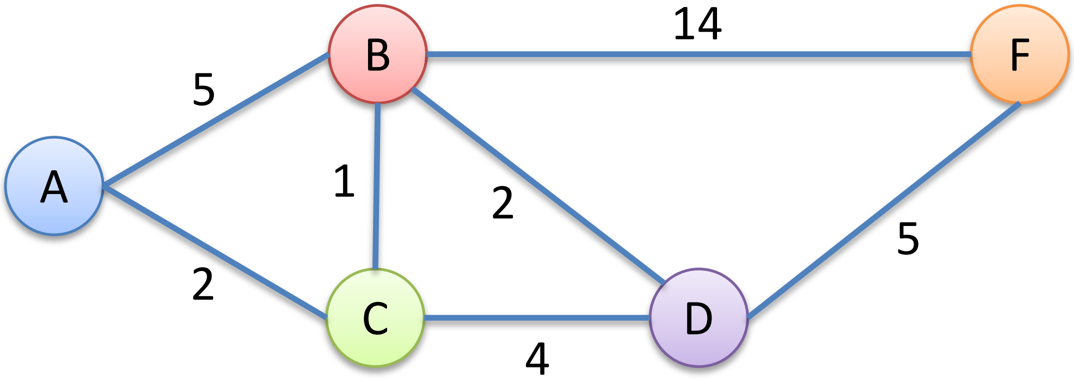
Router C

Via→ ↓ To	A	B	D
A	2	6	
B	7	1	6
D		3	4
F		15	9

Router D

Via→ ↓ To	B	C	F
A	7	6	
B	2	5	19
C	3	4	
F	16		5

Distance Vector – Round 2



B will send to neighbors (A, C, D, F):
I can get to A in 3, C in 1, D in 2, and F in 7.

Router F

Via→ ↓ To	B	D
A	17	
B	14	7
C	15	9
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	12	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	19
F			7	14

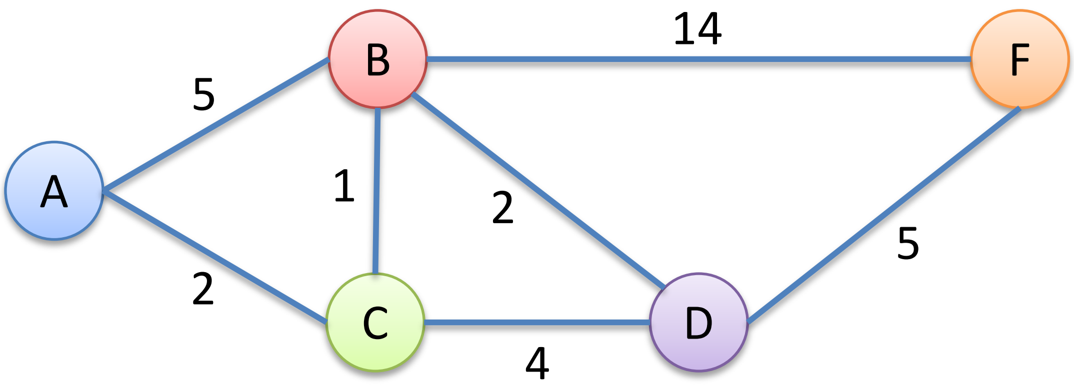
Router C

Via→ ↓ To	A	B	D
A	2	4?	
B	7	1	6
D		3	4
F		8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	
B	2	5	19
C	3	4	
F	9?		5

Distance Vector – Round 2



C will send to neighbors (A, B, D):
I can get to A in 2, B in 1, D in 3, and F in 9.

Router F

Via→ ↓ To	B	D
A	17	
B	14	7
C	15	9
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	5
F	12	11

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		4?	2	19
F		10	7	14

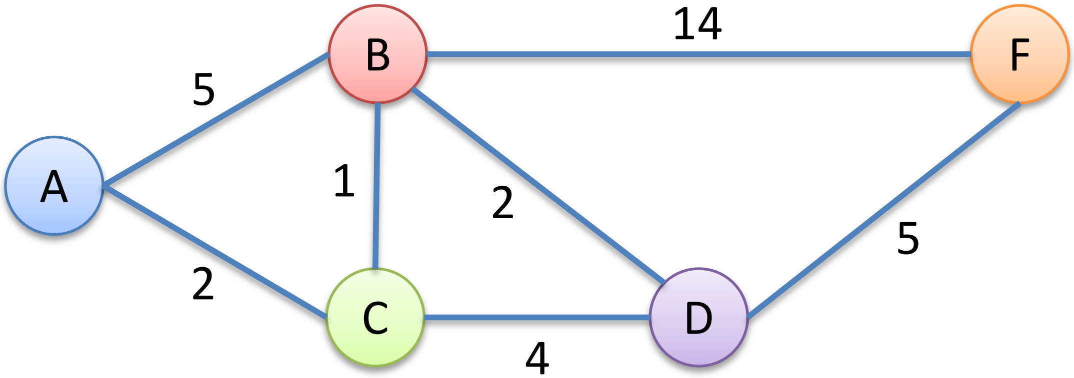
Router C

Via→ ↓ To	A	B	D
A	2	4?	
B	7	1	6
D		3	4
F		8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	
B	2	5	19
C	3	4	
F	9?	13?	5

Distance Vector – Round 2



This process repeats for a while…

Router F

Via→ ↓ To	B	D
A	17	
B	14	7
C	15	9
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	5
F	12	11

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		4?	2	19
F		10	7	14

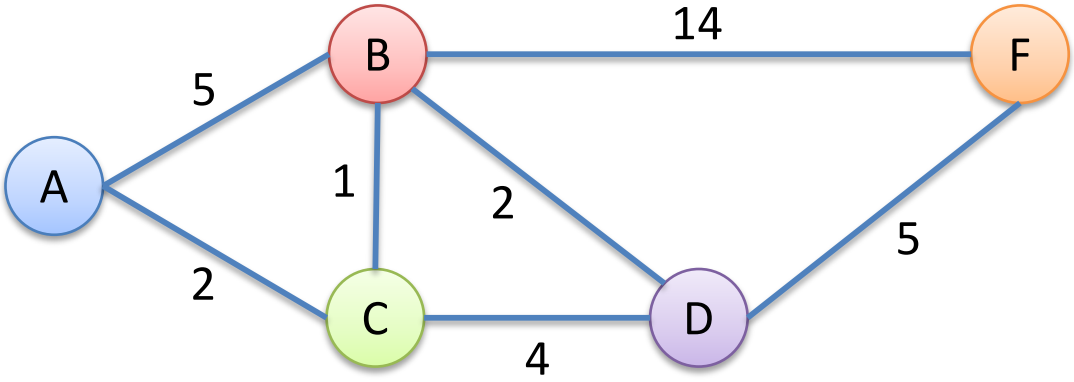
Router C

Via→ ↓ To	A	B	D
A	2	4?	
B	7	1	6
D		3	4
F		8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	
B	2	5	19
C	3	4	
F	9?	13?	5

Distance Vector – Convergence



Eventually, we reach a converged state.

Router F

Via→ ↓ To	B	D
A	17	10
B	14	7
C	15	8
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	5
F	12	10

Router B

Via→ ↓ To	A	C	D	F
A	5	3	7	24
C	7	1	4	22
D	10	4	2	19
F	15	9	7	14

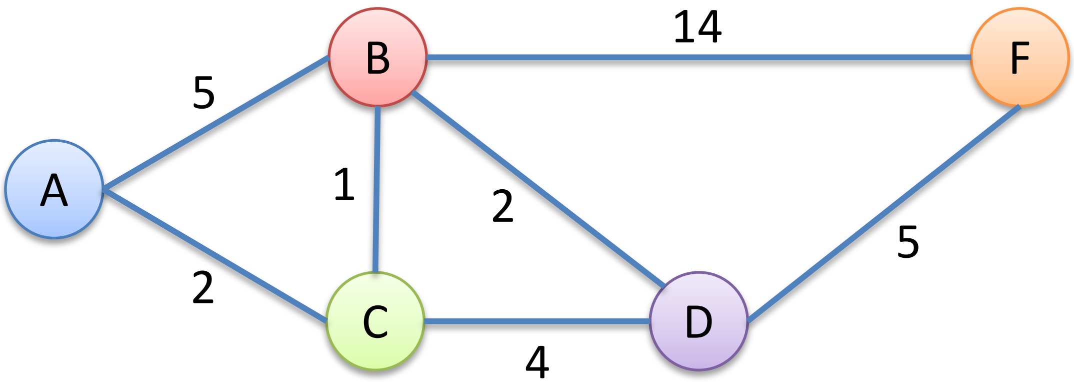
Router C

Via→ ↓ To	A	B	D
A	2	4	9
B	7	1	6
D	7	3	4
F	12	8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	15
B	2	5	12
C	3	4	13
F	9	12	5

Distance Vector – Convergence



Final forwarding tables:

Router F

Via→ ↓ To	B	D
A	17	10
B	14	7
C	15	8
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	5
F	12	10

Router B

Via→ ↓ To	A	C	D	F
A	5	3	7	24
C	7	1	4	22
D	10	4	2	19
F	15	9	7	14

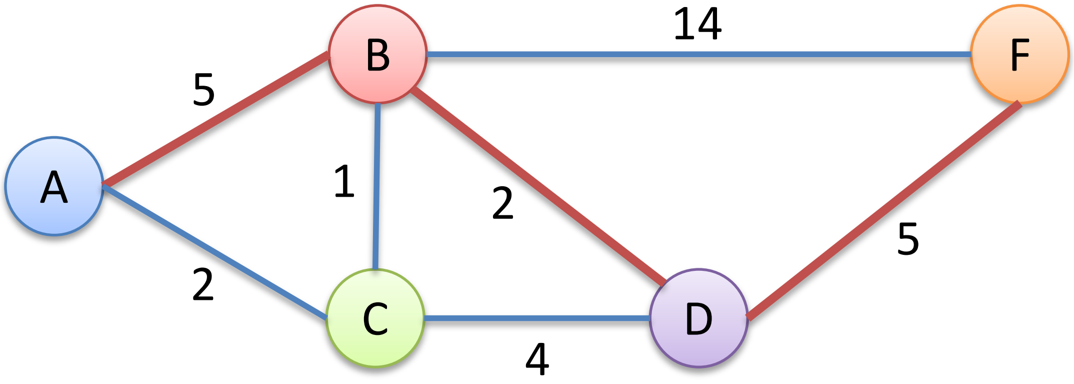
Router C

Via→ ↓ To	A	B	D
A	2	4	9
B	7	1	6
D	7	3	4
F	12	8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	15
B	2	5	12
C	3	4	13
F	9	12	5

Of the links in red below, for how many would a failure cause a loop?



A – 0, B – 1, C – 2, D – 3

Consider the failures independently (not all at the same time).

Router F

Via→ ↓ To	B	D
A	17	10
B	14	7
C	15	8
D	16	5

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	5
F	12	10

Router B

Via→ ↓ To	A	C	D	F
A	5	3	7	24
C	7	1	4	22
D	10	4	2	19
F	15	9	7	14

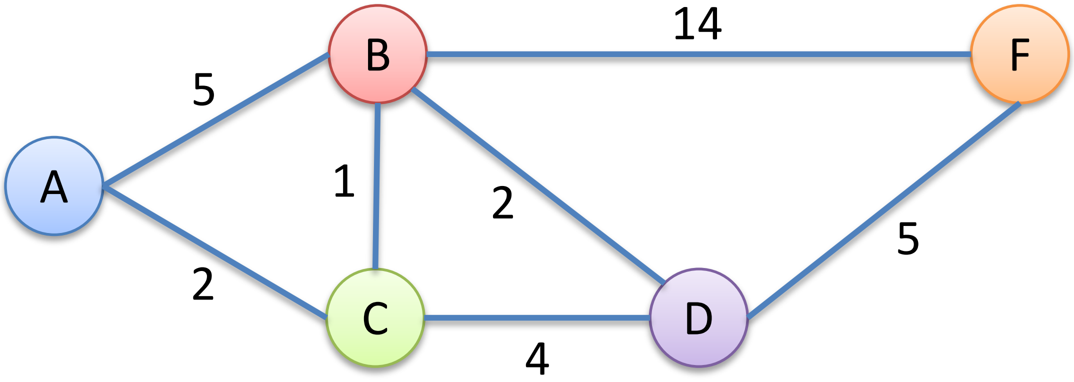
Router C

Via→ ↓ To	A	B	D
A	2	4	9
B	7	1	6
D	7	3	4
F	12	8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	15
B	2	5	12
C	3	4	13
F	9	12	5

Rewind: Distance Vector – Round 2



Router F

Via→ ↓ To	B	D
A	17	
B	14	7
C	15	9
D	16	5

B will send to neighbors (A, C, D, F):
I can get to A in 3, C in 1, D in 2, and F in 7.

