BGP

CS 168 – Spring 2024

Interdomain Routing

- **Inter**domain routing is between autonomous systems (AS)
    - Similar goals as intradomain routing with scalability + policy compliance
    - Autonomous systems want privacy and autonomy
- Border gateway protocol (BGP) is current design
    - Extends on top of DV (with some crucial differences)

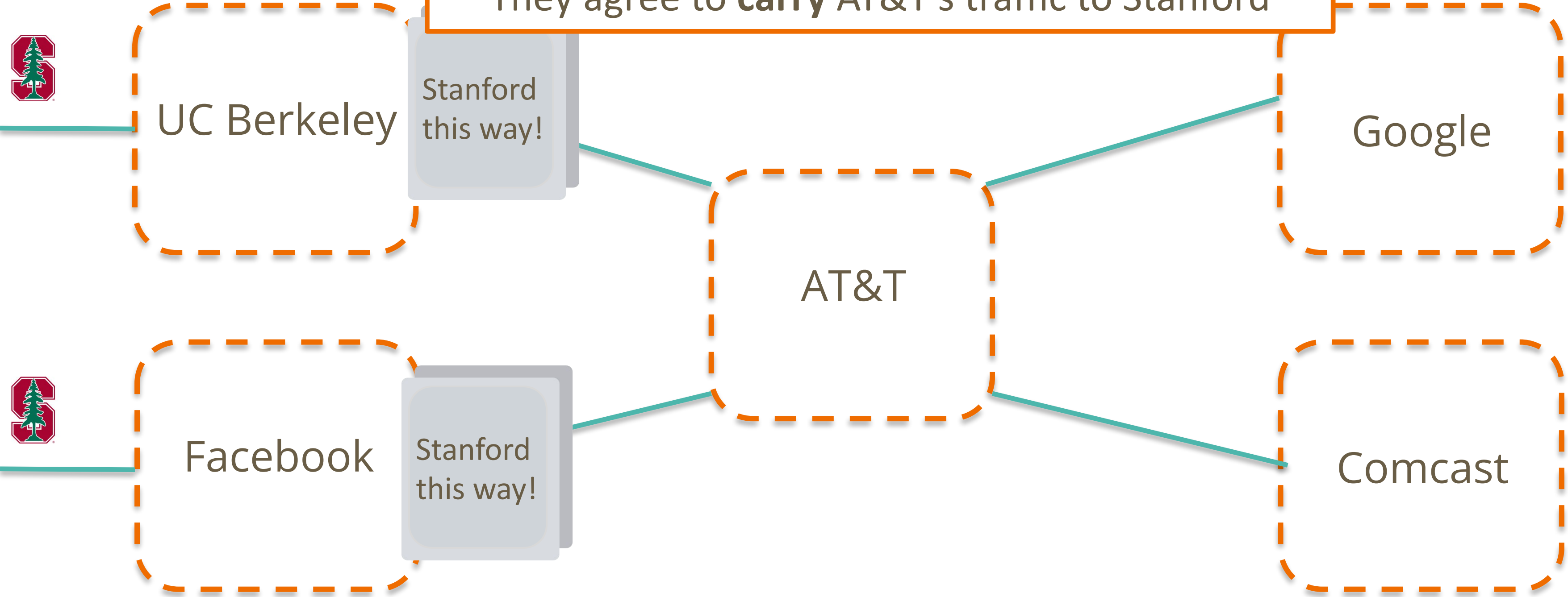
Export & Selection

- If you are an AS:
  - Route Selection
    - **Where you send your packets**
    - Determine how to choose a valid route to a given IP prefix, when multiple paths through ASes
  - Route Export
    - **Which ASes will receive your route**
    - Other ASes will *select* your route and **send traffic to you**

Export & Selection

UCB and Facebook **export** their route to Stanford

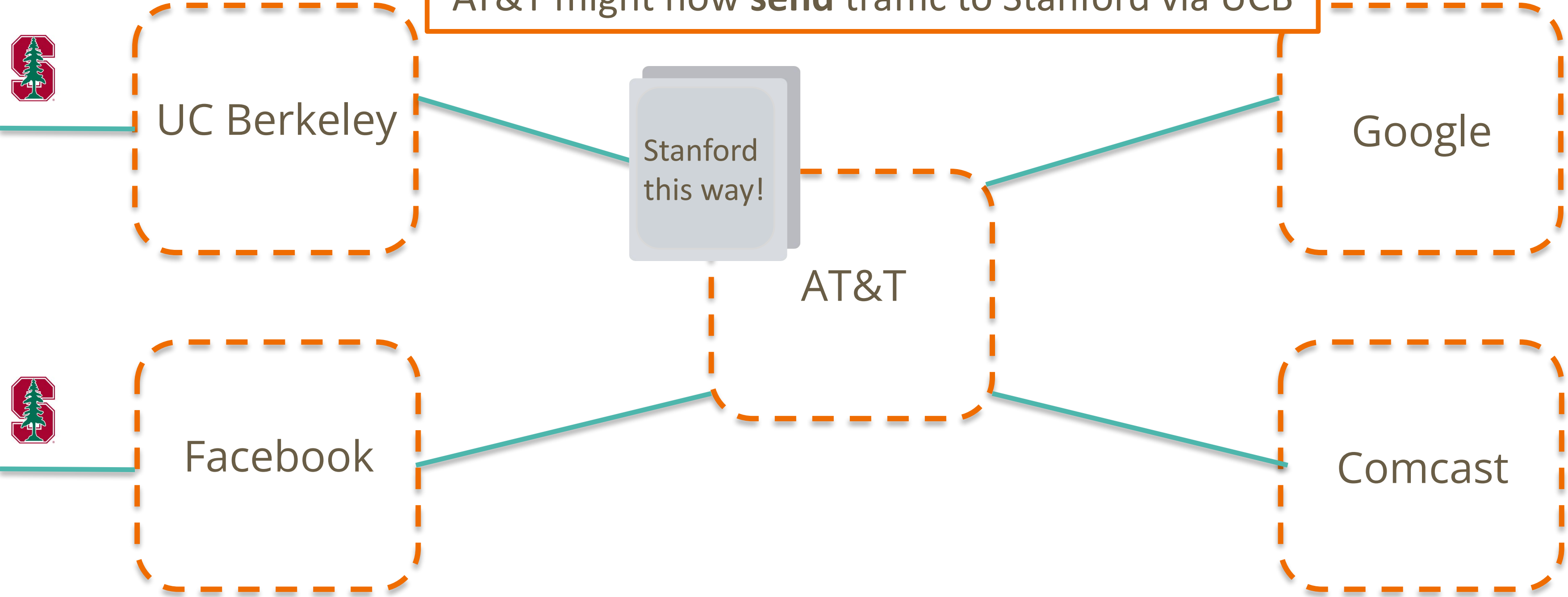
They agree to **carry** AT&T's traffic to Stanford



