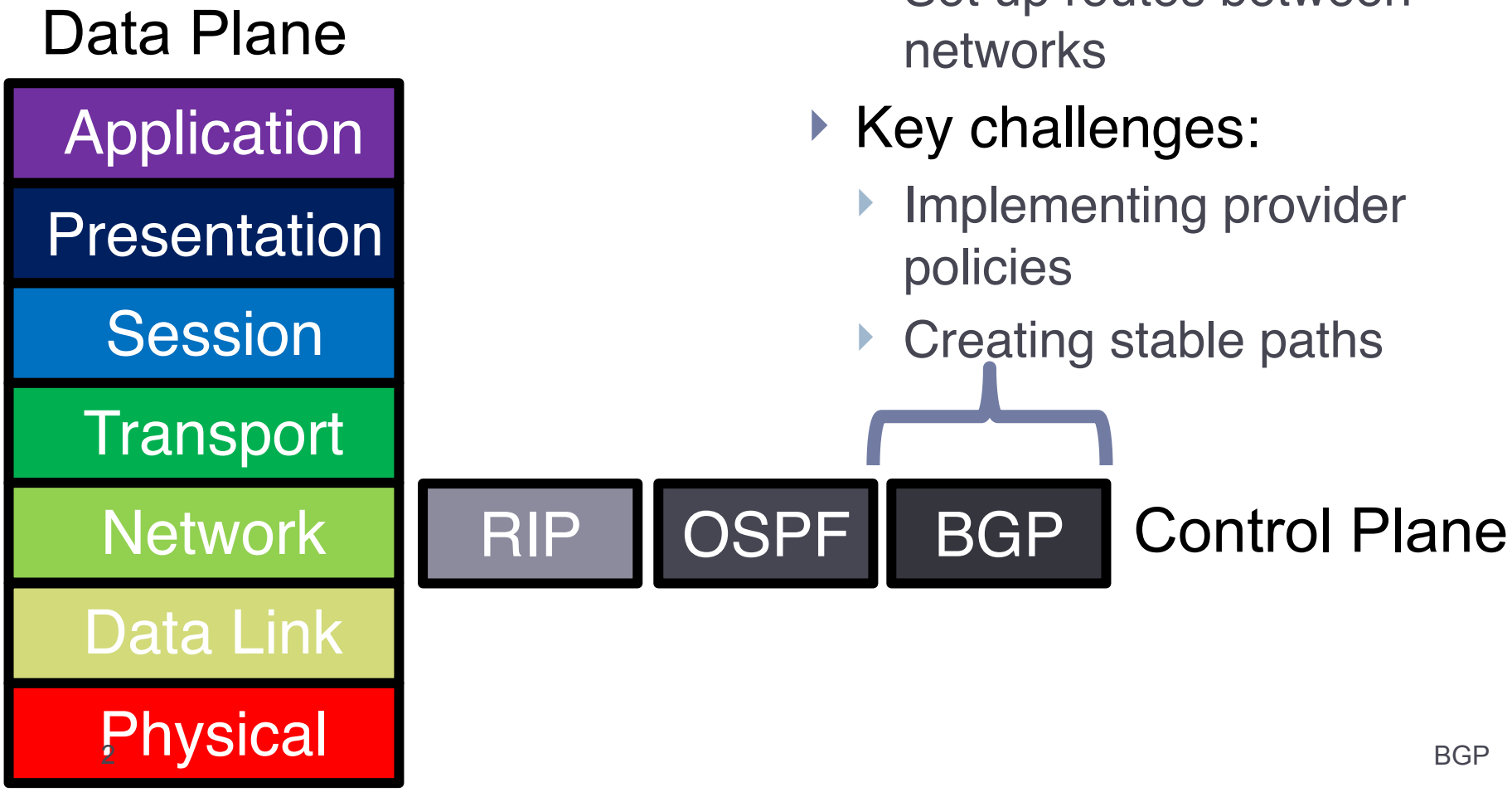
Cristina Nita-Rotaru

CS4700/5700: Network fundamentals

Inter-domain routing.

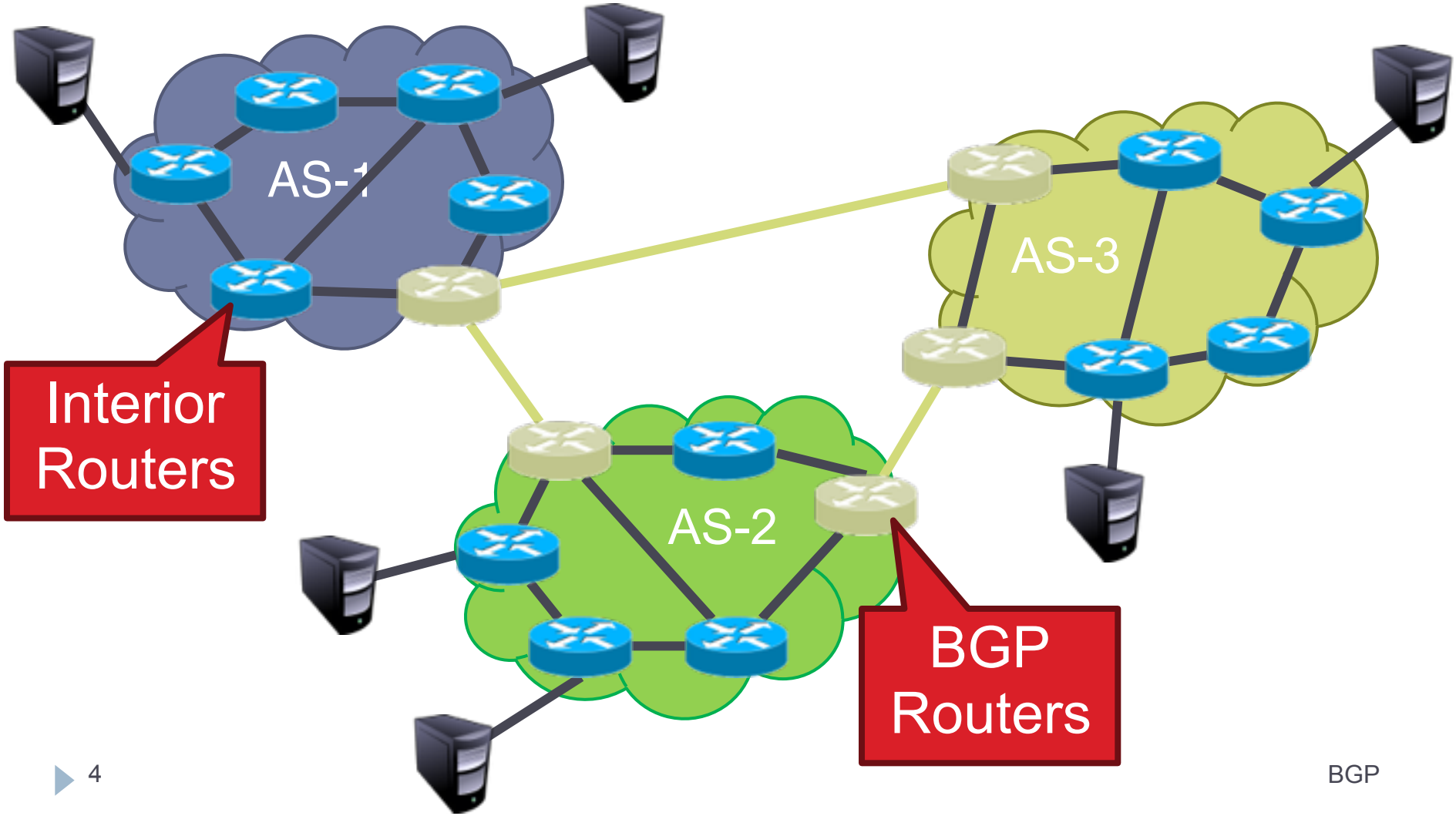
Network Layer, Control Plane

- Function:
  - Set up routes between networks
- Key challenges:
  - Implementing provider policies
  - Creating stable paths



1: BGP

ASs, Revisited



AS Numbers

- Each AS identified by an ASN number
  - 16-bit values (latest protocol supports 32-bit ones)
  - 64512 – 65535 are reserved
- Currently, there are > 20000 ASNs
  - AT&T: 5074, 6341, 7018, …
  - Sprint: 1239, 1240, 6211, 6242, …
  - Northeastern: 156
  - North America ASs → ftp://ftp.arin.net/info/asn.txt

Inter-Domain Routing

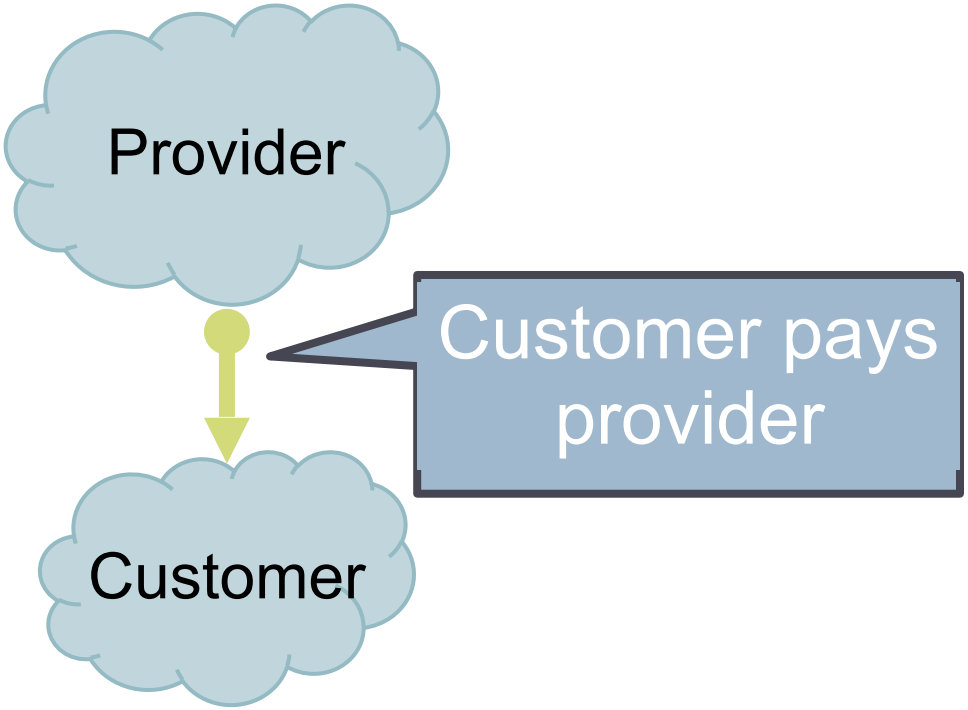
- Global connectivity is at stake!
  - Thus, all ASs must use the same protocol
  - Contrast with intra-domain routing
- What are the requirements?
  - Scalability
  - Flexibility in choosing routes
    - Cost
    - Routing around failures
- Question: link state or distance vector?
  - Trick question: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

BGP

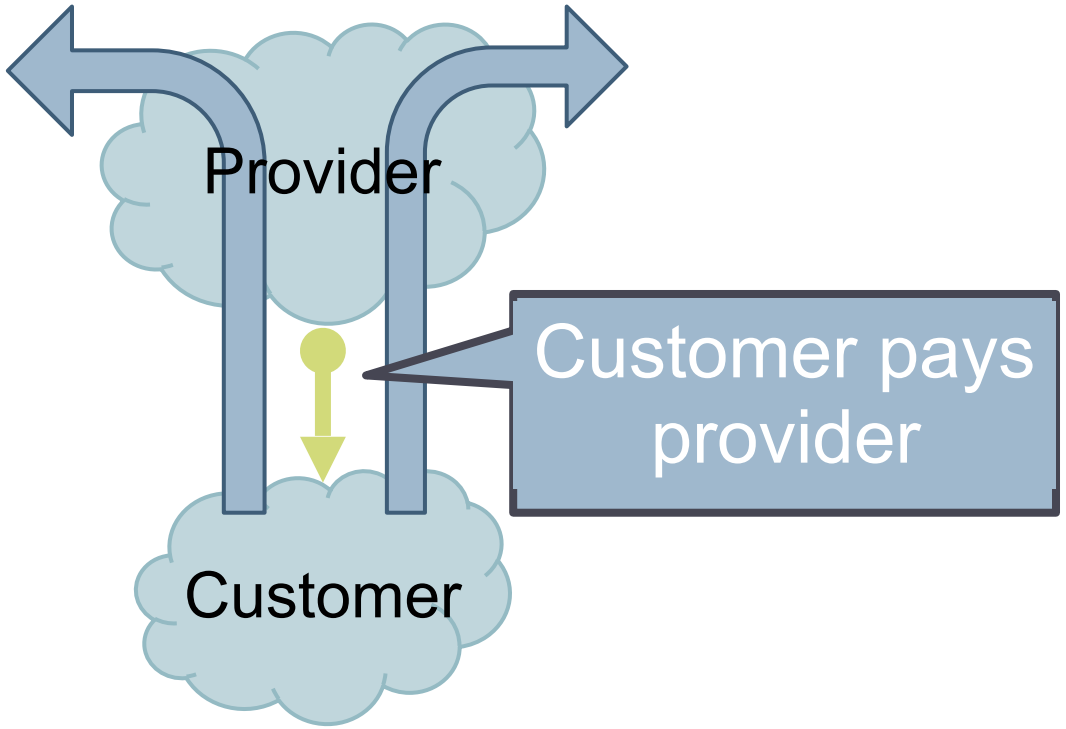
- Border Gateway Protocol
  - De facto inter-domain protocol of the Internet
  - Policy based routing protocol
  - Uses a Bellman-Ford path vector protocol
- Relatively simple protocol, but…
  - Complex, manual configuration
  - Entire world sees advertisements
    - Errors can screw up traffic globally
  - Policies driven by economics
    - How much $$$ does it cost to route along a given path?
    - Not by performance (e.g. shortest paths)

