

Lecture 5: Network Configuration and Defense

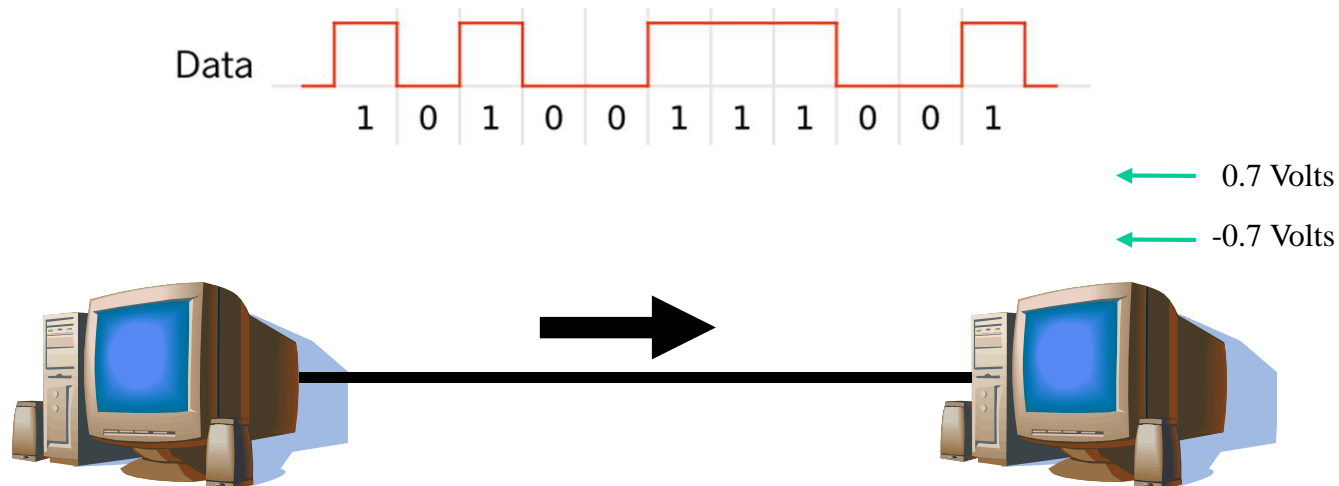
CS 598: Network Security

Matthew Caesar

February 26, 2013

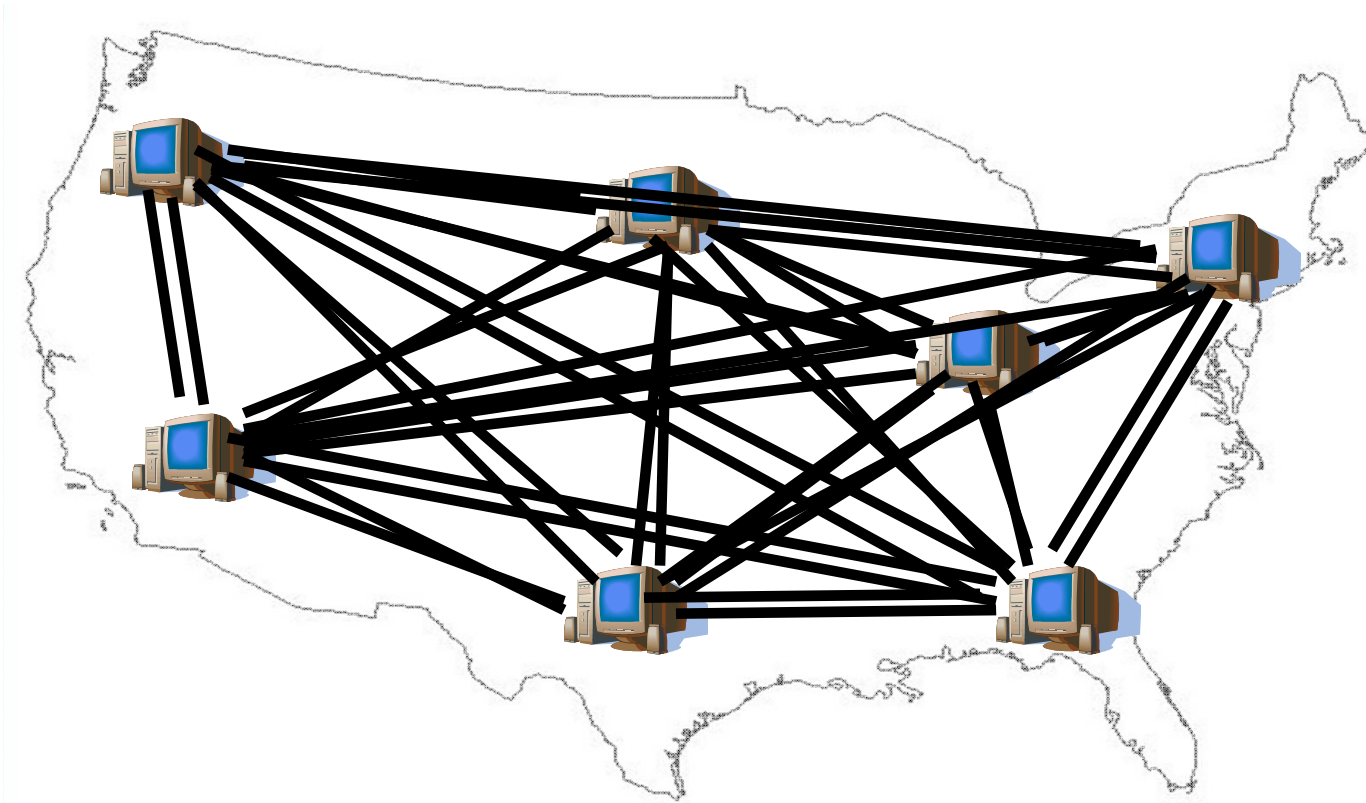
Part 1: How the Internet works

How can two hosts communicate?



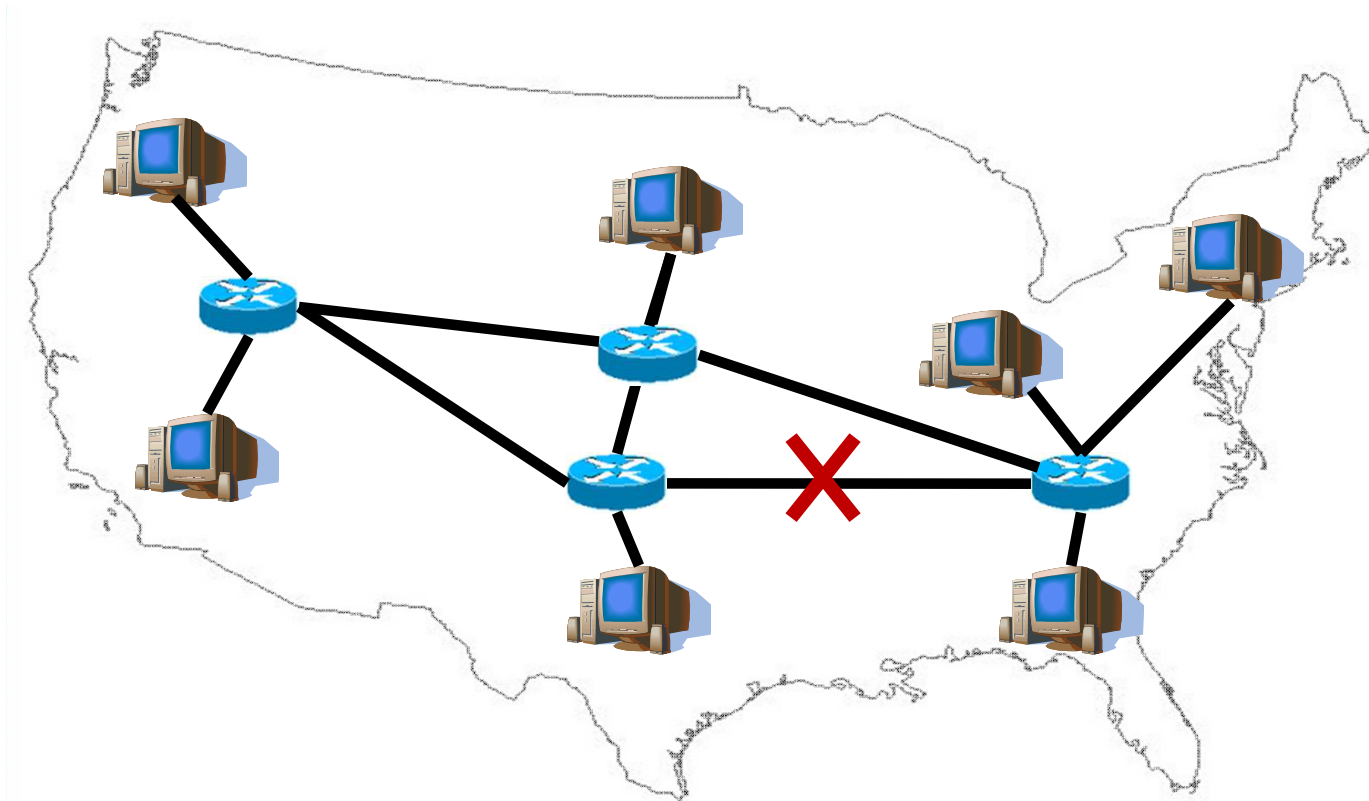
- Encode information on modulated “Carrier signal”
 - Phase, frequency, and amplitude modulation, and combinations thereof
 - Ethernet: self-clocking Manchester coding ensures one transition per clock
 - Technologies: copper, optical, wireless

How can many hosts communicate?



- Naïve approach: full mesh
- Problem:
 - Obviously doesn't scale to the 570,937,778 hosts in the Internet (estimated, Aug 2008)

How can many hosts communicate?



- Multiplex traffic with routers
- Goals: make network robust to failures, maintain spare capacity, reduce operational costs
 - More on “topology” later in this lecture

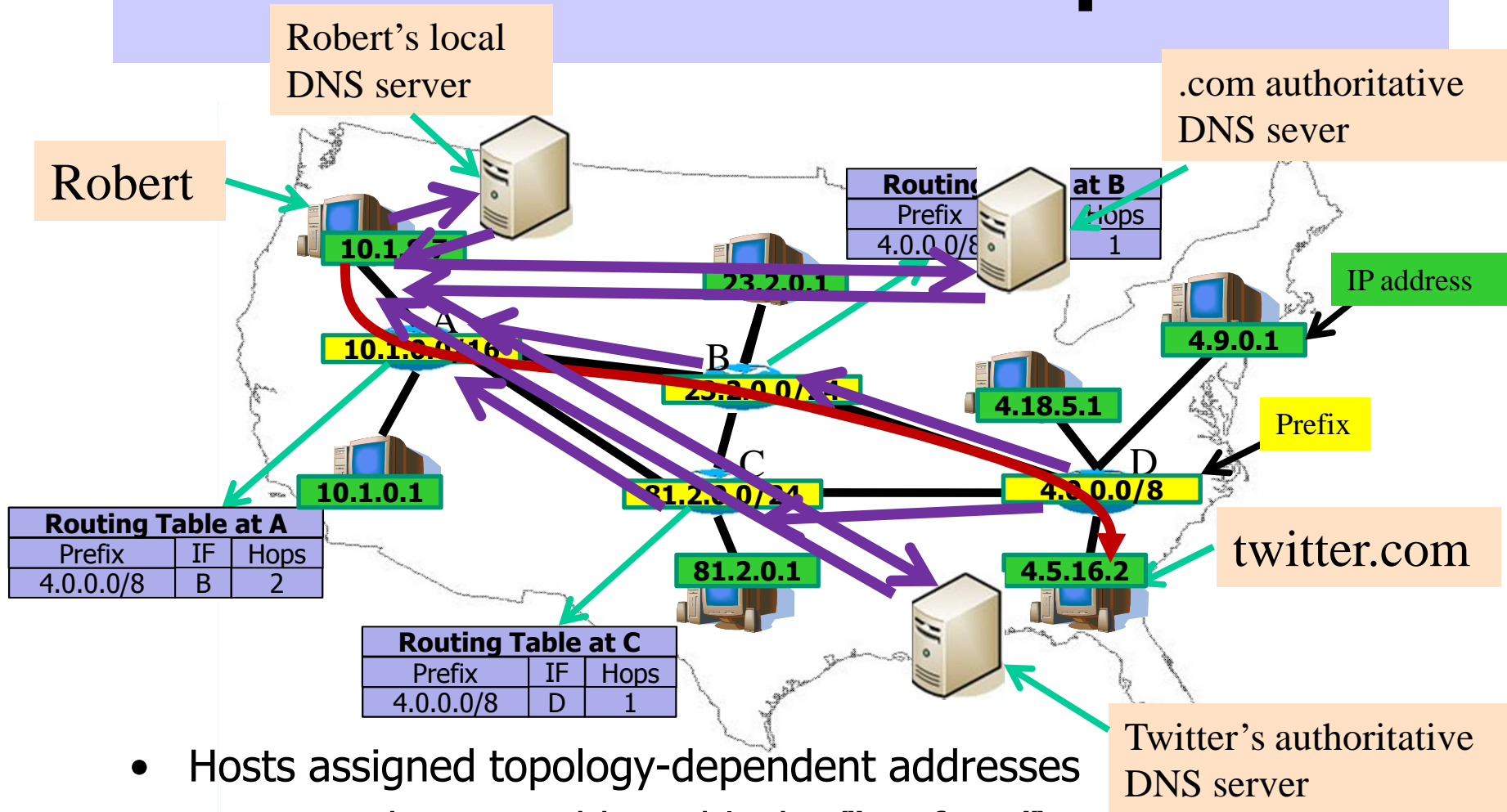
Complete Network Assets : XO Communications



LEGEND

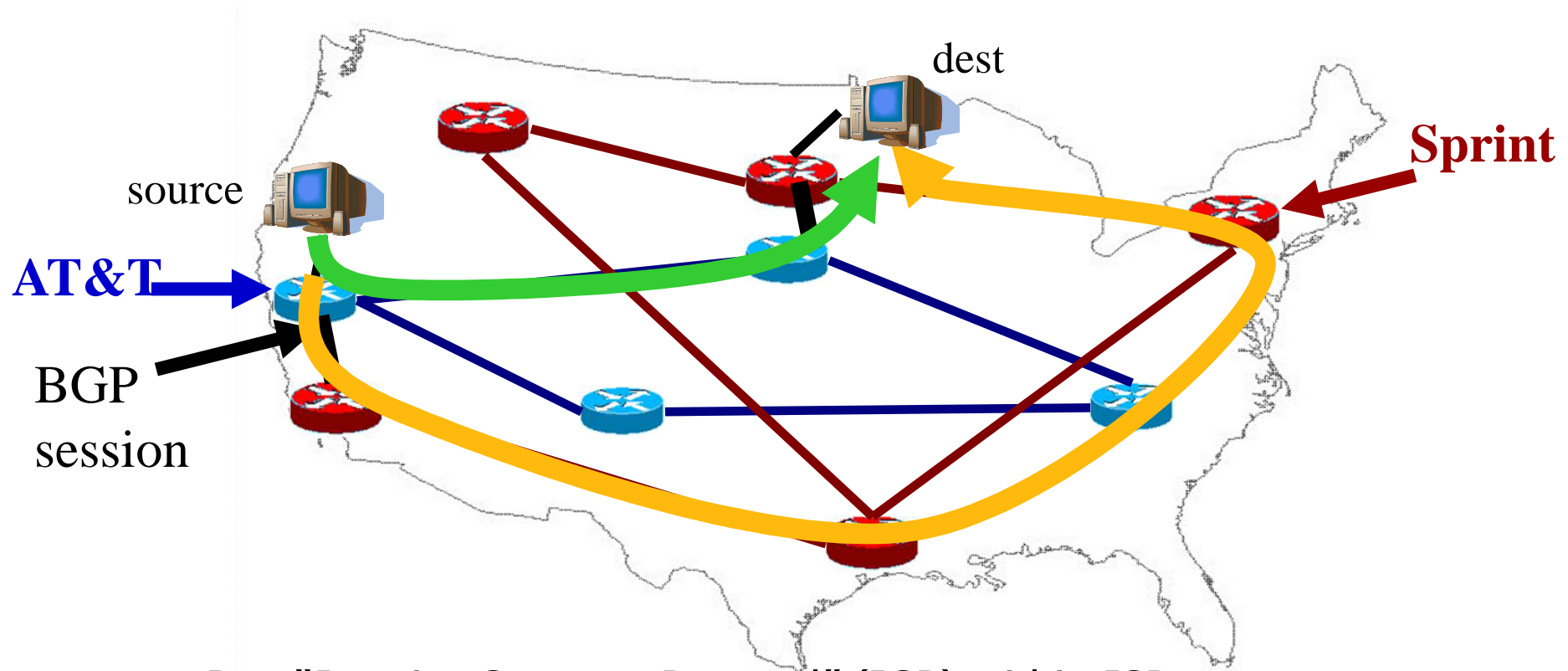
OC-12 Market Uplinks	Data Center IP OC-12c Uplink	Core IP Node	Class 5 Voice Switch	Local Voice Footprint
OC-3 Market Uplinks	OC-48 IP Backbone	Metro IP Node	Sonus Gateway	XO Market
Diversely Routed OC-48Transport	OC-48 IP Market Uplink	Private Peering IP Node	Longhaul Termination (All Bandwidths)	Network Management Center
OC-192 BLSR Rings	OC-192 Backbone Circuit	Public Peering IP Node	Longhaul Termination (OC-48 & Above Only)	Private Line Backbone
GigE	Peering Backbone Circuit	Data Center		

How can routers find paths?