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	12	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	19
F			7	14

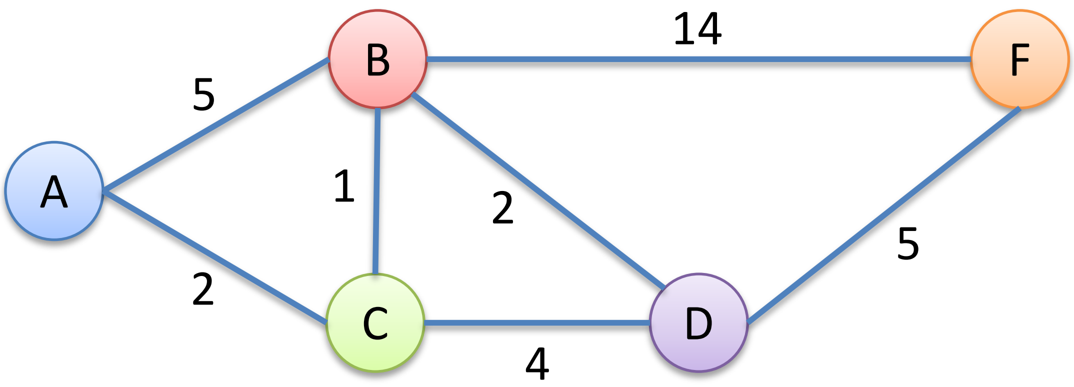
Router C

Via→ ↓ To	A	B	D
A	2	4?	
B	7	1	6
D		3	4
F		8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	
B	2	5	19
C	3	4	
F	9?		5

Rewind: Distance Vector – Round 2



Router F

Via→ ↓ To	B	D
A	17	
B	14	7
C	15	9
D	16	5

Poisoned reverse: Don’t advertise a lower value to a neighbor if you go through that neighbor to get there!

No!

∞

Router A

Via→ ↓ To	B	C
B	5	3
C	6	2
D	7	6
F	12	

Router B

Via→ ↓ To	A	C	D	F
A	5	3		
C	7	1	6	
D		5	2	19
F			7	14

Router C

Via→ ↓ To	A	B	D
A	2	~~4?~~	
B	7	1	6
D		3	4
F		8	9

Router D

Via→ ↓ To	B	C	F
A	5	6	
B	2	5	19
C	3	4	
F	9?		5

Loop-prevention

- Route poisoning helps prevent loops, but doesn’t guarantee loop free.
- Other mechanisms help too
- There will always be a window of vulnerability

Summary

Link State

+ Fast convergence (reacts to events quickly)
+ Small window of inconsistency

– Large number of messages sent on events
– Large routing tables as network size grows

Distance Vector

+ Distributed (small tables)
+ No flooding (fewer messages)

– Slower convergence
– Larger window of inconsistency

Real Protocols

Link State

- Open Shortest Path First (OSPF)
- Intermediate system to intermediate system (IS-IS)

Distance Vector

- Routing Information Protocol (RIP)
- Interior Gateway Routing Protocol (IGRP – Cisco)
- Border Gateway Protocol (BGP) (sort of)

Internet/inter-AS Routing

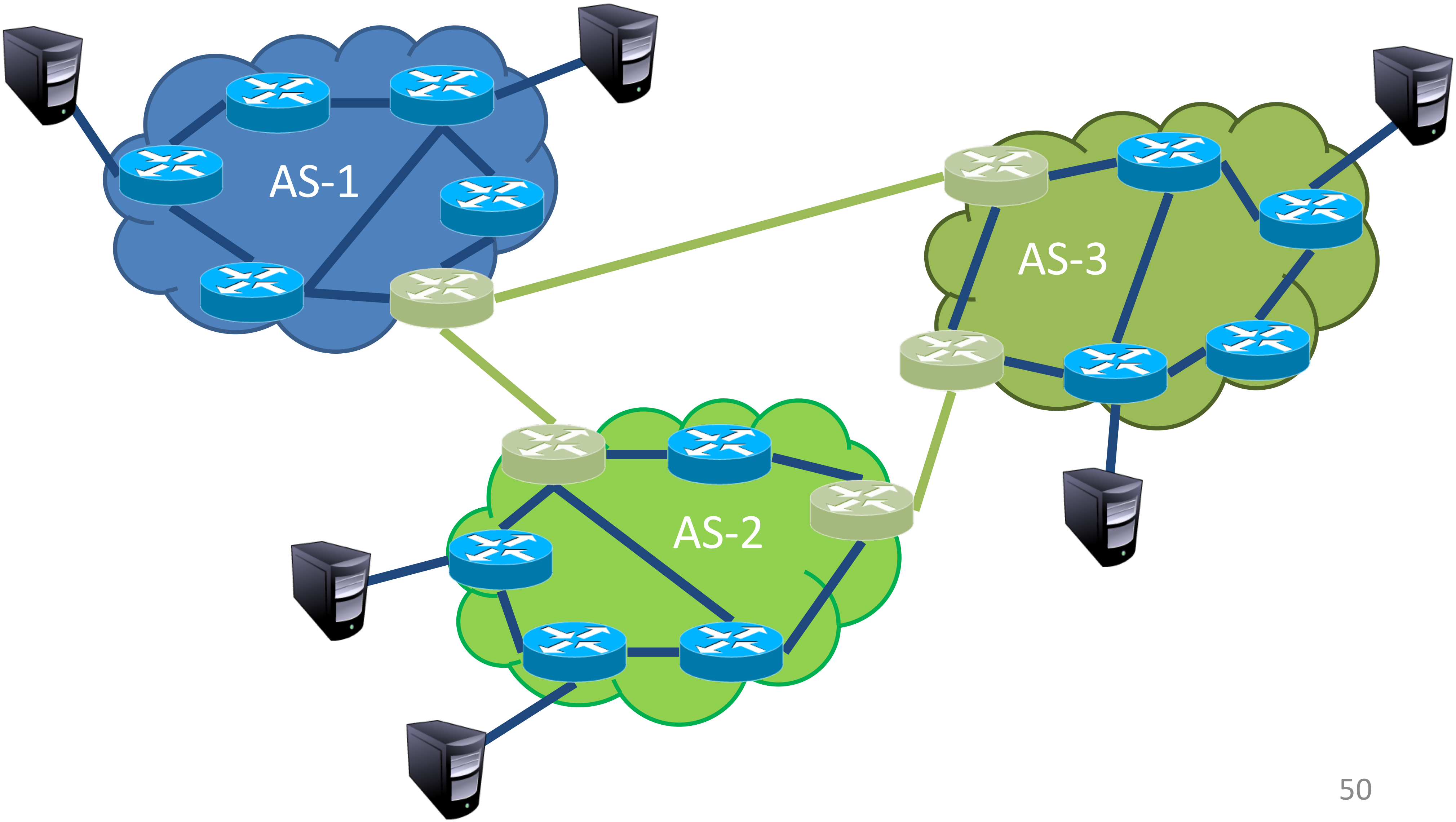
Goal:

Get traffic from one AS to another.

Inter-Domain Routing

- Global connectivity is at stake!
  - Thus, all ASs must use the same protocol
  - Contrast with intra-domain routing
- What are the requirements?
  - Scalability
  - Flexibility in choosing routes
    - Cost
    - Routing around failures
- Question: link state or distance vector?
  - Trick question: BGP is a path vector protocol

Hierarchical routing: Autonomous Systems



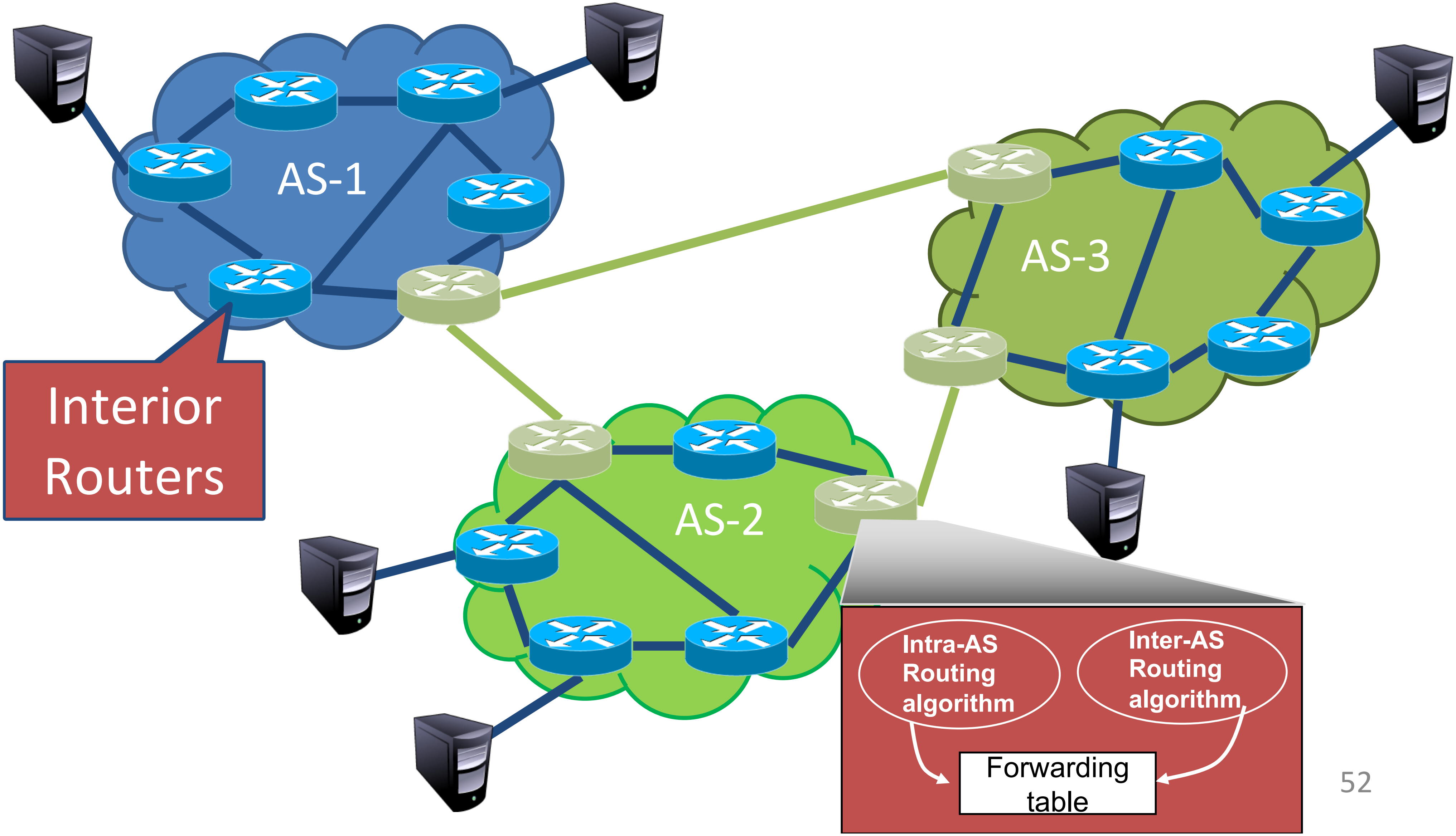
Hierarchical routing

- We aggregate routers into regions, "autonomous systems" (AS)
- Routers in same AS run same routing protocol
  - "intra-AS" or "interior" routing protocol
  - routers in different AS can run different intra-AS routing protocol