BGP Relationships

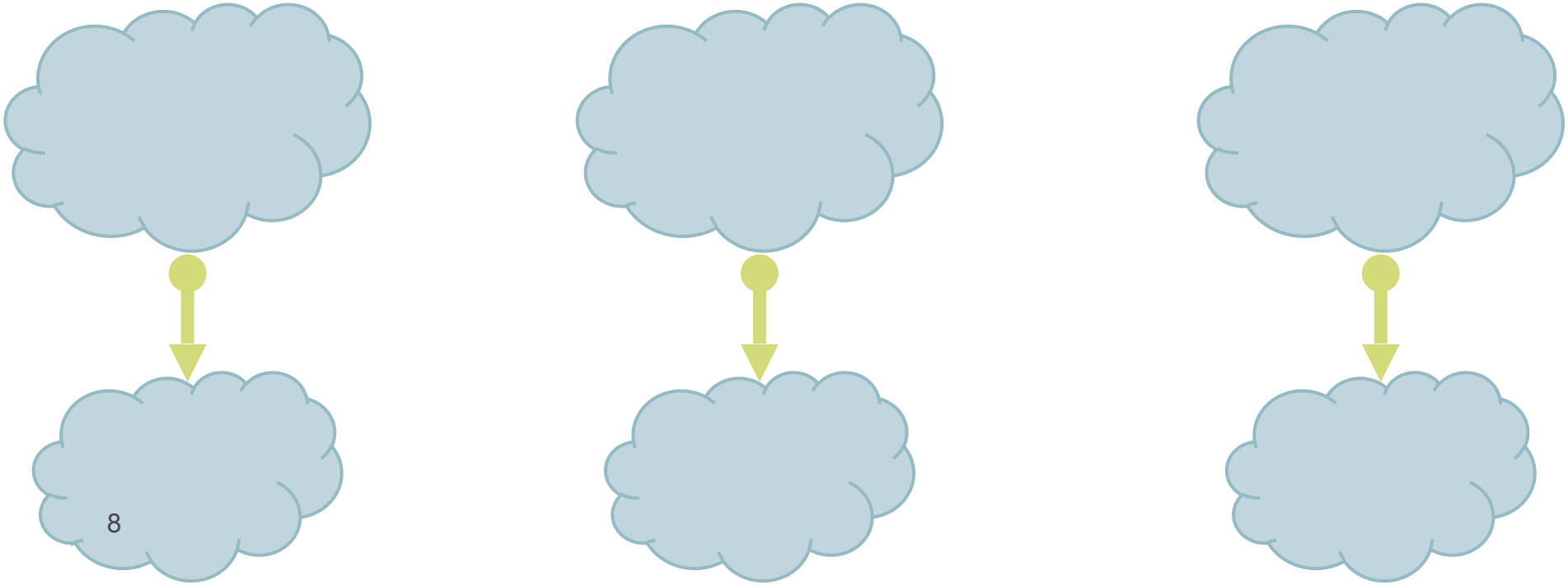
BGP Relationships



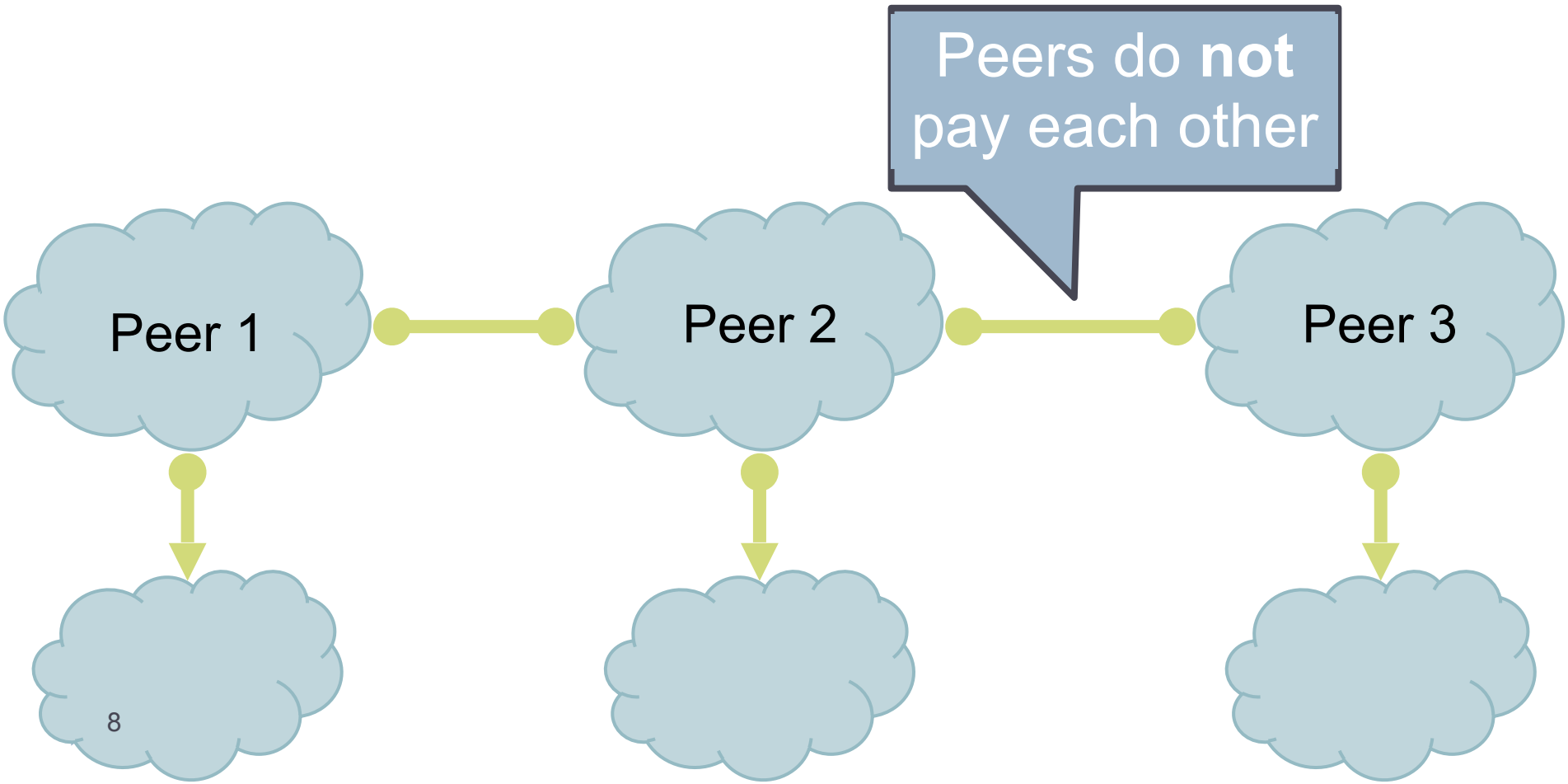
BGP Relationships



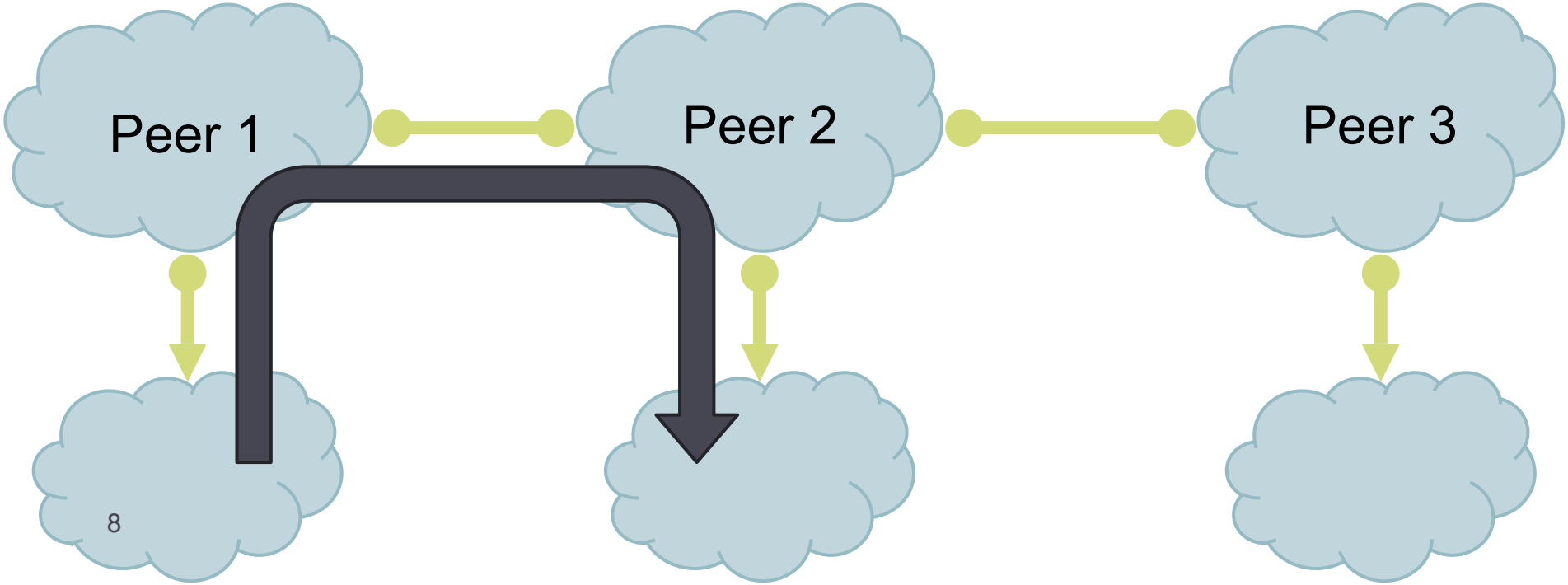
BGP Relationships



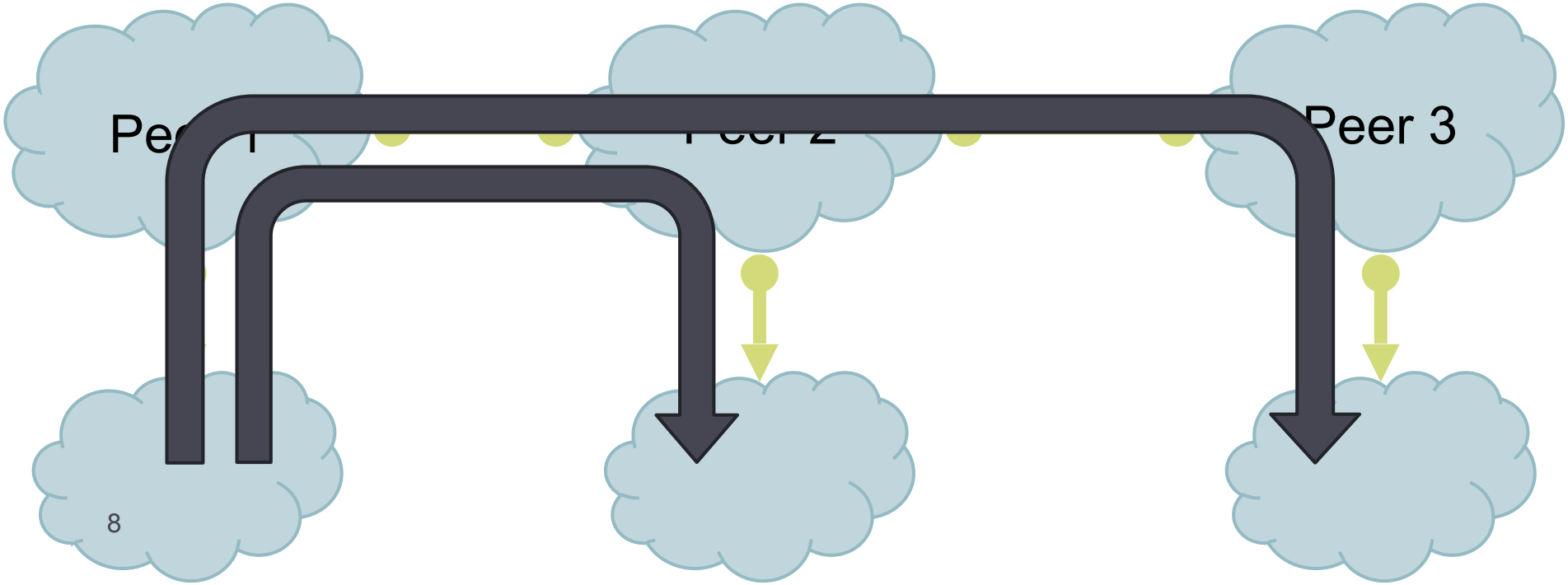
BGP Relationships



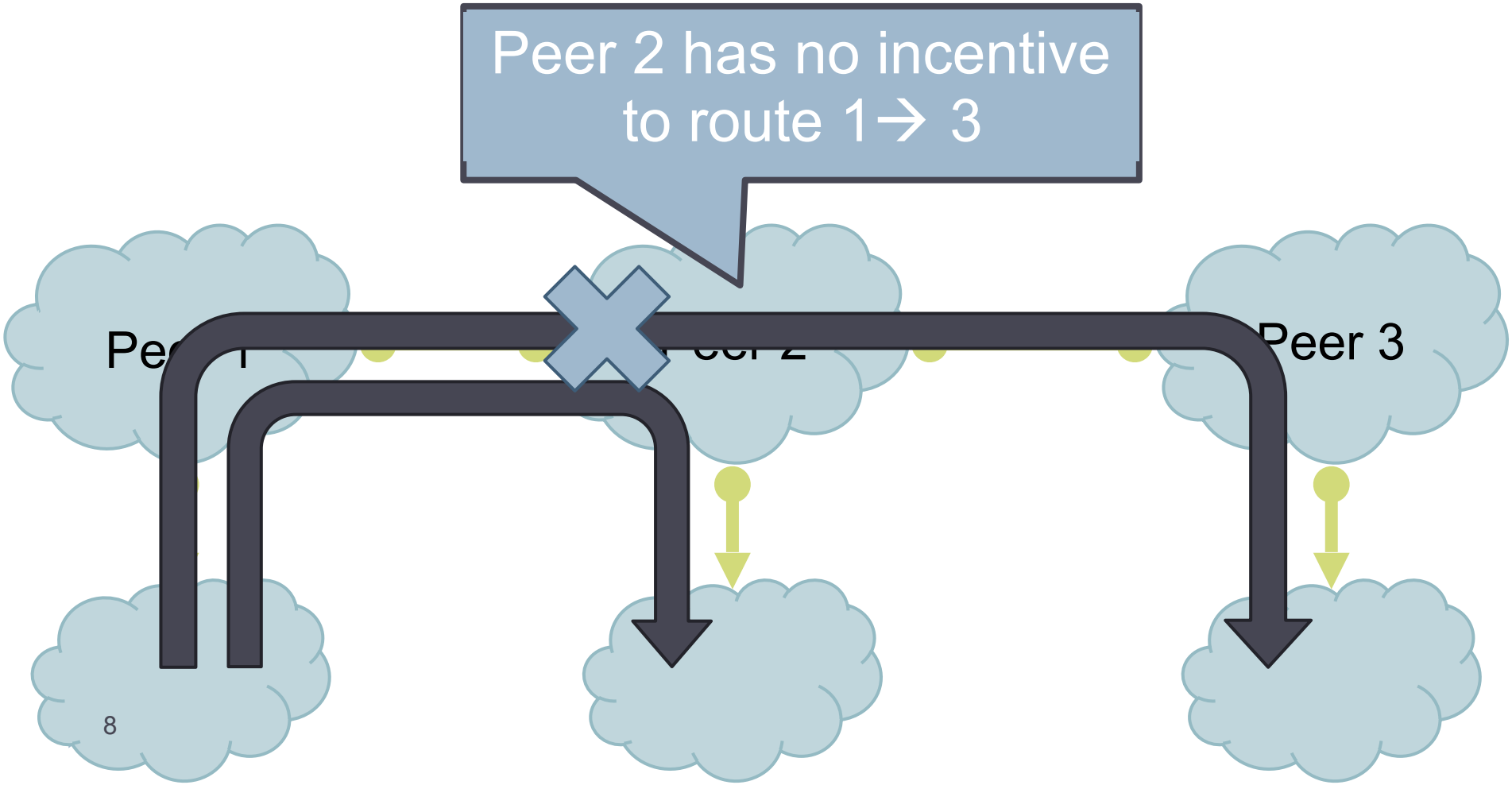
BGP Relationships



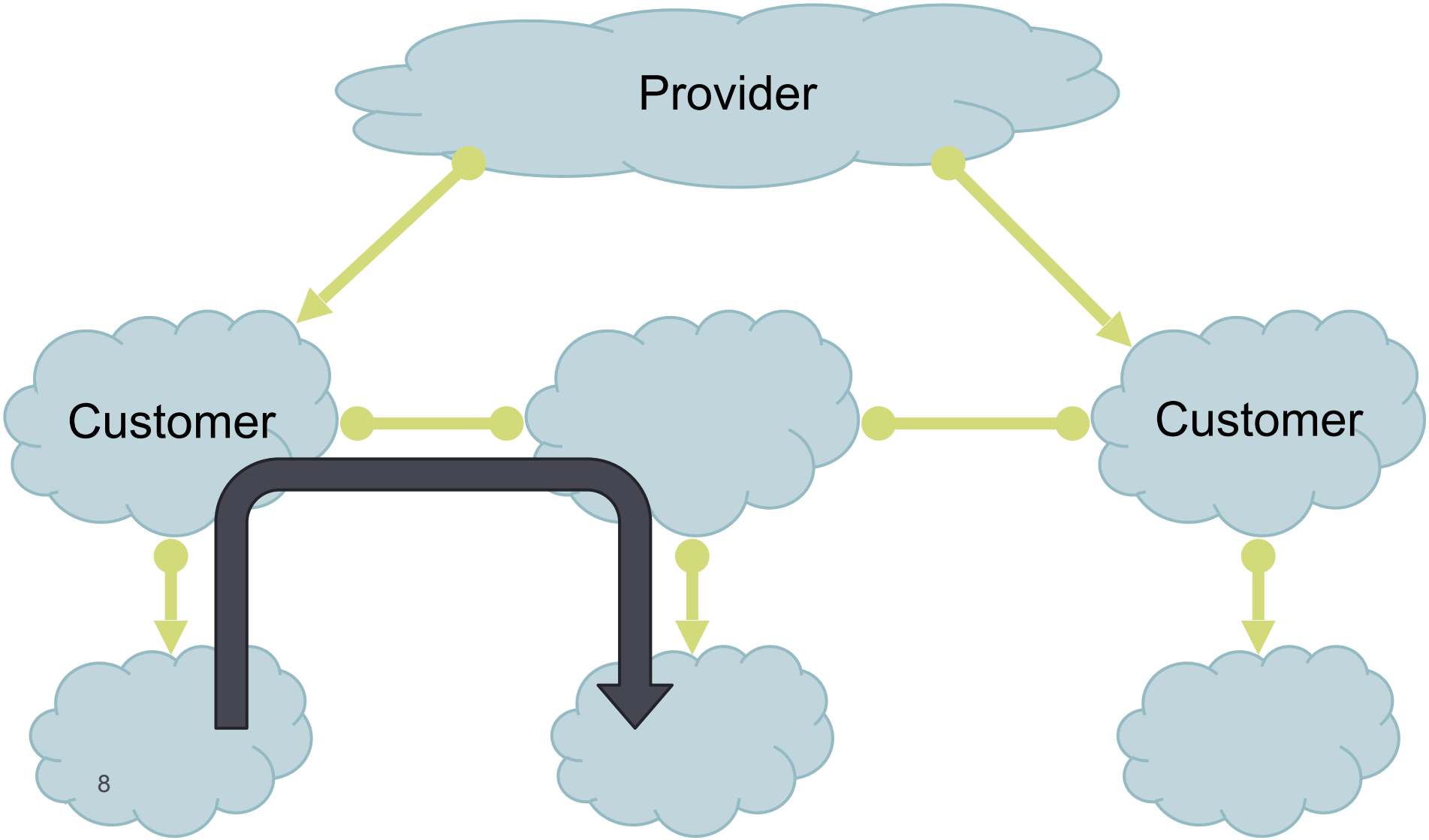
BGP Relationships



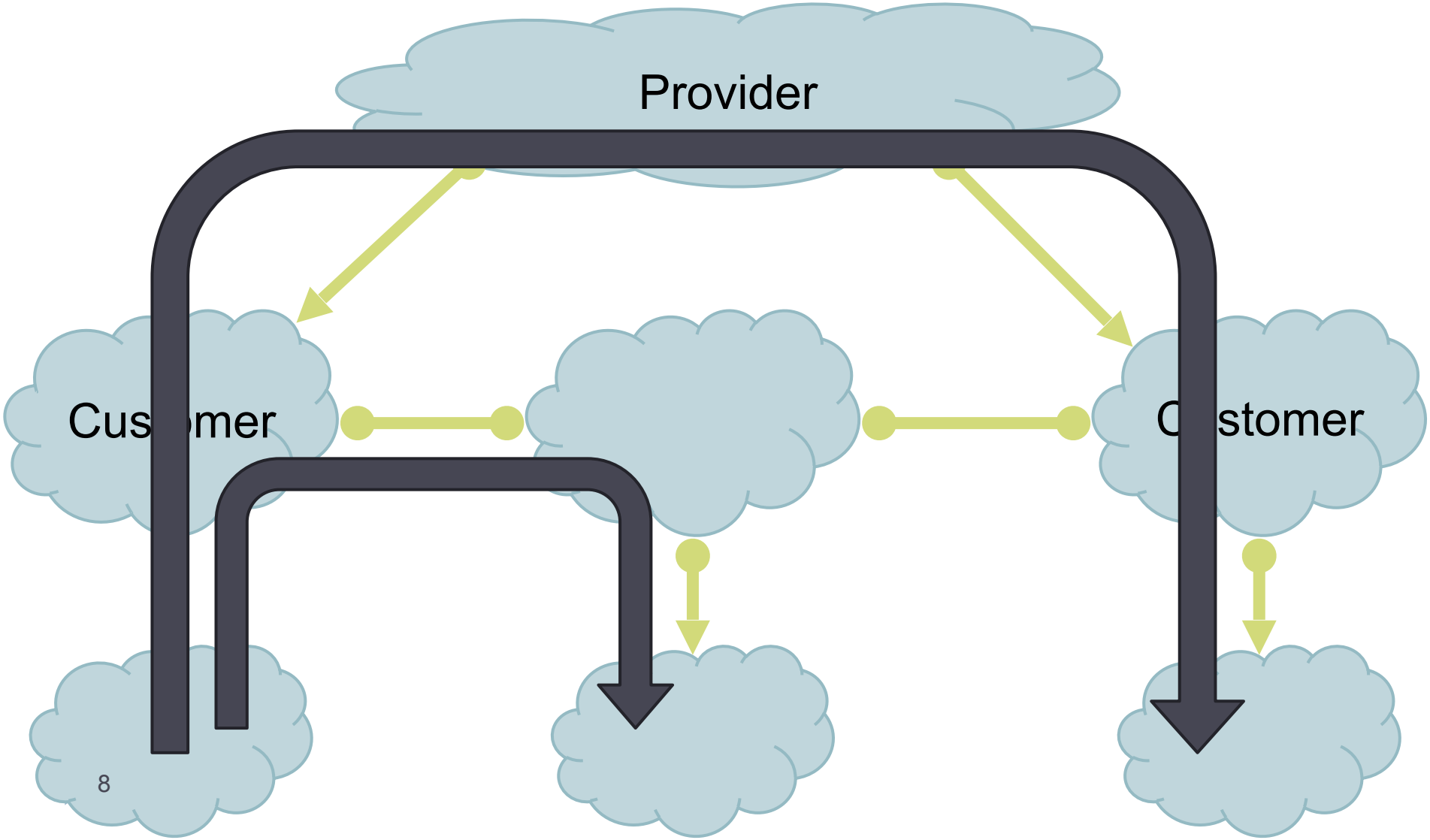
BGP Relationships



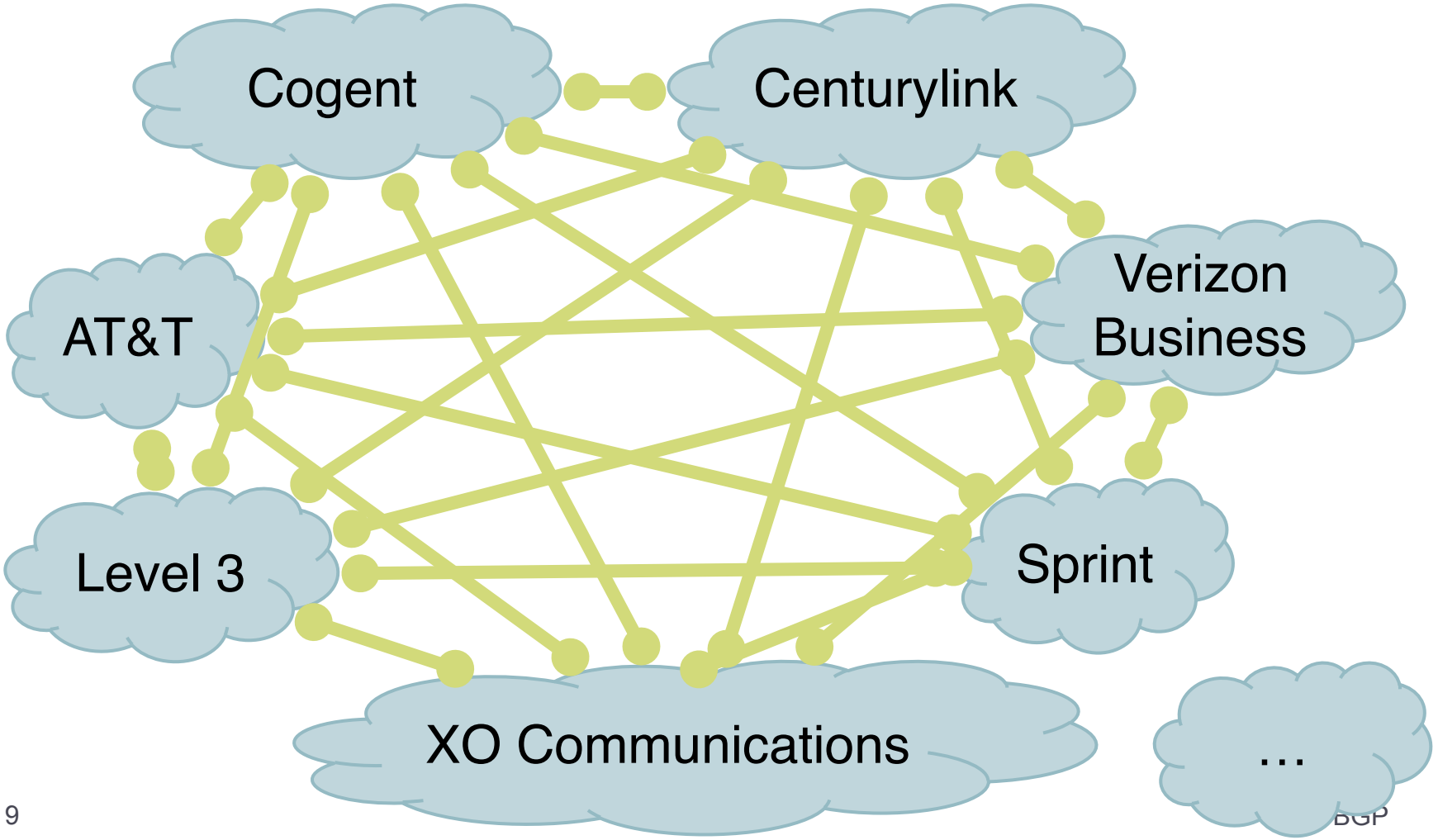
BGP Relationships



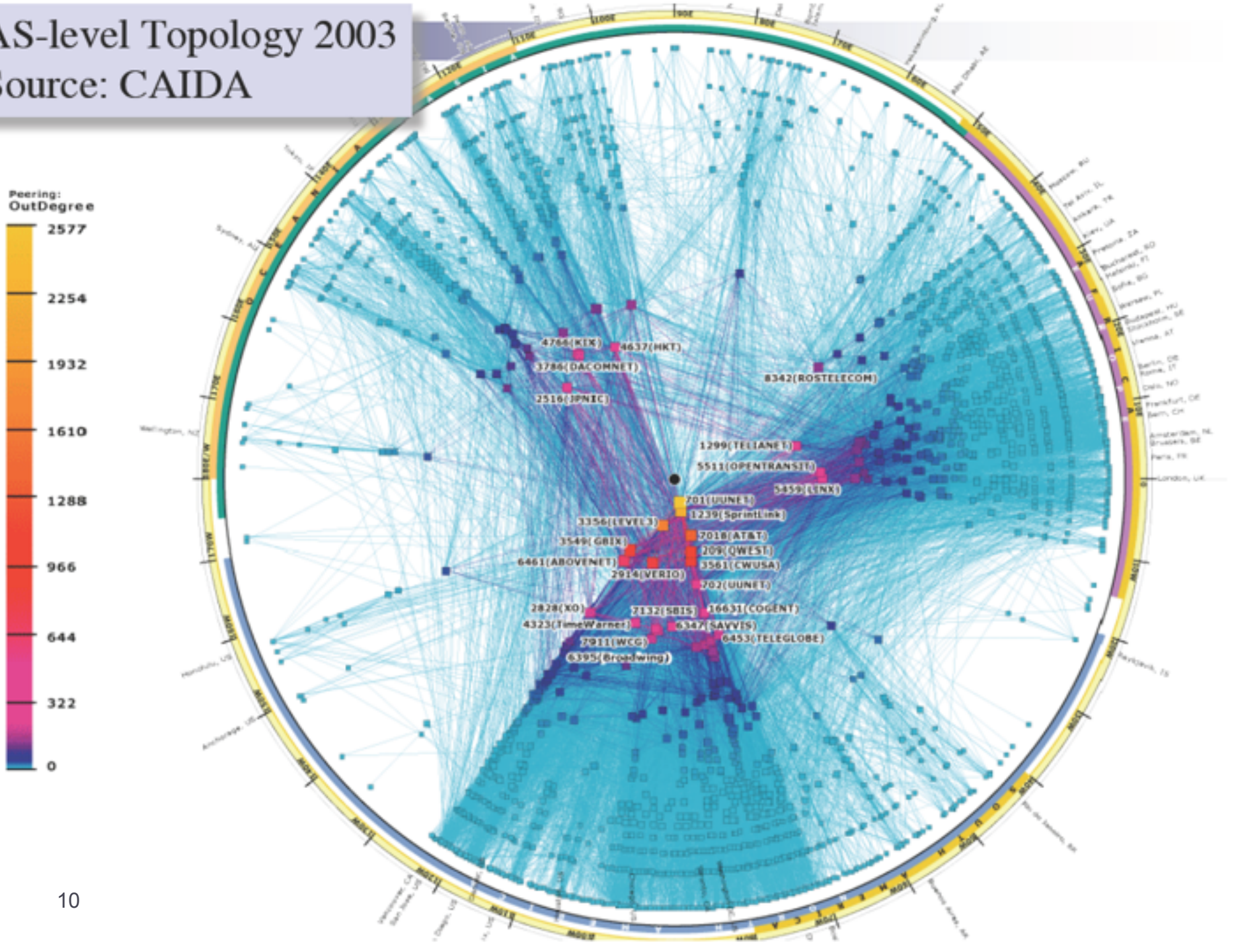
BGP Relationships