- Hosts assigned topology-dependent addresses
- Routers advertize address blocks ("prefixes")
- Routers compute "shortest" paths to prefixes
- Map IP addresses to names with DNS

Intra- vs. Inter-domain routing

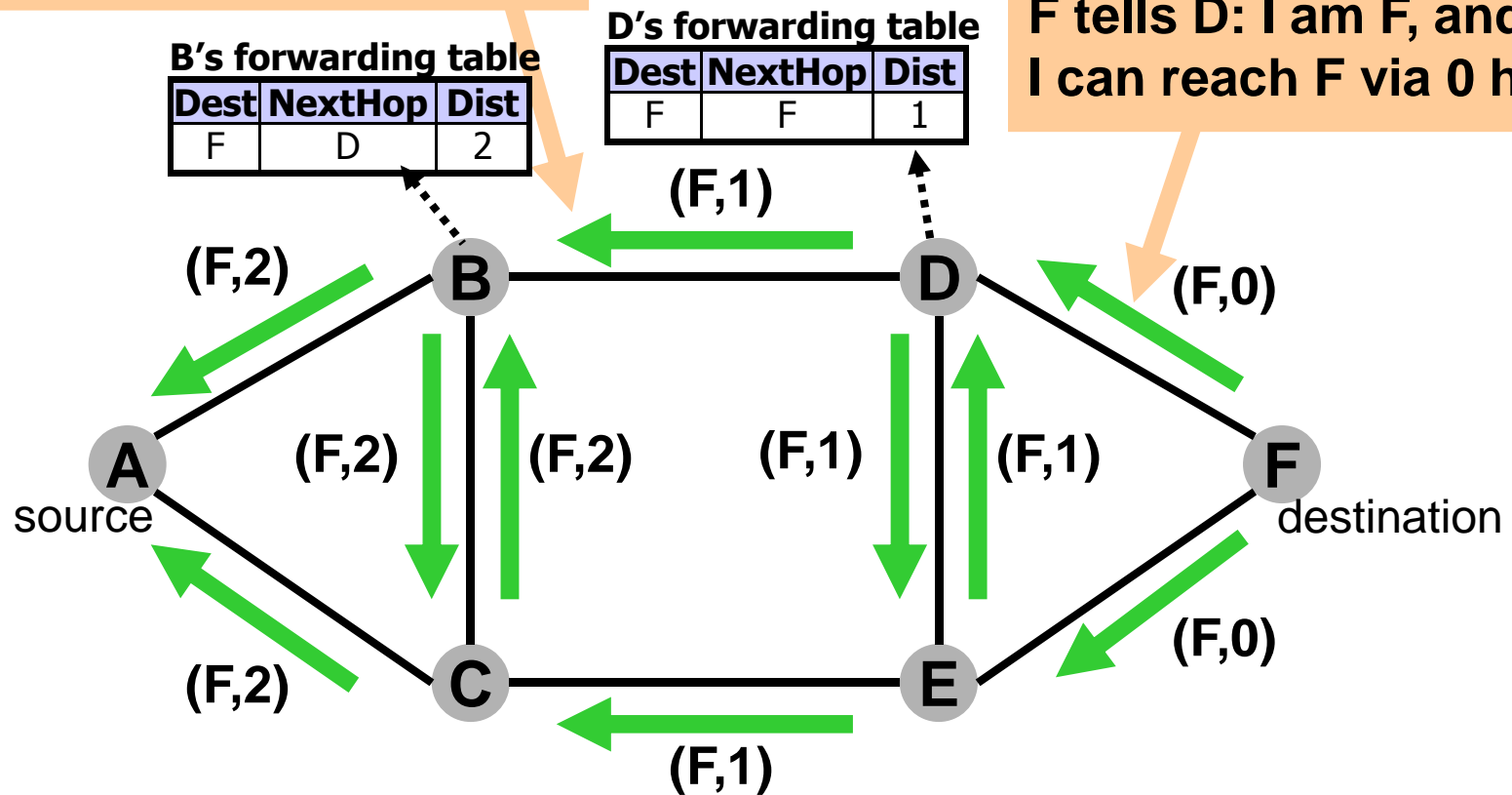


- Run "Interior Gateway Protocol" (IGP) within ISPs
 - OSPF, IS-IS, RIP
- Use "Border Gateway Protocol" (BGP) to connect ISPs
 - To reduce costs, peer at exchange points (AMS-IX, MAE-EAST)

Distance vector: update propagation

D tells B: I am D, and I can reach F via 1 hop

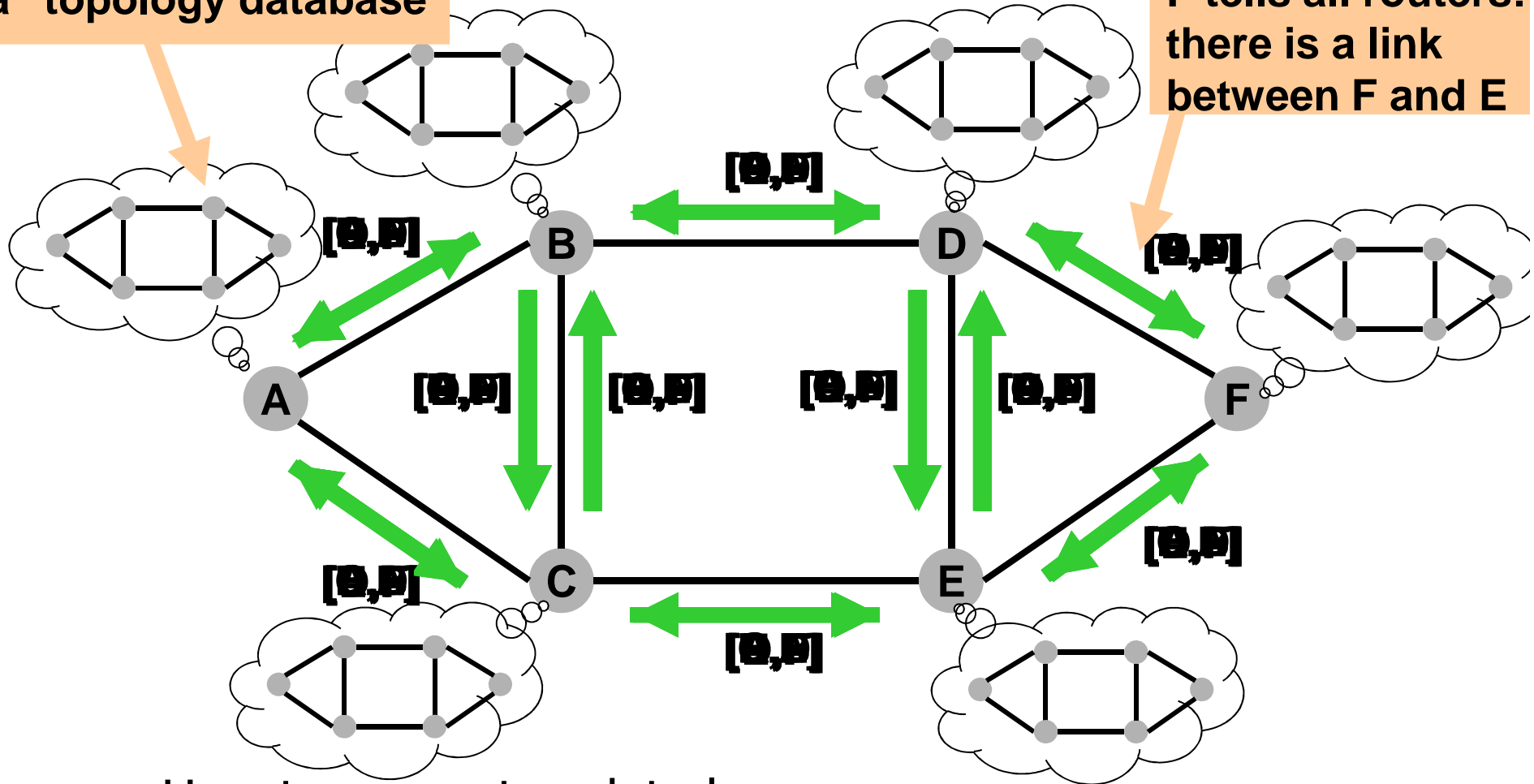
F tells D: I am F, and I can reach F via 0 hops



Link state: update propagation

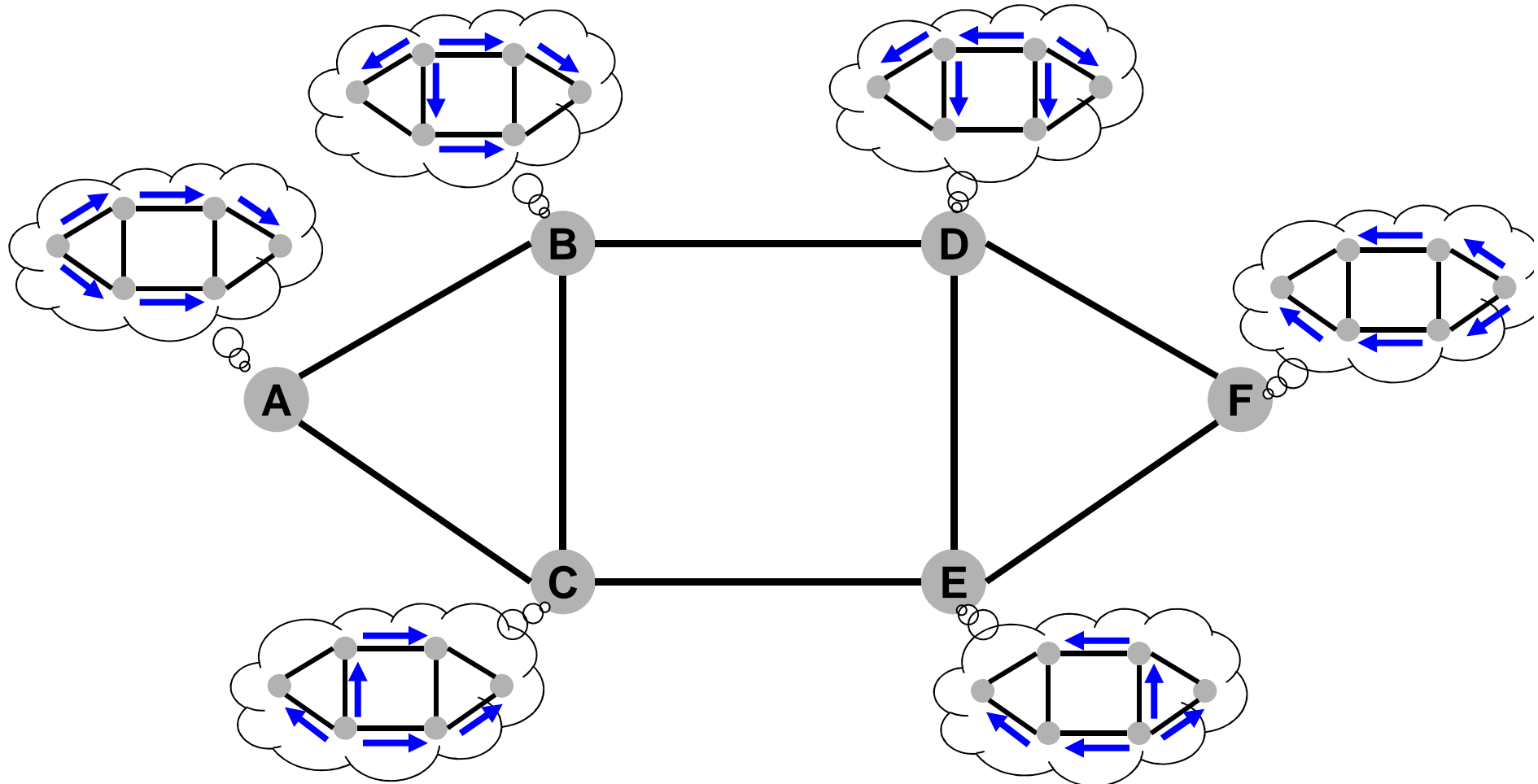
Each node maintains a “topology database”

F tells all routers: there is a link between F and E



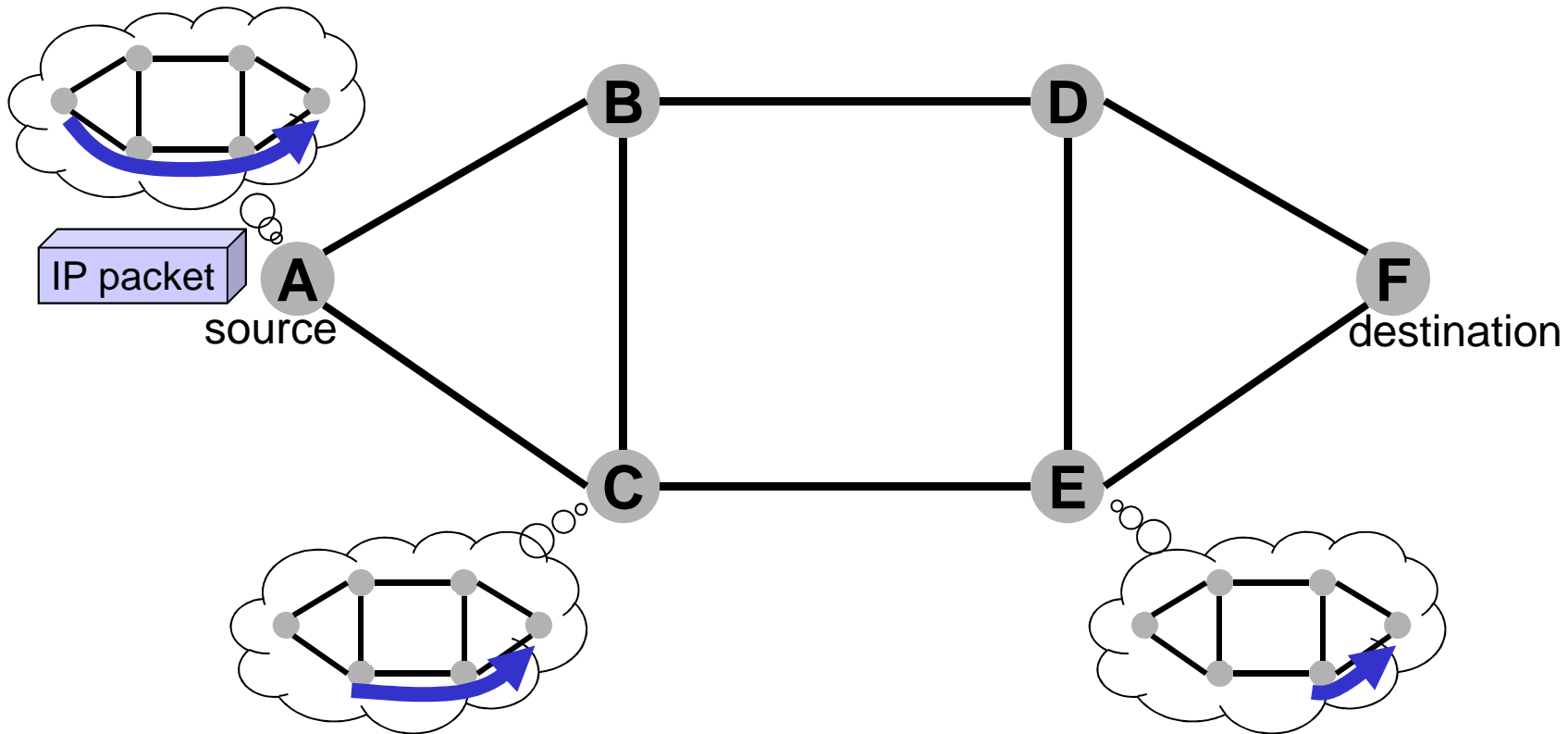
- How to prevent update loops:
- How to bring up new node:

Link state: route computation



- Each router computes shortest path tree, rooted at that router
- Determines next-hop to each dest, publish to forwarding table
- Operators can assign link costs to control path selection

Link-state: packet forwarding

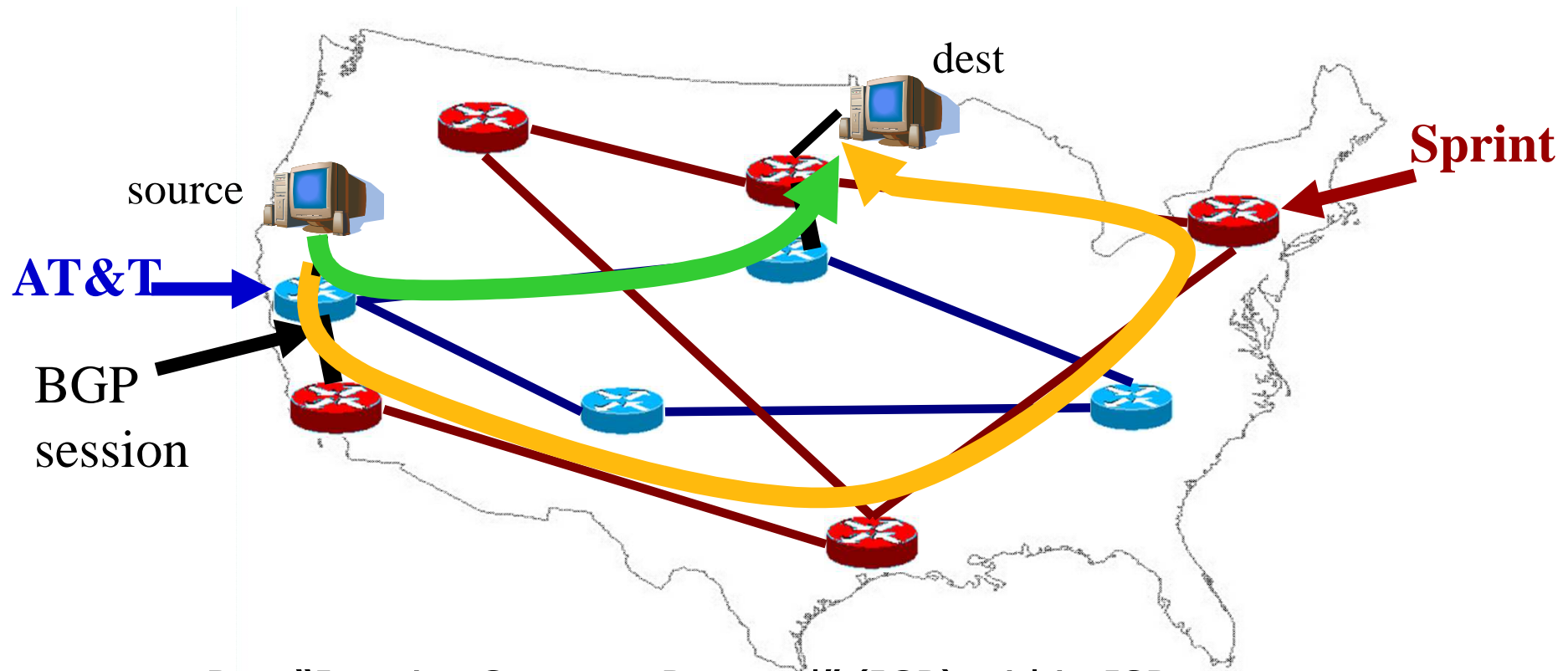


- Downsides of link-state:
 - Lesser control on policy (certain routes can't be filtered), more cpu
 - Increased visibility (bad for privacy, but good for diagnostics)