Export & Selection

AT&T **selects** route through UC Berkeley

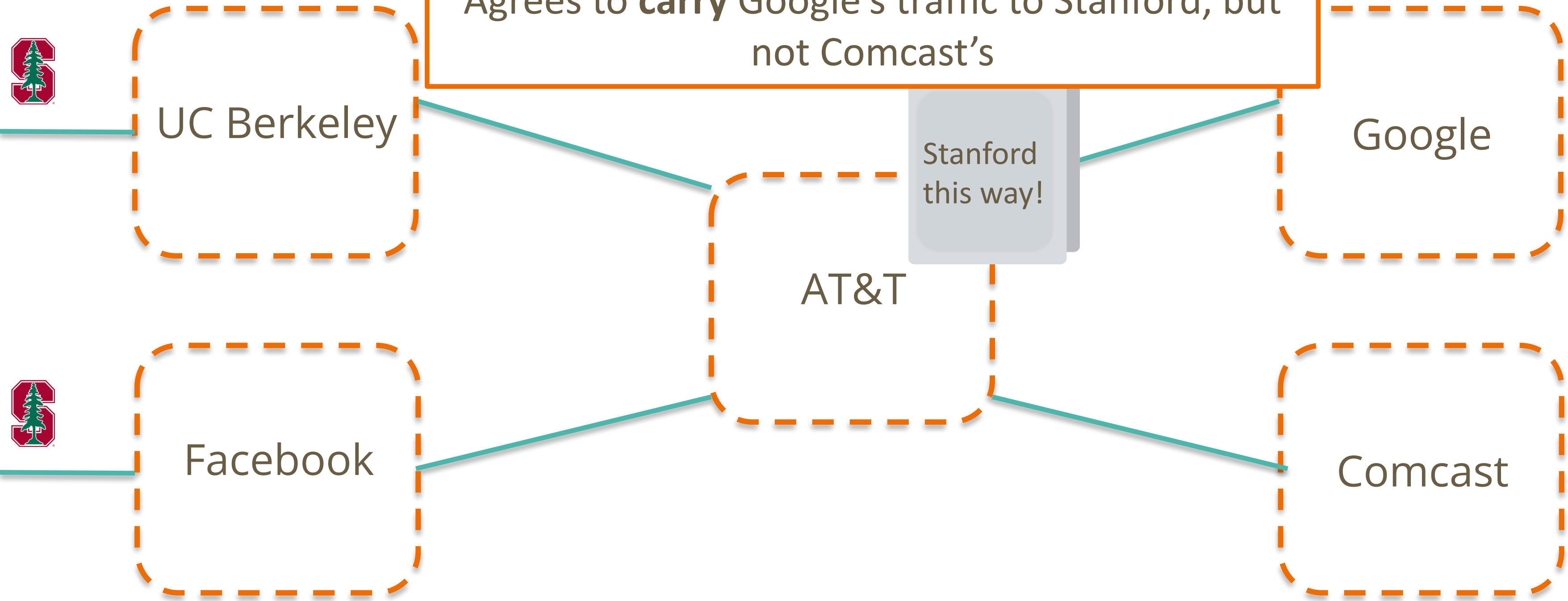
AT&T might now **send** traffic to Stanford via UCB



Export & Selection

AT&T **exports** route to Google only

Agrees to **carry** Google's traffic to Stanford, but not Comcast's



Types of ASes (domains)

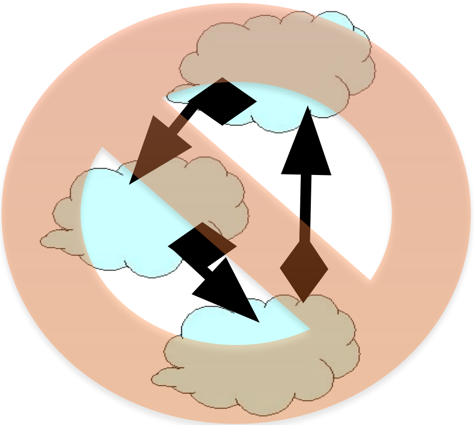
- **Stub**: only sends/receives traffic for its users
  - companies, universities, etc.
- **Transit**: carries traffic for other ASes
  - Global ISPs (Tier 1): fully connected mesh
  - Regional ISPs (Tier 2)
  - Local ISPs (Tier 3)
- Lower tiers buy service from higher tiers
- What's the relationship between AS and ISP?
  - All ISPs are ASes, but not all ASes are ISPs
  - E.g. UC Berkeley is not an ISP but it is an AS

Business Relationship among ASes

- Two ASes will **connect** only if they have business relationship:
  - **Customer-Provider**
    - *Provider* B carries *customer* A's traffic **for a fee**
  - **Peers**
    - *Peers* A, B carry each other's traffic **for free**
- What roles can a global ISP (Tier 1) have?
  - Provider to Tier 2 or Tier 3
  - Peer to other global ISP (tier 1)
  - Not a customer!

Business Relationship Restrictions

- The graph of **peering** relations can be ***cyclic***
  - The peer of my peer can also be my peer
  - For example, global ISPs all peer with each other
- The graph of **customer-provider** relations must be ***acyclic***



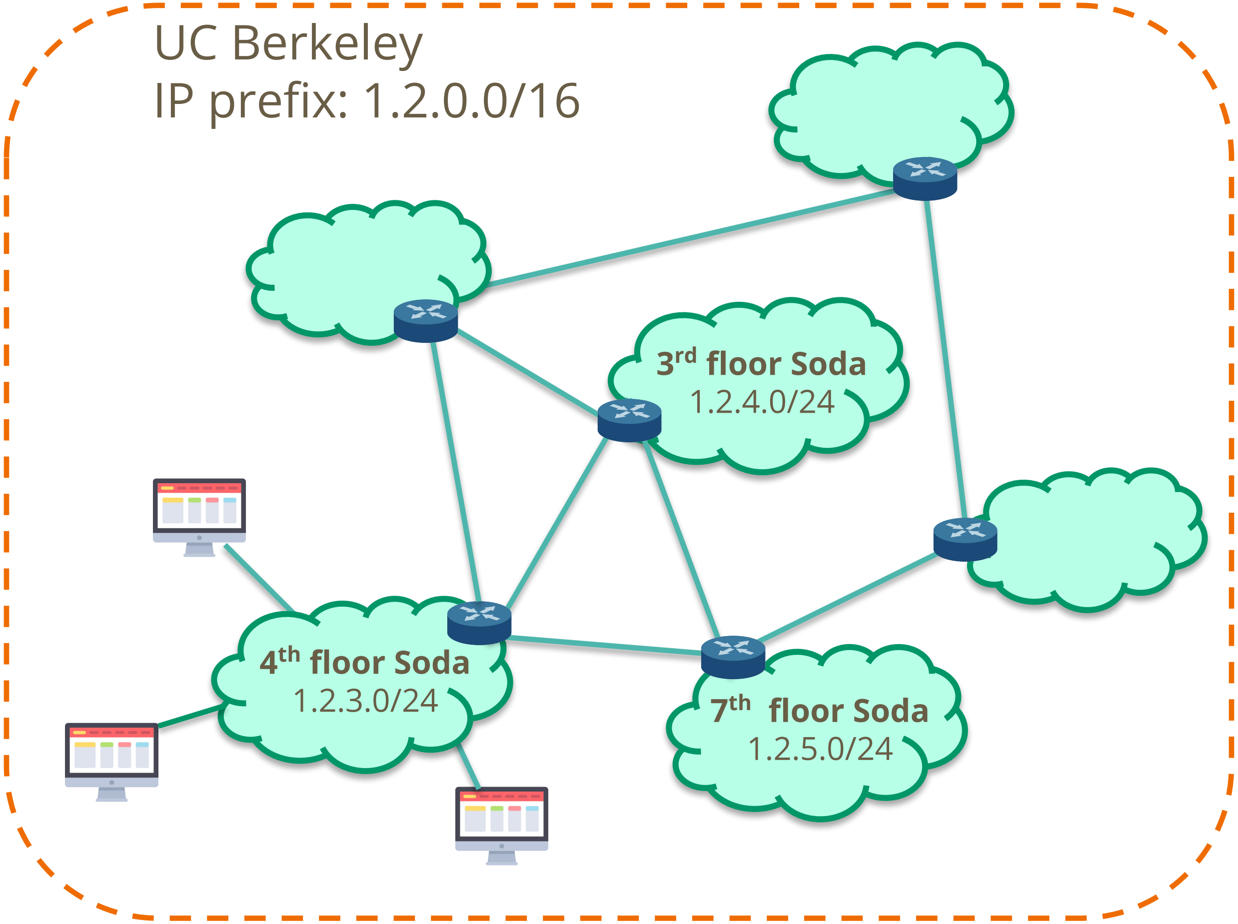
The Big Picture

How does this fit with what we’ve learned so far?