Tier-1 ISP Peering



AS-level Topology 2003
Source: CAIDA



Peering/Interconnection Wars

Peer

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

Don't Peer

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

Peering/Interconnection Wars

Peer

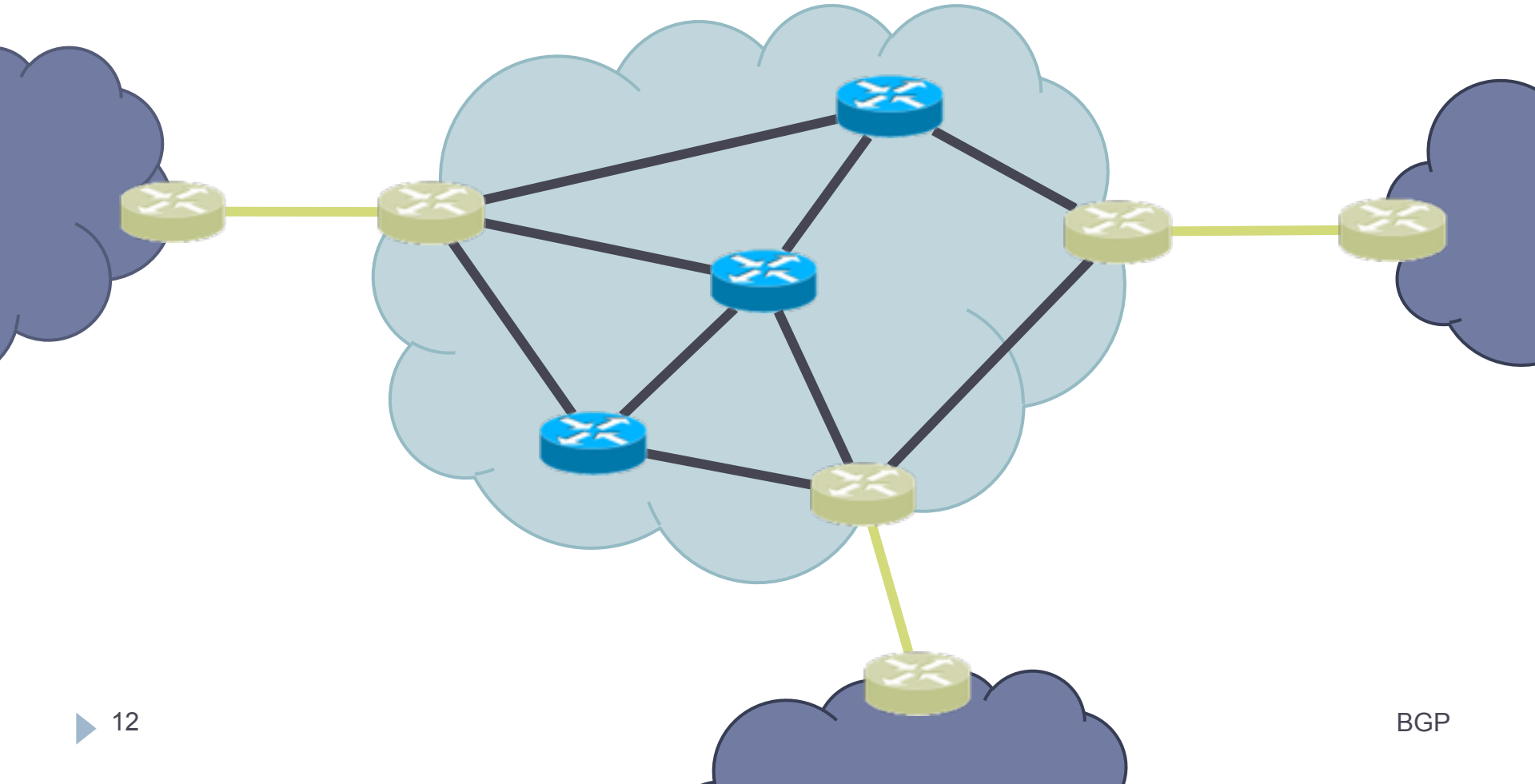
- Reduce upstream costs
- Improve end-to-end performance
- May be the only way to connect to parts of the Internet

Don't Peer

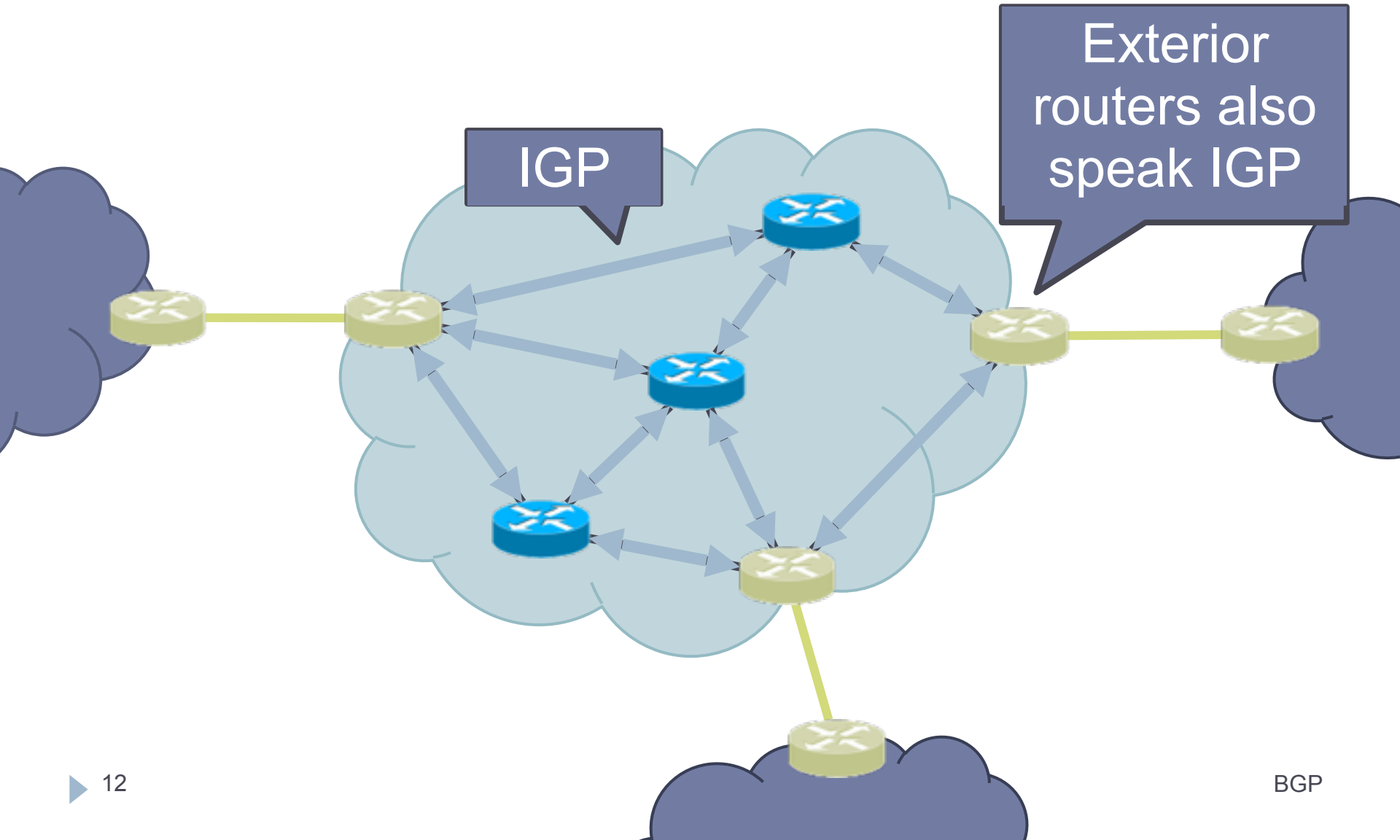
- You would rather have customers
- Peers are often competitors
- Peering agreements require periodic renegotiation