Shortest-path forwarding isn't enough

- In the real world, ISPs want to influence path selection
 - Load balance traffic, prefer cheaper paths, avoid untrusted routes, give preferential service, block reachability, limit external control over path selection decisions
- One trick: change the “cost” used to compute shortest paths
- Another trick: filter routes from being received from/advertised to certain neighbors

Intra- vs. Inter-domain routing



- Run "Interior Gateway Protocol" (IGP) within ISPs
 - OSPF, IS-IS, RIP
- Use "Border Gateway Protocol" (BGP) to connect ISPs
 - To reduce costs, peer at exchange points (AMS-IX, MAE-EAST)

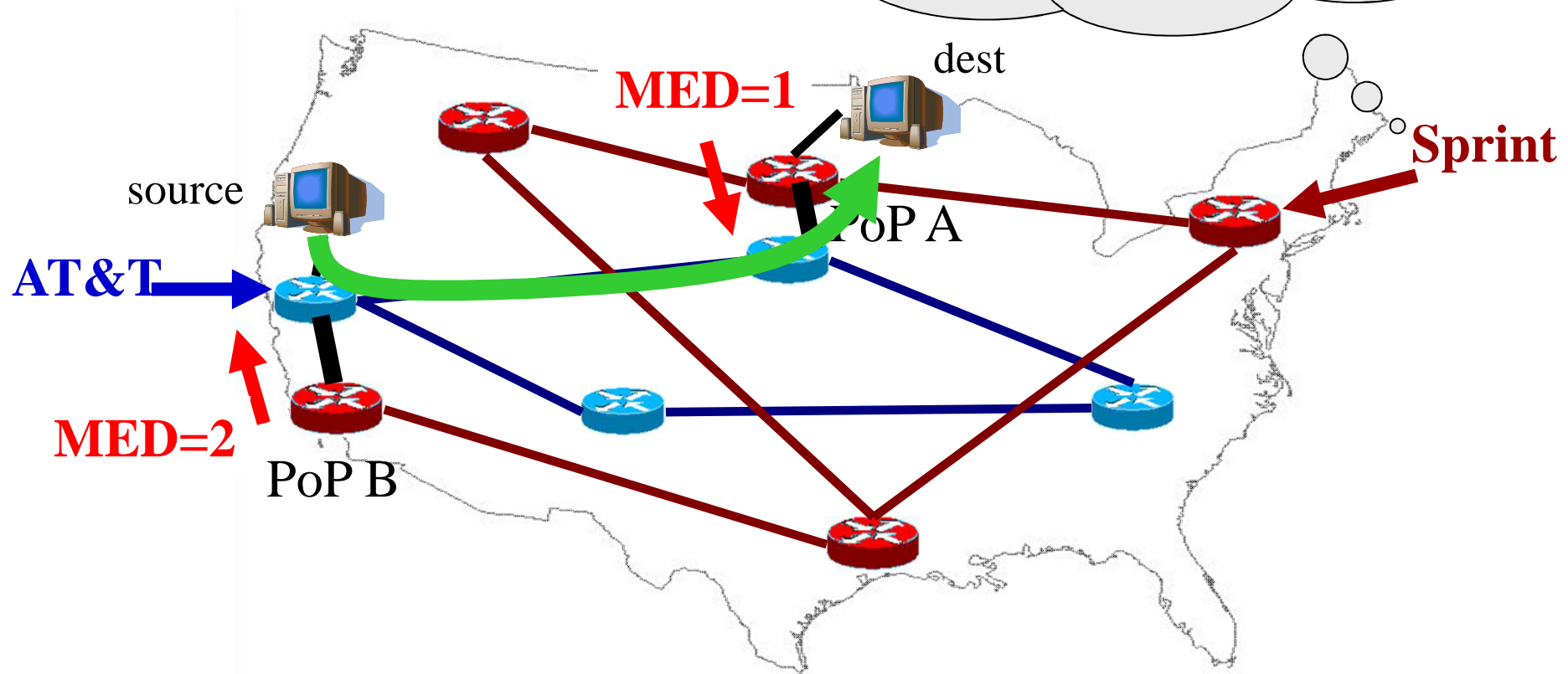
Changing the “cost” of paths

Step	Attribute	Controlled by local or neighbor AS?
1.	Highest LocalPref	local
2.	Lowest AS path length	neighbor
3.	Lowest origin type	neither
4.	Lowest MED	neighbor
5.	eBGP-learned over iBGP-learned	neither
6.	Lowest IGP cost to border router	local
7.	Lowest router ID (to break ties)	neither

- ISPs have a lot of different kinds of policies
 - Could make cost a linear combination of different metrics
 - More expressive: have several “costs” per link
- Main idea: append “attributes” to updates
- Can set preferences (or filter the route) based on set of attributes contained in update
 - Hard-coded “decision process” orders importance of attributes
 - This process can be influenced by changing values of attributes

Example: Using MED to influence traffic across multiple ingress points

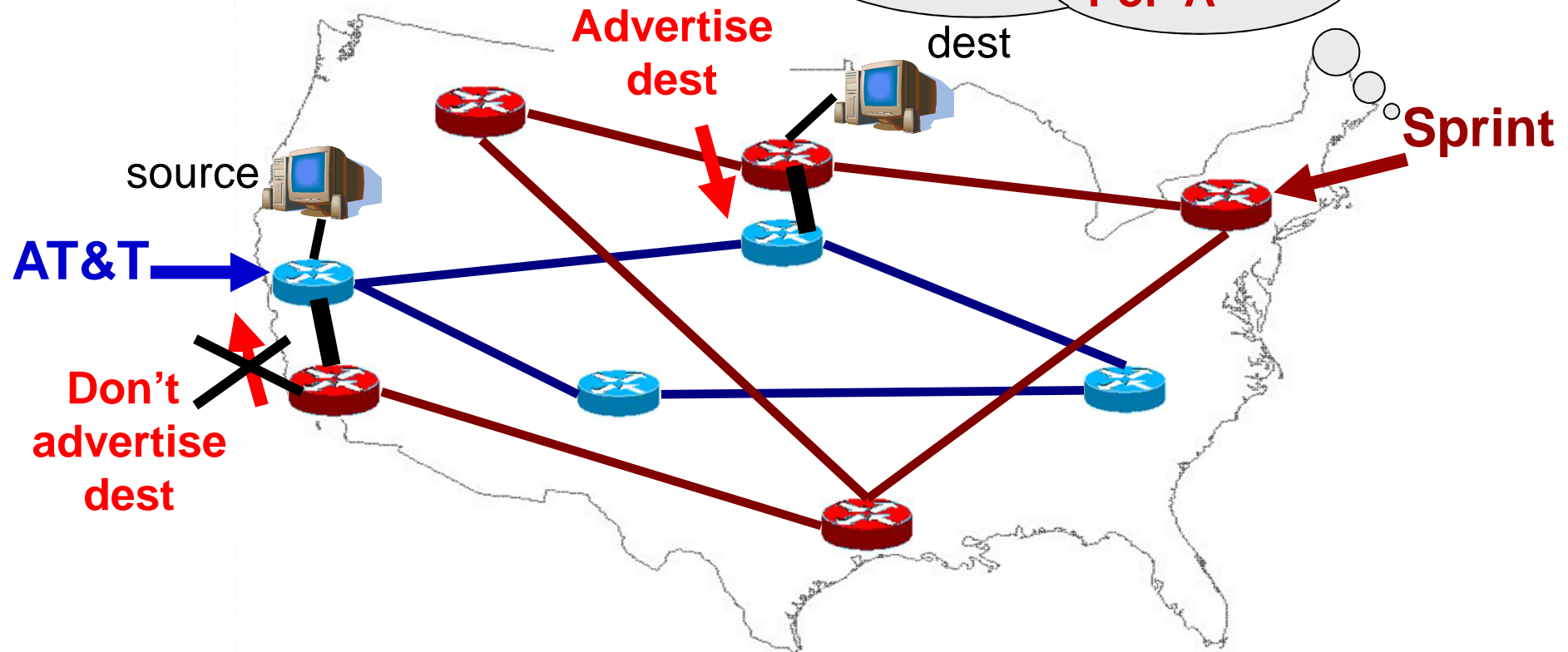
I would like AT&T to route traffic to me via PoP A



- MED: "multi-exit discriminator"
 - tell neighboring ISP which ingress peering points I prefer
 - Local ISP can choose to filter MED on import

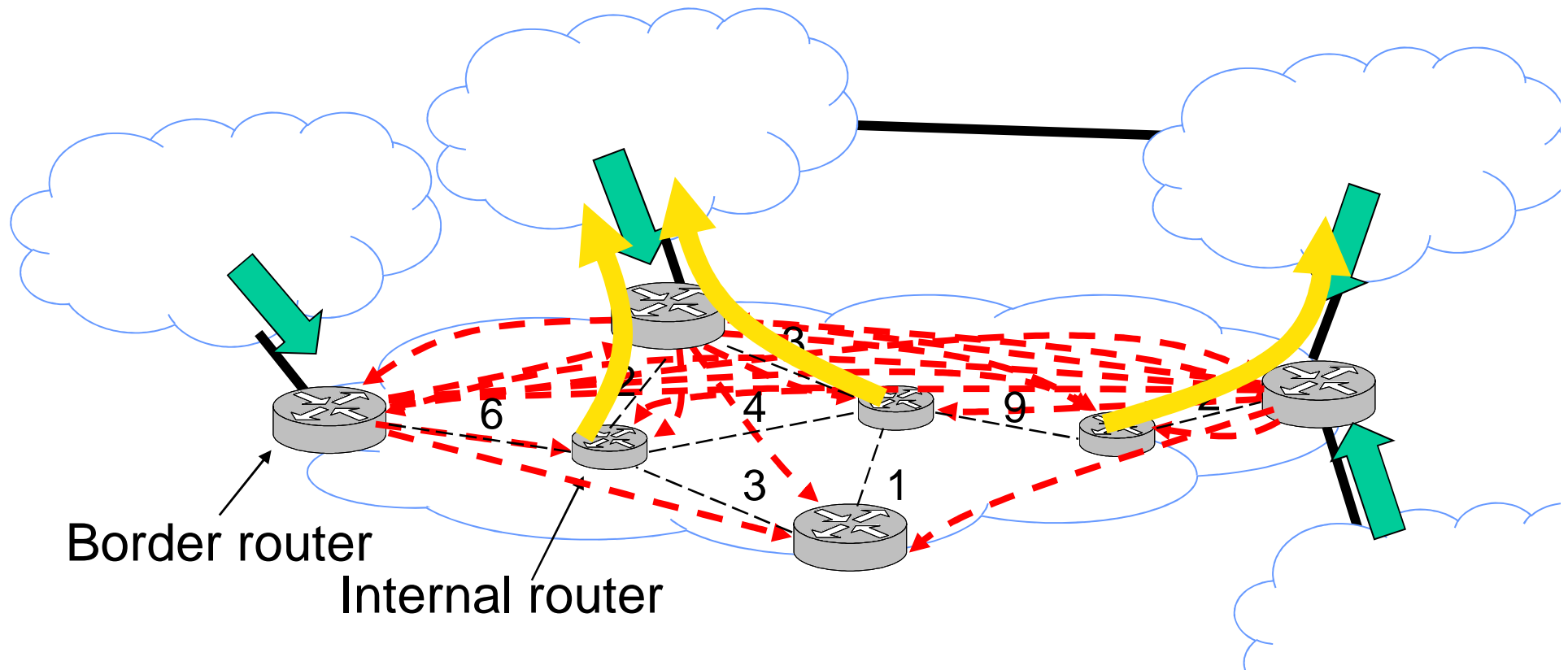
Different peering adver.

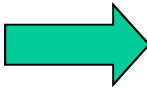

AT&T isn't listening to my
MEDs, but I would REALLY
like AT&T to route to me via
PoP A



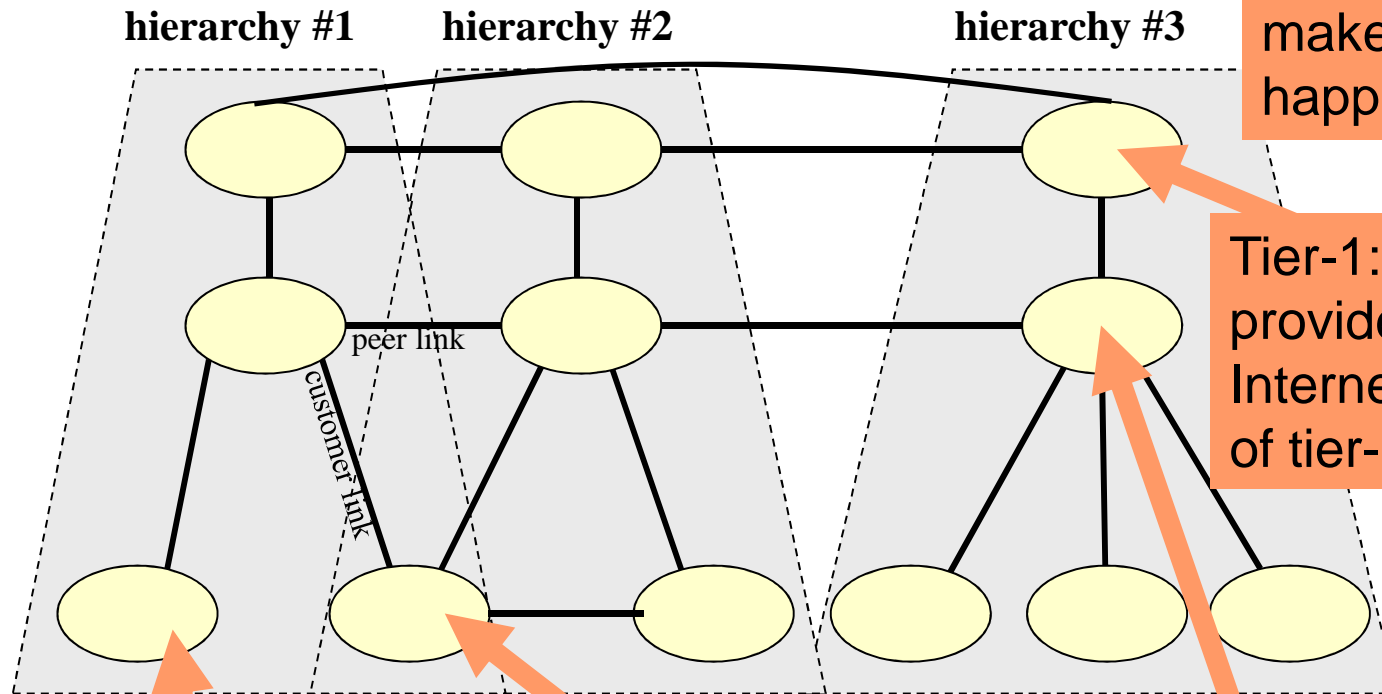
- Sprint can trick AT&T into routing over longer distance!
- Consistent export: make sure your neighbor is advertising the same set of prefixes at all peering points
- ISPs sometimes sign SLAs with consistent export clause

How inter- and intra-domain routing work together



1. Provide internal reachability (**IGP**) -----
2. Learn routes to external destinations (**eBGP**) 
3. Distribute externally learned routes internally (**iBGP**) 
4. Select closest egress (**IGP**) -----

Policies between ISPs: Types of ASes



Tier-1s must be connected in a full mesh (Why? Who makes sure that happens?)