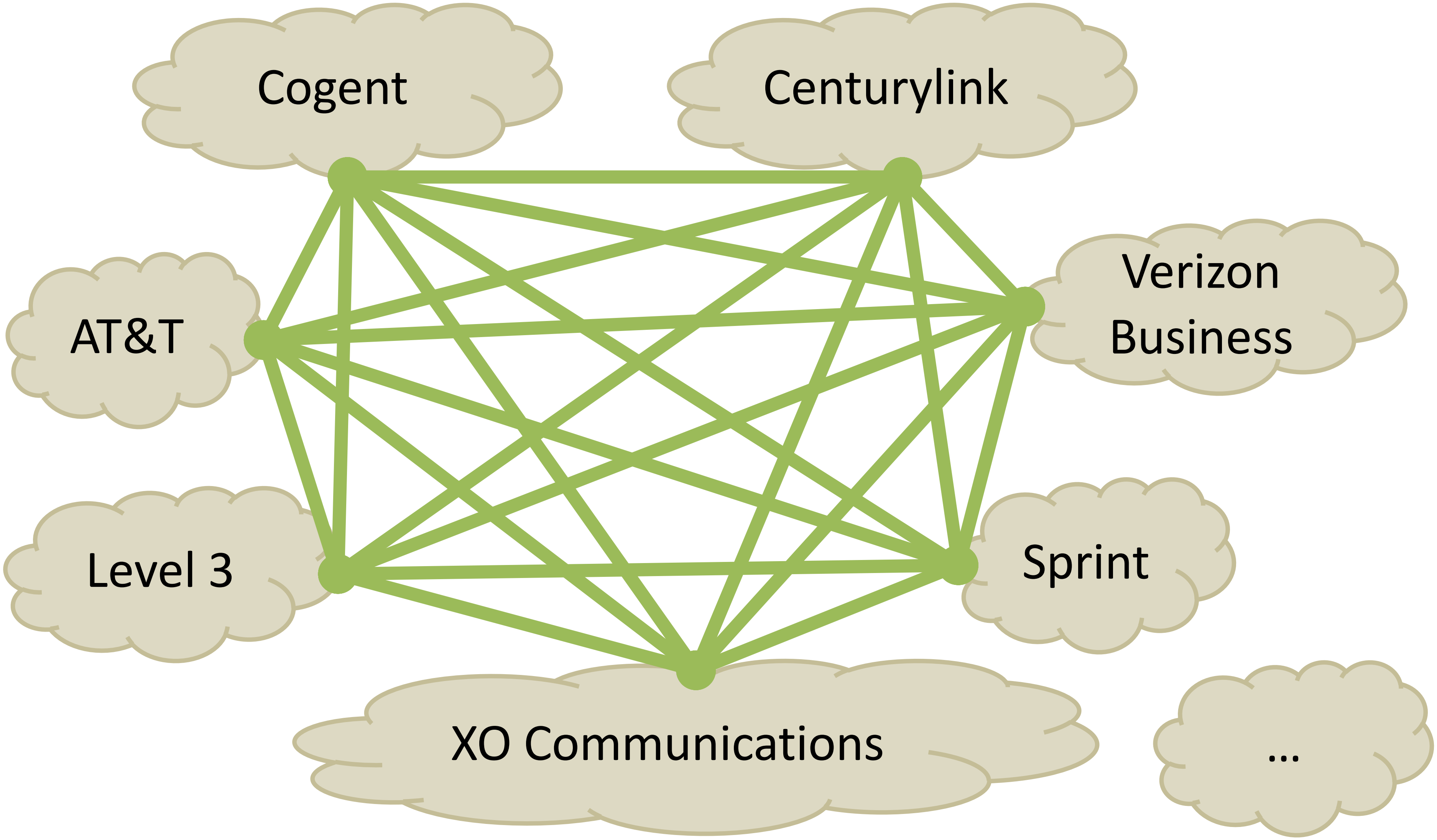
Gateway (or border) router:

- at "edge" of its own AS
- has link to router in another AS

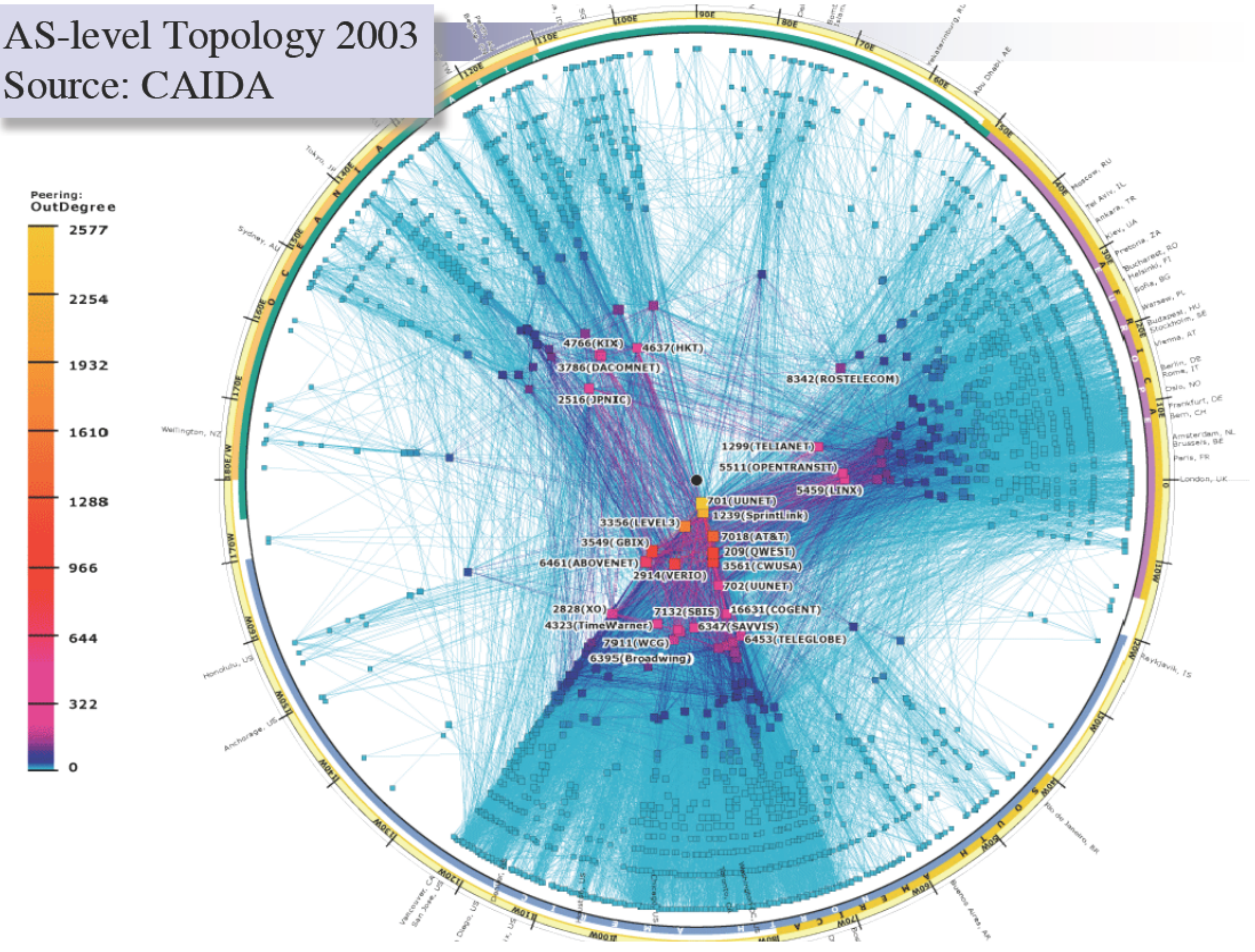
Hierarchical routing: Interconnected ASes



Tier-1 ISP Peering



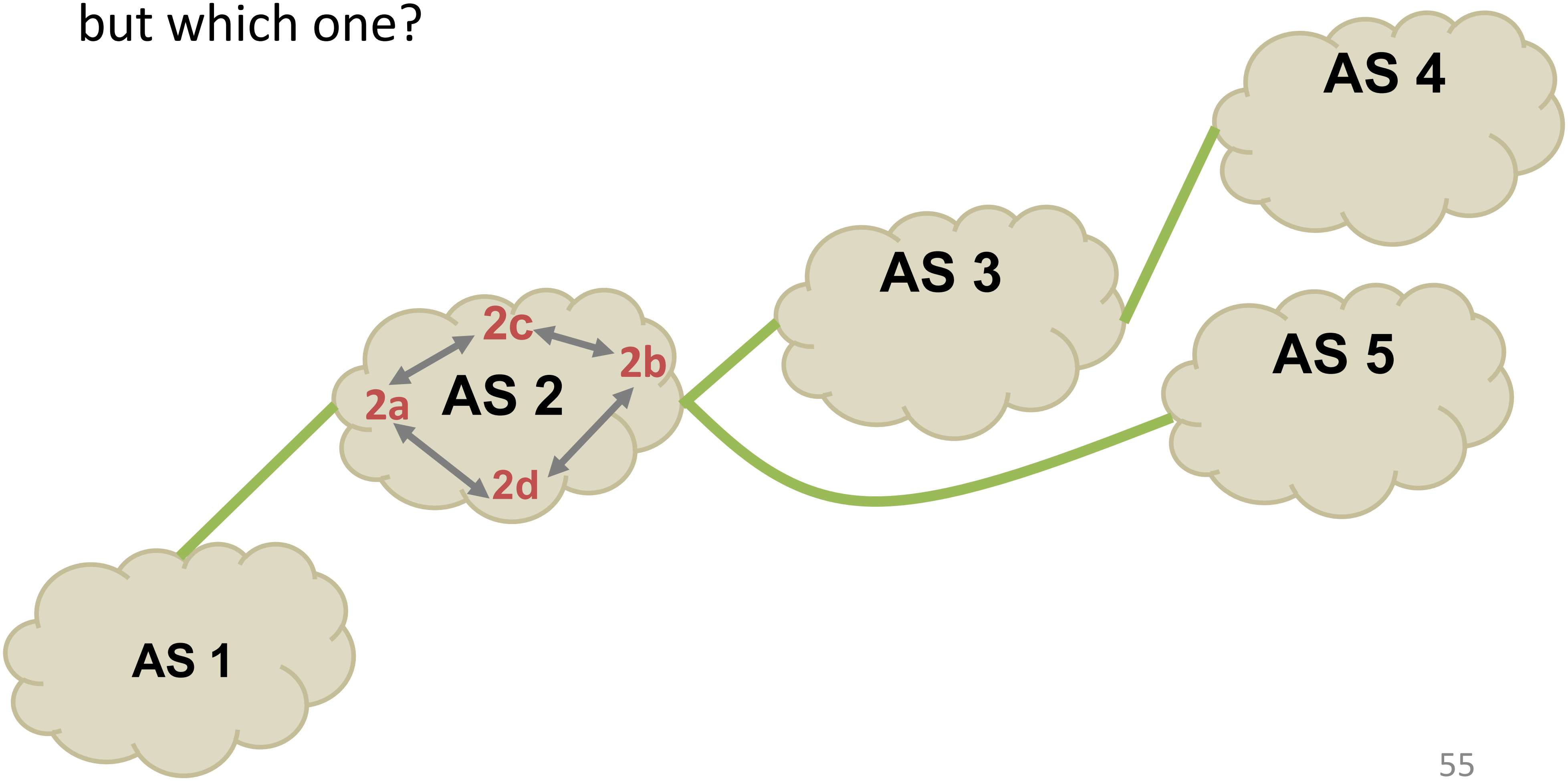
AS-level Topology 2003
Source: CAIDA



Inter-domain (Inter-ISP) Routing

Suppose router in AS2 receives a datagram destined outside of AS2:

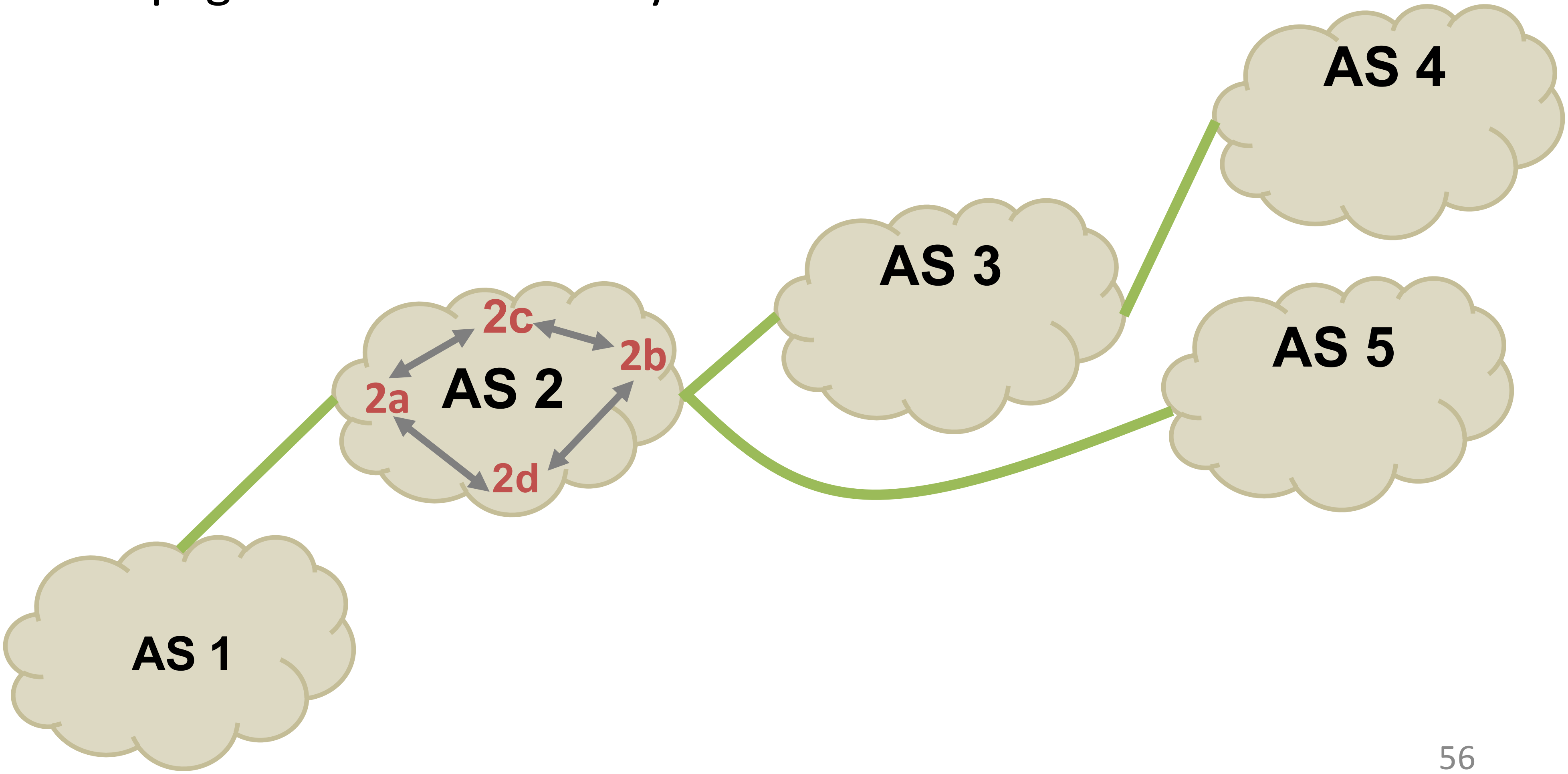
- Router should forward packet to gateway router, but which one?



Inter-domain (Inter-ISP) Routing

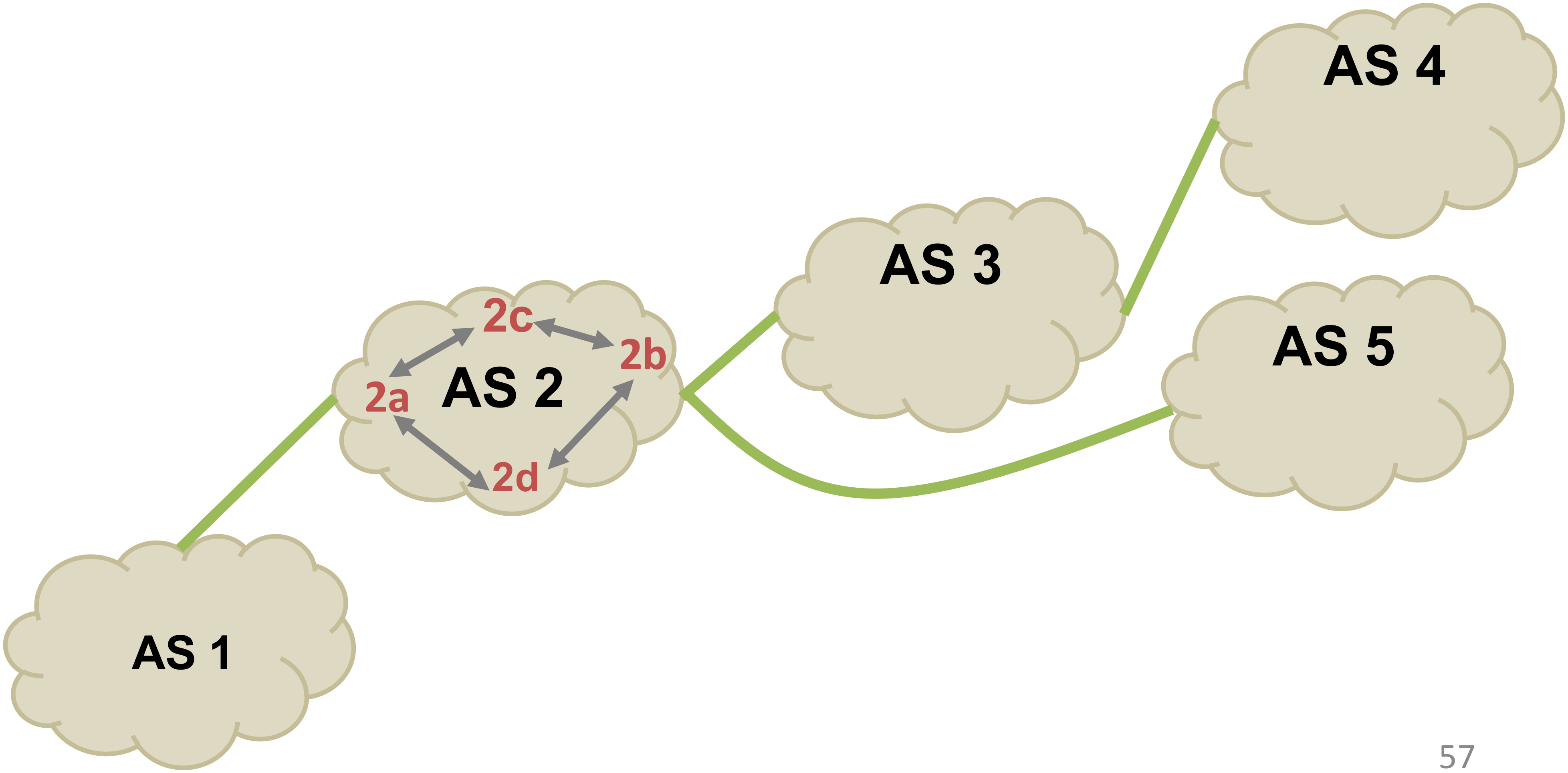
AS2 must:

1. Learn destinations reachable through AS2
2. Propagate this reachability info to all routers in AS2



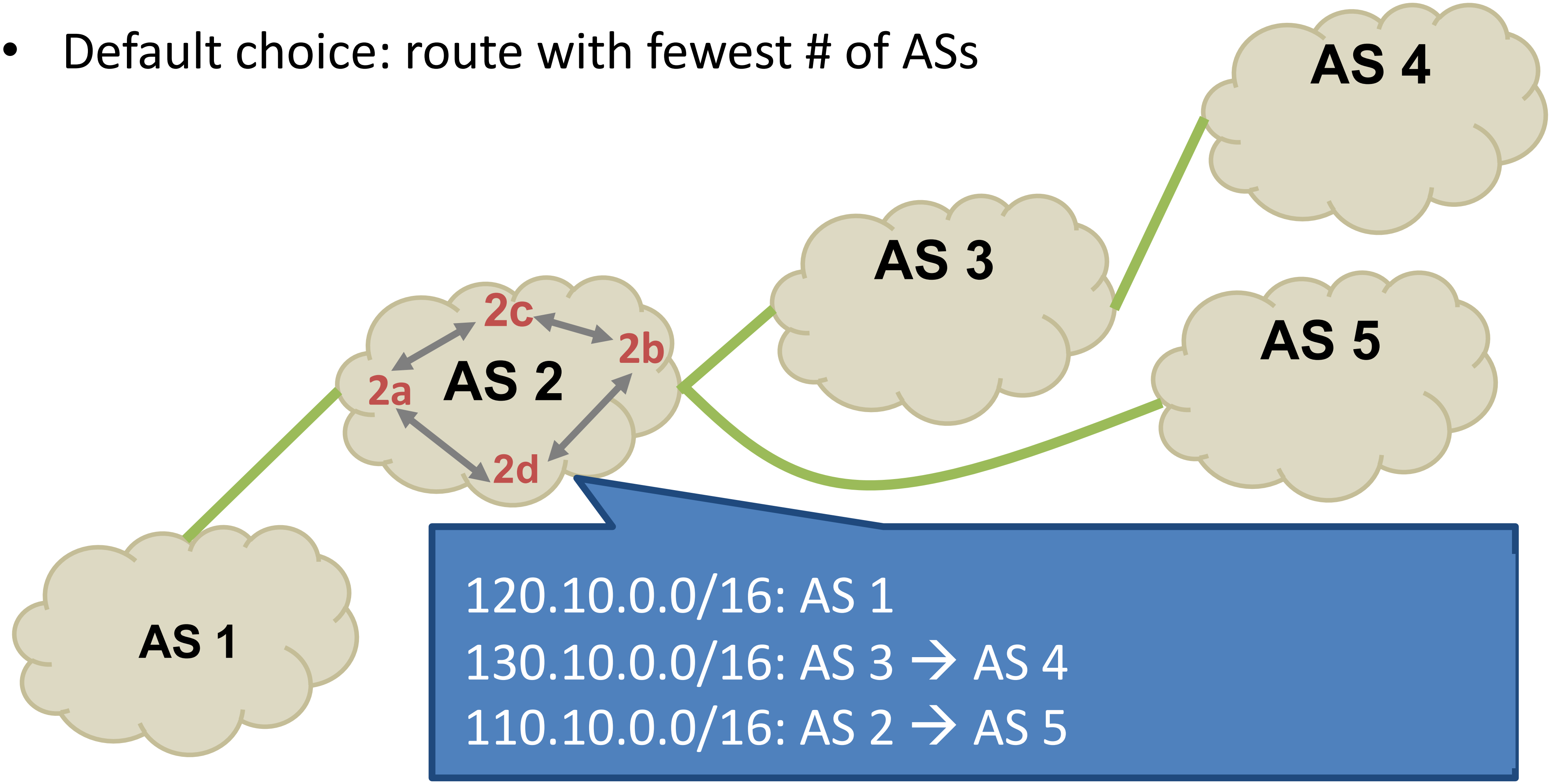
Path Vector Protocol

- Key idea: advertise the entire path
  - Distance vector: send *distance metric* per dest d
  - Path vector: send the *entire path* for each dest d



Path Vector Protocol

- AS-path: sequence of ASs a route traverses
  - Like distance vector, plus additional information
- Used for loop detection and to apply policy
- Default choice: route with fewest # of ASs



120.10.0.0/16: AS 1
130.10.0.0/16: AS 3 → AS 4
110.10.0.0/16: AS 2 → AS 5

If an external destination is reachable from multiple gateways, a router inside the AS should forward packets for that destination to…