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

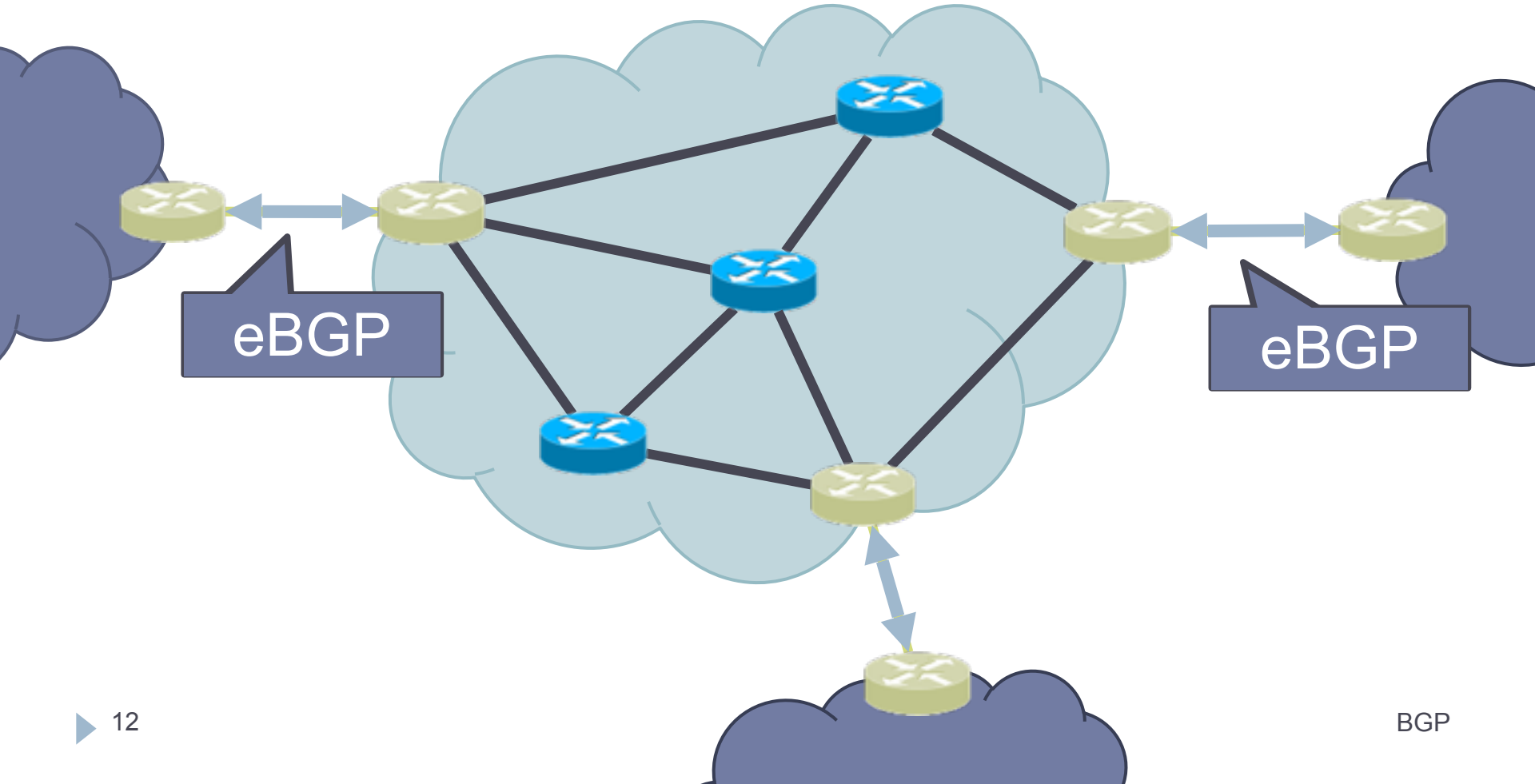
Two Types of BGP Neighbors



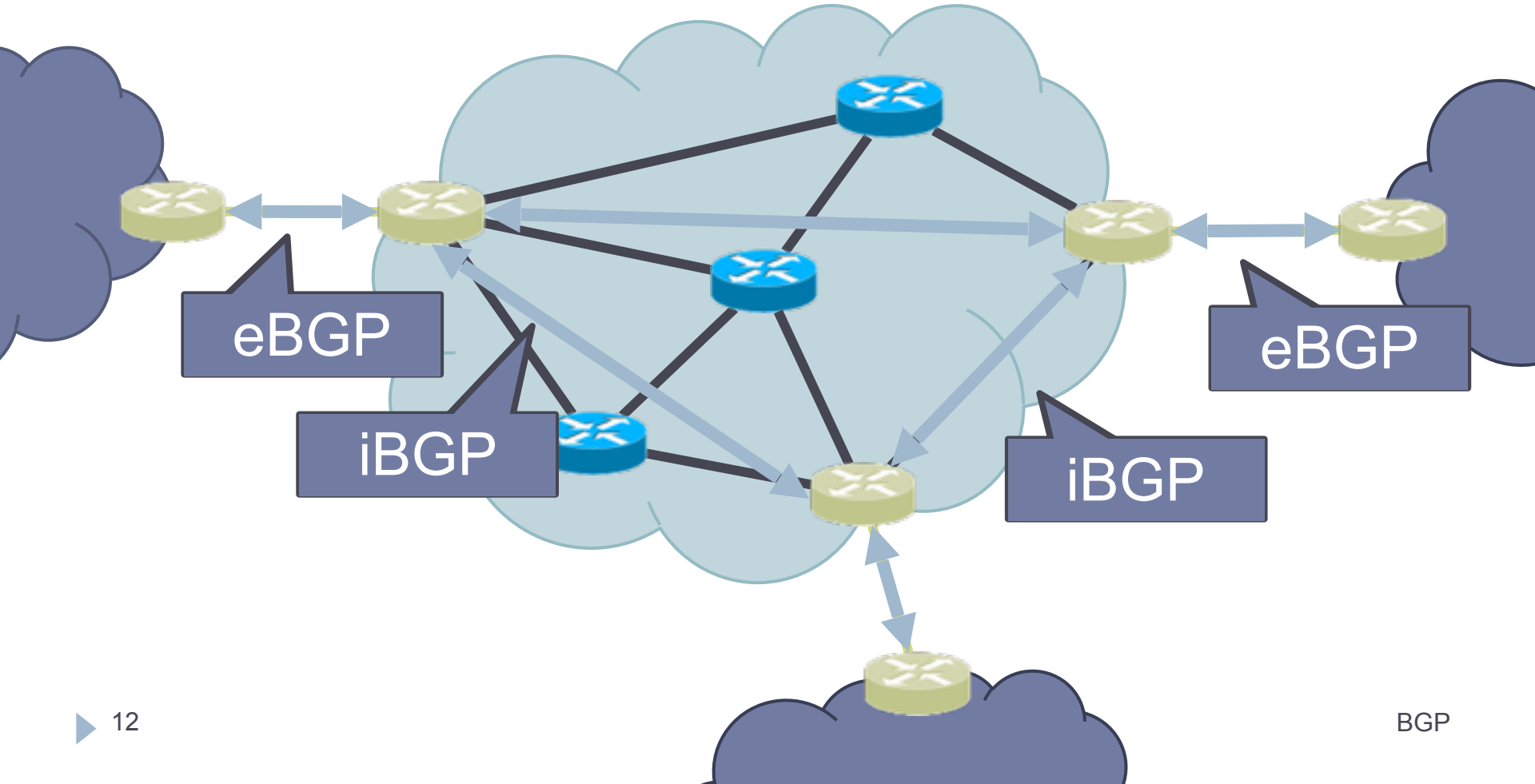
Two Types of BGP Neighbors



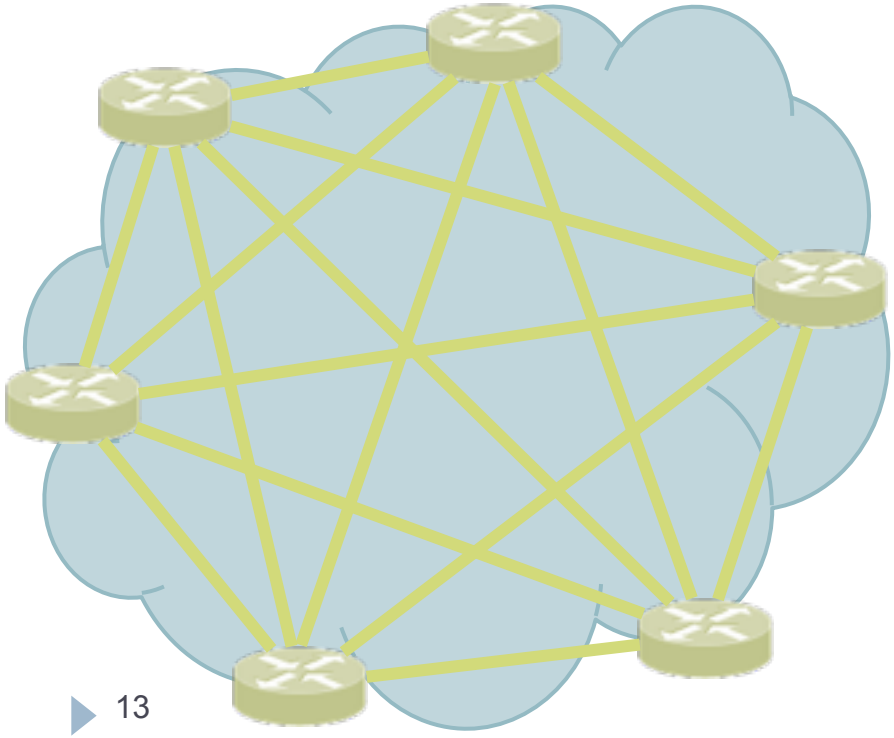
Two Types of BGP Neighbors



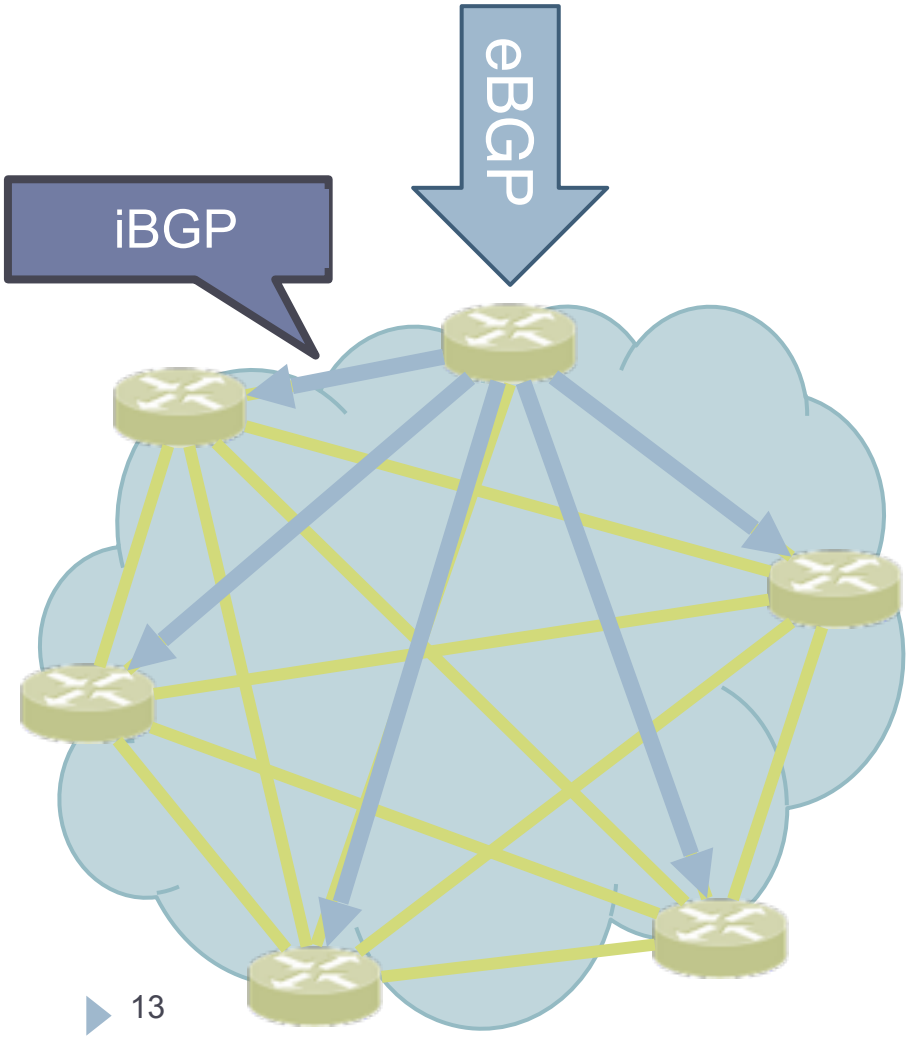
Two Types of BGP Neighbors



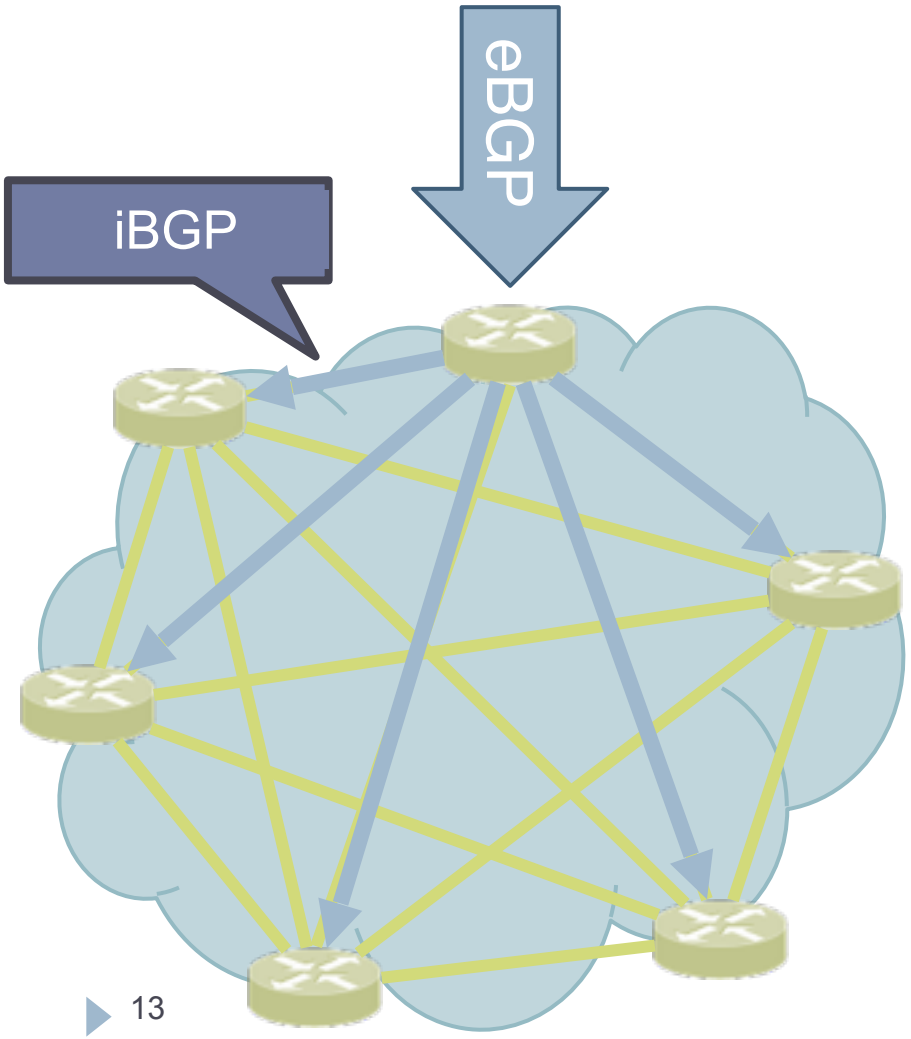
Full iBGP Meshes



Full iBGP Meshes

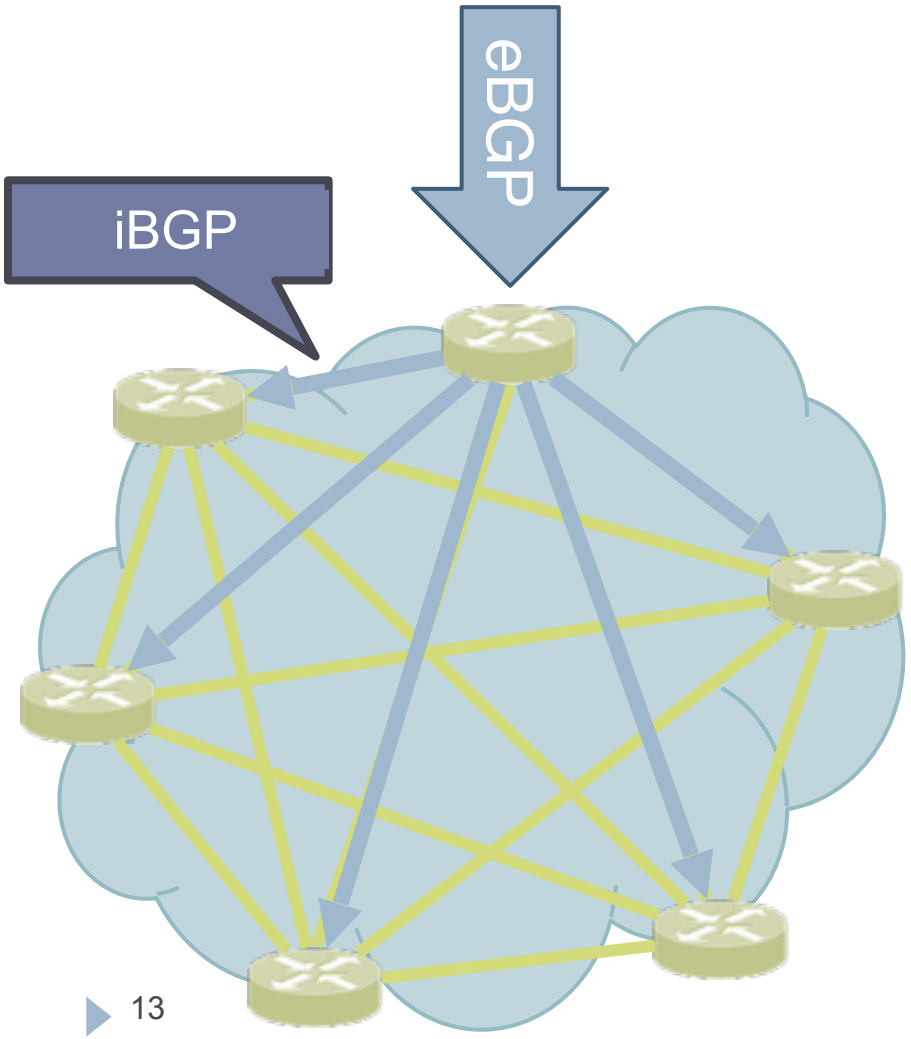


Full iBGP Meshes



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

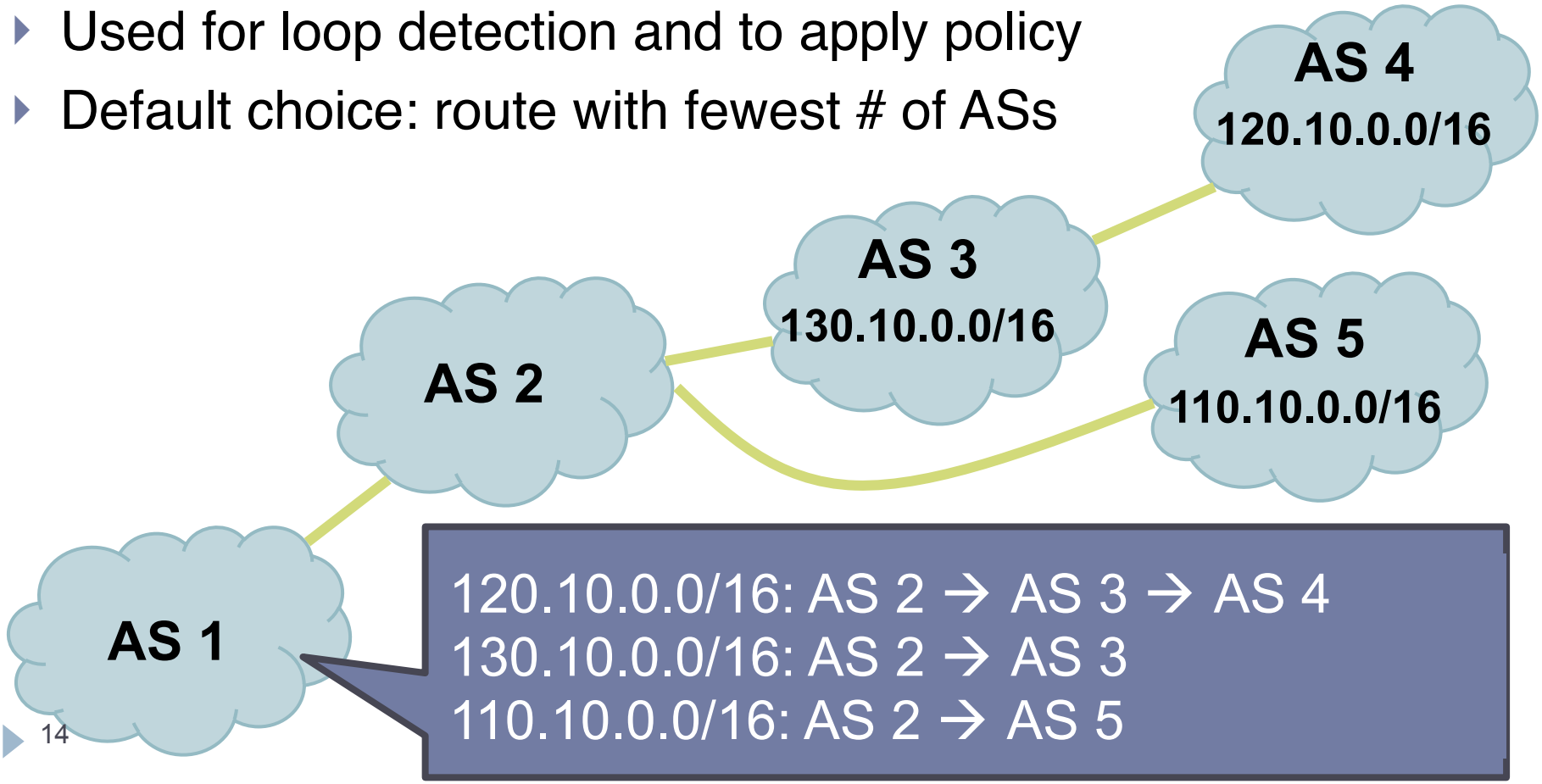
Full iBGP Meshes



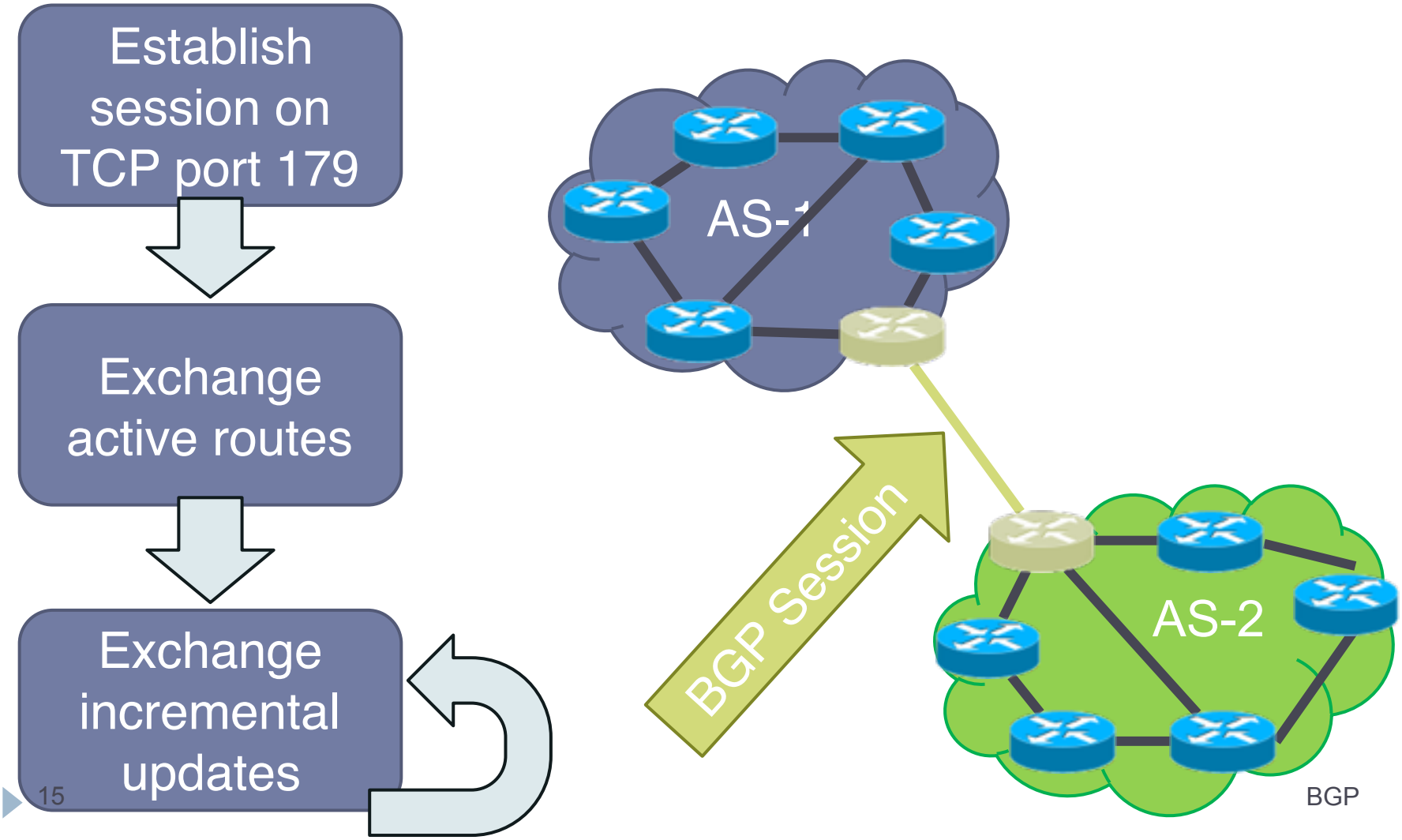
- Question: why do we need iBGP?
  - OSPF does not include BGP policy info
  - Prevents routing loops within the AS
- iBGP updates do not trigger announcements

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



BGP Operations (Simplified)



Four Types of BGP Messages

- Open: Establish a peering session.