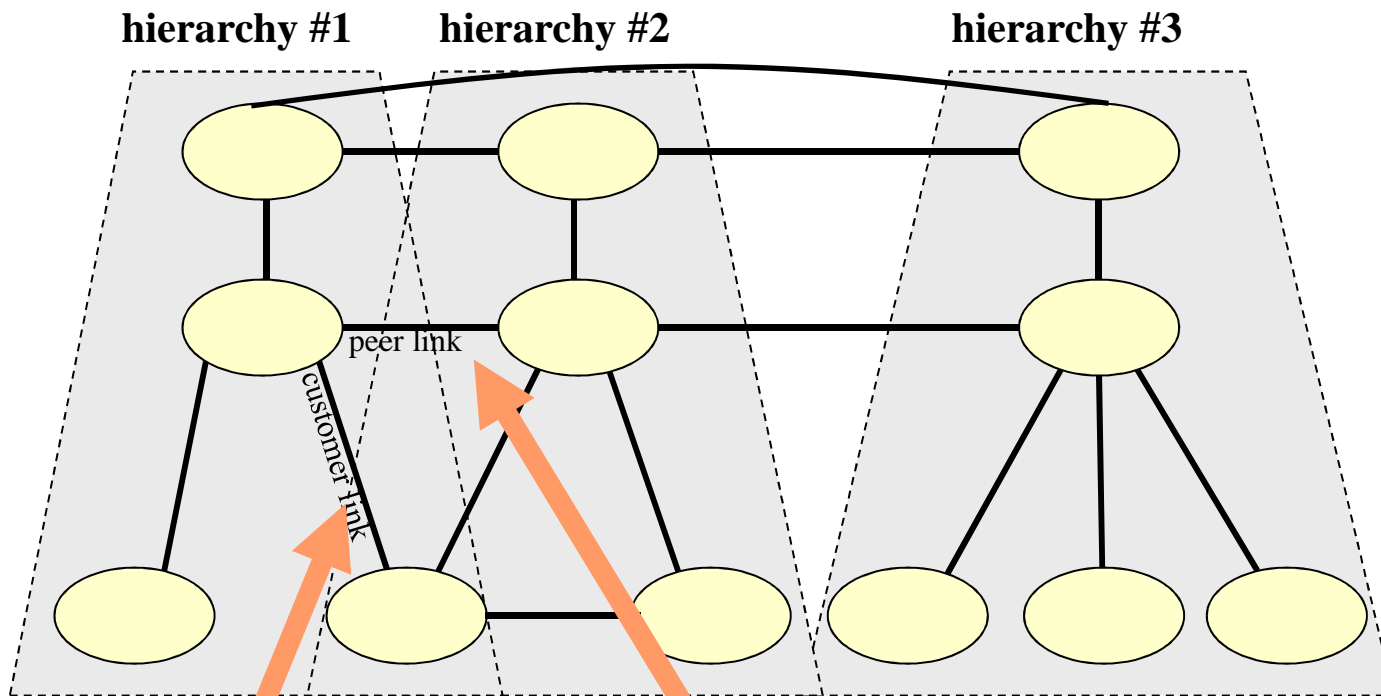
Tier-1: ISP with no providers (core of Internet is clique of tier-1s)

Stub: ISP with no customers

Multihomed: ISP with more than one provider

Transit: ISP that forward traffic between other ISPs

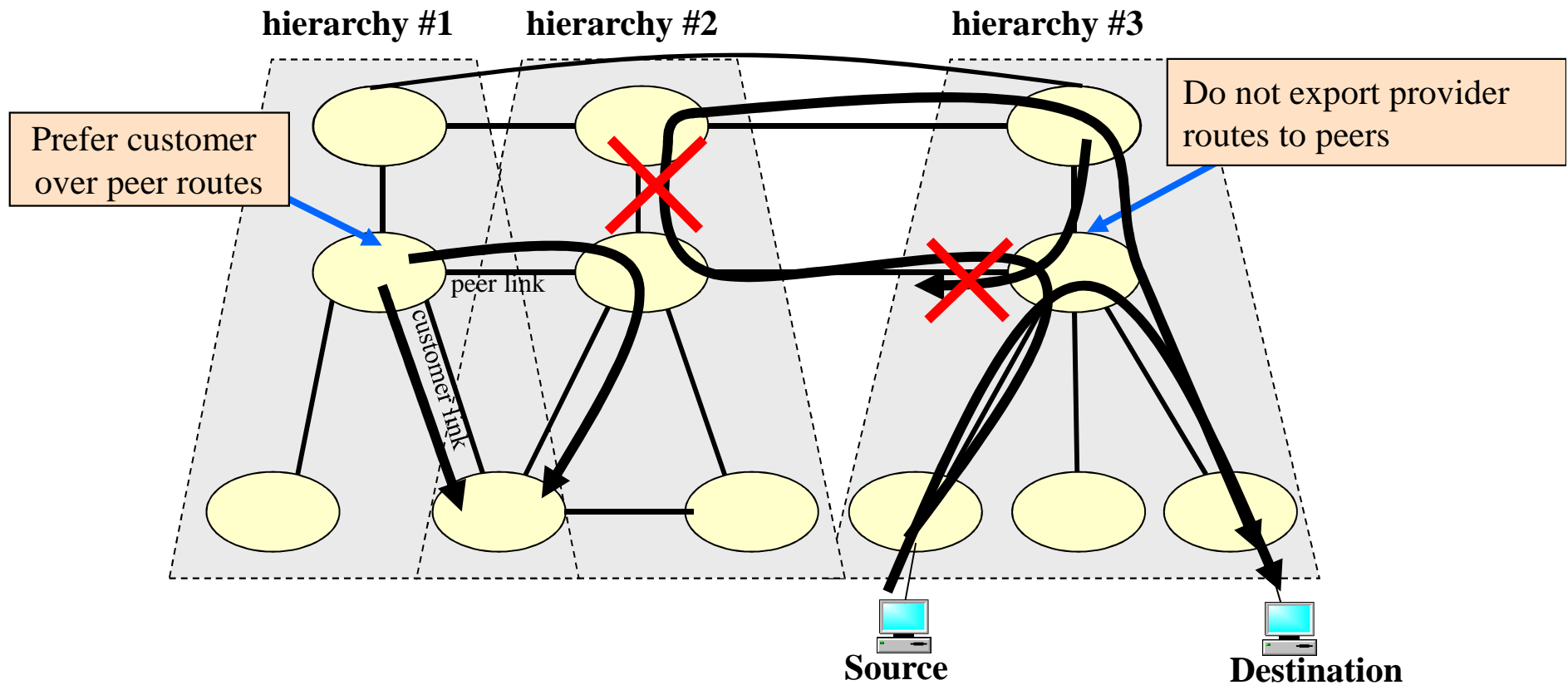
Policies between ISPs: Types of AS relationships



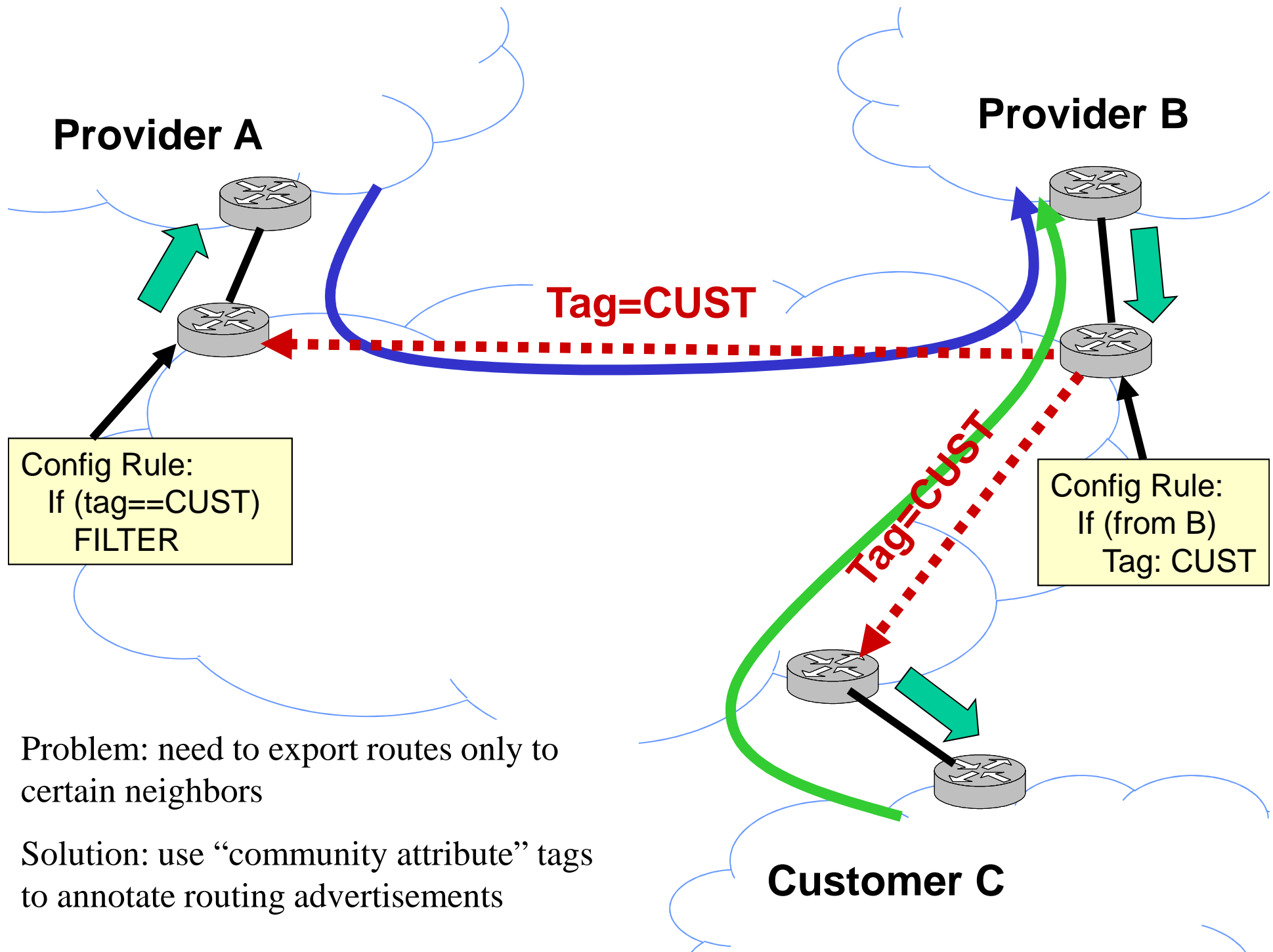
Provider-customer:
customer pays
provider money to
transit traffic

Peer link: ISPs form link out
of mutual benefit, typically
no money is exchanged

AS relationships influence routing policies



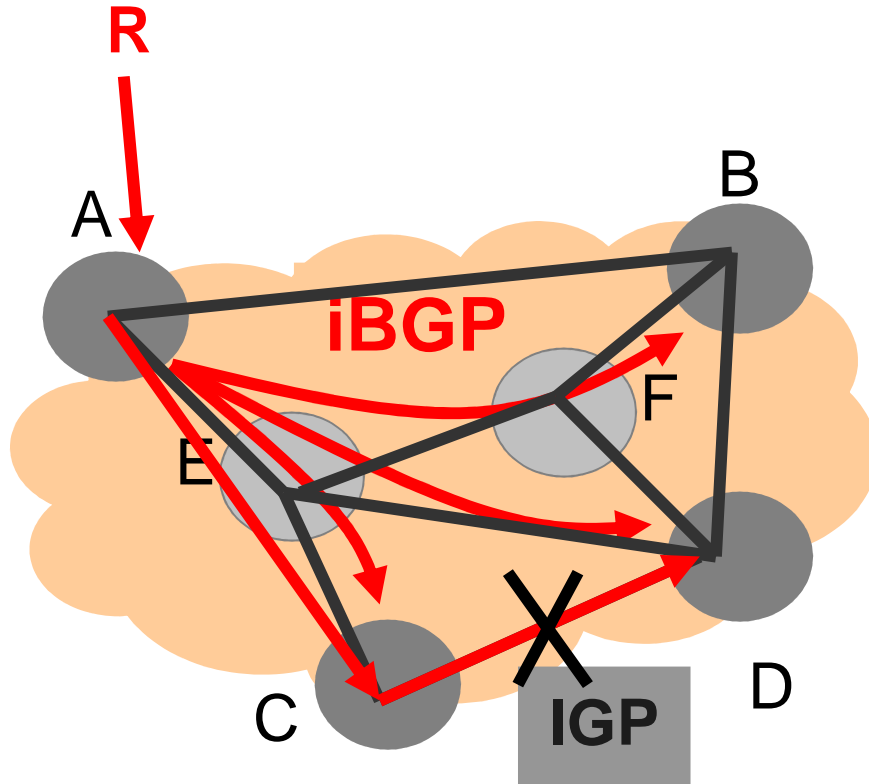
- Example policies: peer, provider/customer
- Also trust issues, security, scalability, traffic engineering



Problem: need to export routes only to certain neighbors

Solution: use "community attribute" tags to annotate routing advertisements

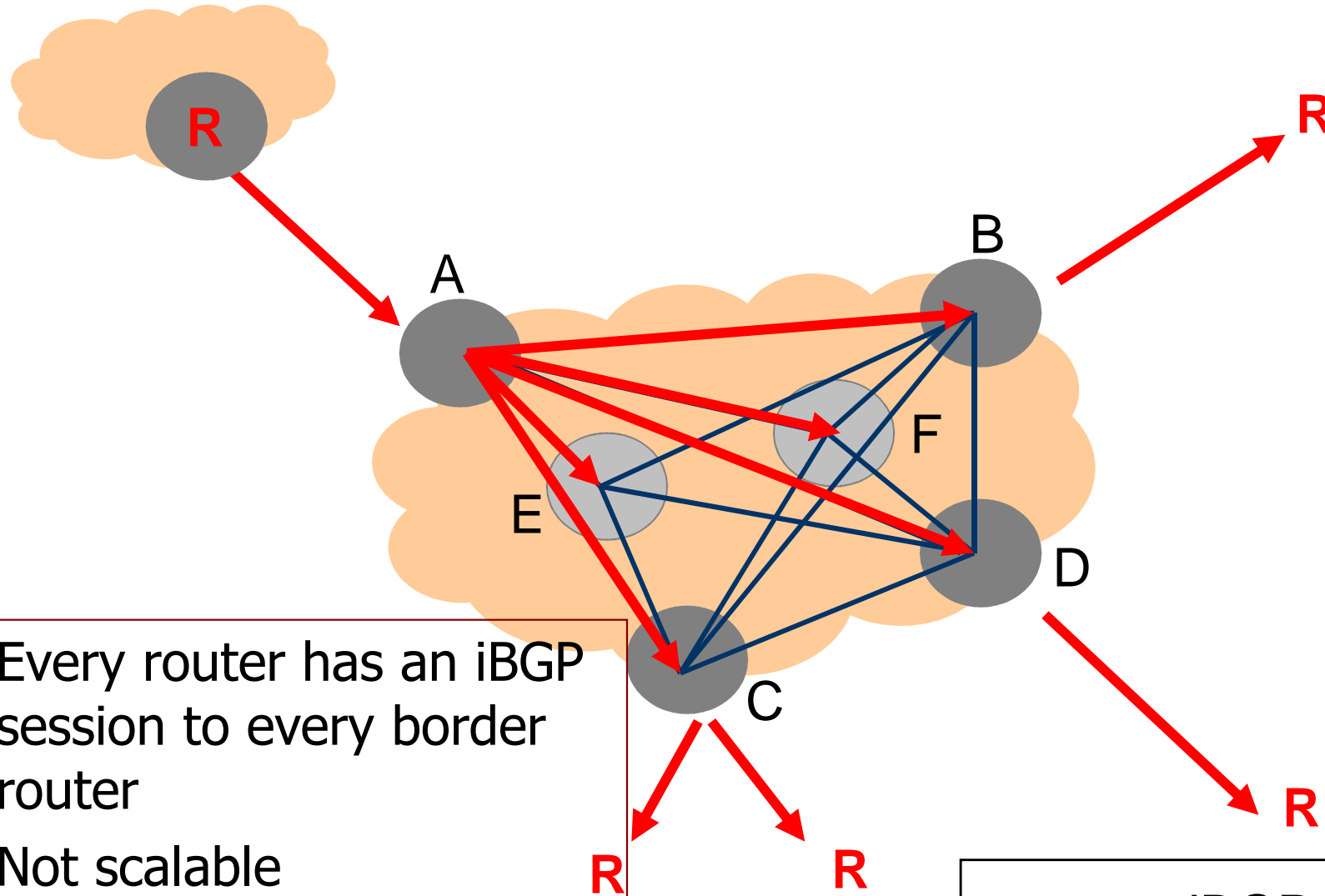
Background - iBGP



→ Route

- iBGP sessions run on TCP
- Overlay over the intra-domain routing protocol (IGP) like OSPF
- Routing messages and data packets forwarded via IGP within AS
- Routes from iBGP session not propagated to another iBGP session