Three parts of Gateway Protocols

- **eBGP**
  - Between border routers in **different ASes**
  - Learn about **external routes**
- **iBGP**
  - Between border routers and other routers **within a single AS**
  - Learn **which border router** to use to reach external destinations
- **IGP**
  - The **protocol** used for **intradomain** routing (e.g. OSPF).
    - Shortest path to **subnet in the same AS**
    - Shortest path to **border router** for given external network
  - Just a different name for L3 routing as we've talked about earlier

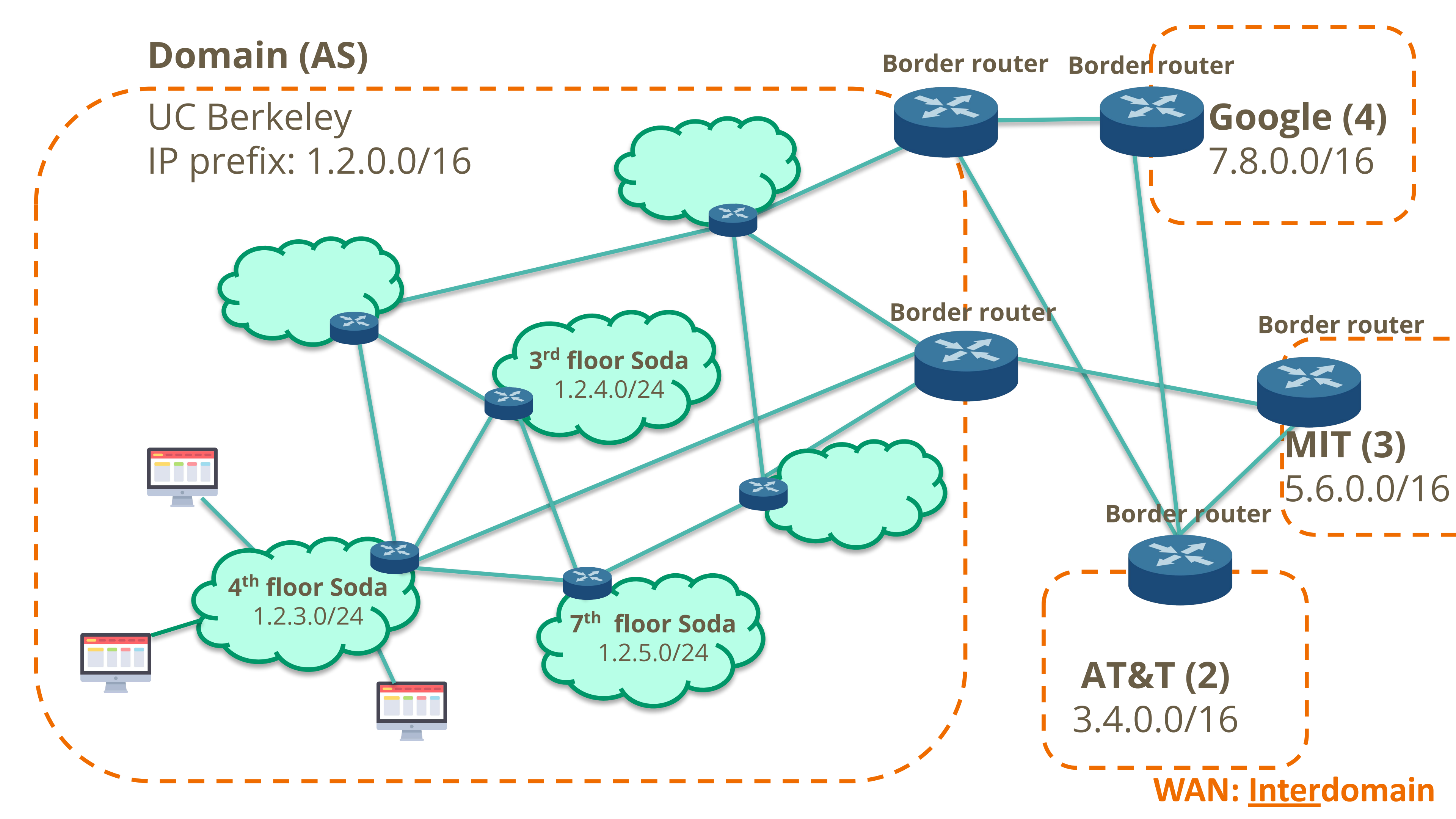
Domain (AS)