- Keep Alive: Handshake at regular intervals.
- Notification: Shuts down a peering session.
- Update: Announce new routes or withdraw previously announced routes.

Four Types of BGP Messages

- Open: Establish a peering session.
- Keep Alive: Handshake at regular intervals.
- Notification: Shuts down a peering session.
- Update: Announce new routes or withdraw previously announced routes.

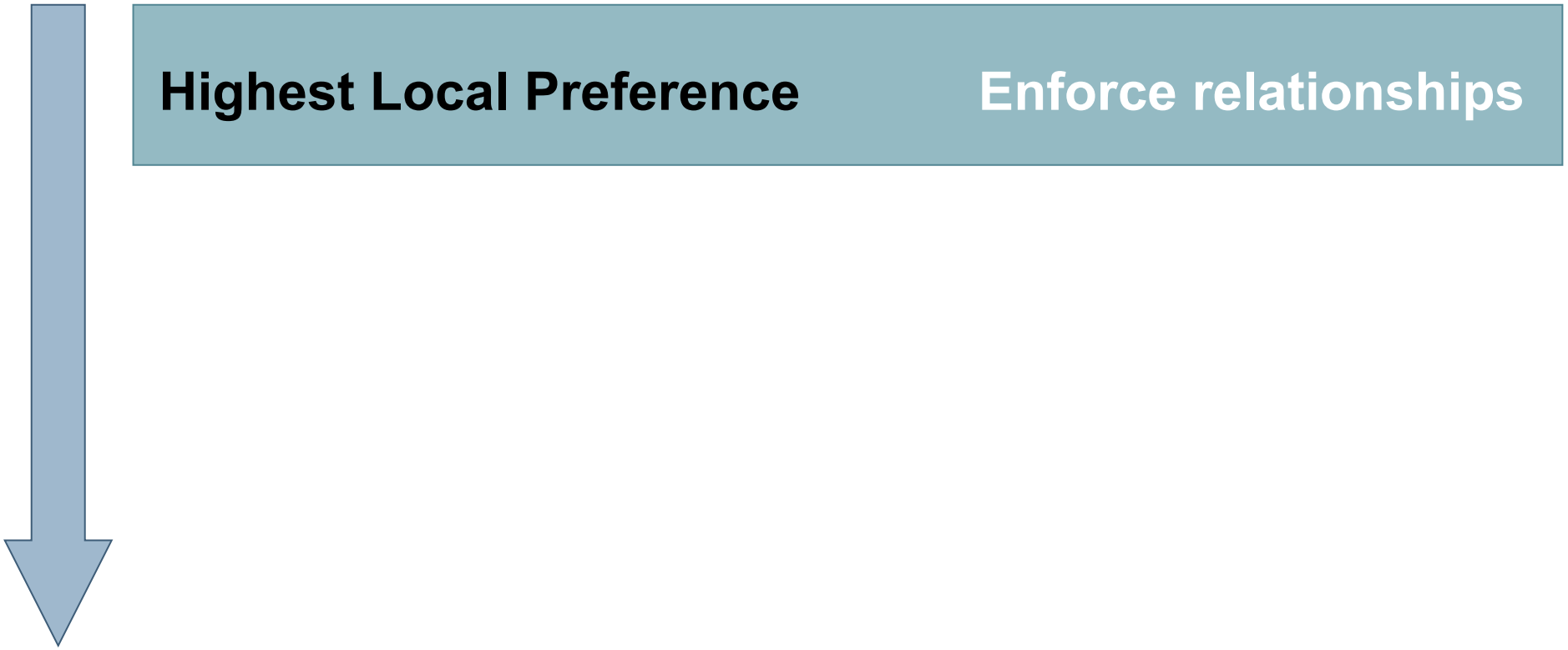
announcement = IP prefix + attributes values

BGP Attributes

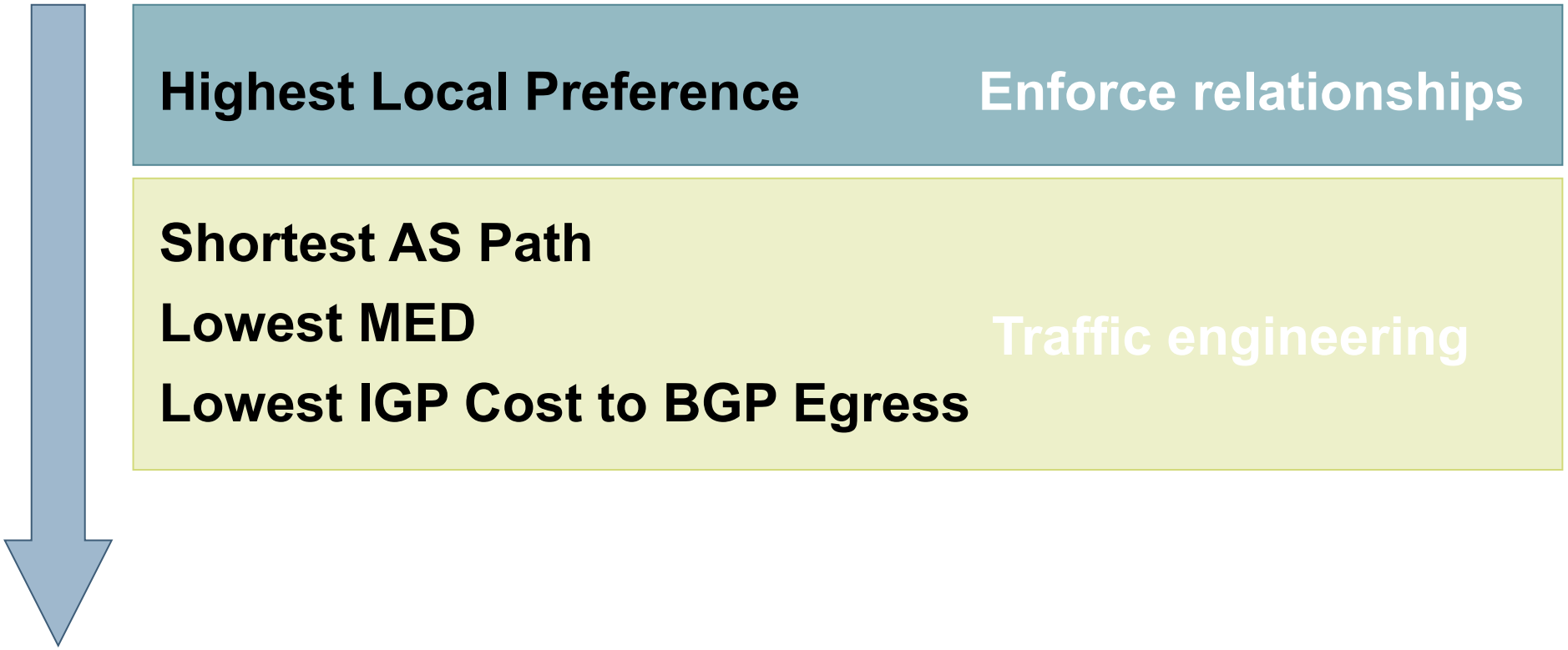
- Attributes used to select “best” path
  - LocalPref
    - Local preference policy to choose most preferred route
    - Overrides default fewest AS behavior
  - Multi-exit Discriminator (MED)
    - Specifies path for external traffic destined for an internal network
    - Chooses peering point for your network
  - Import Rules
    - What route advertisements do I accept?
  - Export Rules
    - Which routes do I forward to whom?