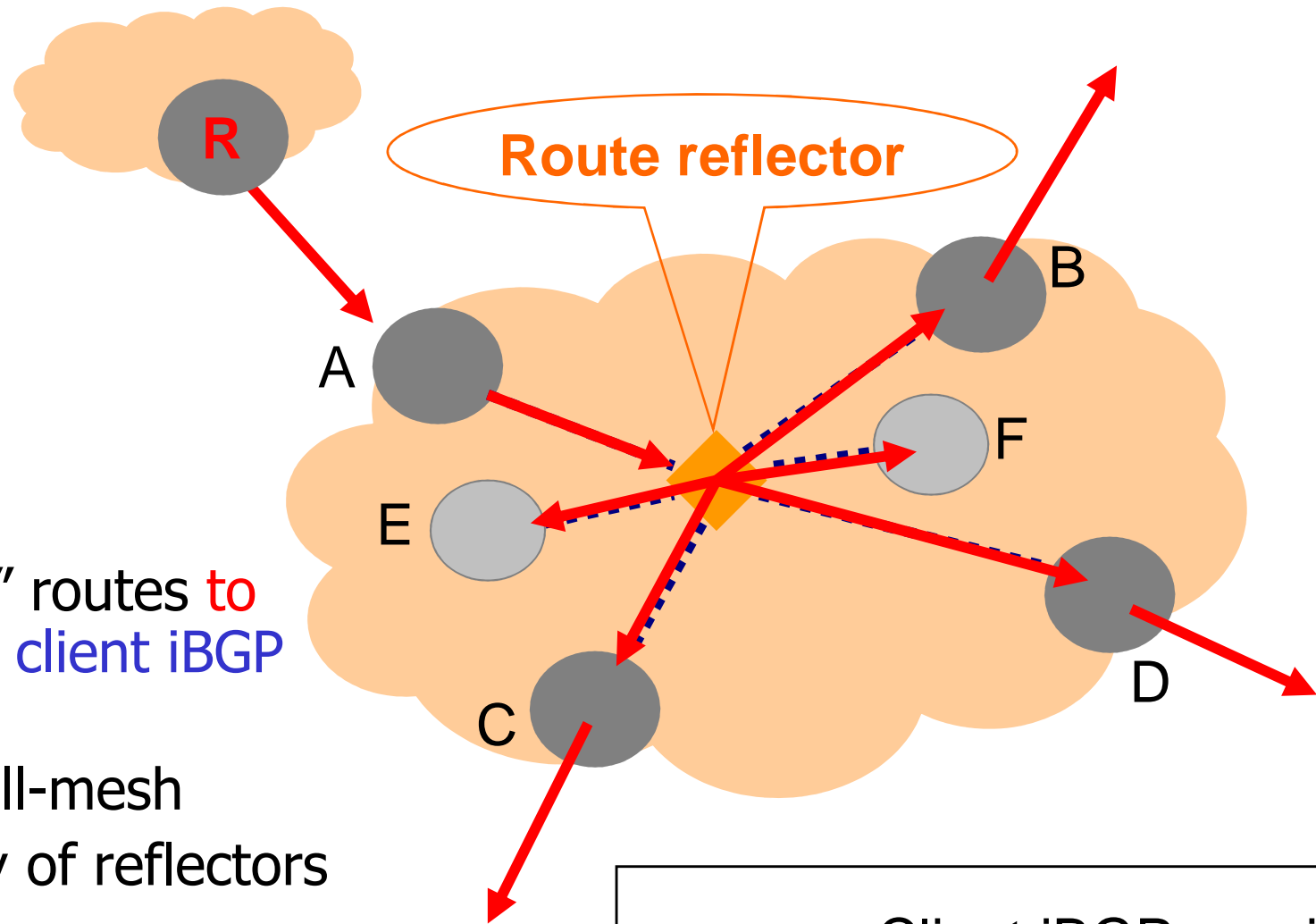
Approach#1: Full-mesh iBGP



- Every router has an iBGP session to every border router
- Not scalable



Approach#2: Route reflection



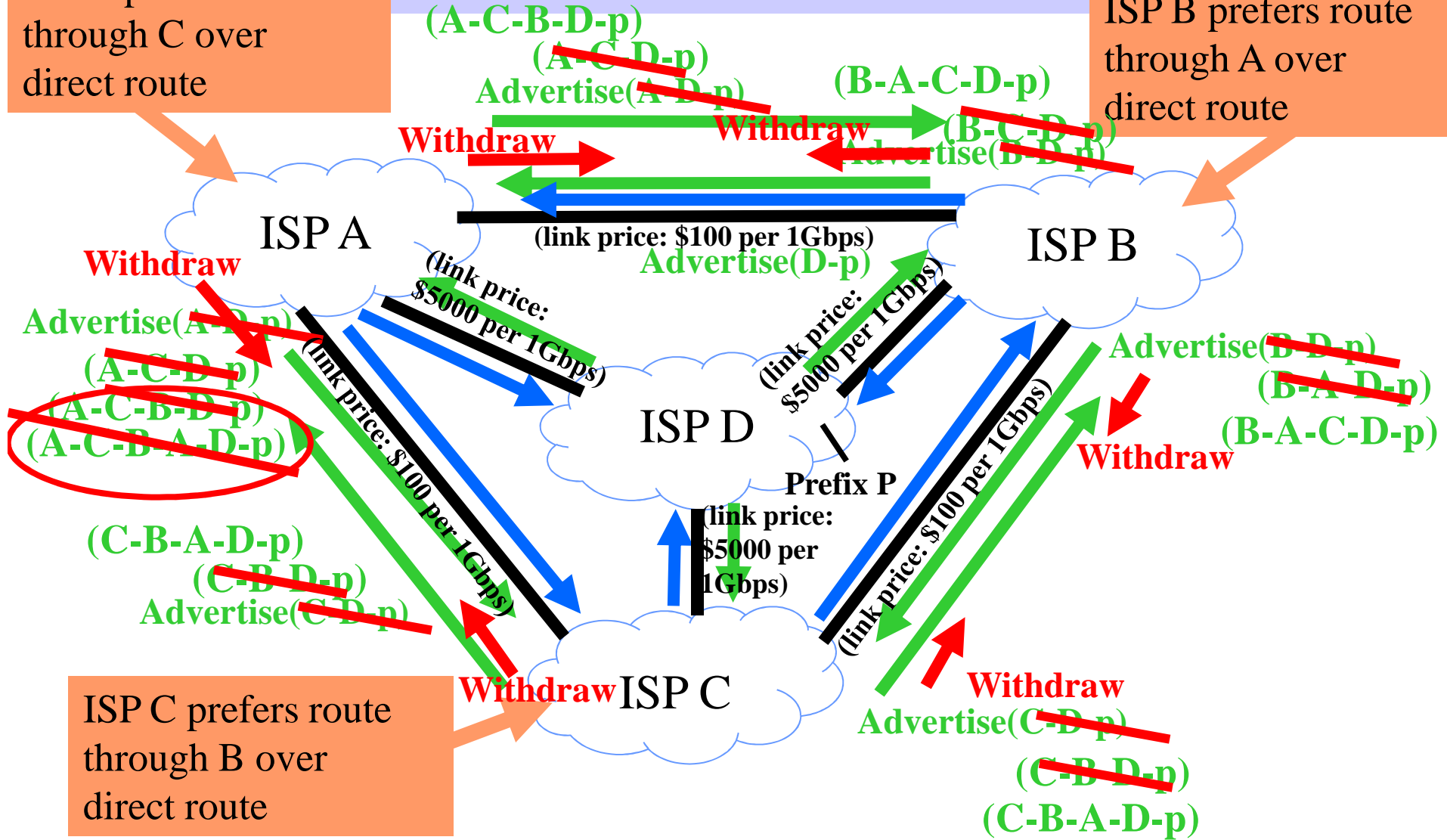
- “Reflects” routes **to** and **from** client iBGP sessions
- Avoids full-mesh
- Hierarchy of reflectors



Policy disputes

ISP A prefers route through C over direct route

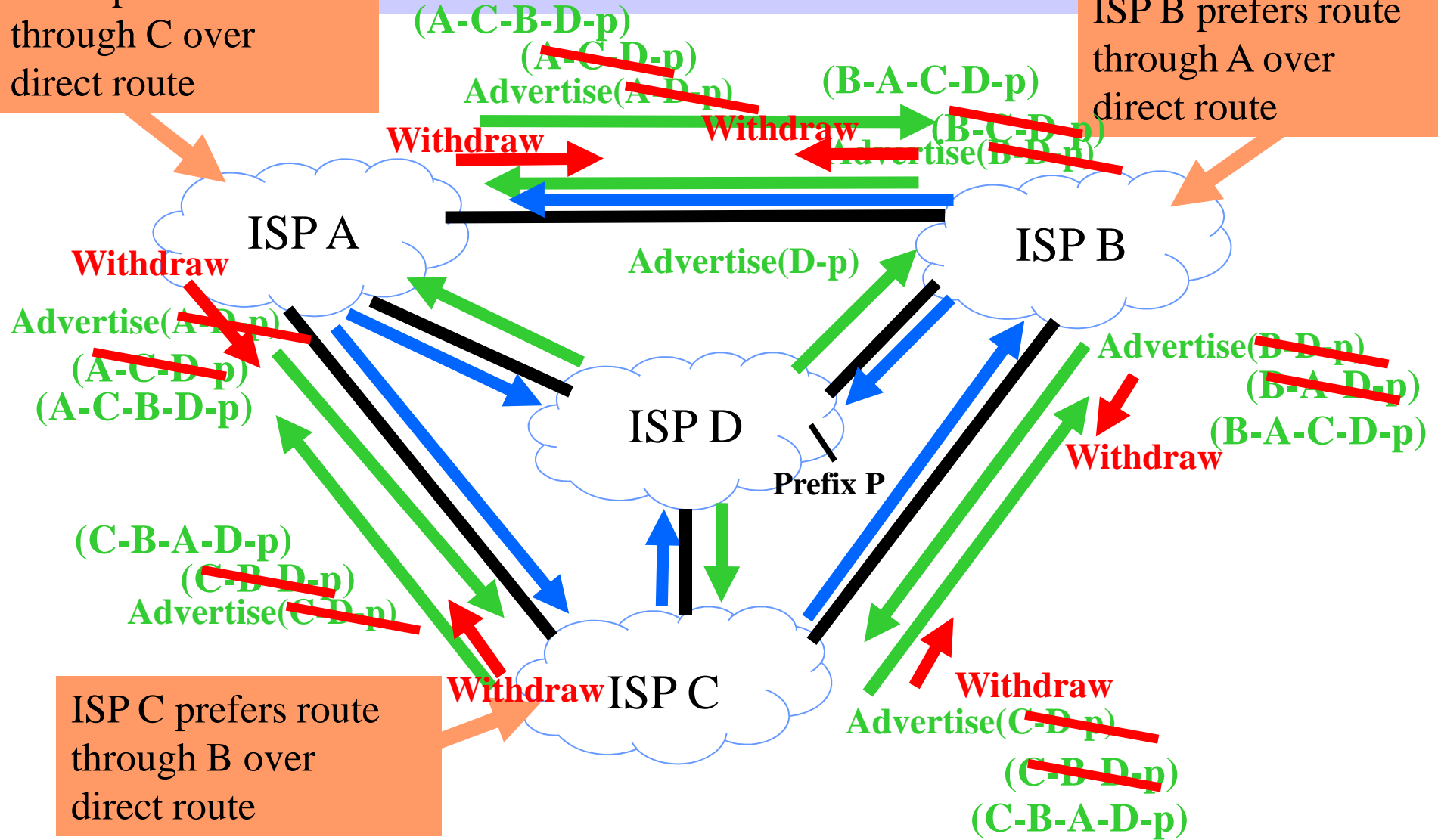
ISP B prefers route through A over direct route



Policy disputes

ISP A prefers route through C over direct route

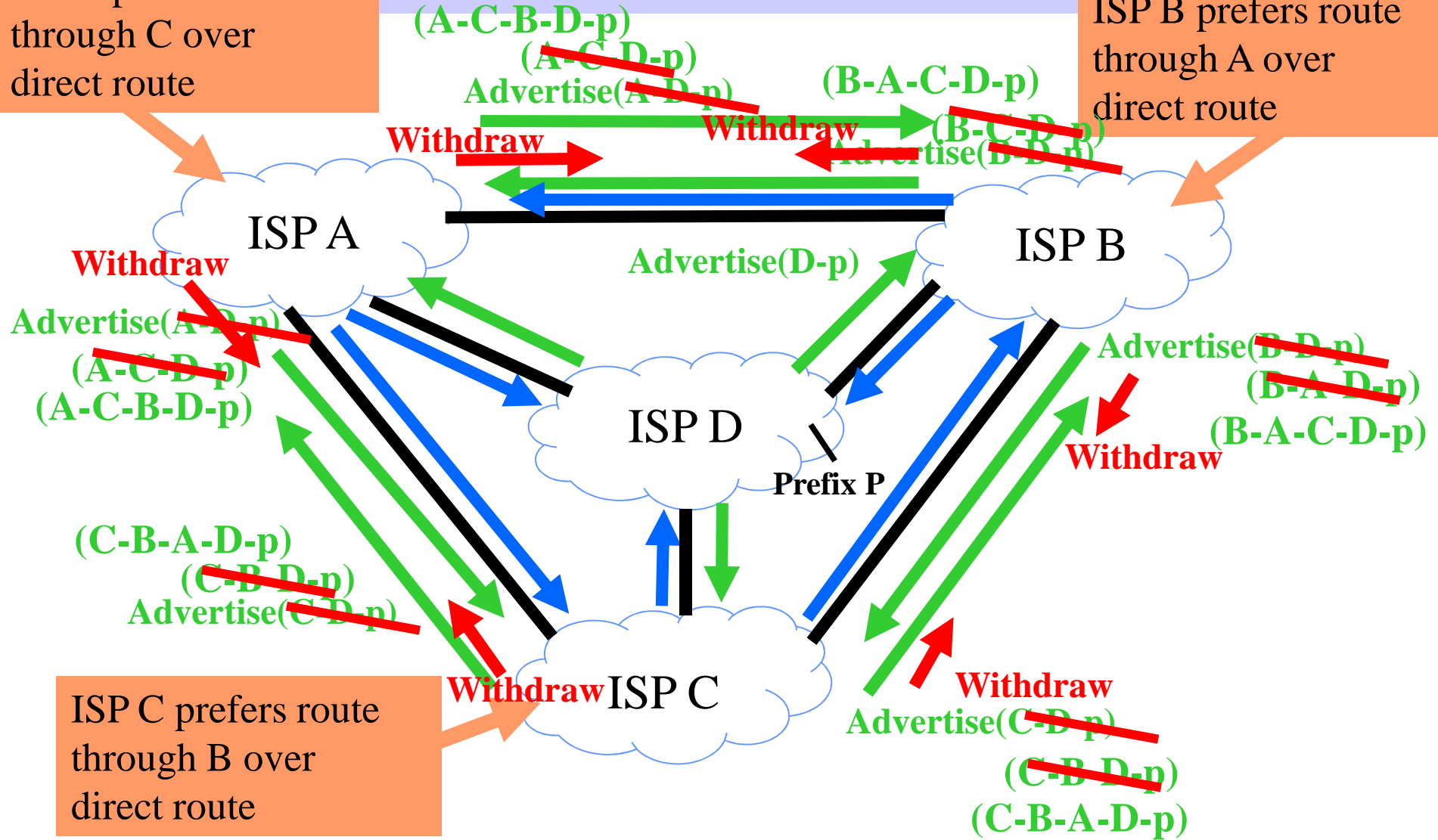
ISP B prefers route through A over direct route



Policy disputes

ISP A prefers route through C over direct route

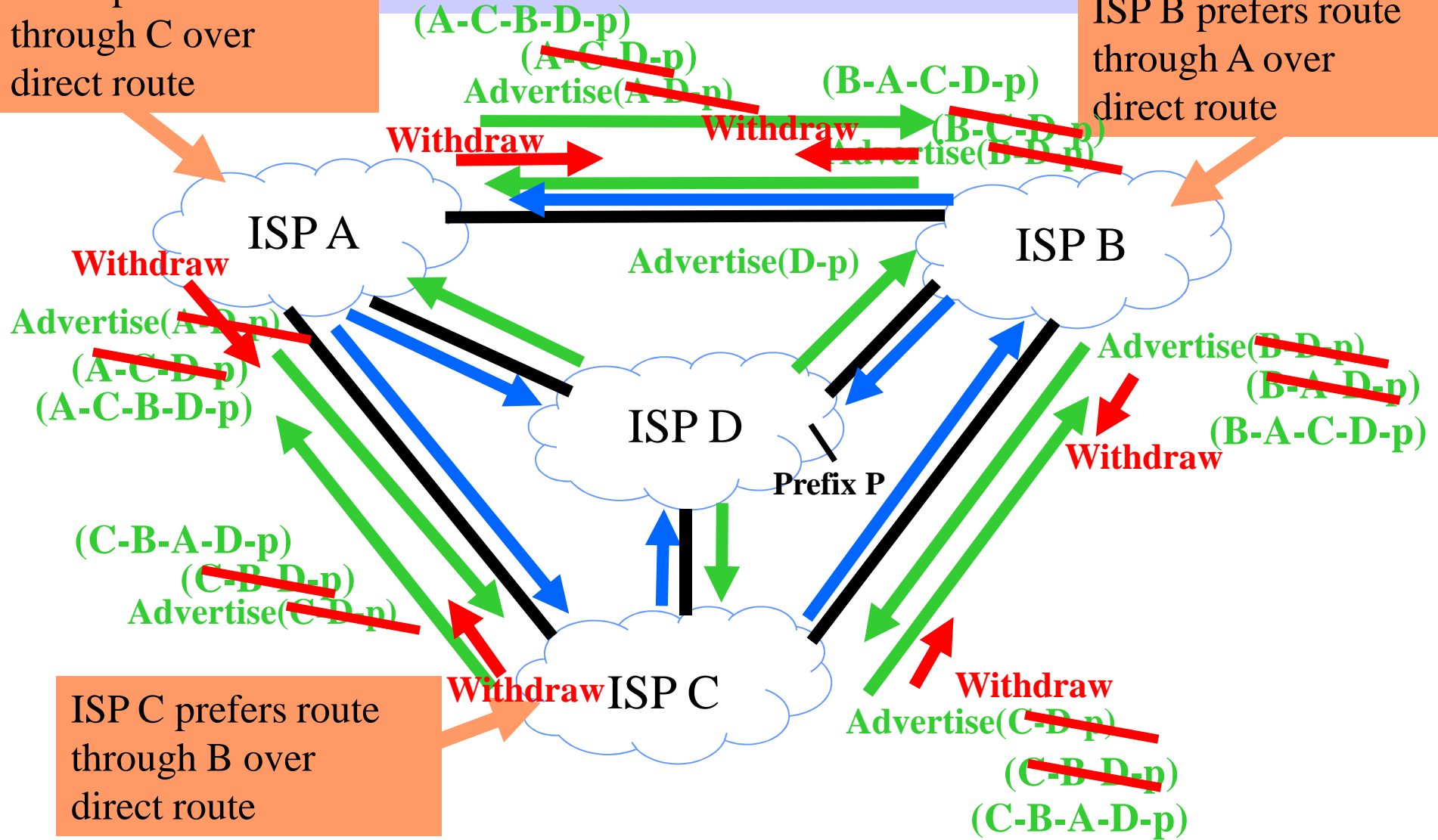
ISP B prefers route through A over direct route



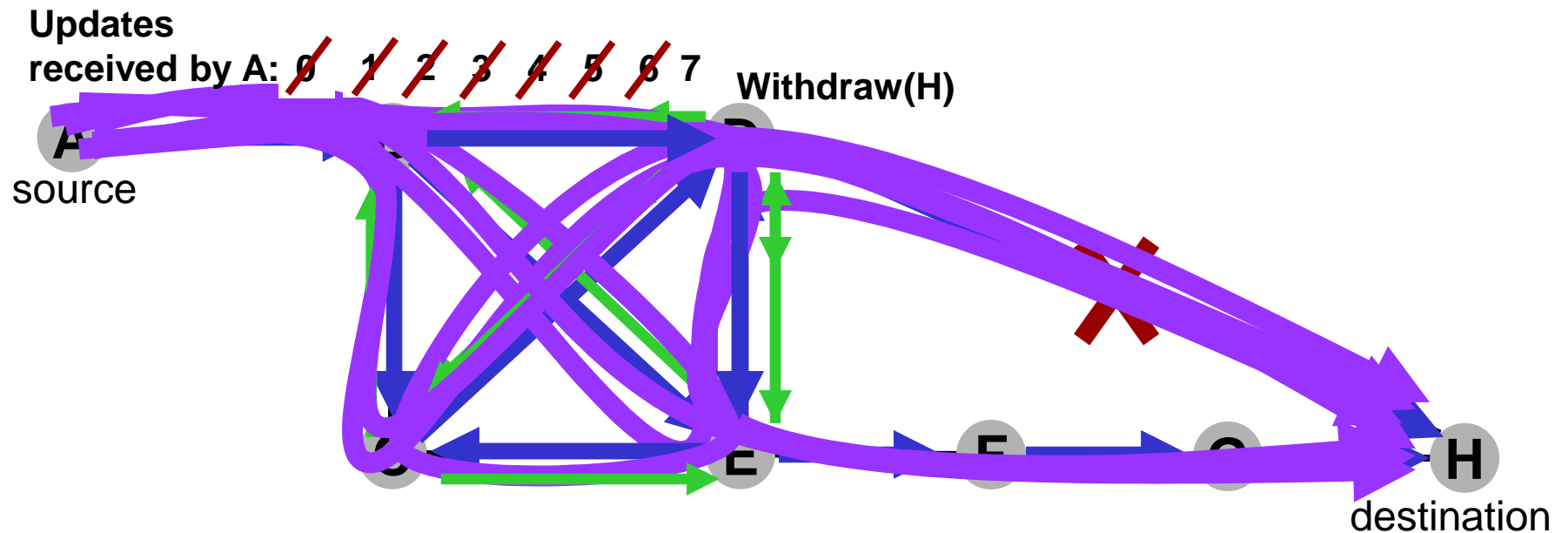
Policy disputes

ISP A prefers route through C over direct route

ISP B prefers route through A over direct route



Distance vector: convergence



- How many updates would link-state require?
- Is link-state better or worse than distance vector?
- Which should be used for intra-domain routing?
What about inter-domain routing?