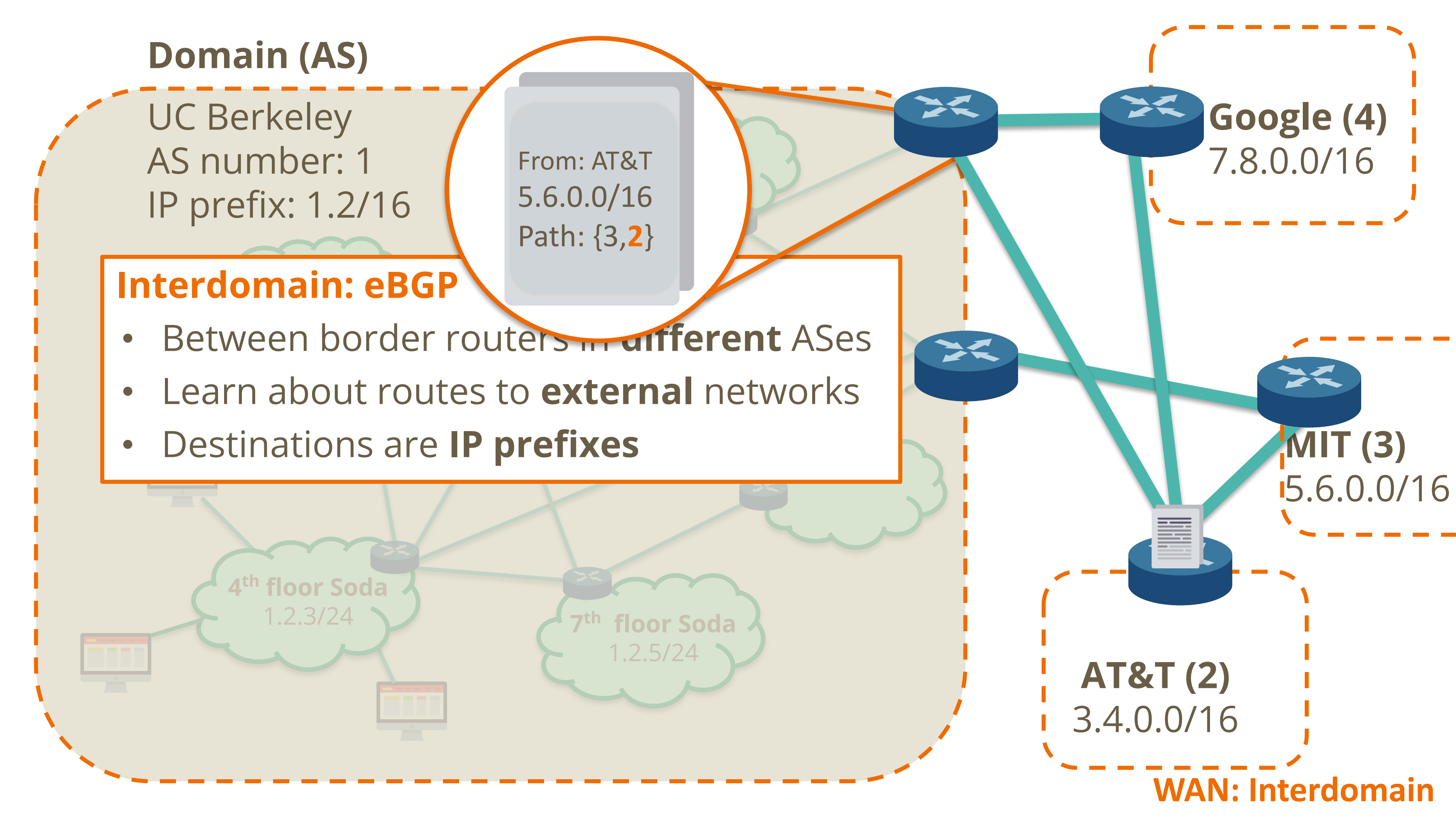


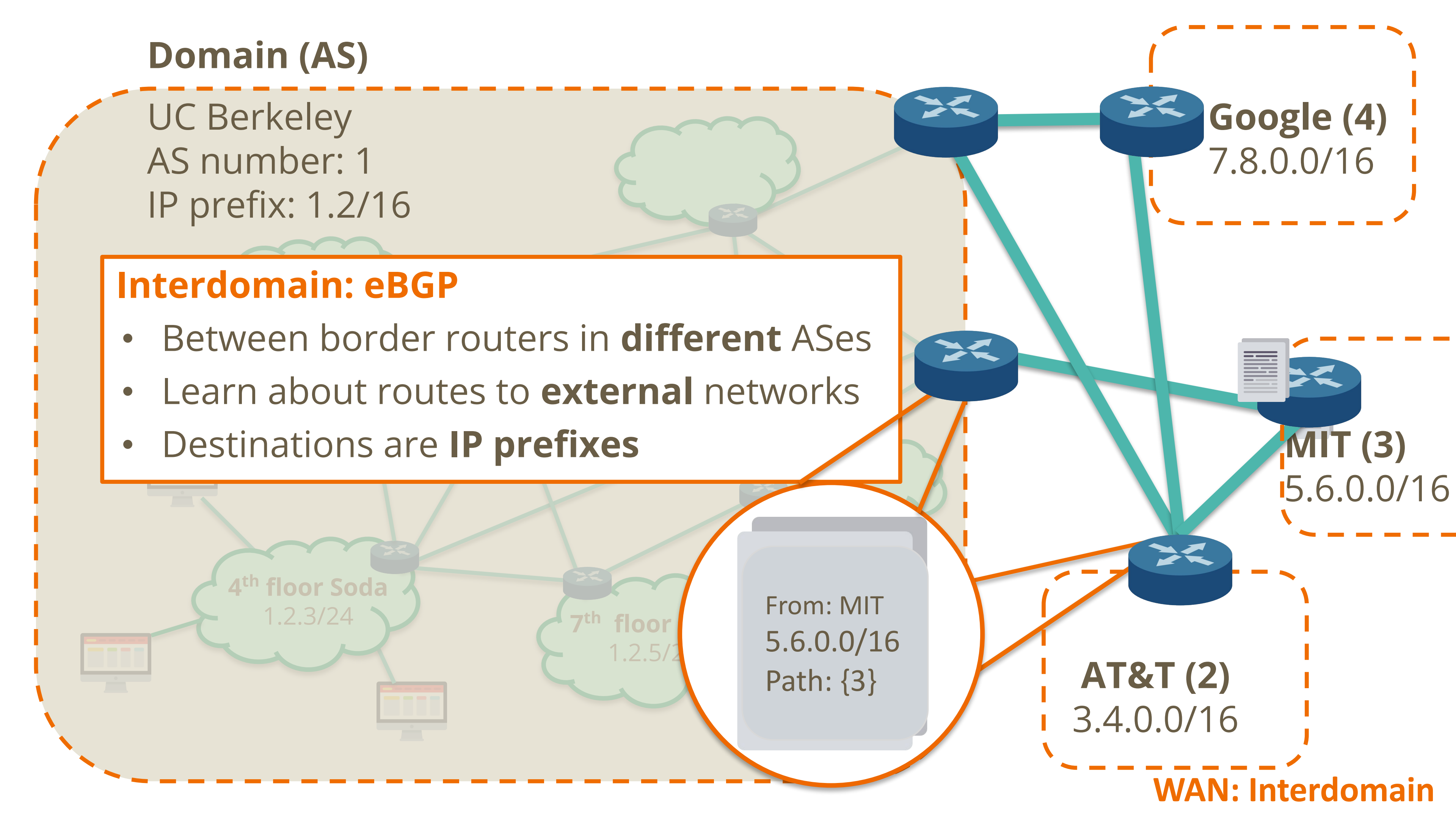
L3: Intradomain

- Destinations are **IP addresses**
- **IGP**: exchange info about paths to local destinations
  - DV, LS, etc.

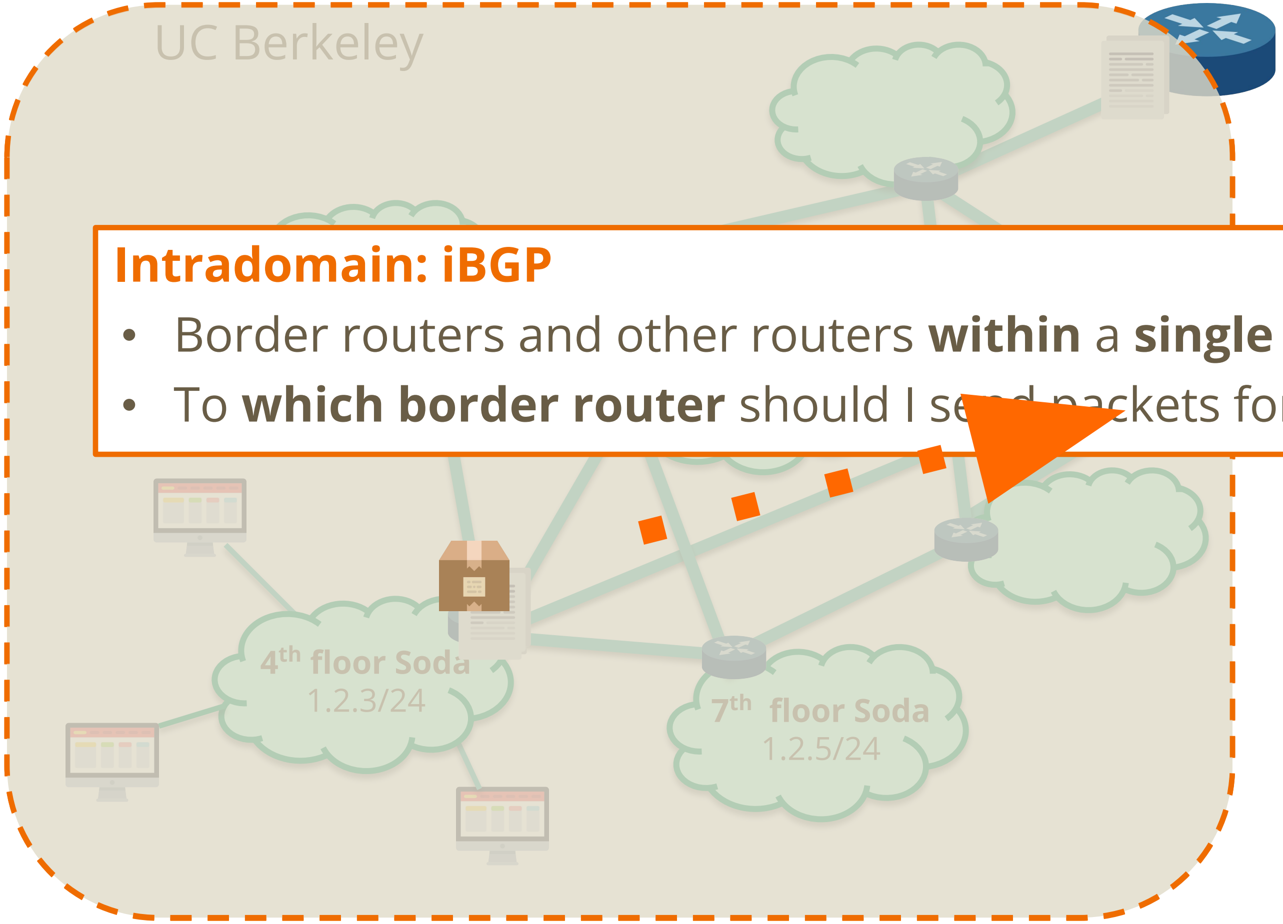
LAN: Intradomain


Domain (AS)
UC Berkeley
IP prefix: 1.2.0.0/16
Border router
Border router
Google (4)
7.8.0.0/16
3rd floor Soda
1.2.4.0/24
Border router
Border router
MIT (3)
5.6.0.0/16
4th floor Soda
1.2.3.0/24
7th floor Soda
1.2.5.0/24
Border router
AT&T (2)
3.4.0.0/16
WAN: Interdomain