Route Selection Summary

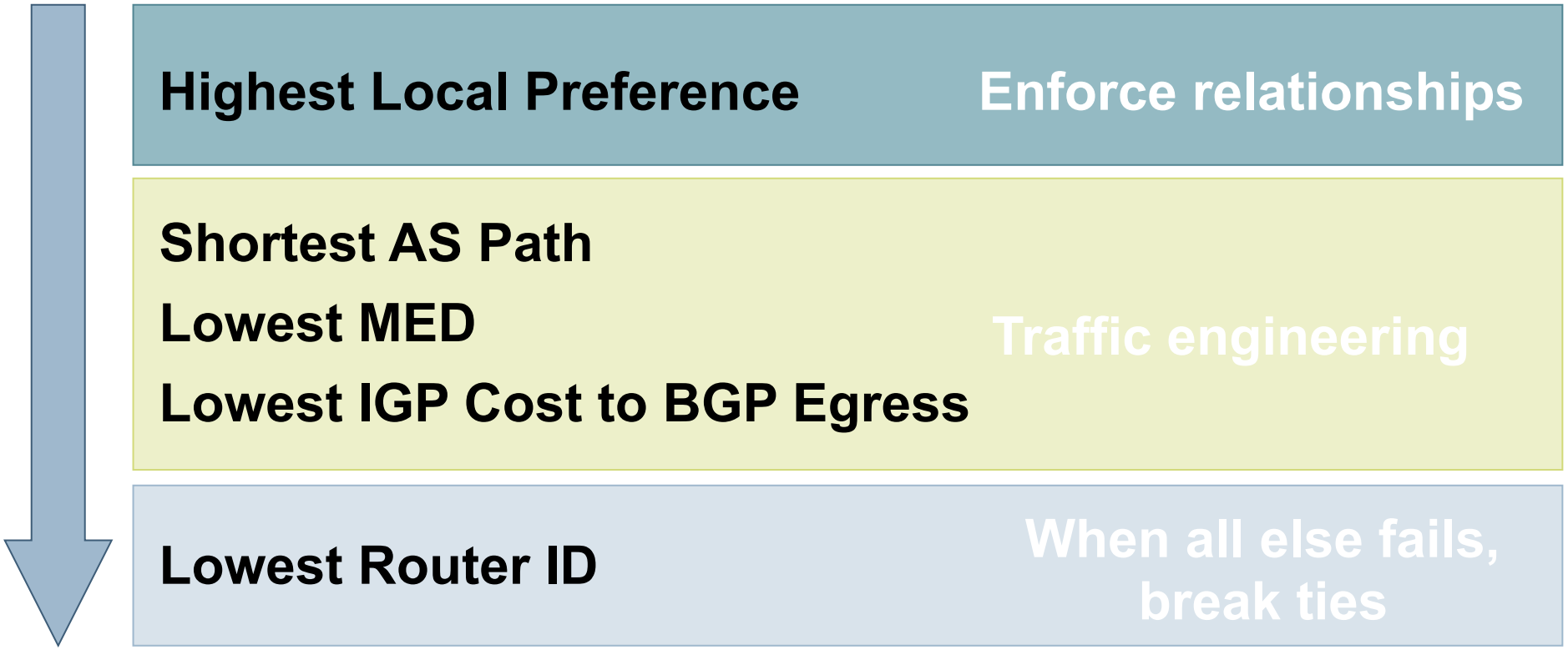
Route Selection Summary



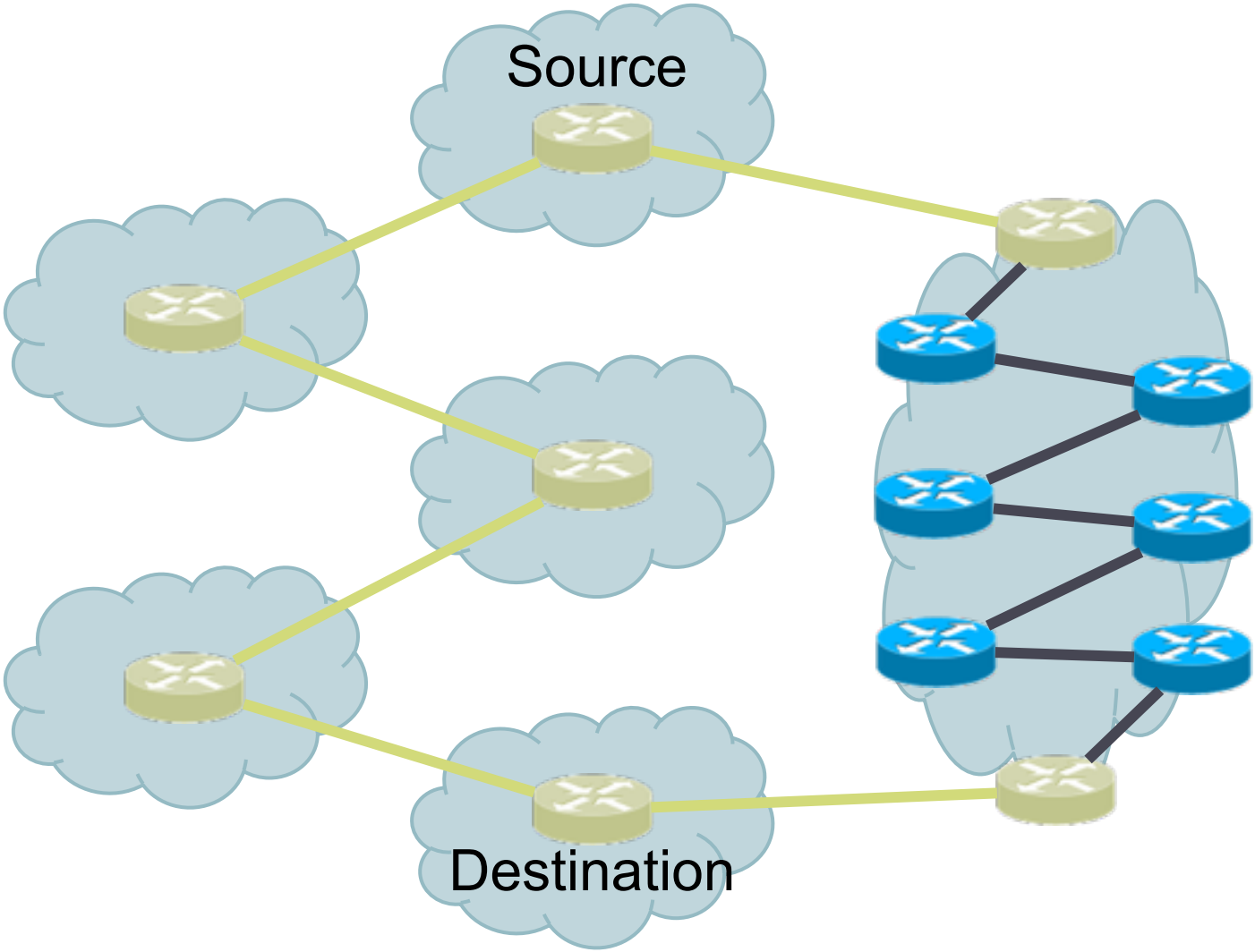
Route Selection Summary



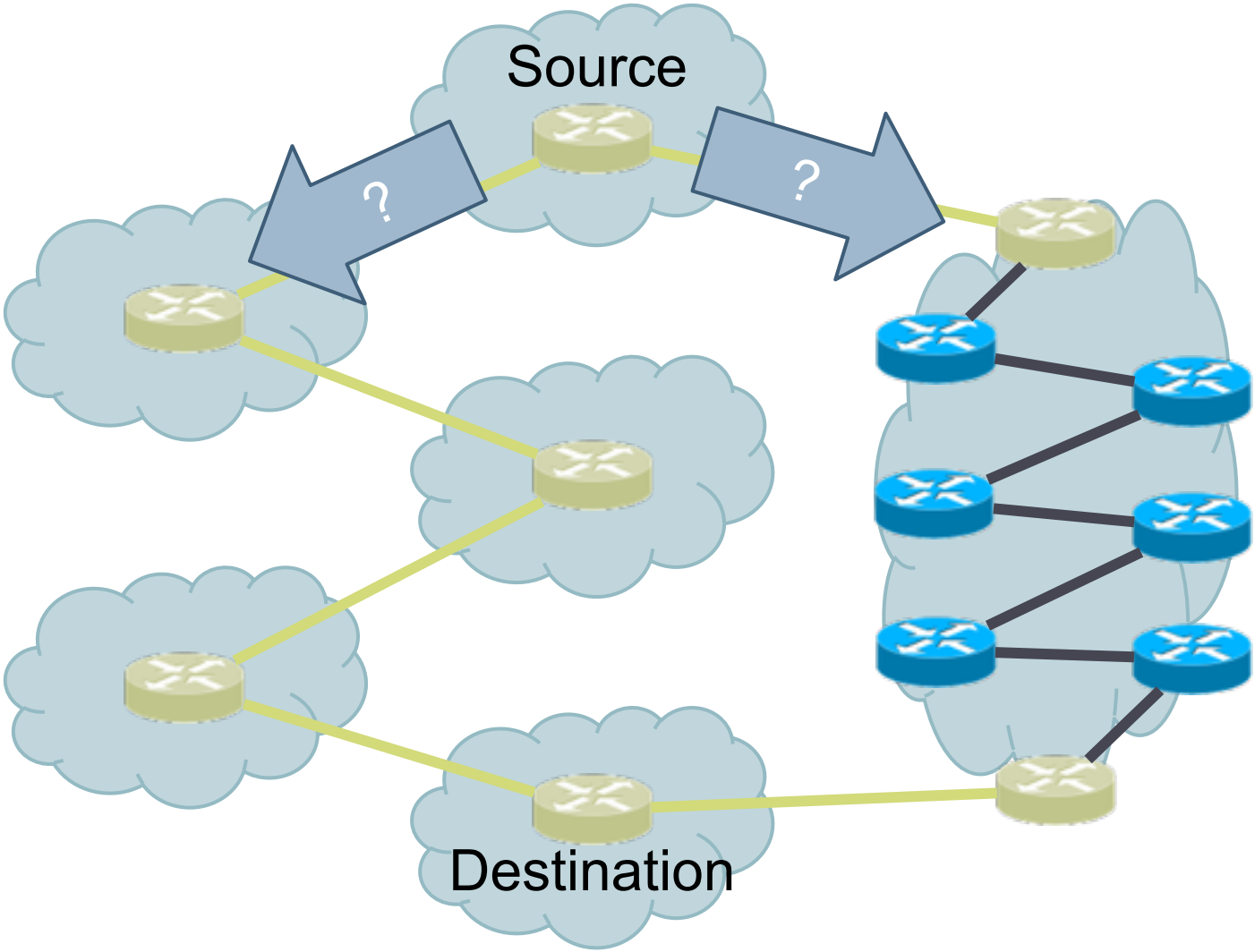
Route Selection Summary



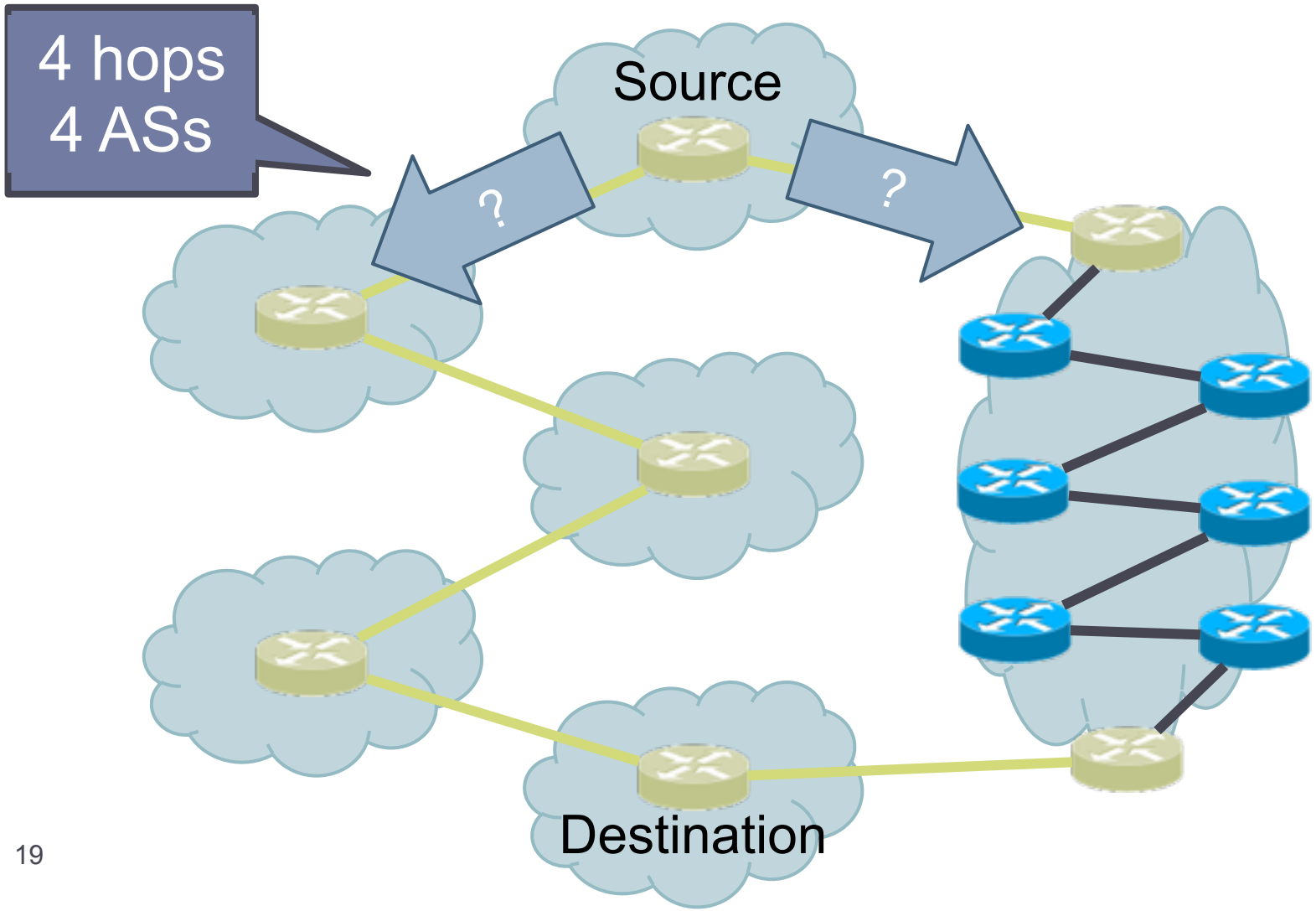
Shortest AS Path != Shortest Path



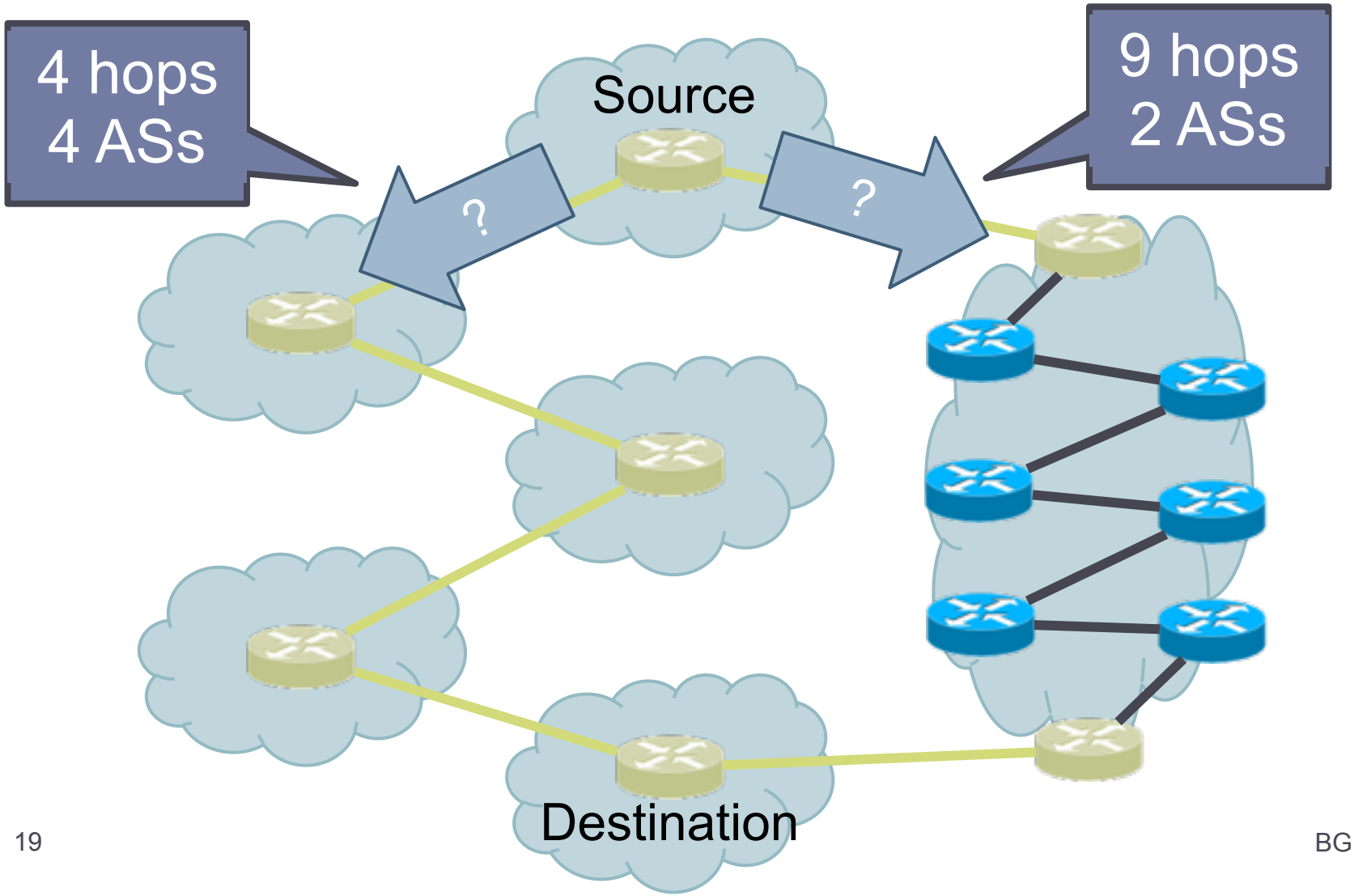
Shortest AS Path != Shortest Path