How can ISPs control network usage?

Wow, AS7007!

From: Stephen A. Misel (no email)

I happened to be in one of our 7505 routers this afternoon when POP -- all of a sudden most of the internet disappeared! I immediately thought it was me, but looked around and saw this AS7007 broadcasting MY routes! It wasn't for all of our network space -- We have several /18's here, and it seemed only the first /24 of each CIDR was affected. When I found a workstation at the end of the /18, we got the whois info for 7007 -- Florida Internet Exchange, and called them.

They claimed to have a customer broadcasting some bad routing information and unplugged their router. A few moments later, the internet stabilized and I started seeing real routes.

Correct me if I'm wrong, but:

(1) We're going to read about this in EVERY computer magazine, newspaper and TV as "the end of the internet?"

- Challenges:
 - When problems occur, hard to tell who/what's the cause
 - No single entity in charge, allows for organic growth but harder to optimize routes or resolve disputes
 - Misconfigurations, cross-protocol interactions

ISP Dispute Causing Connectivity Issues for Customers

Posted by Zonk on Wednesday March 19, @06:31PM

from the [make-up-you-two-or-i'm-turning-this-interweb-around](#) dept.

[Don't Believe in Imaginary Property](#) writes

"A peering dispute between Telia and Cogent is causing routing and connectivity problems for many internet users. Cogent shut down their connections to Telia over what they described as a 'contract dispute' over the size and location of their peering points. Telia attempted to route around the problem, but Cogent blocked that, too. This has caused a lot of trouble for sites which are not multi-homed. [Groklaw](#), for example, is on a Cogent network (MCNC.demarc.cogentco.com), so any Europeans connecting via Telia can't get through."

► [communications](#), [internet](#), [it](#) ([tagging beta](#))

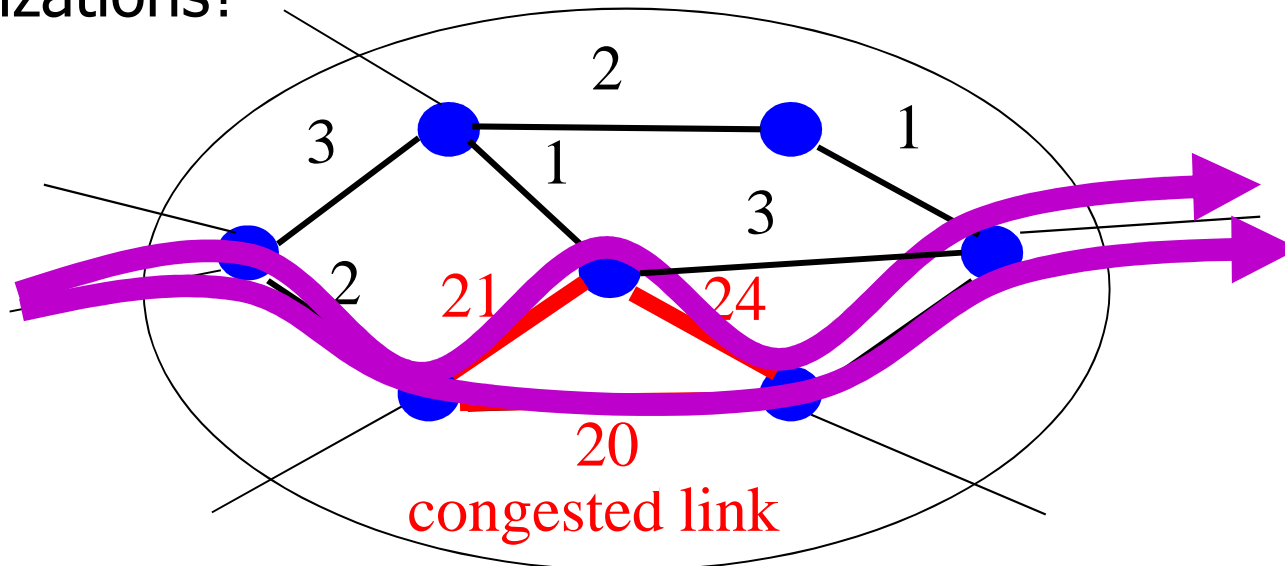


Do IP Networks Manage Themselves?

- In some sense, yes:
 - TCP senders send less traffic during congestion
 - Routing protocols adapt to topology changes
- But, does the network run *efficiently*?
 - Congested link when idle paths exist?
 - High-delay path when a low-delay path exists?
- How should routing adapt to the traffic?
 - Avoiding congested links in the network
 - Satisfying application requirements (e.g., delay)
- ... essential questions of traffic engineering

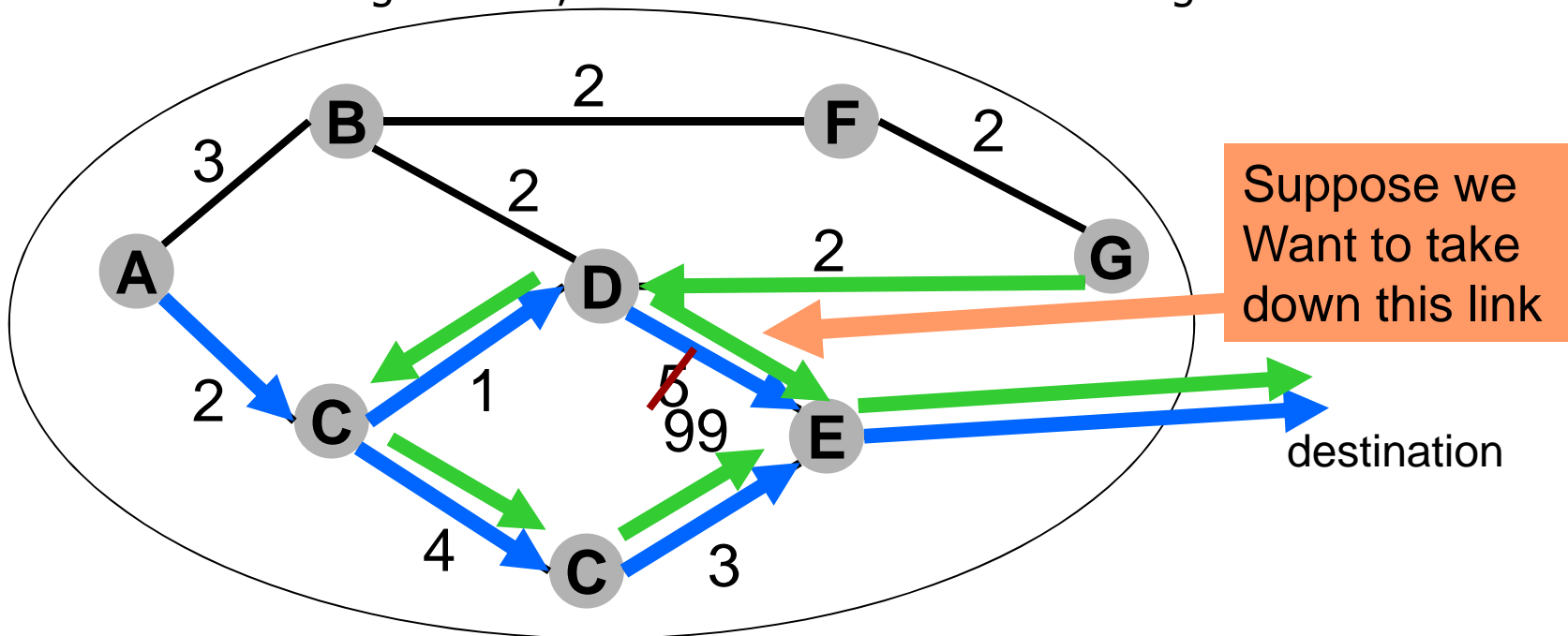
Original ARPAnet Routing (1969)

- Shortest-path routing based on congestion
 - Leads to oscillations
- Maybe provision over longer timescales?
 - But, how to predict future load? And what about path changes?
- Also, how to assign link weights based on desired utilizations?



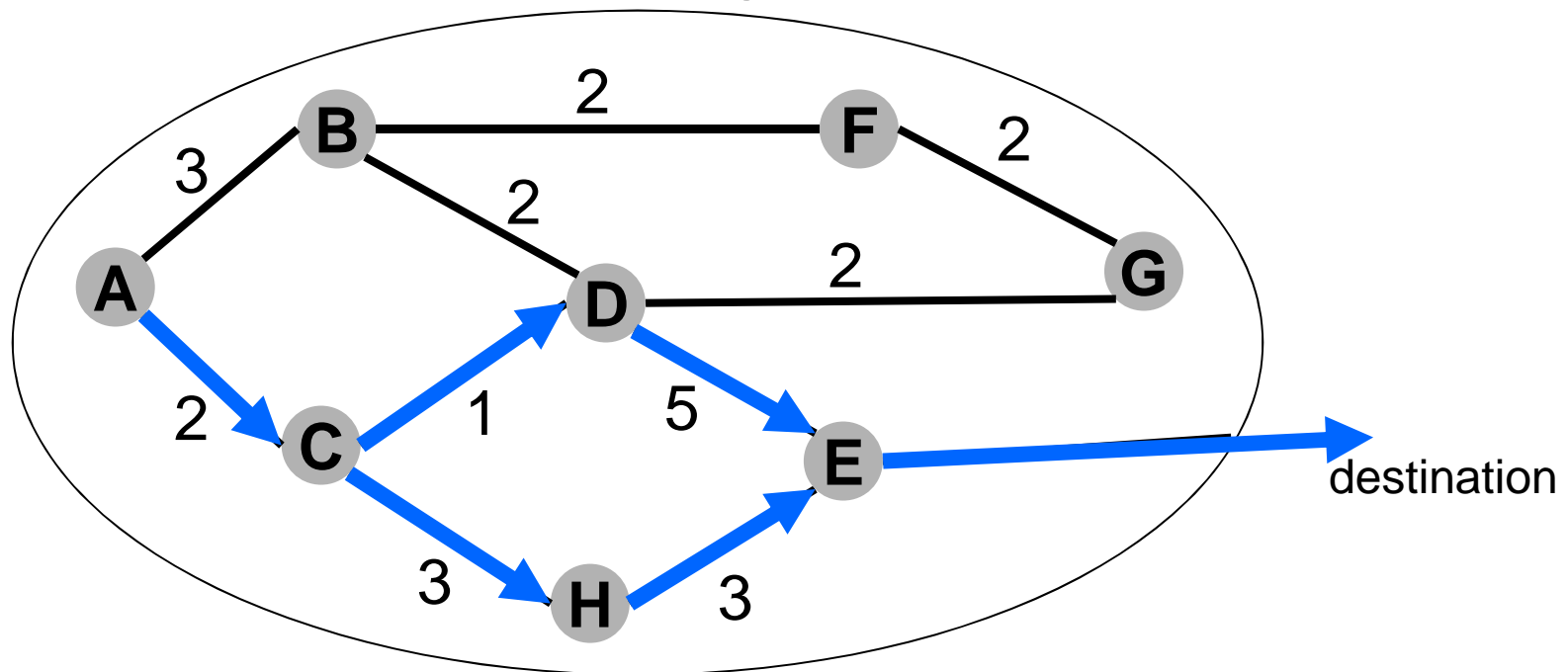
“Costing out” of equipment

- Increase cost of link to high value
 - Triggers immediate flooding of LSAs
- Leads to new shortest paths avoiding the link
 - While the link still exists to forward during convergence
- Then, can safely disconnect the link
 - New flooding of LSAs, but no influence on forwarding



Equal-Cost Multi-Path (ECMP)

- Multiple shortest paths
 - Router can compute multiple shortest paths
 - Forwarding table has multiple outgoing links
 - Router load balances traffic evenly over the links
 - Downside: packet reordering. Fix:



Network Measurement and Monitoring

Motivating Scenarios

- New job: boss tells you to run the network. Problem: previous guy who ran the network quit, and there's no documentation!
- 20% of staff suddenly can't reach external Internet. Where is the problem? How to fix it?
- Backbone is starting to get congested. Where should I provision capacity?
- Network operator is blocking/censoring my traffic – how can I circumvent?

First question: what do you have access to?

- End hosts only
 - Active: Ping, traceroute, packet-pair probing
 - Passive: snooping on traffic, tcpdump/wireshark
 - ...
- Network infrastructure
 - Trace routing updates, put traces on links, collect SNMP data...
 - ...

Internet Measurement and Monitoring: Motivation

- Need to understand what's going on in your network
 - Attacks, outages, performance issues, weak points, forensics
- Understanding helps fix these problems
 - How to provision, defend, fix and improve your network; diagnosing problems in neighbors, verifying SLAs are met
- But it's a harder problem than you might think
 - Vast amounts of information, lack of global visibility, difficulty in deploying and instrumenting measurement infrastructure, correlating and time synchronizing different measurement

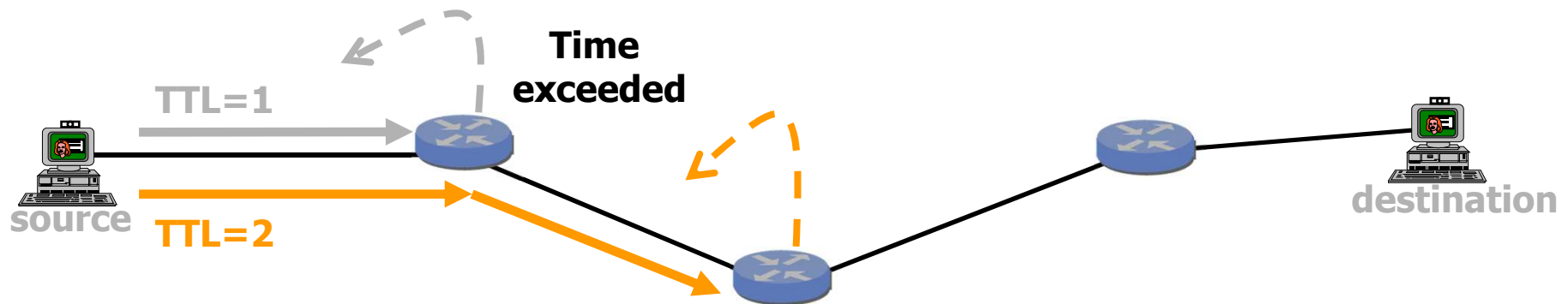
What do you want to measure?

- Internet infrastructure
 - Physical device properties, topology
- Internet traffic
 - Packets, flows, data
- Internet applications
 - DNS, web, P2P, online games, streaming, etc

Types of Measurement: Infrastructure

First question: What do you control?

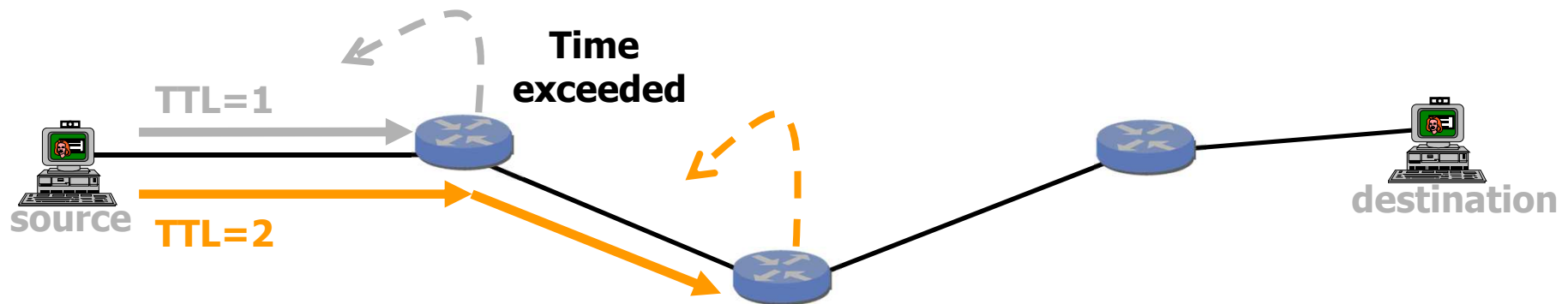
- End hosts only
 - Use *traceroute*, *ping*
- Traceroute tool exploits this TTL behavior



Send packets with TTL=1, 2, 3, ... and record source of "time exceeded" message

Finding links in a path with traceroute

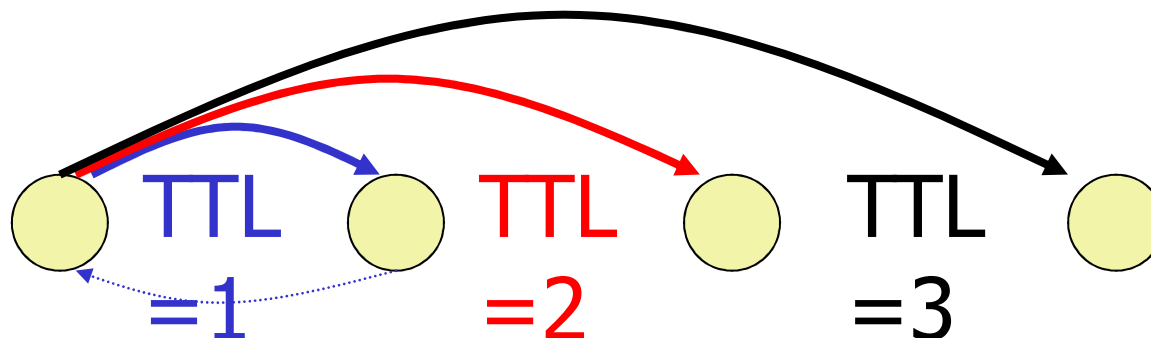
- Time-To-Live field in IP packet header
 - Source sends a packet with a TTL of n
 - Each router along the path decrements the TTL
 - “TTL exceeded” sent when TTL reaches 0
- Traceroute tool exploits this TTL behavior



Send packets with TTL=1, 2, 3, ... and record source of "time exceeded" message

Problems with Traceroute

- Can't unambiguously identify one-way outages
 - Failure to reach host : failure of *reverse* path?
- ICMP messages may be filtered or rate-limited
- IP address of “time exceeded” packet may be the *outgoing* interface of the *return* packet