A. The closest gateway that can reach the destination.

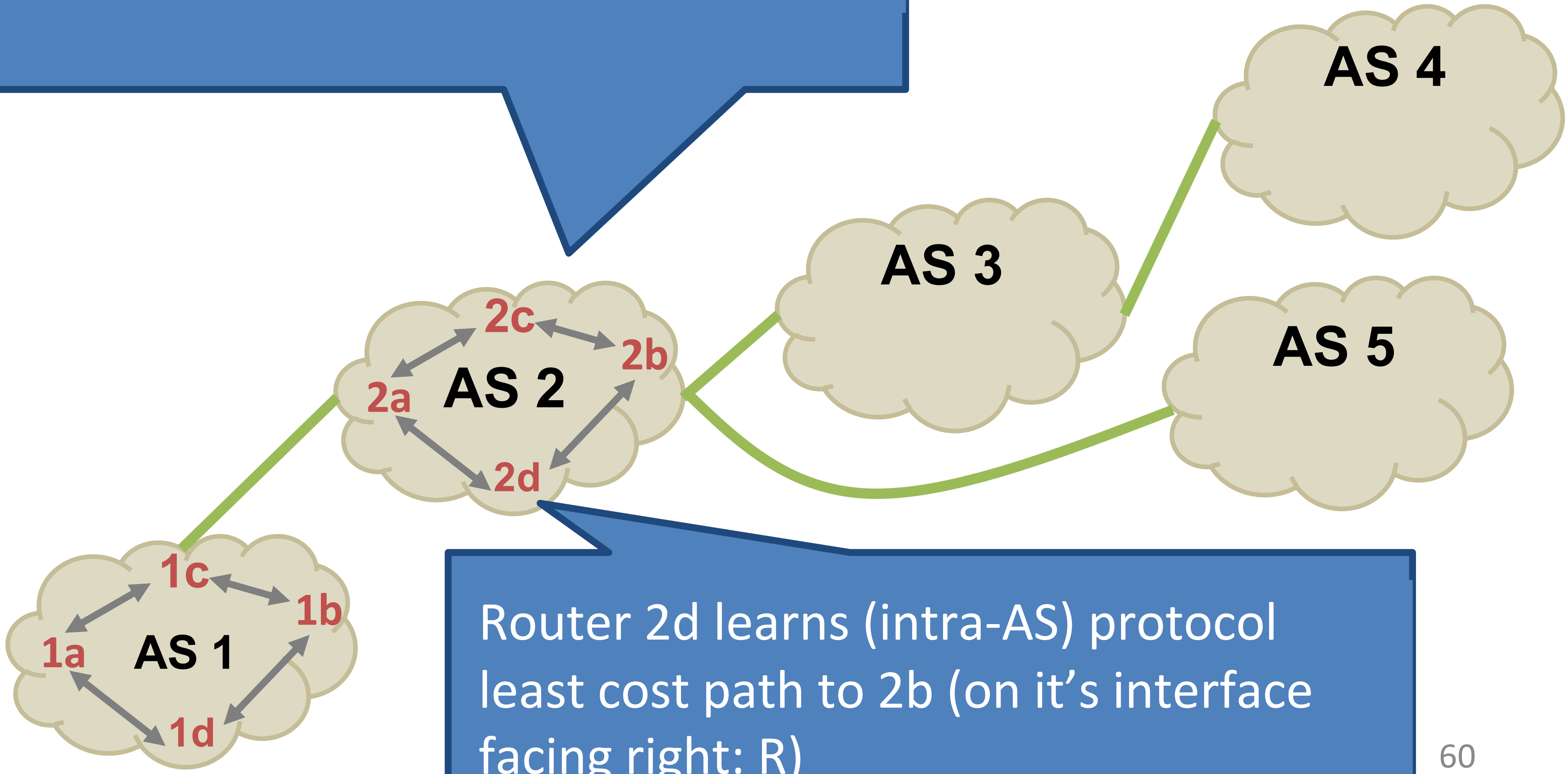
B. The gateway that has the least-cost external path to the destination.

C. The gateway that has the least-cost path for both the internal and external path.

D. Somewhere else.

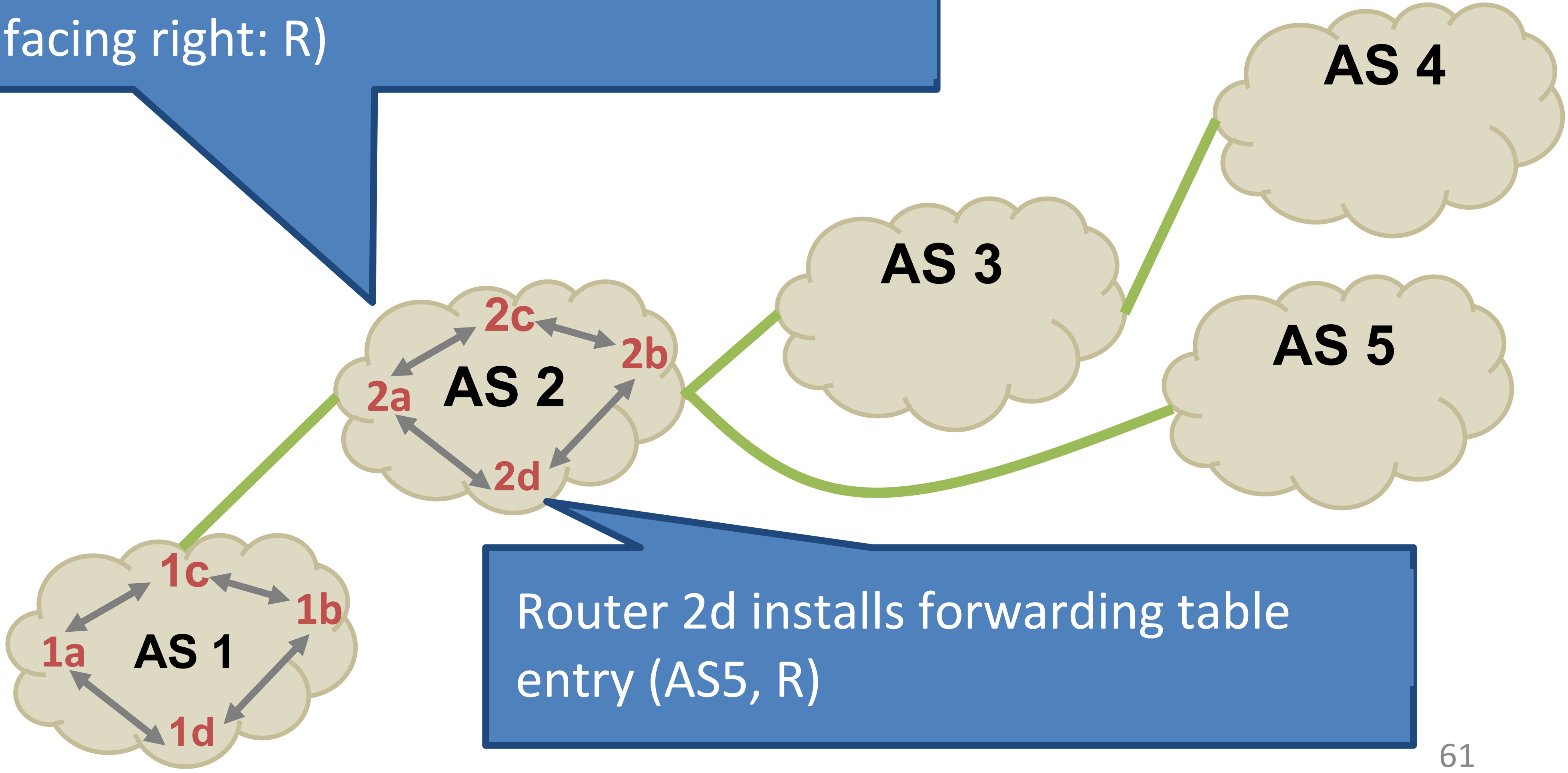
Building the forwarding table in router 2d, for path to AS 5

AS2 learns (Inter-AS) protocol that AS5 is reachable through AS3 (via gateway 2b)



Router 2d learns (intra-AS) protocol least cost path to 2b (on it's interface facing right: R)

Building the forwarding table in router 2d, for path to AS 5



Router 2d learns (intra-AS) protocol least cost path to 2b (on it's interface facing right: R)

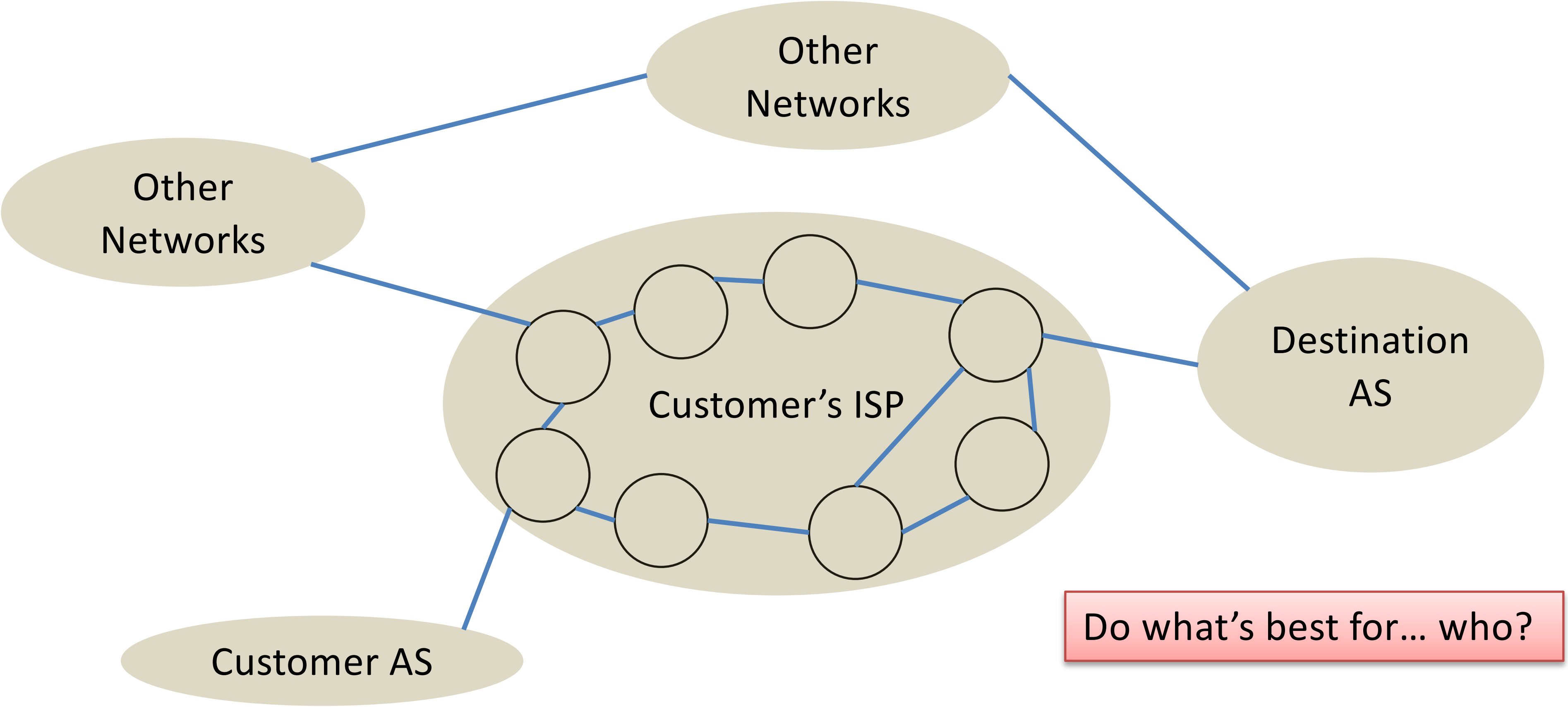
Router 2d installs forwarding table entry (AS5, R)

Why do we need different Intra and Interdomain AS routing ?

A. Scalability
B. Performance
C. A and B
D. More than just A and B

Routing Policy

- How should the ISP route the customer's traffic to the destination?



Do what's best for… who?