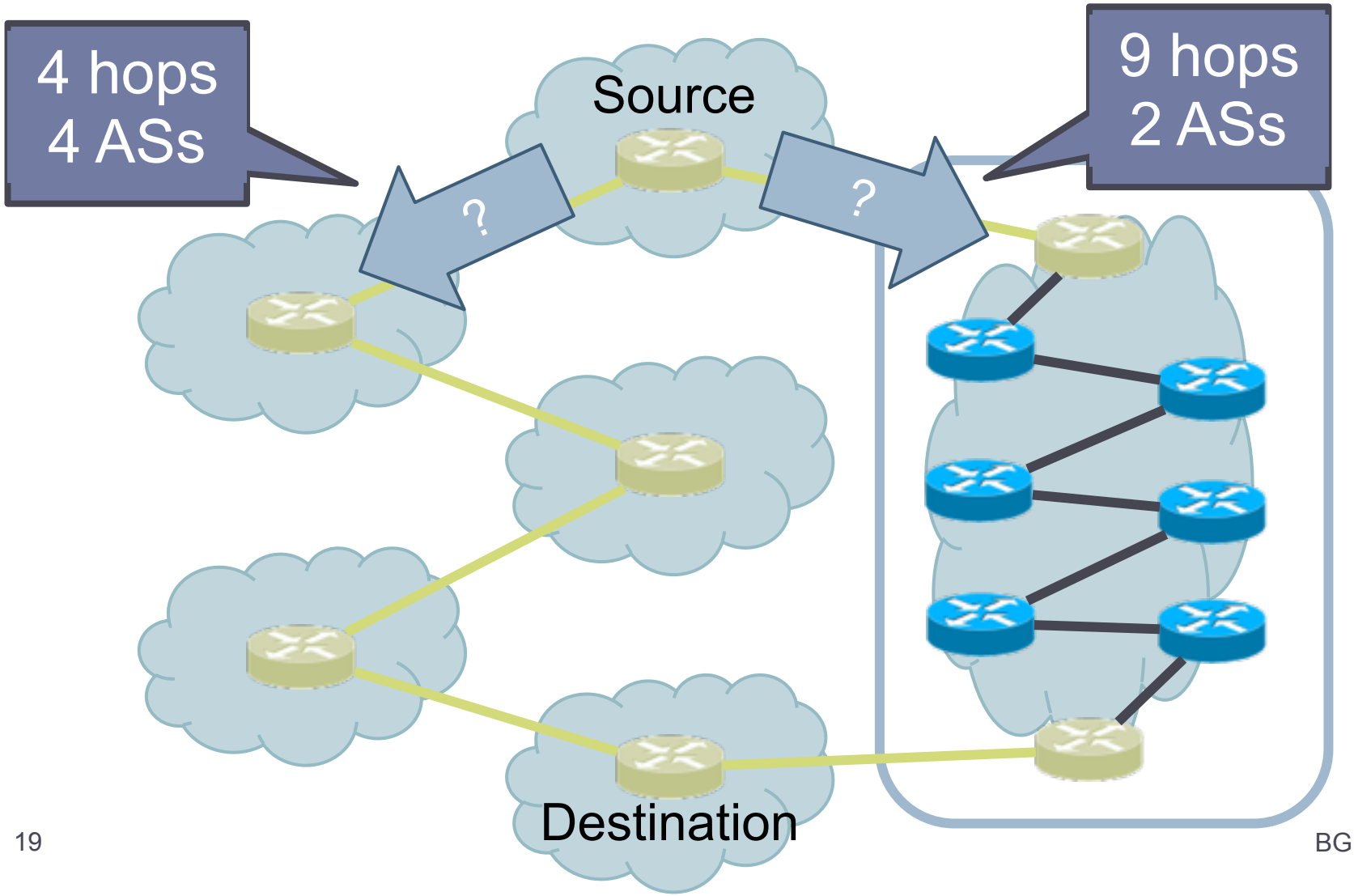
Shortest AS Path != Shortest Path



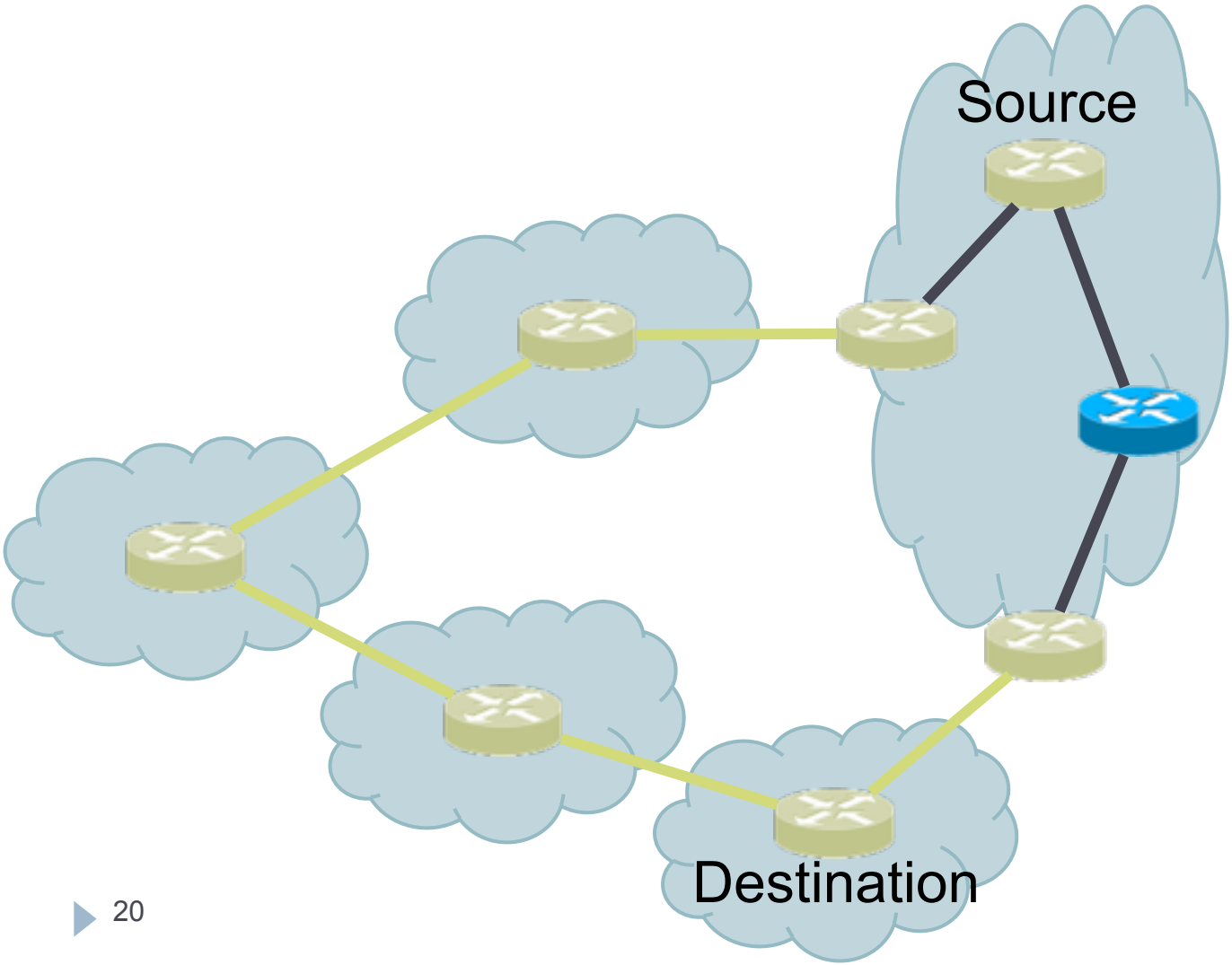
Shortest AS Path != Shortest Path



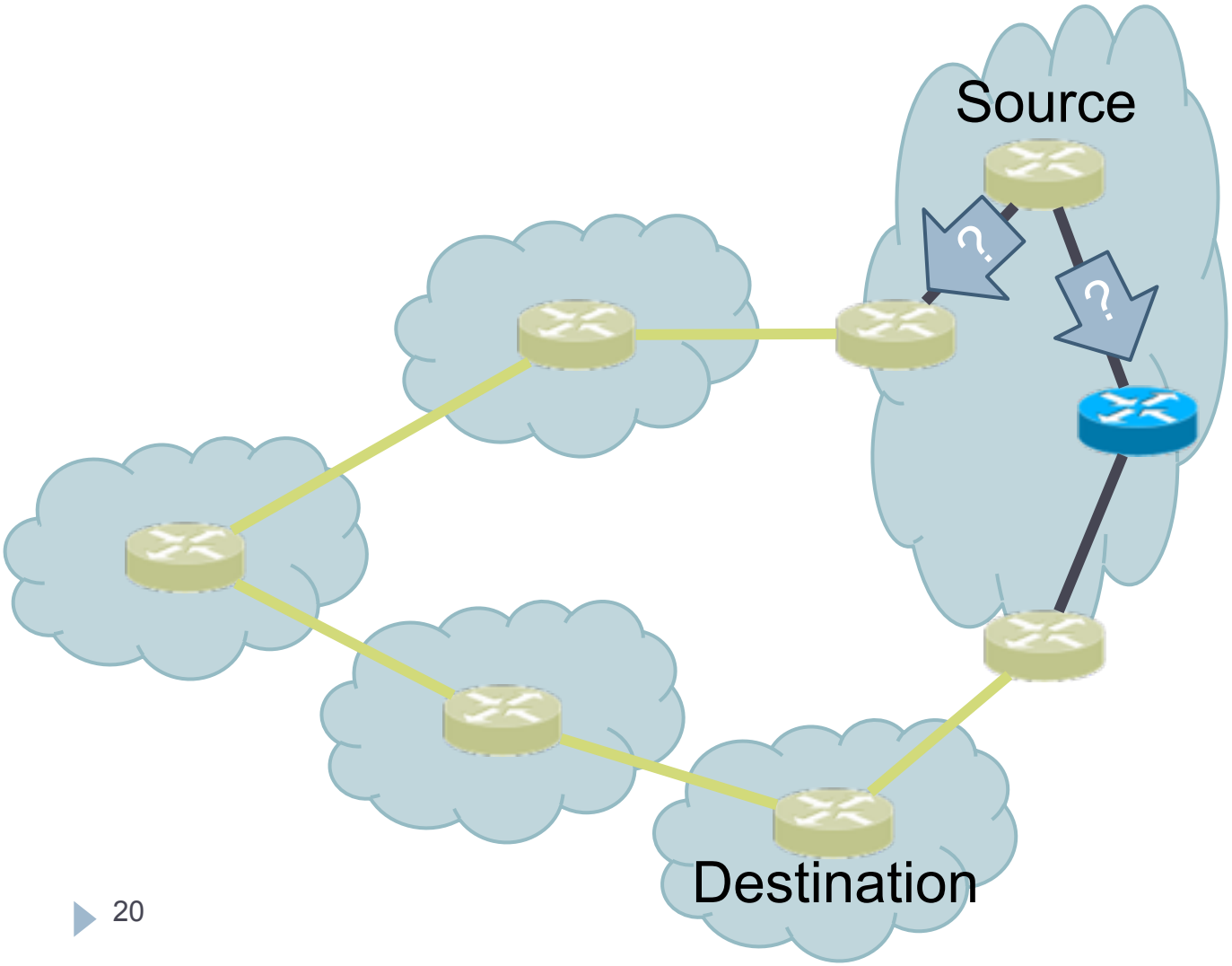
Shortest AS Path != Shortest Path



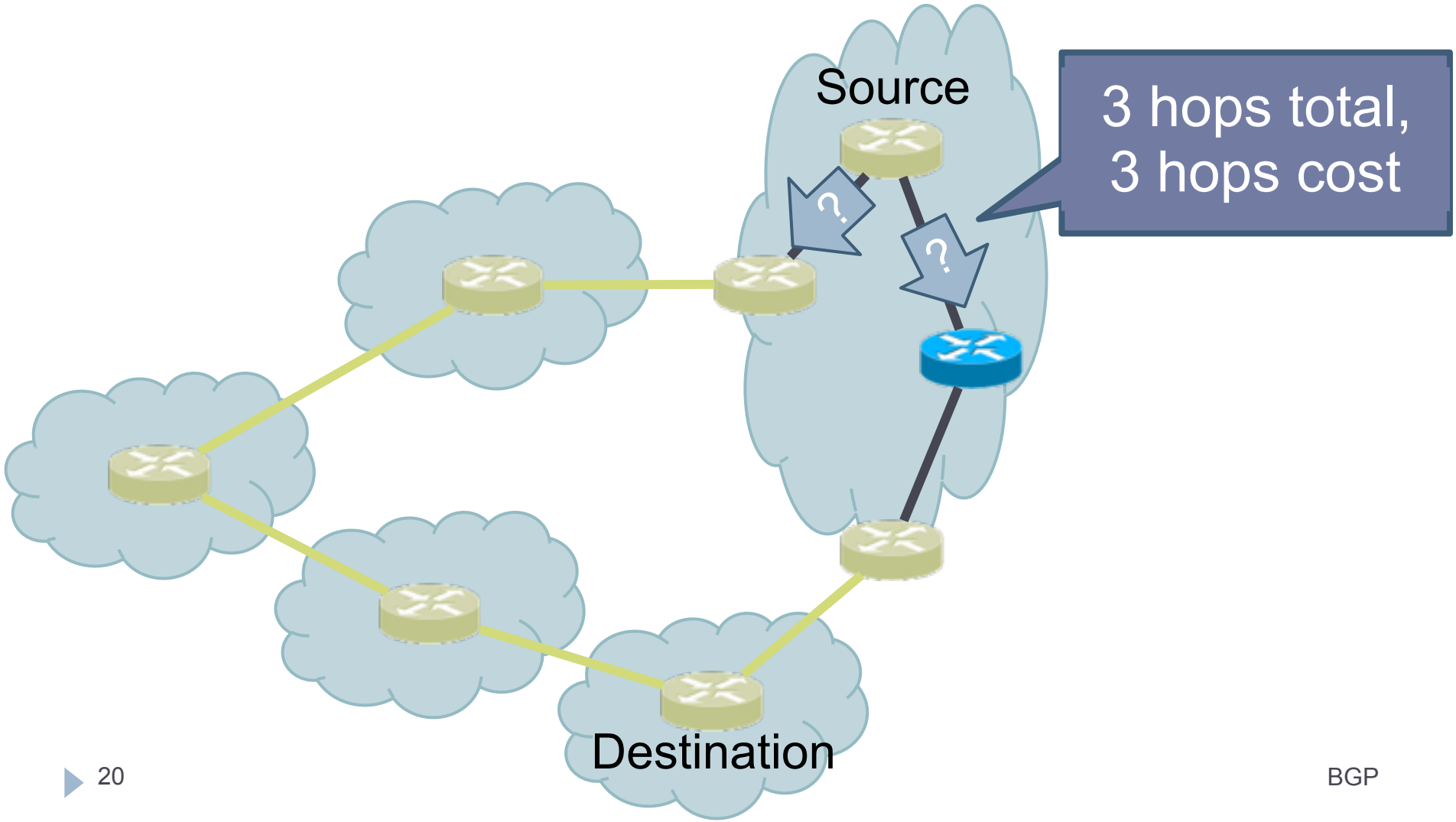
Hot Potato Routing



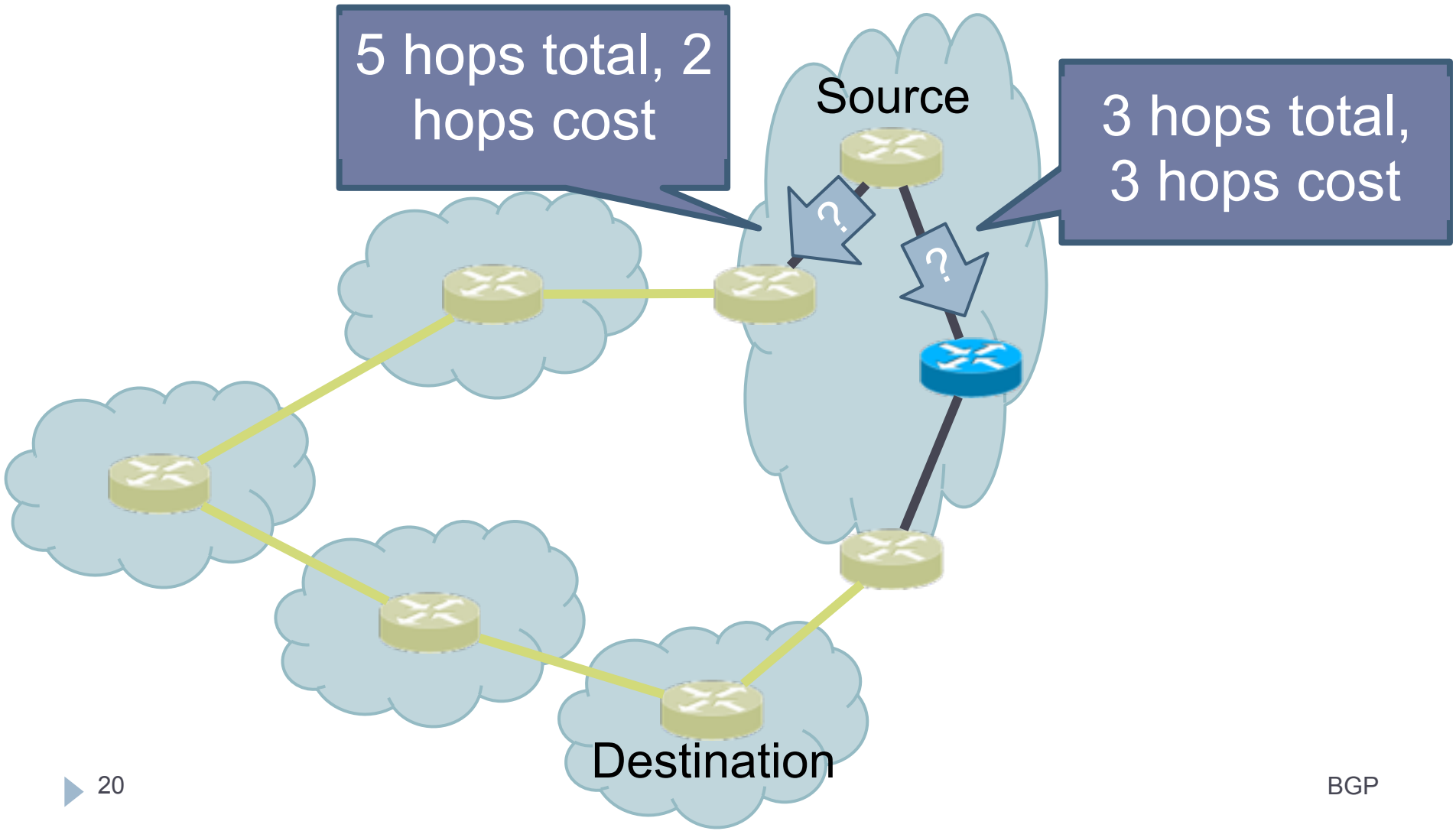
Hot Potato Routing



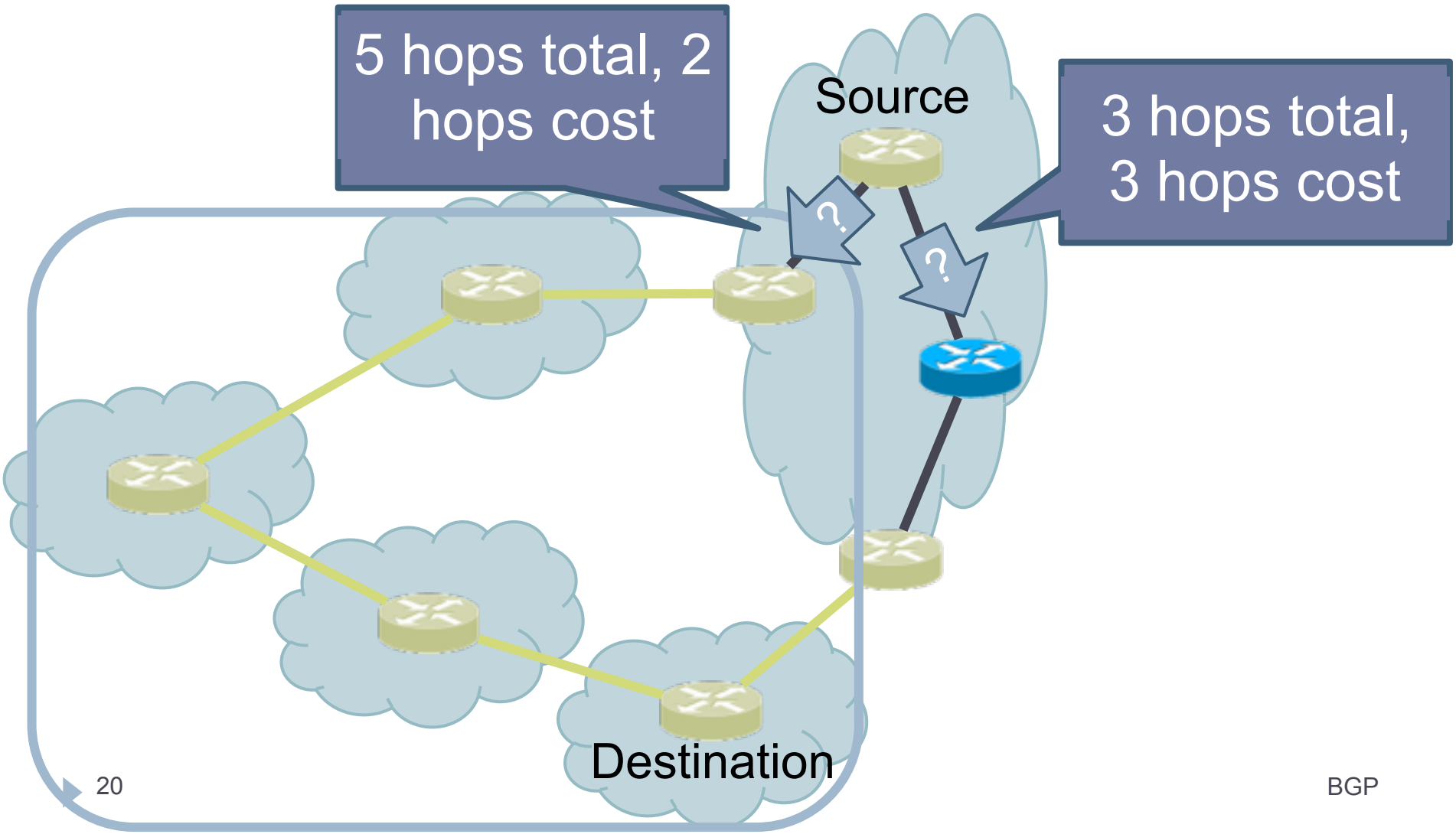