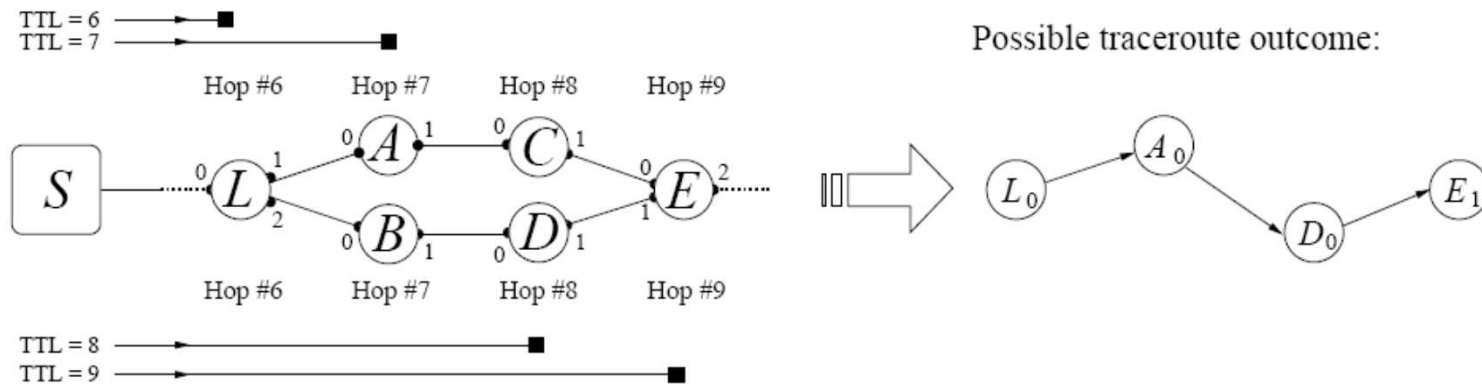


More Caveats: Topology Measurement

- Routers have multiple interfaces
- Measured topology is a function of vantage points
- **Example:** Node degree
 - Must “alias” all interfaces to a single node (PS 2)
 - Is topology a function of vantage point?
 - Each vantage point forms a tree
 - See Lakhina *et al.*

Less Famous Traceroute Pitfall

- Host sends out a sequence of packets
 - Each has a different destination port
 - Load balancers send probes along different paths
 - Equal cost multi-path
 - Per flow load balancing



Applications of traceroute

- Network troubleshooting
 - Identify forwarding loops and black holes
 - Identify long and convoluted paths
 - See how far the probe packets get
- Network topology inference
 - Launch traceroute probes from many places
 - ... toward many destinations
 - Join together to fill in parts of the topology
 - ... though traceroute undersamples the edges

Challenges of traceroute

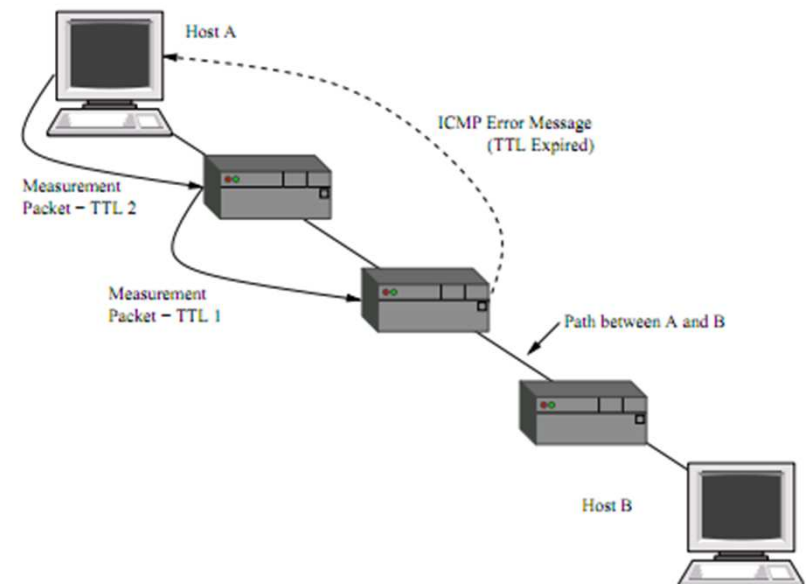
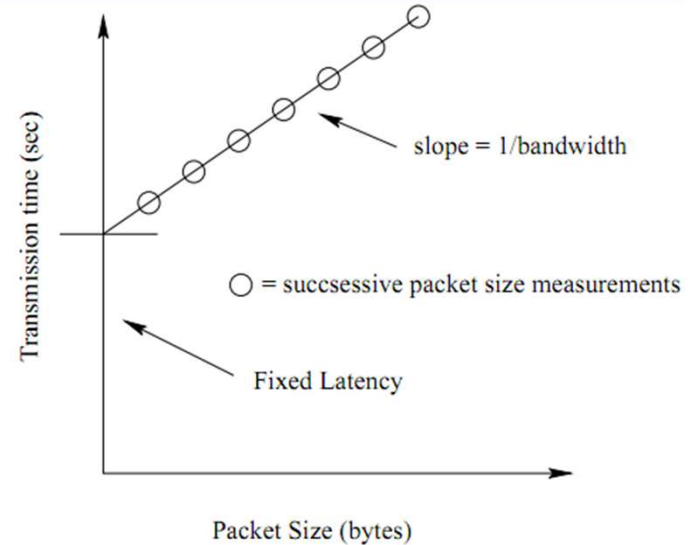
- Can be fooled by load balancing in the network
 - Successive probes may traverse different paths
- Non-participating network elements
 - Some routers and firewalls don't reply
- Inaccurate delay information
 - Includes processing delays on the router CPU
- Round-trip vs. one-way measurements
 - Paths may have asymmetric properties
- Interfaces, not routers
 - Returns IP address of interfaces, not routers
- Traceroute may reveal false loops
 - Path change that leads to a longer path
 - Causing later probe packets to hit same nodes

Measuring bandwidth: What is “bandwidth”, anyway?

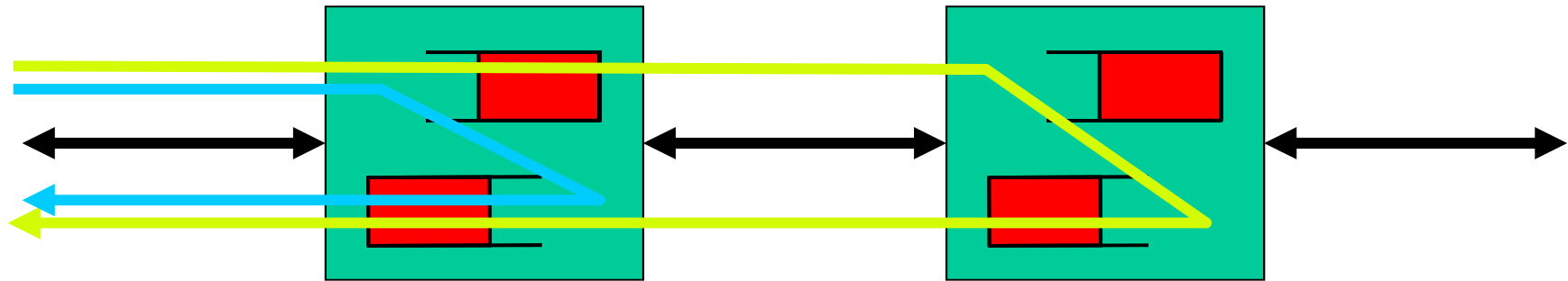
- Link vs. path bandwidth:
 - **Link bandwidth:** rate at which bits can be sent over a single link
 - **Path bandwidth:** minimum of link bandwidths along the path
 - **Bottleneck link:** the link on the path with the minimum bandwidth
- Capacity vs available bandwidth
 - **Capacity:** total bits per second that could be sent
 - **Available bandwidth:** amount of bandwidth “left over” after cross traffic

Estimating bandwidth: Single-packet estimation

- Observation: transmission time of a packet is a function of link bandwidth
 - Transmission time = (Packet size) / (bandwidth) + latency
- Idea: send varying packet sizes, measure transmission time to infer bandwidth
 - Repeat across hops using traceroute-style TTL expiry trick
- Downside:
 - IP limits max packet size
 - Errors accumulate over links in multihop case



Example: Pathchar



$$rtt(i+1) = rtt(i) + d + L/c + \epsilon$$

i : initial TTL value

c : link capacity

L : packet size

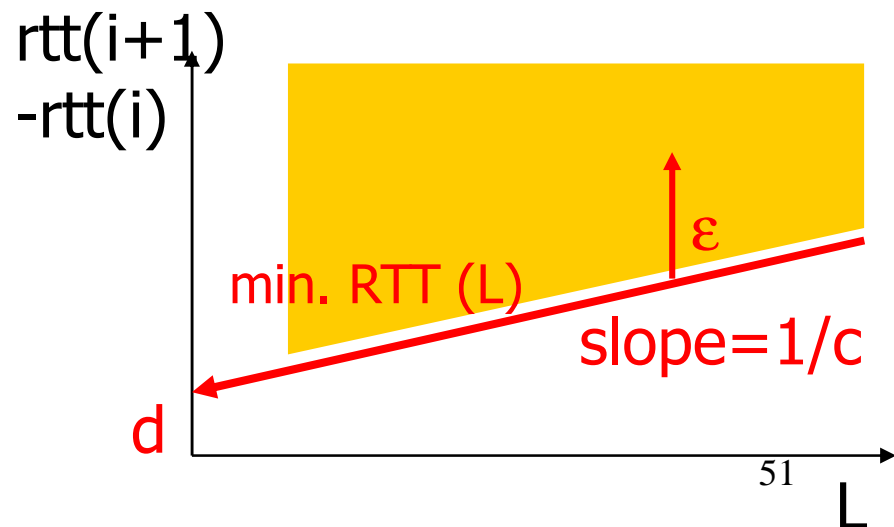
Three delay components:

d : propagation delay

L/c : transmission delay

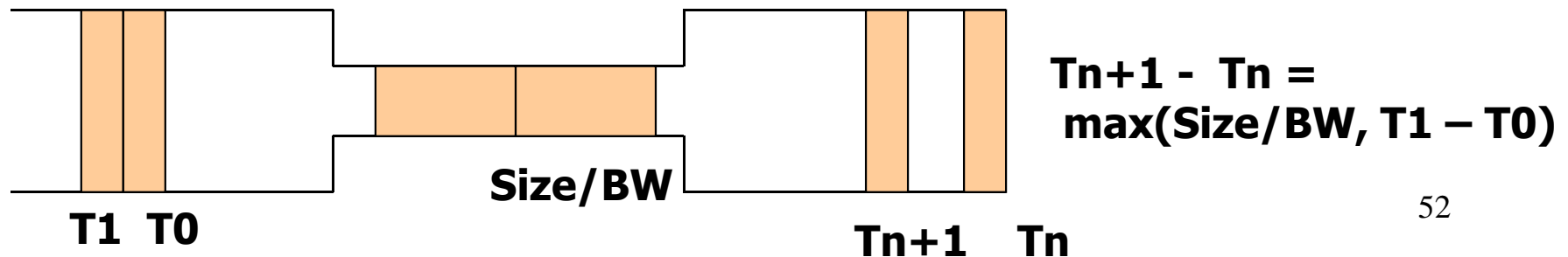
ϵ : queueing delay + noise

How to infer d, c ?



Estimating bandwidth: Packet pair technique

- Packet-pair: send two packets back-to-back, measure difference in time when they arrive at the destination
 - Difference in time caused by serialization delay at intermediate links
 - Many variants: packet trains, packet trails
- Downsides:
 - Measure path, not link capacity



Routing Data

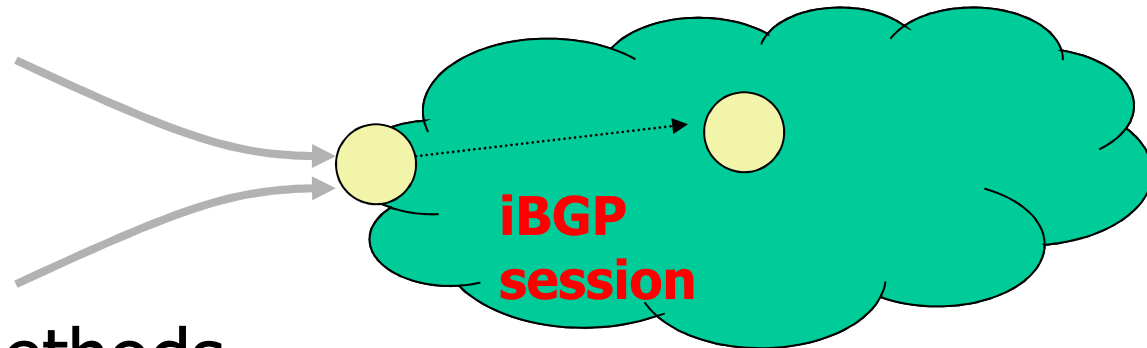
- IGP
- BGP

– Collection methods

- eBGP (typically “multihop”)
- iBGP

– Table dumps: Periodic, complete routing table state (direct dump from router)

– Routing updates: Continuous, incremental, best route only



BGP Routing Updates: Example

TIME: 07/06/06 19:49:52
TYPE: BGP4MP/STATE_CHANGE
PEER: 18.31.0.51 AS65533
STATE: Active/Connect

TIME: 07/06/06 19:49:52
TYPE: BGP4MP/STATE_CHANGE
PEER: 18.31.0.51 AS65533
STATE: Connect/Opensent

TIME: 07/06/06 19:49:52
TYPE: BGP4MP/STATE_CHANGE
PEER: 18.31.0.51 AS65533
STATE: Opensent/Active

TIME: 07/06/06 19:49:55
TYPE:
BGP4MP/MESSAGE/Update
FROM: 18.168.0.27 AS3
TO: 18.7.14.168 AS3

WITHDRAW
12.105.89.0/24
64.17.224.0/21
64.17.232.0/21
66.63.0.0/19
89.224.0.0/14
198.92.192.0/21
204.201.21.0/24

Accuracy issue: Old versions of Zebra would not process updates during a table dump...buggy timestamps.

BGP Routing Updates: Example

```
~/code/caesar/utils/routing: > bunzip2 -cf  
rib.20030402.1152.bz2 | rba | head -n  
30
```

```
TIME: 04/02/03 11:52:00  
TYPE: TABLE_DUMP/INET  
VIEW: 0  
SEQUENCE: 1  
PREFIX: 3.0.0.0/8  
FROM: 217.75.96.60 AS16150  
ORIGINATED: 04/02/03 11:27:17  
ORIGIN: IGP  
ASPATH: 16150 8434 3257 1239 7018 80  
NEXT_HOP: 217.75.96.60  
COMMUNITY: 3257:3000 3257:3030  
3257:3032 3257:5031 16150:65305  
16150:65317 16150:65321  
STATUS: 0x1
```

```
TIME: 04/02/03 11:52:00  
TYPE: TABLE_DUMP/INET  
VIEW: 0  
SEQUENCE: 2  
PREFIX: 3.0.0.0/8  
FROM: 147.28.255.2 AS3130  
ORIGINATED: 04/01/03 14:34:03  
ORIGIN: IGP  
ASPATH: 3130 2914 7018 80  
NEXT_HOP: 147.28.255.2  
MULTI_EXIT_DISC: 20  
COMMUNITY: 2914:420 2914:2000  
2914:3000 3130:200 3130:300  
STATUS: 0x1
```

```
~/code/caesar/utils/routing: >
```

Types of Measurement: Traffic

Outline

- Netflow
- Heavy hitter detection

Granularities of traffic measurement

- **Packet-level:**
 - Tcpdump: software based
 - Special hardware packet collectors
- **Flow-level:**
 - Cisco Netflow; other vendors have similar facility
 - 5-tuple flow: srcIP, dstIP, scrPort, dstPort, protocol
 - use a time-out value to “terminate” a flow
 - statistics collected: start/end time, packet/byte counts
 - Sampling may be used for scalability
- **Link-level:**
 - SNMP traffic statistics, often over 5-min interval
 - IETF MIB (management information base)
 - Byte counts, packet counts, etc.
- pros and cons of each?

Simple Network Management Protocol (SNMP)

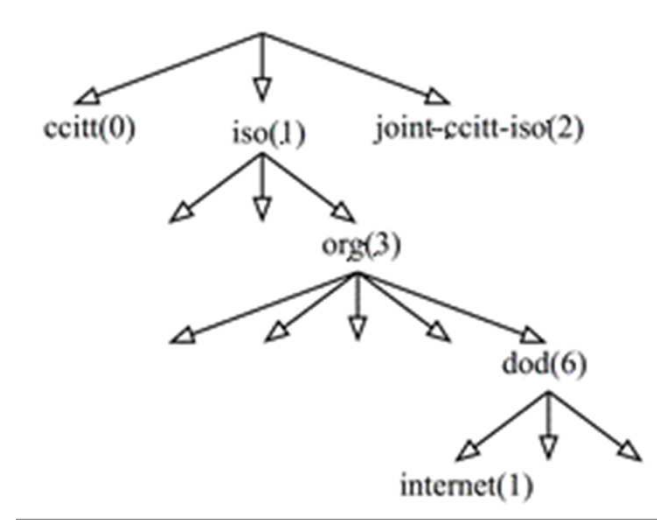
- Mechanism for remote management and monitoring of network devices (routers, bridges, servers, etc.)
- Key idea: all operations done by manipulating **values** of **variables**
 - Standardized, extensible set of variables, organized as a hierarchical tree
 - Protocol for requesting, returning, setting, and notifying of changes (traps) of values of variables

SNMP Protocol

- Messages use UDP, ports 161 (requests/responses) and 162 (notifications)
- Message types:
 - GetRequest: request values of variables from device
 - GetNextRequest: request value of variable following the one supplied
 - GetResponse: return values
 - SetRequest: instruct device to set values of variables
 - Trap: from device - notify monitor / manager of value change
- Management Information Base stores variables
 - Standardized structure enables general toolkits (net-SNMP, HP OpenView)

How to identify variables in SNMP

- ASN.1 Object identifiers
- Variables identified by globally unique strings of digits
 - ex: 1.3.6.1.4.1.3.5.1.1
 - name space is hierarchical; tree on next slide
 - in above, 1 stands for iso, 3 stands for org, 6 stands for dod, 1 stands for internet, 4 stands for private, etc.