Domain (AS)
UC Berkeley
AS number: 1
IP prefix: 1.2/16
From: AT&T
5.6.0.0/16
Path: {3,2}
Interdomain: eBGP
• Between border routers in different ASes
• Learn about routes to external networks
• Destinations are IP prefixes
4th floor Soda
1.2.3/24
7th floor Soda
1.2.5/24
Google (4)
7.8.0.0/16
MIT (3)
5.6.0.0/16
AT&T (2)
3.4.0.0/16
WAN: Interdomain


Domain (AS)
UC Berkeley
AS number: 1
IP prefix: 1.2/16
Interdomain: eBGP
• Between border routers in **different** ASes
• Learn about routes to **external** networks
• Destinations are **IP prefixes**
4th floor Soda
1.2.3/24
7th floor
From: MIT
5.6.0.0/16
Path: {3}
Google (4)
7.8.0.0/16
MIT (3)
5.6.0.0/16
AT&T (2)
3.4.0.0/16
WAN: Interdomain

Domain (AS)



Intradomain: iBGP

- Border routers and other routers **within** a **single AS**
- To **which border router** should I send packets for MIT?

Basic Messages in BGP

- **Open**: establishes BGP session
- **Notification**: report unusual conditions
- **Update**:
  - Format <IP prefix: route attributes>
  - Inform neighbors of new routes (***announcements***)
  - Inform neighbors of old routes that are no longer active (***withdrawal***)
- **Keepalive**:
  - Inform neighbors that this BGP session is still alive

What's this?

BGP Route Attributes

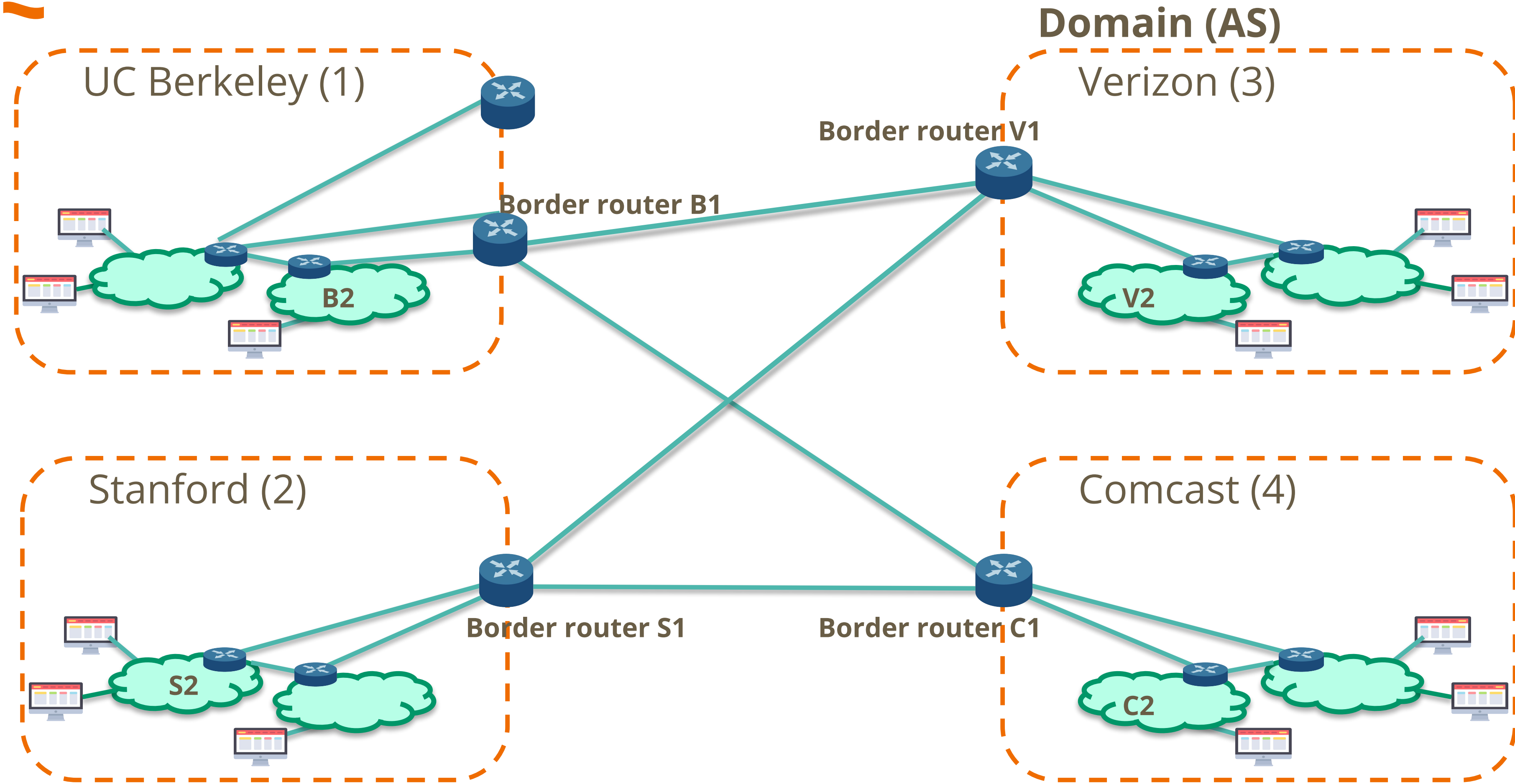
Attributes: Parameters used in route selection

- **Local** attributes
  - ASes keep them private
  - Not included in eBGP route announcements
  - E.g. LOCAL_PREF
- **Nonlocal** attributes:
  - propagated with eBGP route announcements
  - E.g. AS_PATH

Route Selection in Priority Order

Priority	Rule	Remarks
1	LOCAL PREF	Pick **highest** LOCAL PREF
2	ASPATH	Pick **shortest** ASPATH length
3	IGP path	Lowest IGP cost to next hop (egress router)
4	MED	**Lowest** MED preferred
5	Router ID	Smallest next-hop router’s IP address as tie-breaker