Which routes a BGP router advertises will depend on…

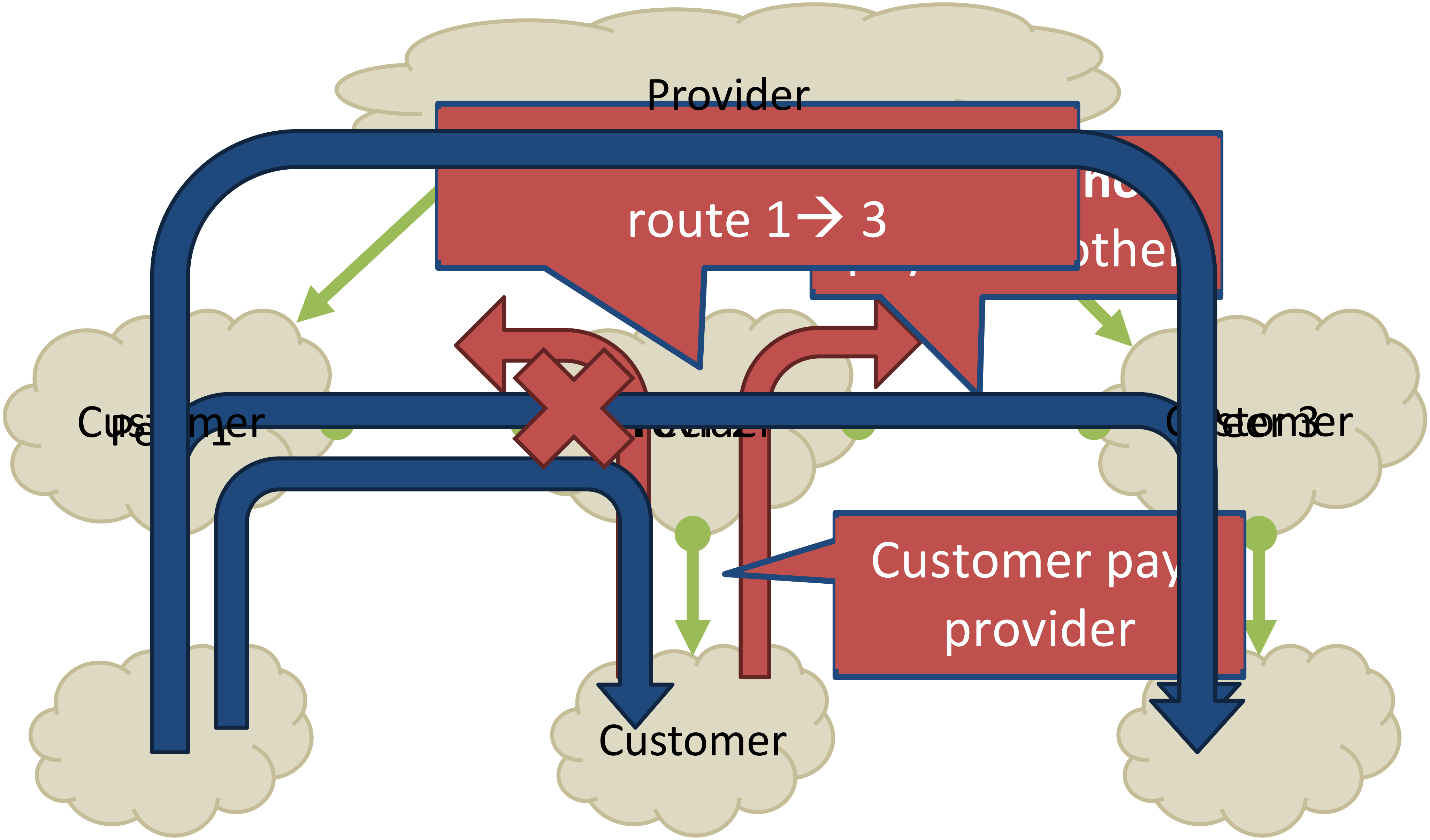
A. which ISPs have contractual agreements.

B. the shortest path to a subnet/prefix.

C. which subnets are customers of an ISP.

D. More than one of the above. (which?)

BGP Relationships



Provider

route 1→ 3

Customer pays provider

Peering/Interconnection Wars

- Peer

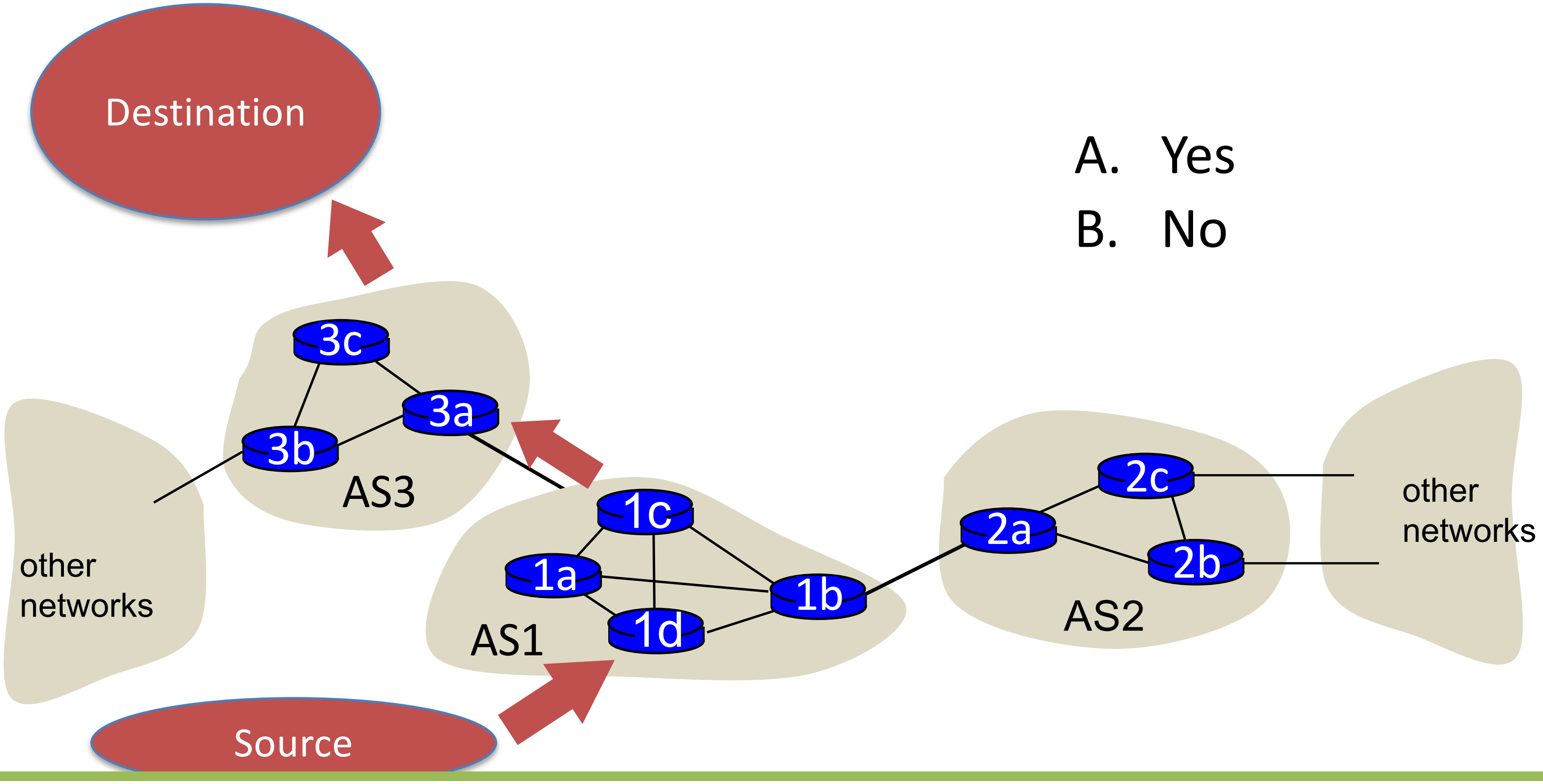
- Don't Peer

- Reduce upstream costs
- Improve end-to-end performance
- May be the only way to connect to parts of the Internet

- You would rather have customers
- Peers are often competitors
- Peering agreements require periodic renegotiation

Peering struggles in the ISP world are extremely contentious, agreements are usually confidential

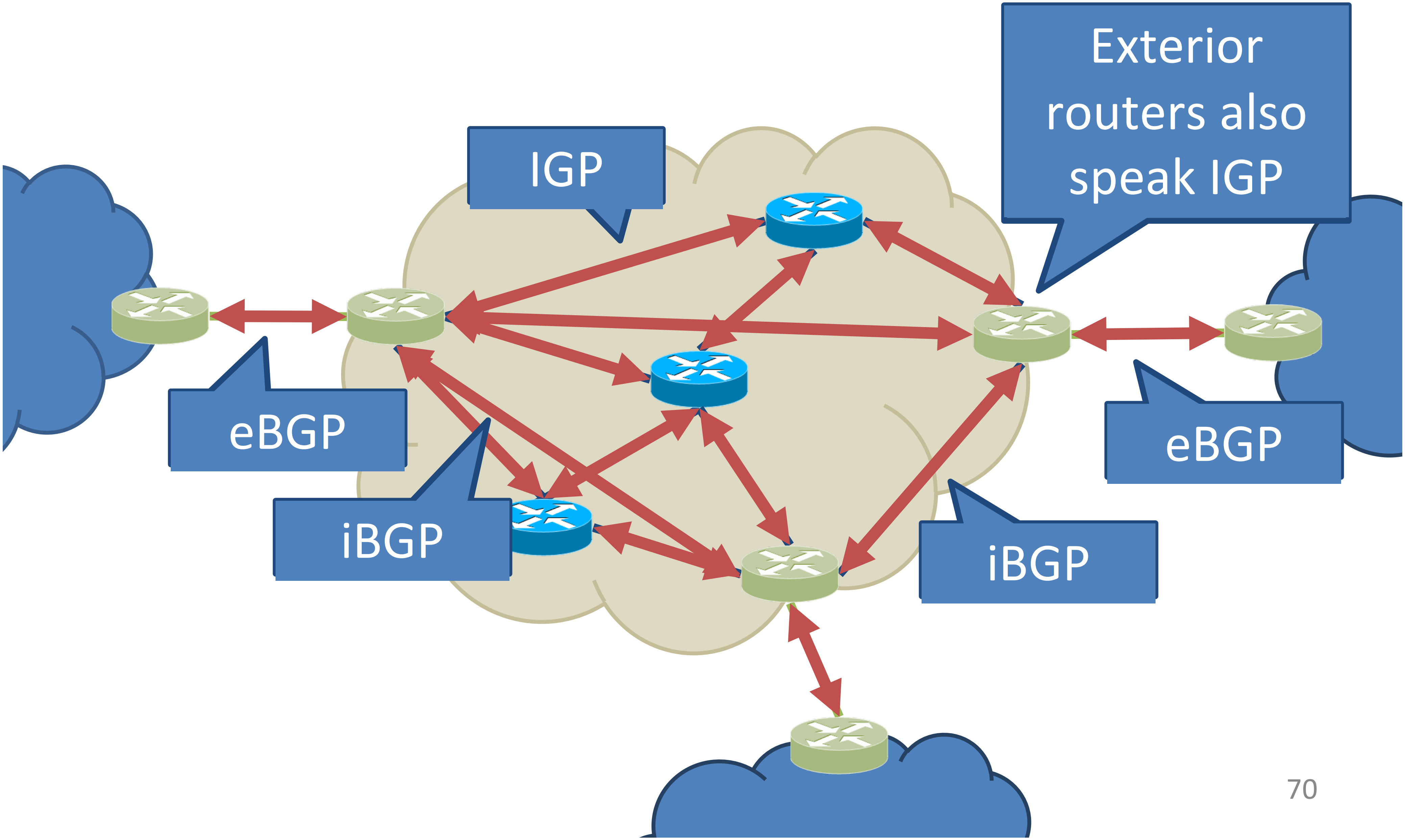
Border routers: exchange AS reachability, Internal routers: exchange intra-AS reachability., Is this sufficient to route from source to destination?