ManageEngine MibBrowser Free Tool

File Edit View Operations Help

Download More Free Tools

Loaded MibModules

- IANAifType-MIB
- RFC1213-MIB
 - org
 - dod
 - internet
 - directory
 - mgmt
 - mib-2
 - system
 - interfaces
 - at
 - ip
 - ipForwarding
 - ipDefaultTTL
 - ipInReceives
 - ipInHdrErrors
 - ipInAddrErrors
 - ipForwDatagrams
 - ipInUnknownProtos
 - ipInDiscards
 - ipInDelivers
 - ipOutRequests
 - ipOutDiscards
 - ipOutNoRoutes
 - ipReasmTimeout
 - ipReasmReqds
 - ipReasmOKs
 - ipReasmFails
 - ipFragOKs
 - ipFragFails
 - ipFragCreates
 - ipAddrTable
 - ipRouteTable
 - ipNetToMediaTable
 - ipRoutingDiscards
 - icmp

Host: localhost Port: 161

Community: ***** Write Community:

Set Value:

Object ID: .iso.org.dod.internet.mgmt.mib-2.ip.ipInDiscards

Loading MIBs .\mibs\RFC1213-MIB .\mibs\IF-MIB
MIB(s) Loaded Successfully..

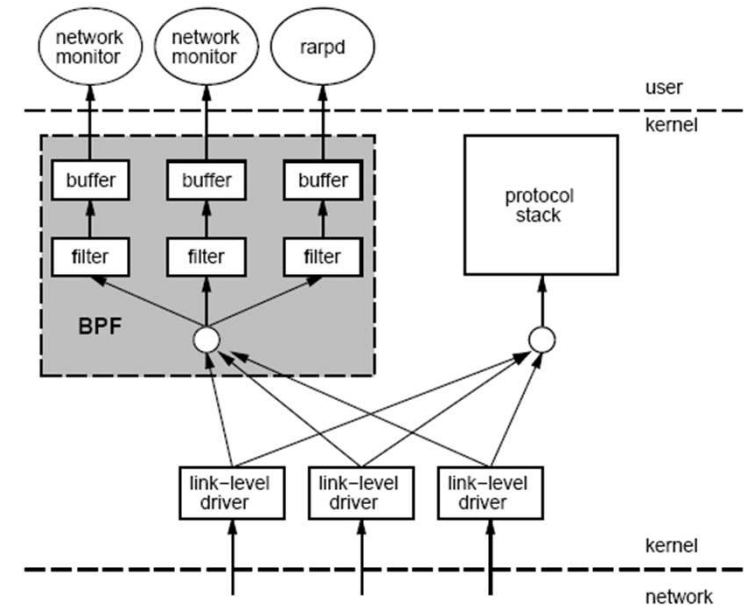
Description: MultiVar

Syntax	Counter	Status	mandatory
Access	read-only	Reference	
Index			
Object ID	.1.3.6.1.2.1.4.8		
Description	were encountered to prevent their continued processing, which were discarded (e.g., for lack of buffer space). Note that this counter does not include any datagrams discard		

Global View

Packet Capture: tcpdump/bpf

- Put interface in promiscuous mode
- Use bpf to extract packets of interest



Accuracy Issues

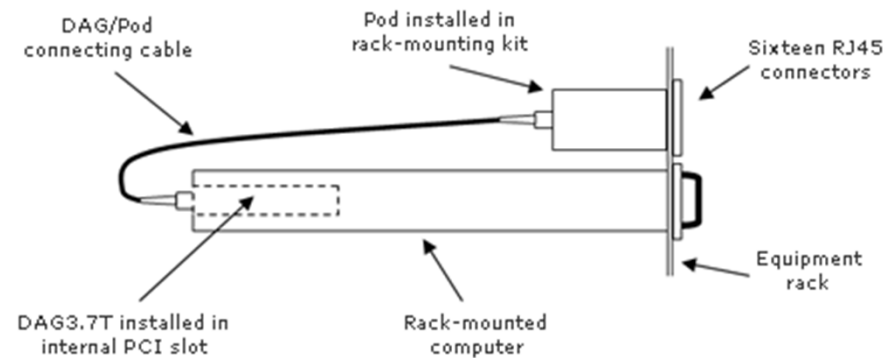
- Packets may be dropped by filter
 - Failure of tcpdump to keep up with filter
 - Failure of filter to keep up with dump speeds

Question: How to recover lost information from packet drops?

Packet Capture on High-Speed Links

Example: Endace OC3Mon

- Rack-mounted PC
- Optical splitter
- Data Acquisition and Generation (DAG) card



Traffic Flow Statistics

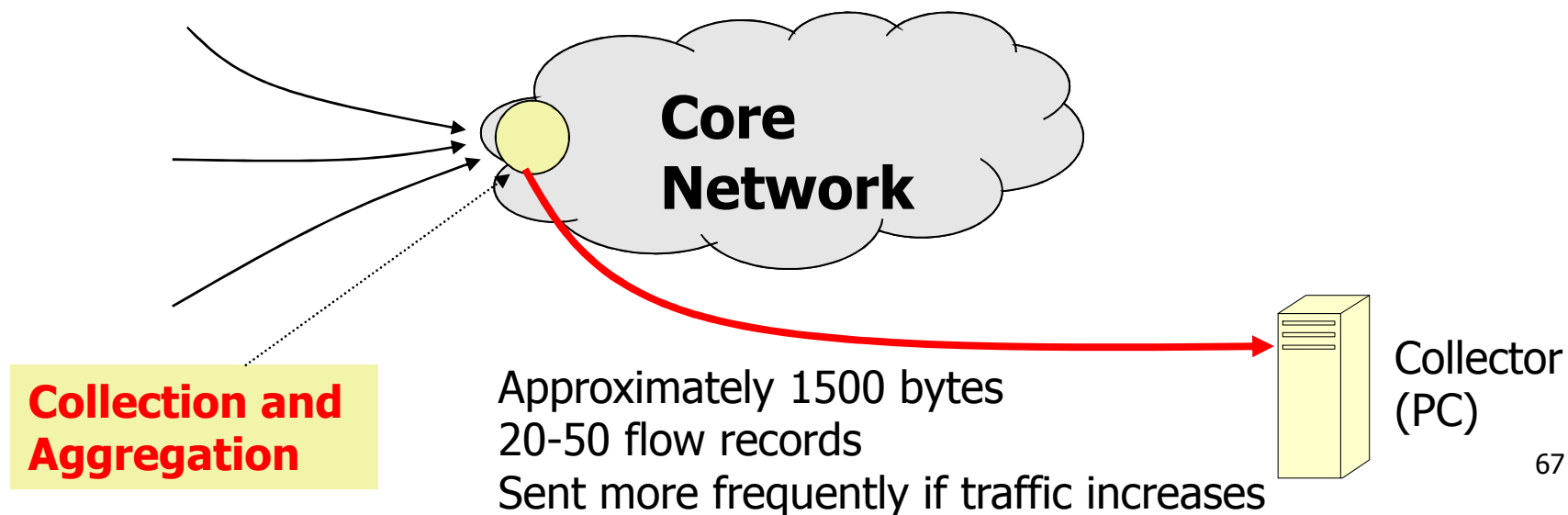
- *Flow monitoring* (e.g., Cisco Netflow)
 - Statistics about groups of related packets (e.g., same IP/TCP headers and close in time)
 - Recording header information, counts, and time
- More detail than SNMP, less overhead than packet capture
 - Typically implemented directly on line card₆₅

What is a flow?

- **Source IP address**
- **Destination IP address**
- **Source port**
- **Destination port**
- **Layer 3 protocol type**
- TOS byte (DSCP)
- Input logical interface (ifIndex)

Cisco Netflow

- Basic output: “Flow record”
 - Most common version is v5
 - Latest version is v10 (RFC 3917)
- Current version (10) is being standardized in the IETF (*template-based*)
 - More flexible record format
 - Much easier to add new flow record types



Flow Record Contents

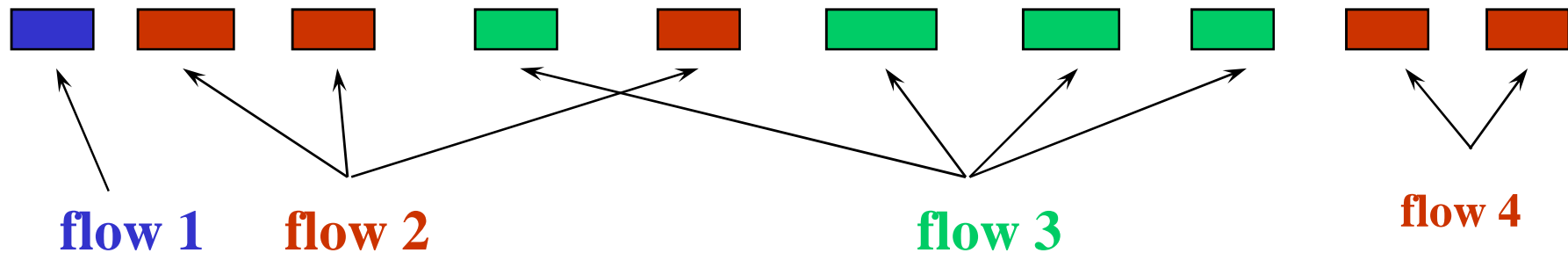
Basic information about the flow...

- Source and Destination, IP address and port
- Packet and byte counts
- Start and end times
- ToS, TCP flags

...plus, information related to routing

- Next-hop IP address
- Source and destination AS
- Source and destination prefix

Aggregating Packets into Flows



- **Criteria 1:** Set of packets that “belong together”
 - Source/destination IP addresses and port numbers
 - Same protocol, ToS bits, ...
 - Same input/output interfaces at a router (if known)
- **Criteria 2:** Packets that are “close” together in time
 - Maximum inter-packet spacing (e.g., 15 sec, 30 sec)
 - **Example:** flows 2 and 4 are different flows due to time

Netflow Processing

1. Create and update flows in NetFlow Cache

SrcIrf	SrcIPadd	DstIrf	DstIPadd	Protocol	TOS	Flgs	Pkts	SrcPort	SrcMsk	SrcAS	DstPort	DstMsk	DstAS	NextHop	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	00A1	/24	180	00A1	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

2. Expiration

- Inactive timer expired (15 sec is default)
- Active timer expired (30 min (1800 sec) is default)
- NetFlow cache is full (oldest flows are expired)
- RST or FIN TCP Flag

SrcIrf	SrcIPadd	DstIrf	DstIPadd	Protocol	TOS	Flgs	Pkts	SrcPort	SrcMsk	SrcAS	DstPort	DstMsk	DstAS	NextHop	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

3. Aggregation?



e.g. Protocol-Port Aggregation Scheme becomes

Protocol	Pkts	SrcPort	DstPort	Bytes/Pkt
11	11000	00A2	00A2	1528

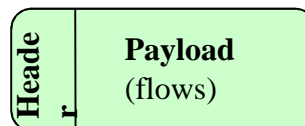
4. Export Version

Non-Aggregated Flows – export **Version 5 or 9**

Aggregated Flows – export **Version 8 or 9**

5. Transport Protocol

Export
Packet

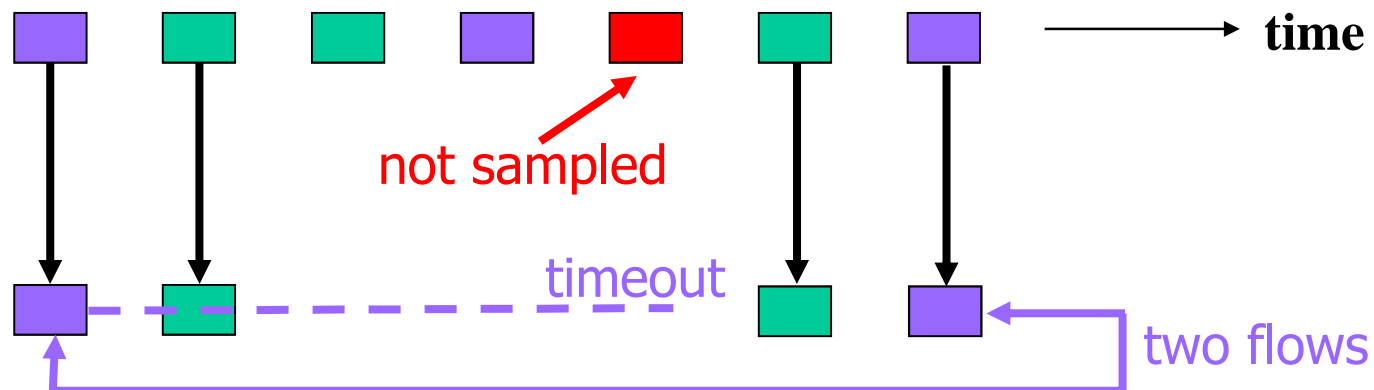


Reducing Measurement Overhead

- **Filtering:** on interface
 - destination prefix for a customer
 - port number for an application (e.g., 80 for Web)
- **Sampling:** before insertion into flow cache
 - Random, deterministic, or hash-based sampling
 - 1-out-of-n or stratified based on packet/flow size
 - *Two types:* packet-level and flow-level
- **Aggregation:** after cache eviction
 - packets/flows with same next-hop AS
 - packets/flows destined to a particular service

Packet Sampling

- Packet sampling before flow creation (Sampled Netflow)
 - 1-out-of-m sampling of individual packets (*e.g.*, $m=100$)
 - Create of flow records over the sampled packets
- Reducing overhead
 - Avoid per-packet overhead on $(m-1)/m$ packets
 - Avoid creating records for a large number of **small flows**
- Increasing overhead (in some cases)
 - May split some **long transfers** into multiple flow records
 - ... due to larger time gaps between successive packets



Problems with Packet Sampling

- Determining size of original flows is tricky
 - For a flow originally of size n , the size of the *sampled* flow follows a binomial distribution
 - Extrapolation can result in big errors
 - Much research in reducing such errors (upcoming lectures)
- Flow records can be lost
- Small flows may be eradicated entirely

Sampling: Flow-Level Sampling

- Sampling of flow records evicted from flow cache
 - When evicting flows from table or when analyzing flows
- Stratified sampling to put weight on “heavy” flows
 - Select all long flows and sample the short flows
- Reduces the number of flow records
 - Still measures the vast majority of the traffic

Flow 1, 40 bytes

← sample with 0.1% probability

Flow 2, 15580 bytes

Flow 3, 8196 bytes

Flow 4, 5350789 bytes

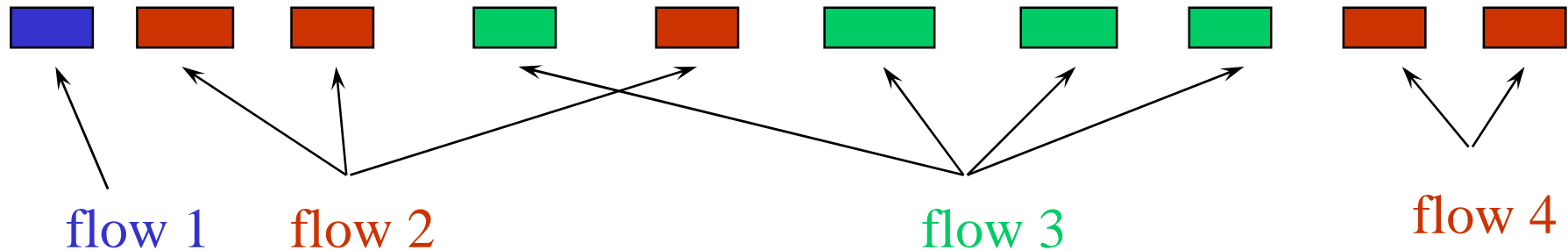
← sample with 100% probability

Flow 5, 532 bytes

Flow 6, 7432 bytes

← sample with 10% probability⁷⁴

Flow Measurement

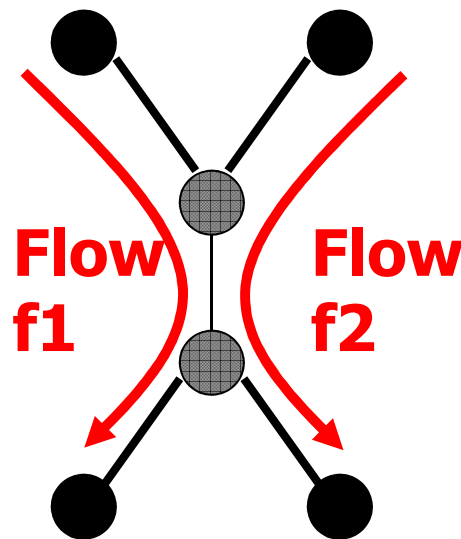


- IP flow abstraction
 - Set of packets with “same” src and dest IP addresses
 - Packets that are “close” together in time (a few seconds)
- Cisco NetFlow
 - Router maintains a cache of statistics about active flows
 - Router exports a measurement record for each flow

Inferring the Path Matrix from the Traffic Matrix

Shared bottleneck detection

- Do two network paths share a common bottleneck (congested link)?
- Hard to figure out if you don't control the topology
- Trick: look for correlation in sending patterns (loss, delay) across the two paths



Types of Measurement: Applications

Where to get application data?

- Web server logs
 - Host, time, URL, response code, content length, ...
 - E.g.,

```
122.345.131.2 - - [15/Oct/1998:00:00:25 -0400]
"GET /images/wwwtlogo.gif HTTP/1.0" 304 -
"http://www.aflcio.org/home.htm" "Mozilla/2.0
(compatible; MSIE 3.02; Update a; AK; AOL 4.0;
Windows 95)" "-"
```
- DNS logs
 - Request, response, time
- Useful for workload characterization, troubleshooting, etc.



-----things to cover

- PASTA principle
- Shared bottlenecks (machiraju)
- Measuring bandwidth (both capacity and available)

Lecture outline

- Background: analysis and modeling (3.6)
- Measurement
 - Infrastructure, Traffic, Applications
- Challenges issues in Internet measurement (4)
 - Instrumentation, processing and capturing issues, databases,
 - Anonymization (8)

Analysis and modeling

Outline

- How ping works (router stack?)
- How traceroute works
- Measuring asymmetric bandwidth/latency