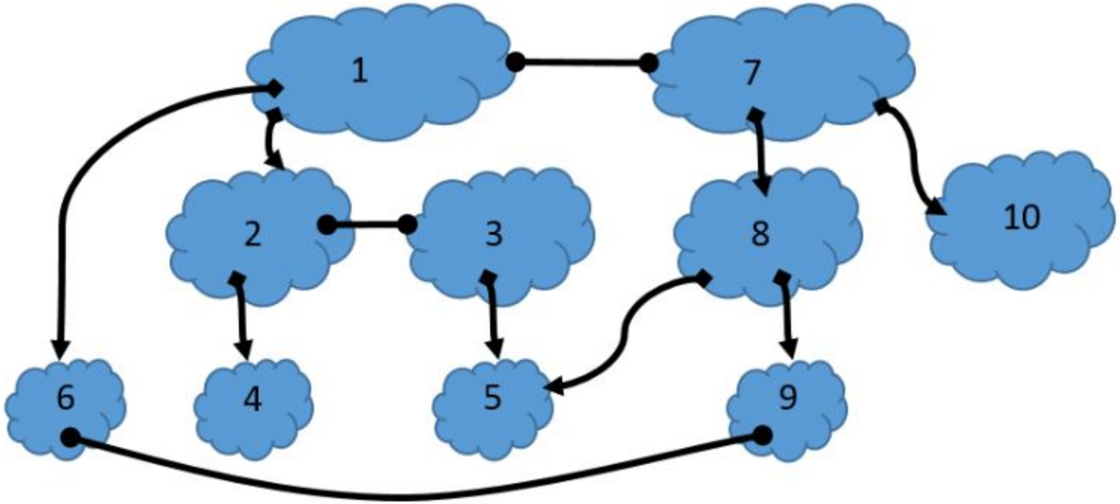
Question 2



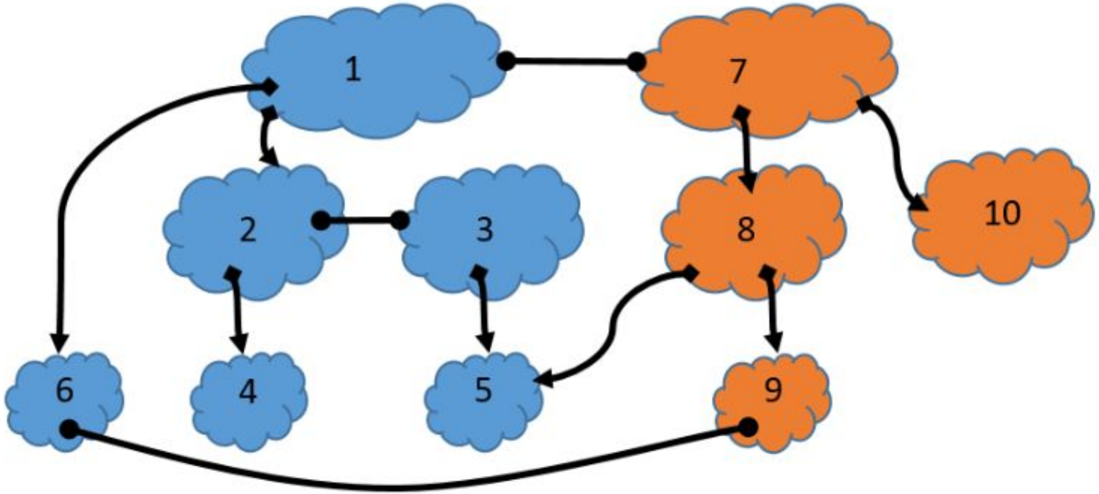
Domain (AS)
UC Berkeley (1)
Verizon (3)
Border router V1
Border router B1
B2
V2
Stanford (2)
Comcast (4)
Border router S1
Border router C1
S2
C2

Worksheet: Q3

Question 3



Question 3

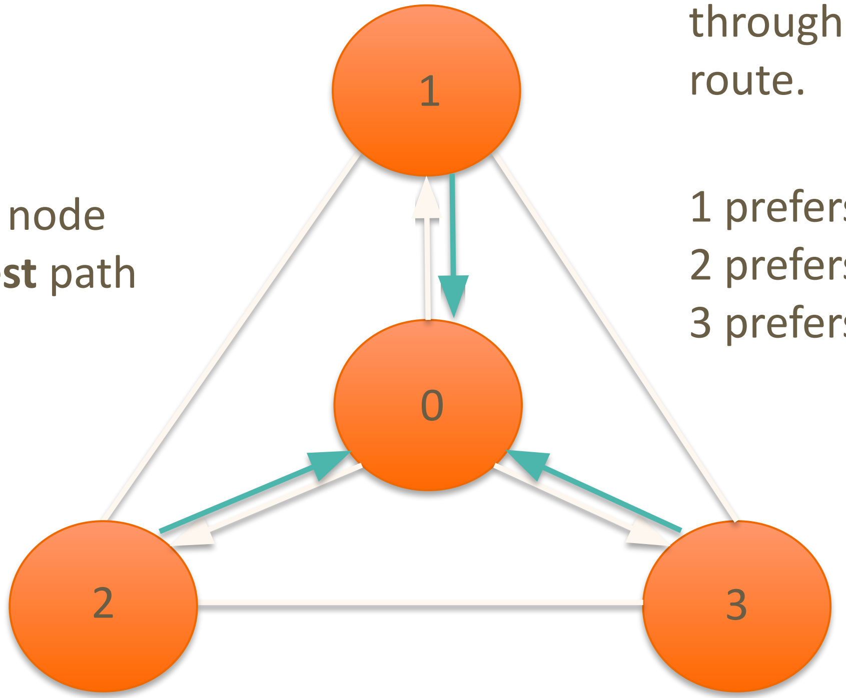


Policy Oscillation

Policy Oscillation

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



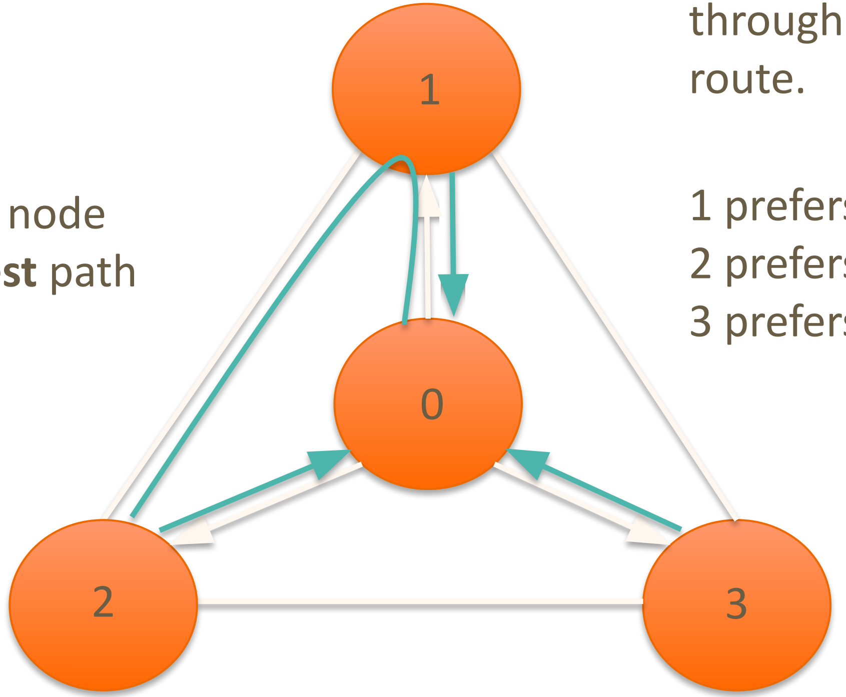
Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