A. Yes

B. No

Internet inter-AS routing: BGP



IGP

Exterior routers also speak IGP

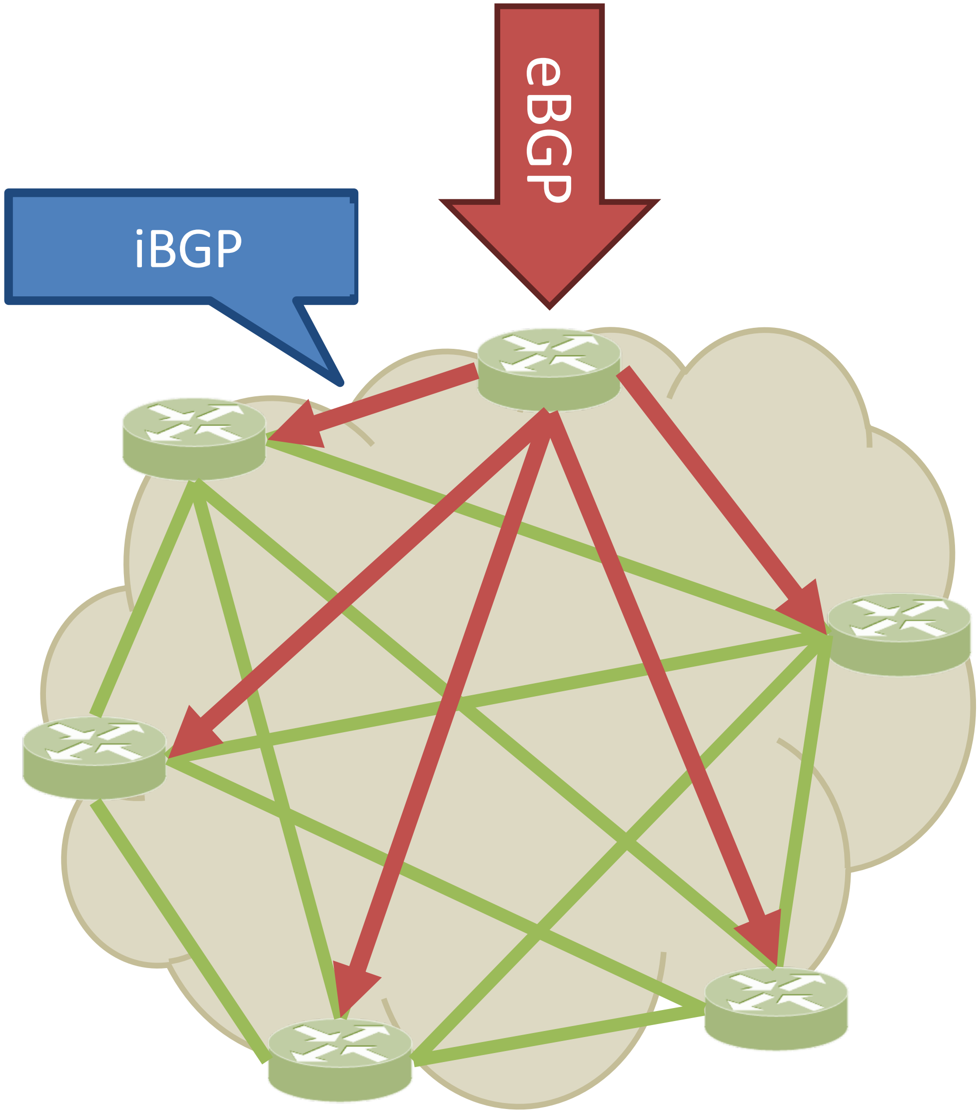
eBGP

eBGP

iBGP

iBGP

Internet inter-AS routing: BGP

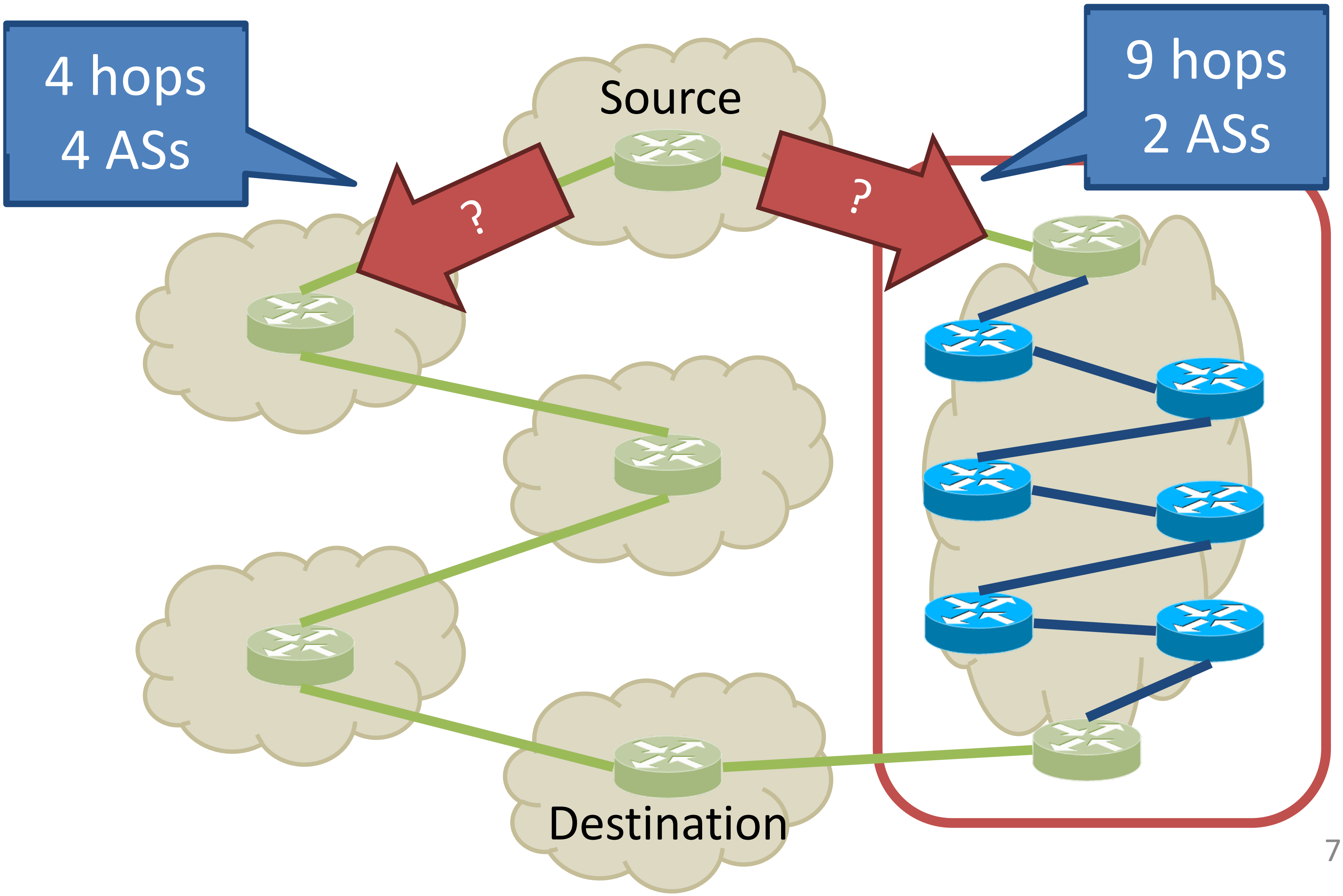


- Question: why do we need iBGP?
  - OSPF does not include BGP policy info
  - Prevents routing loops within the AS
- iBGP updates do not trigger announcements

Internet inter-AS routing: BGP

- BGP (Border Gateway Protocol): *The* de facto inter-domain routing protocol
- BGP provides each AS a means to:
    - external BGP: obtain subnet reachability information from neighboring ASs.
    - internal BGP: propagate reachability information to all AS-internal routers.
    - determine "good" routes to other networks based on reachability information and policy.
- Allows a subnet to advertise its prefix to the rest of the Internet

Shortest AS Path != Shortest Path



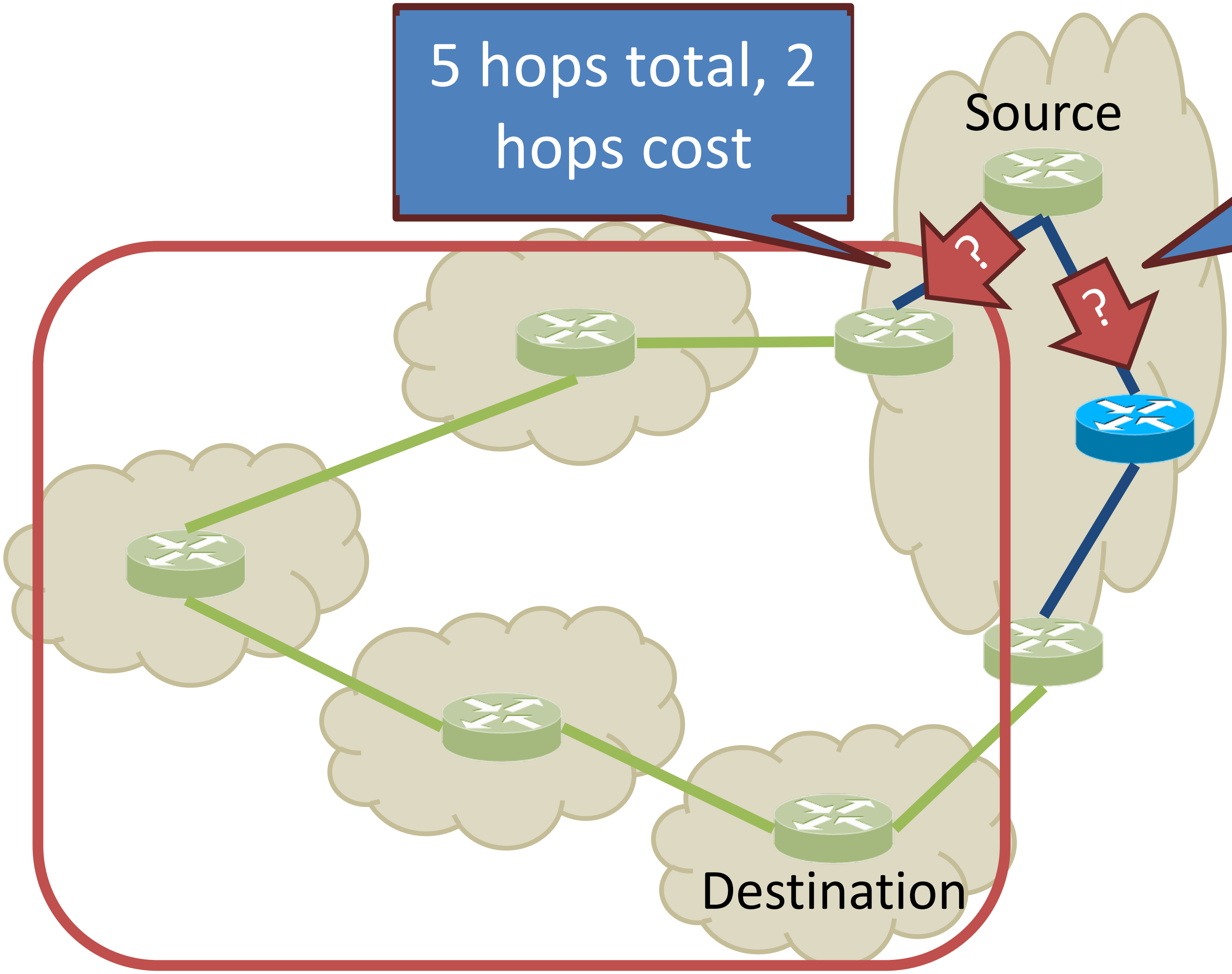
4 hops

4 ASs

9 hops

2 ASs

Hot Potato Routing: get rid of packets ASAP!

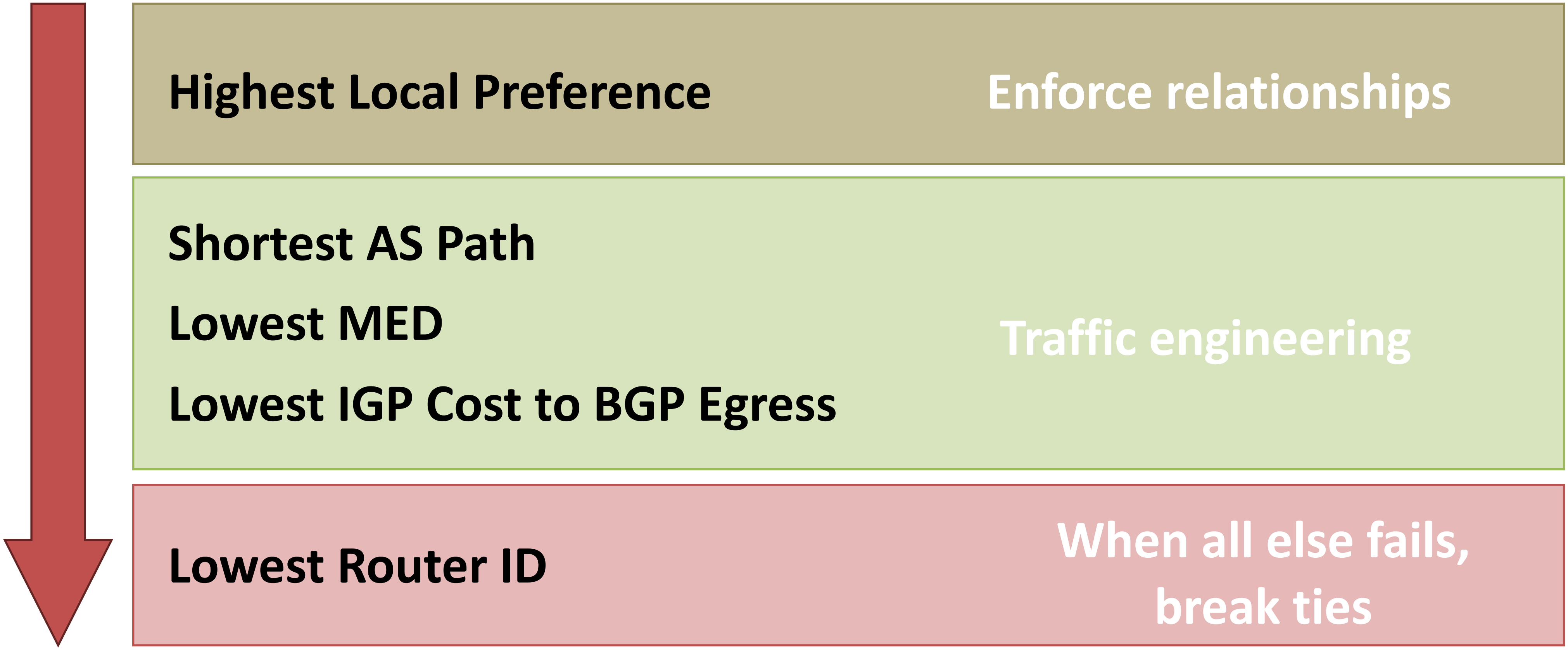


5 hops total, 2 hops cost

3 hops total, 3 hops cost

Best path: get it as close to the destination as we can.