Policy Oscillation 1

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



1 advertises 1⟶0 to 2

Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

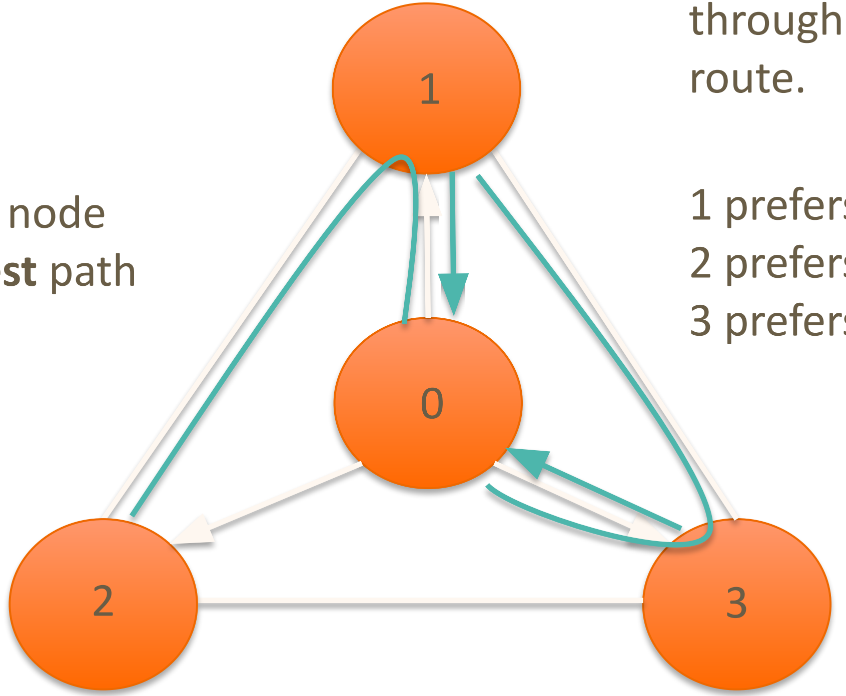
Policy Oscillation 2

Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



3 advertises 3⟶0 to 1

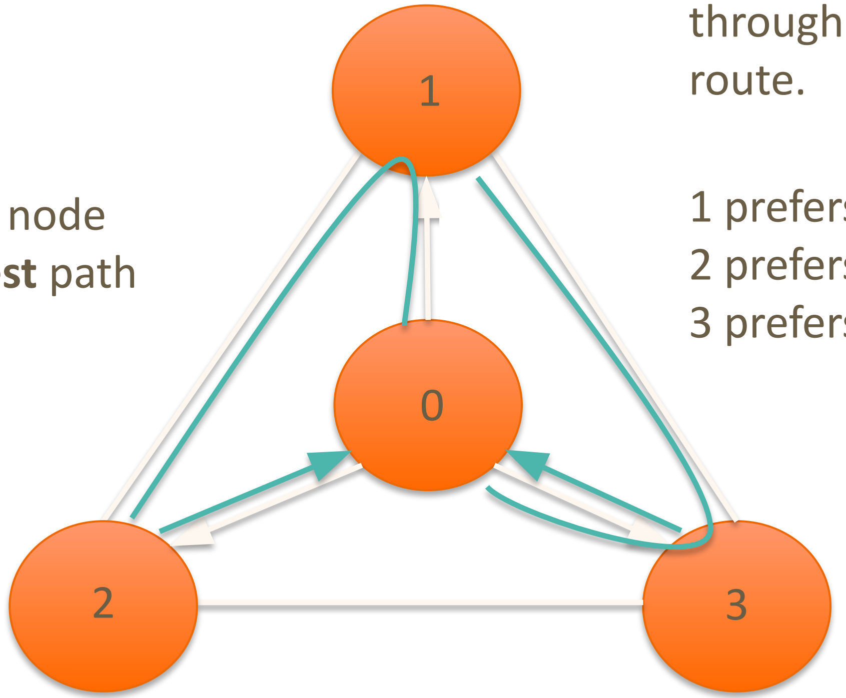
Policy Oscillation 3

Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



1 withdraws its path of 1⟶0 from 2 (because 1 now takes 1->3->0)

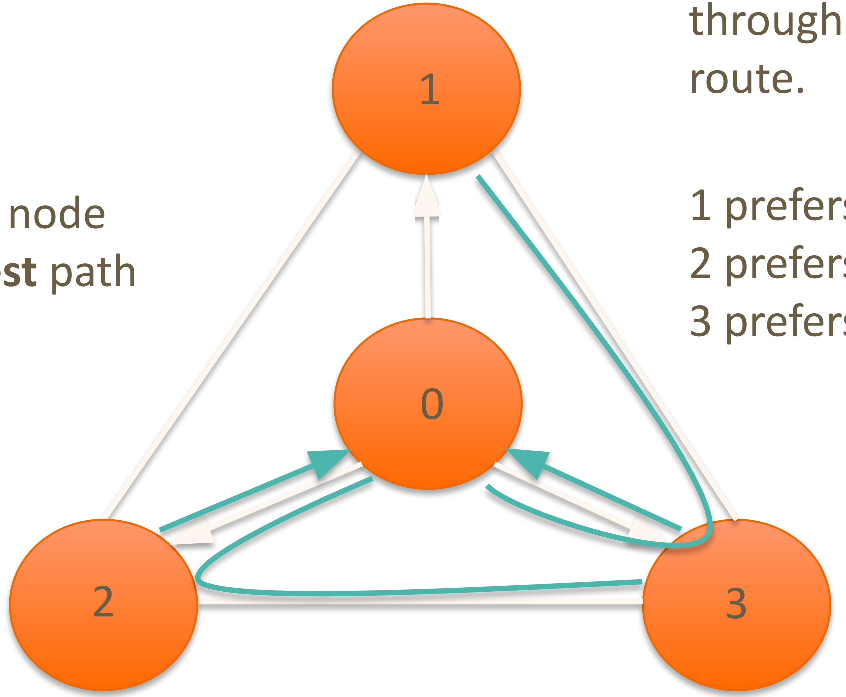
Policy Oscillation 4

Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0

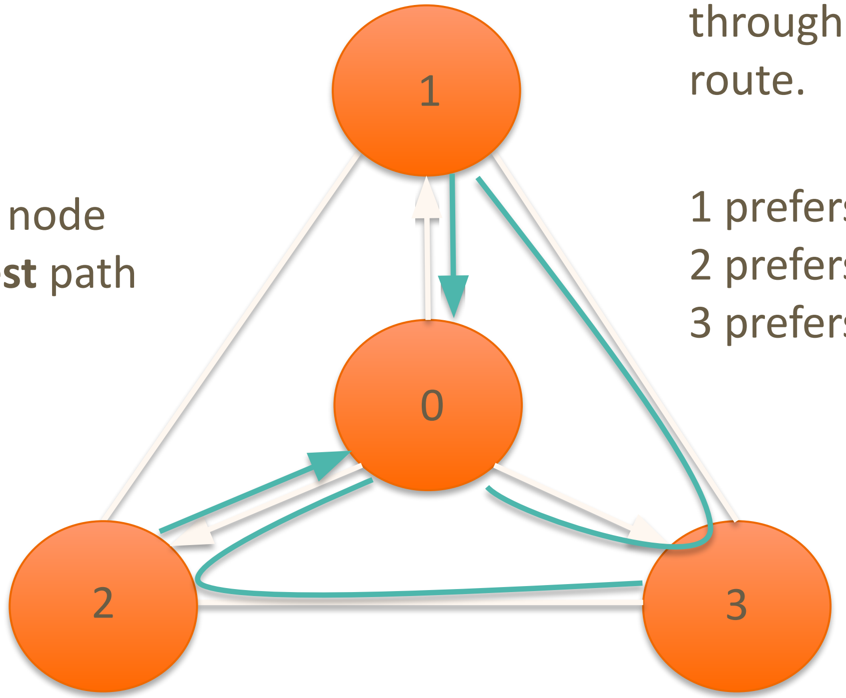


2 now advertises 2⟶0 to 3
(3 would take it as it favors its neighbor)

Policy Oscillation 5

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



Each node **prefers** route through neighbor over direct route.

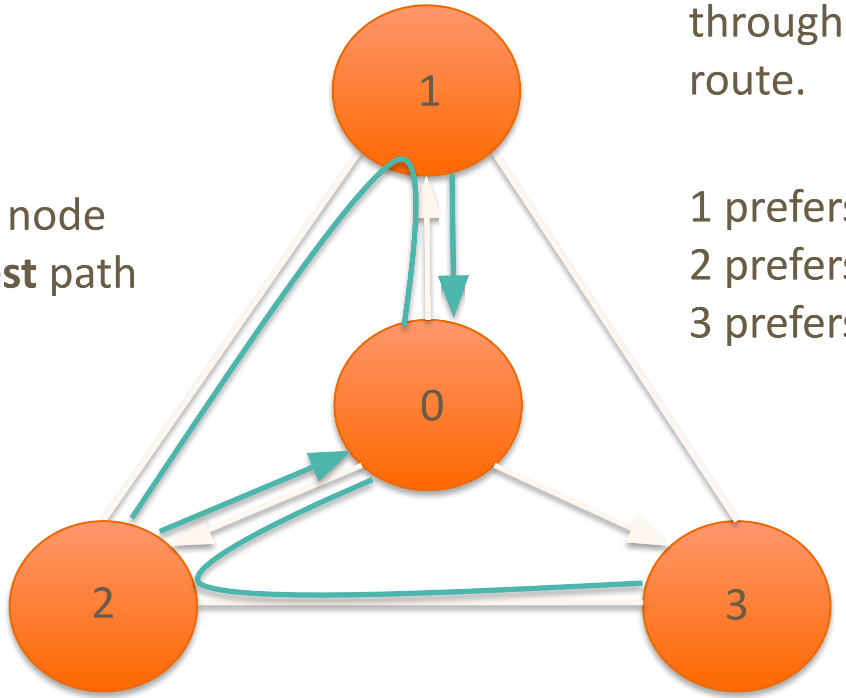
1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

3 now withdraws 3⟶0 from 1

Policy Oscillation 6

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



1 *again* advertises its path 1⟶0

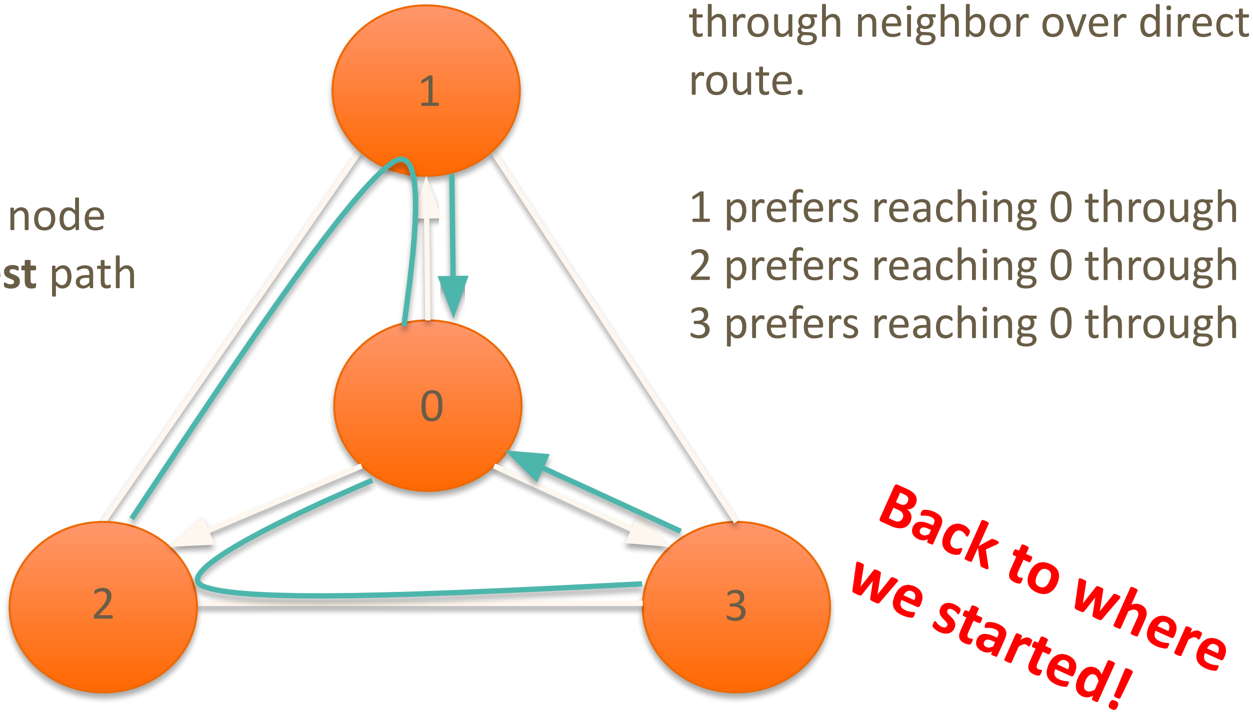
Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

Policy Oscillation 7

Suppose **initially** each node only knows the **shortest** path to 0 (green arrow).

1 knows 1⟶0
2 knows 2⟶0
3 knows 3⟶0



Each node **prefers** route through neighbor over direct route.

1 prefers reaching 0 through 2 or 3
2 prefers reaching 0 through 1 or 3
3 prefers reaching 0 through 1 or 2

2 withdraws its path 2⟶0 from 3

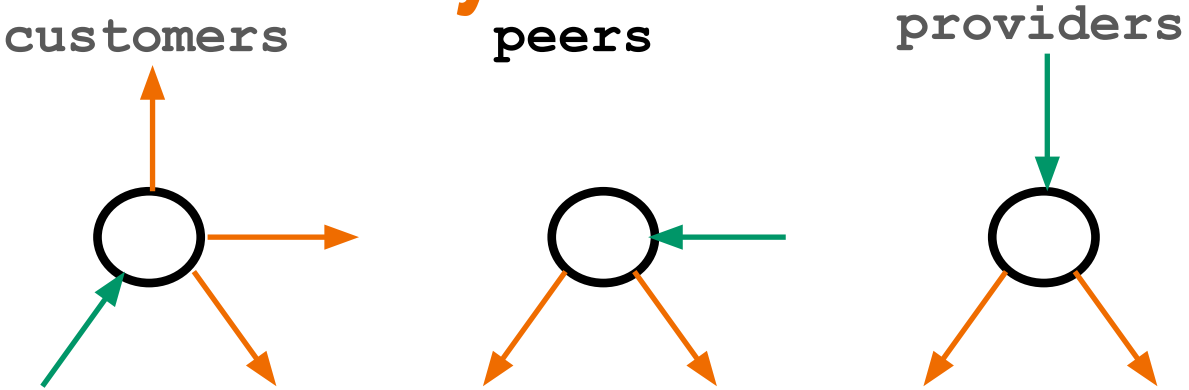
Why doesn't this happen in reality?

Gao-Rexford

Gao-Rexford Policy

Destination prefix advertised by…	Export route to…
Customer	Everyone (providers, peers, other customers)
Peer	Customers
Provider	Customers

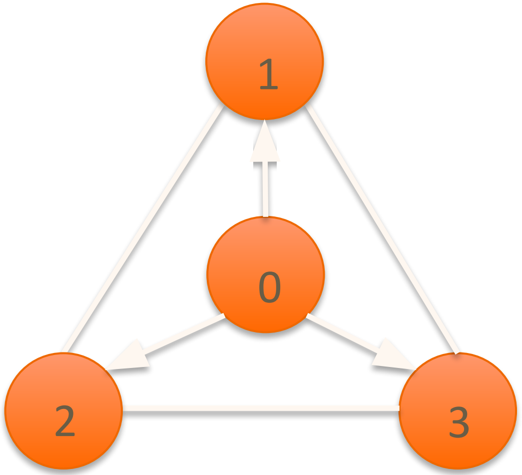
Gao-Rexford Policy Continued



- **Green** arrow is where you learn the route
- **Orange** arrows are where you export the route
- With Gao-Rexford
  - The AS policy graph is a DAG
  - Routes are “valley free”/”single-peaked”

Gao-Rexford avoids Policy Oscillation

- Example shown before did not use Gao-Rexford (why?)
  - 1, 2, and 3 are **peers**
  - 0 is the **provider** to 1, 2, and 3
  - Peers ***don't*** advertise route learned from providers to each other
    - i.e. 1 would never advertise 1->0 (learned from 1's provider 0) to 2 (1's peer)



Destination prefix advertised by…	Export route to…
Peer	Customers