Route Selection Summary



Highest Local Preference

Enforce relationships

Shortest AS Path

Lowest MED

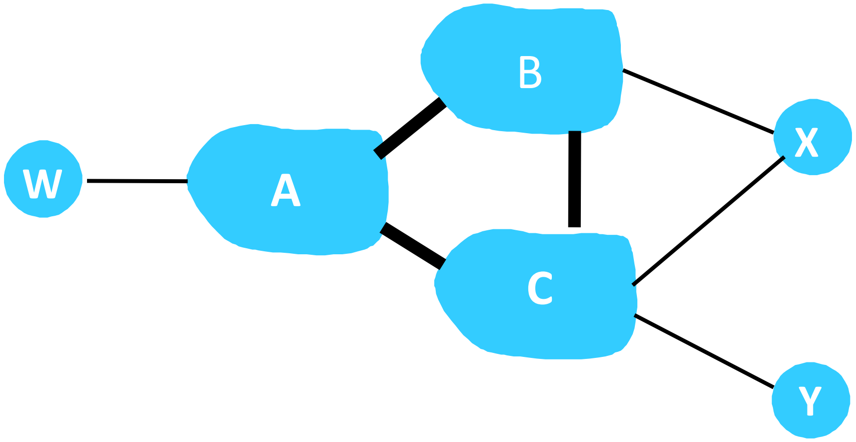
Lowest IGP Cost to BGP Egress

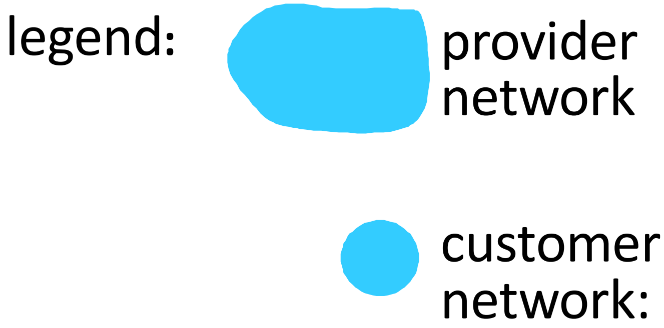
Traffic engineering

Lowest Router ID

When all else fails,
break ties

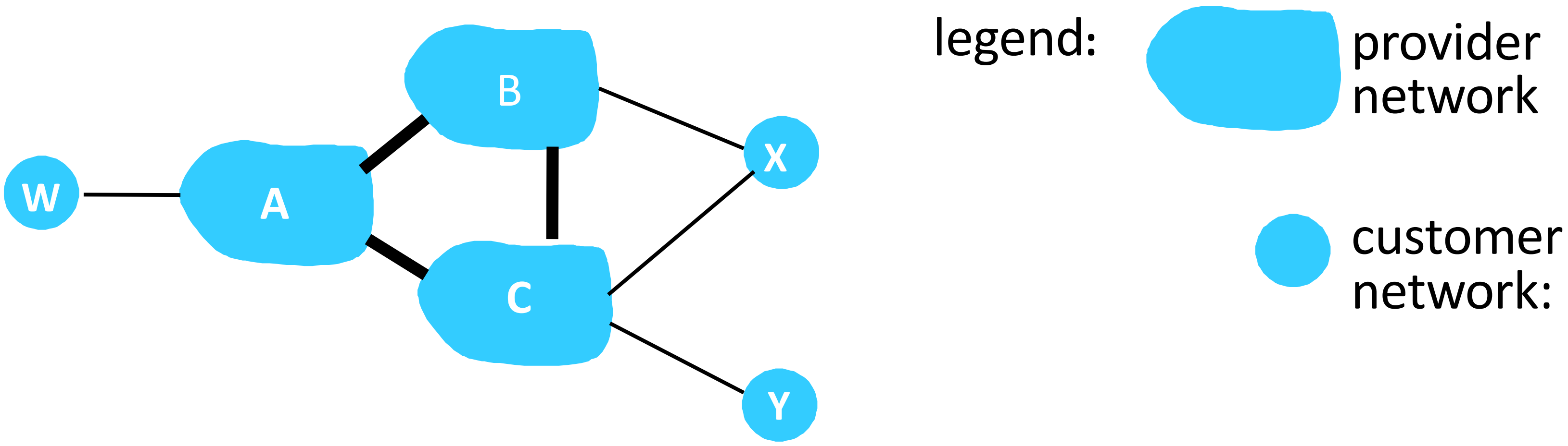
BGP routing policy





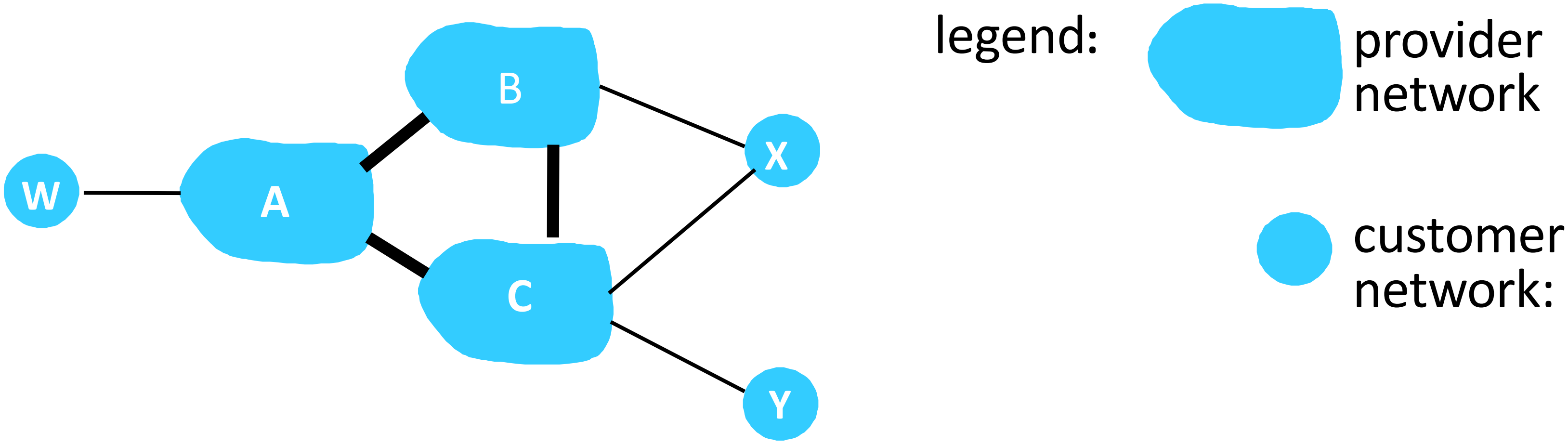
- A,B,C are *provider networks*
- X,W,Y are customer (of provider networks)
- X is *dual-homed:* attached to two networks
  - X does not want to route from B via X to C
  - .. so X will not advertise to B a route to C

BGP routing policy (2)



- A advertises path AW to B
- B advertises path BAW to X
- Should B advertise path BAW to C?
  - 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!

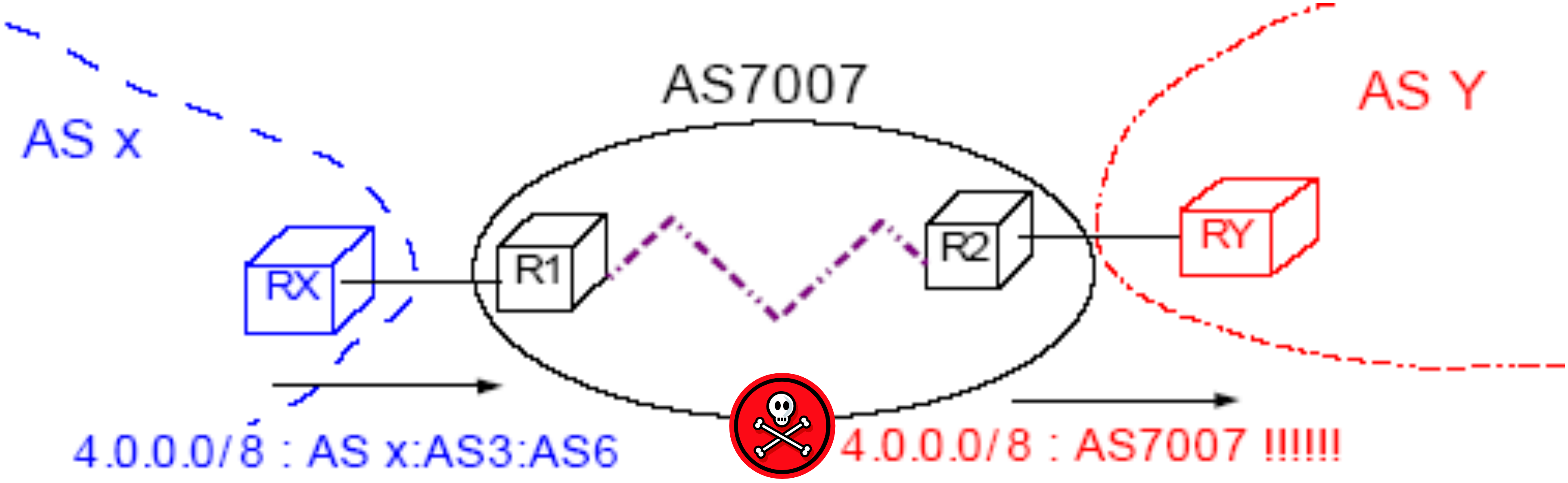
BGP routing policy gone wrong



- x advertises a path to E (that it is not connected to).
- all traffic starts to flow into x from B and C!

Faulty redistribution can be dangerous!

- AS7007 incident (April, 1997):



Hijack large part of the Internet

Summary

- As we've seen before (DNS), a hierarchy can help manage state storage constraints.
  - intra-AS routing: lots of info about local routes
  - inter-AS routing: less info about far away routes

- BGP: the inter-AS routing protocol for the Internet
  - Decisions often contractual

- BGP advertises AS prefixes, including:
  - entire path of ASes along the way
  - which border router heard the advertisement (Next Hop)

Inter-Domain Routing Challenges

- BGP4 is the only inter-domain routing protocol currently in use world-wide
- Issues?
  - Lack of security
  - Ease of misconfiguration
  - Poorly understood interaction between local policies
  - Poor convergence
  - Lack of appropriate information hiding
  - Non-determinism
  - Poor overload behavior

Lots of research into how to fix this

- Security
  - BGPSEC, RPKI
- Misconfigurations, inflexible policy
  - SDN
- Policy Interactions
  - PoiRoot (root cause analysis)
- Convergence
  - Consensus Routing
- Inconsistent behavior
  - LIFEGUARD, among others

Why are these still issues?

- Backward compatibility
- Buy-in / incentives for operators
- Stubbornness

Very similar issues to IPv6 deployment

Why Network Reliability Remains Hard

- Visibility
  - IP provides no built-in monitoring
  - Economic disincentives to share information publicly

- Control
  - Routing protocols optimize for policy, not reliability
  - Outage affecting your traffic may be caused by distant network

- Detecting, isolating and repairing network problems for Internet paths remains largely a slow, manual process