Hot Potato Routing



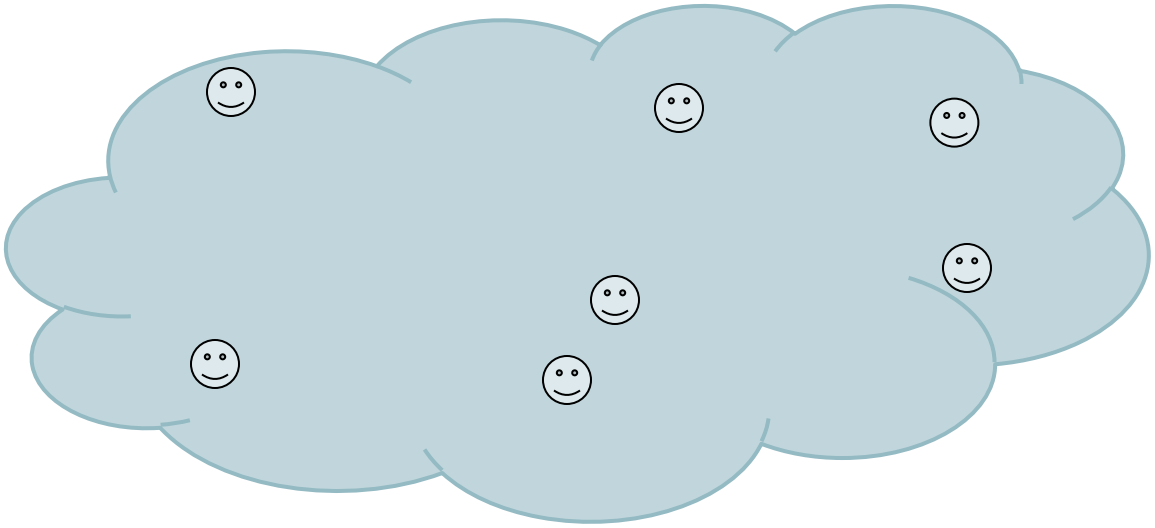
Hot Potato Routing



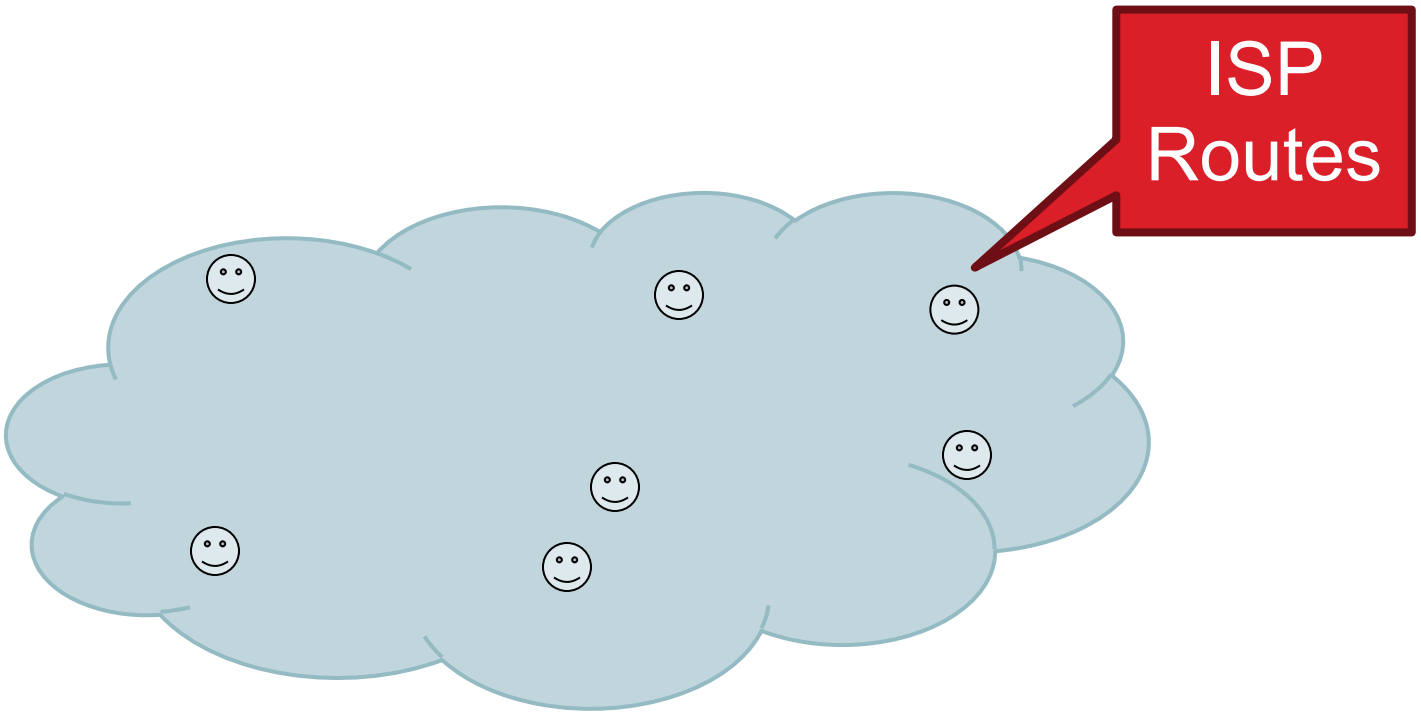
Hot Potato Routing



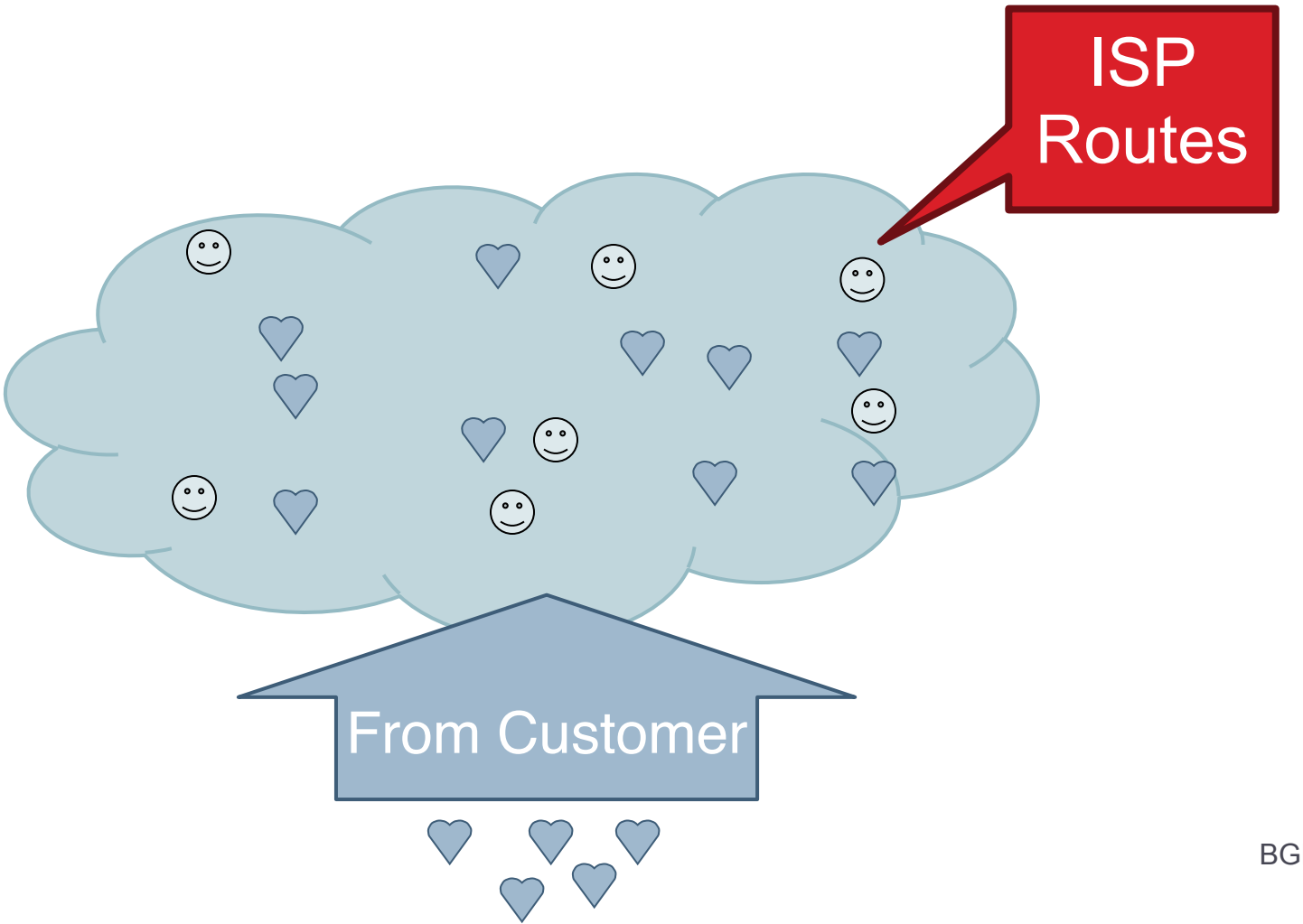
Importing Routes



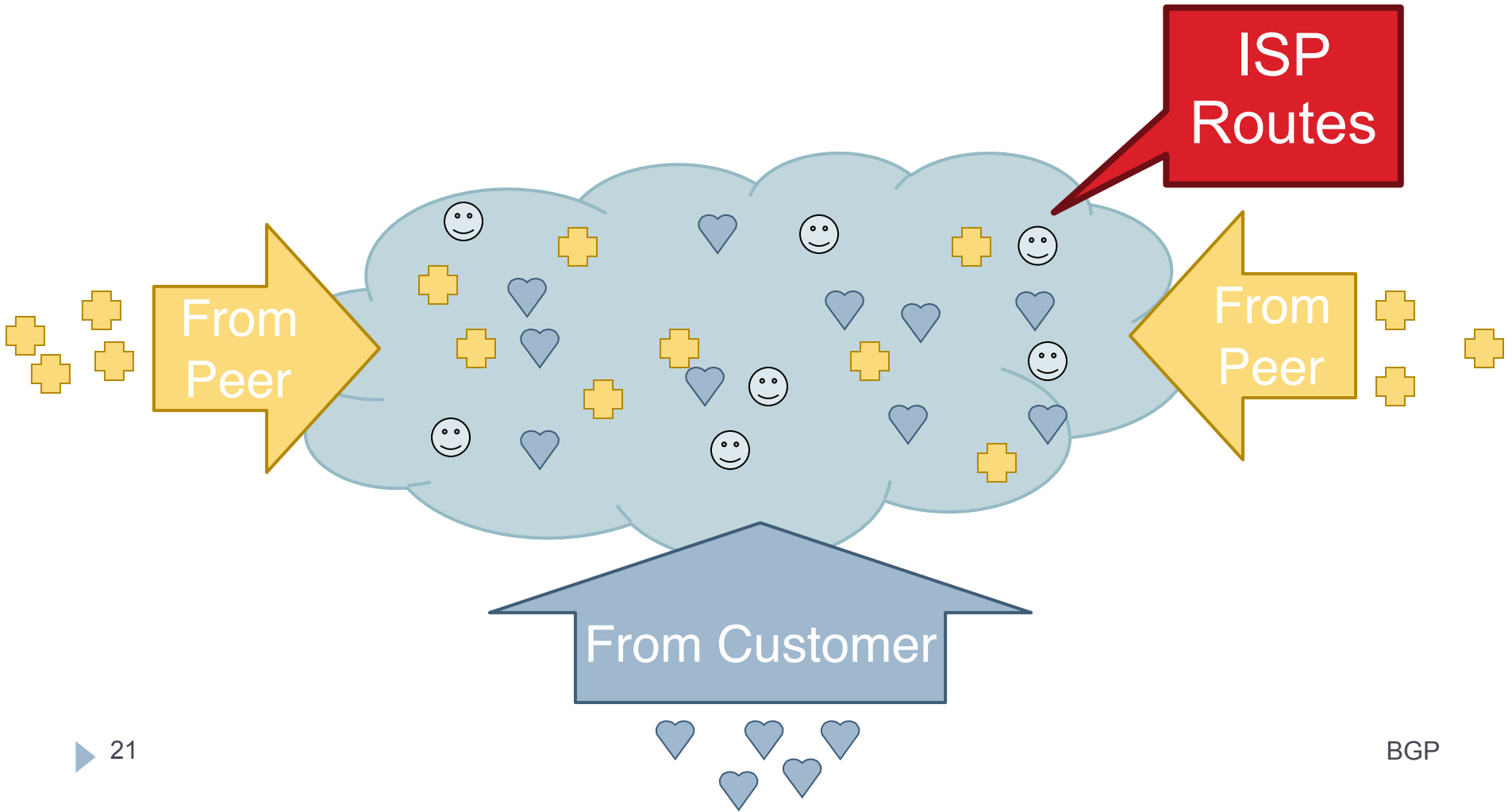
Importing Routes



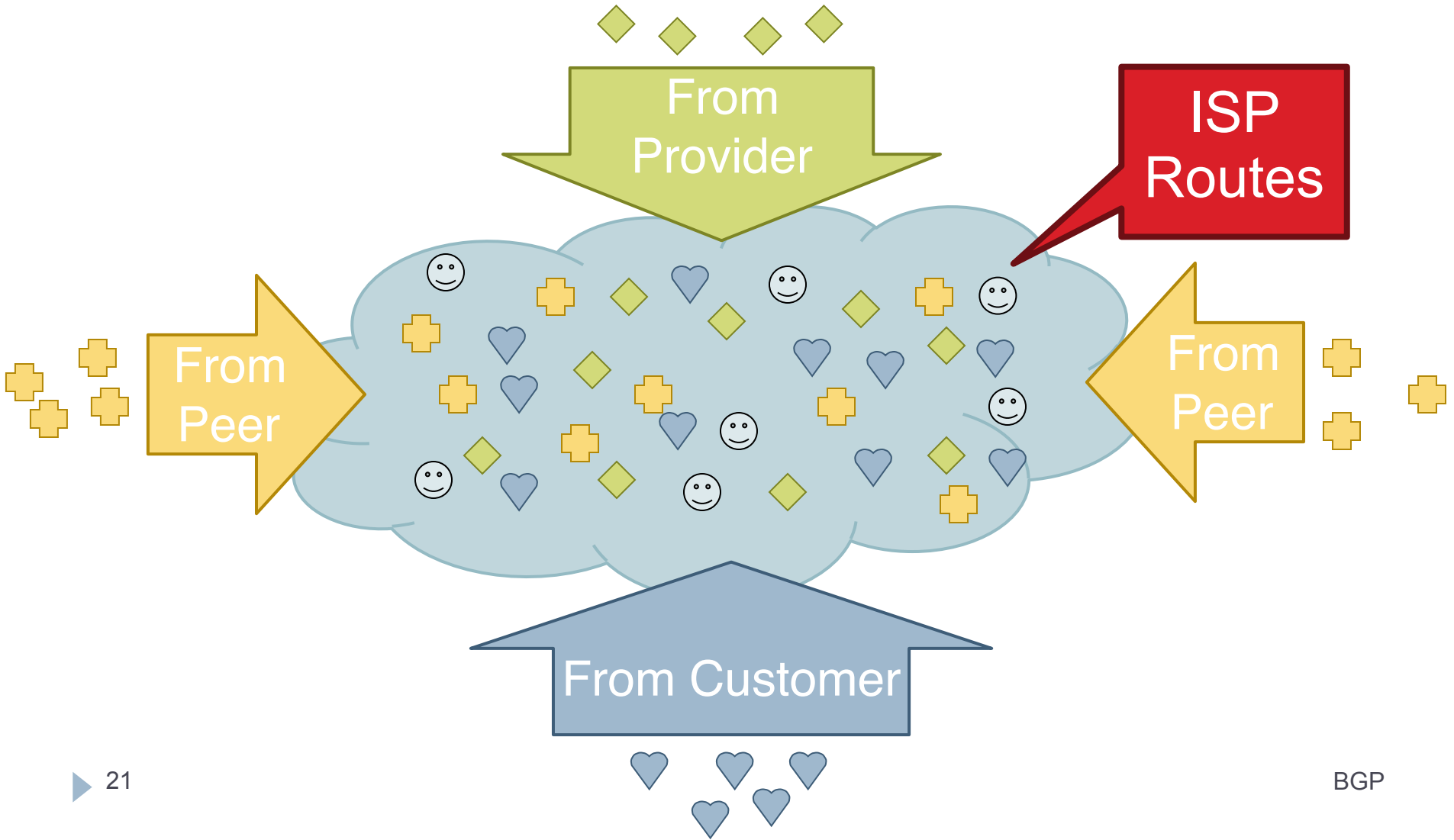
Importing Routes



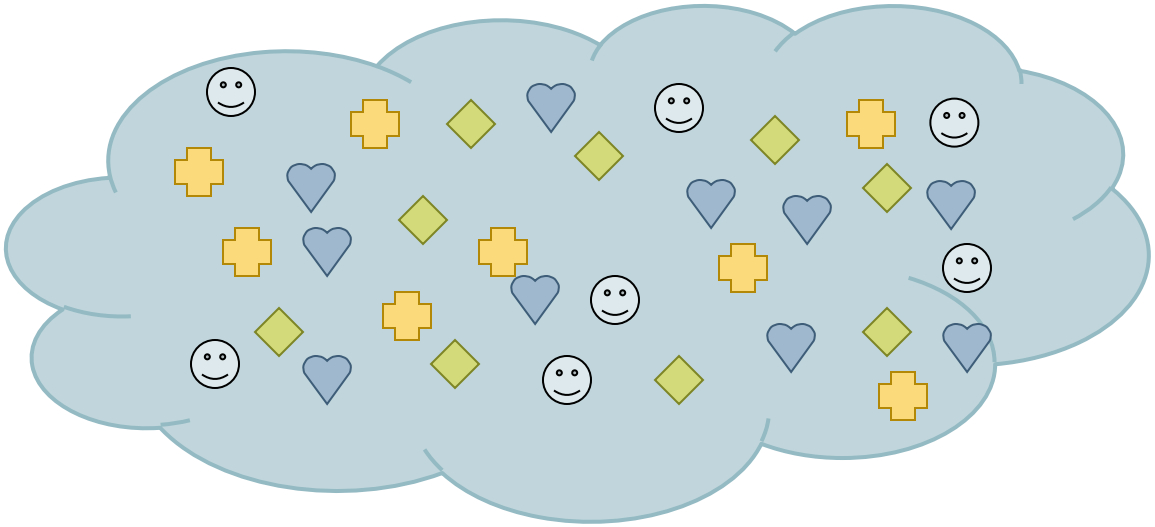
Importing Routes



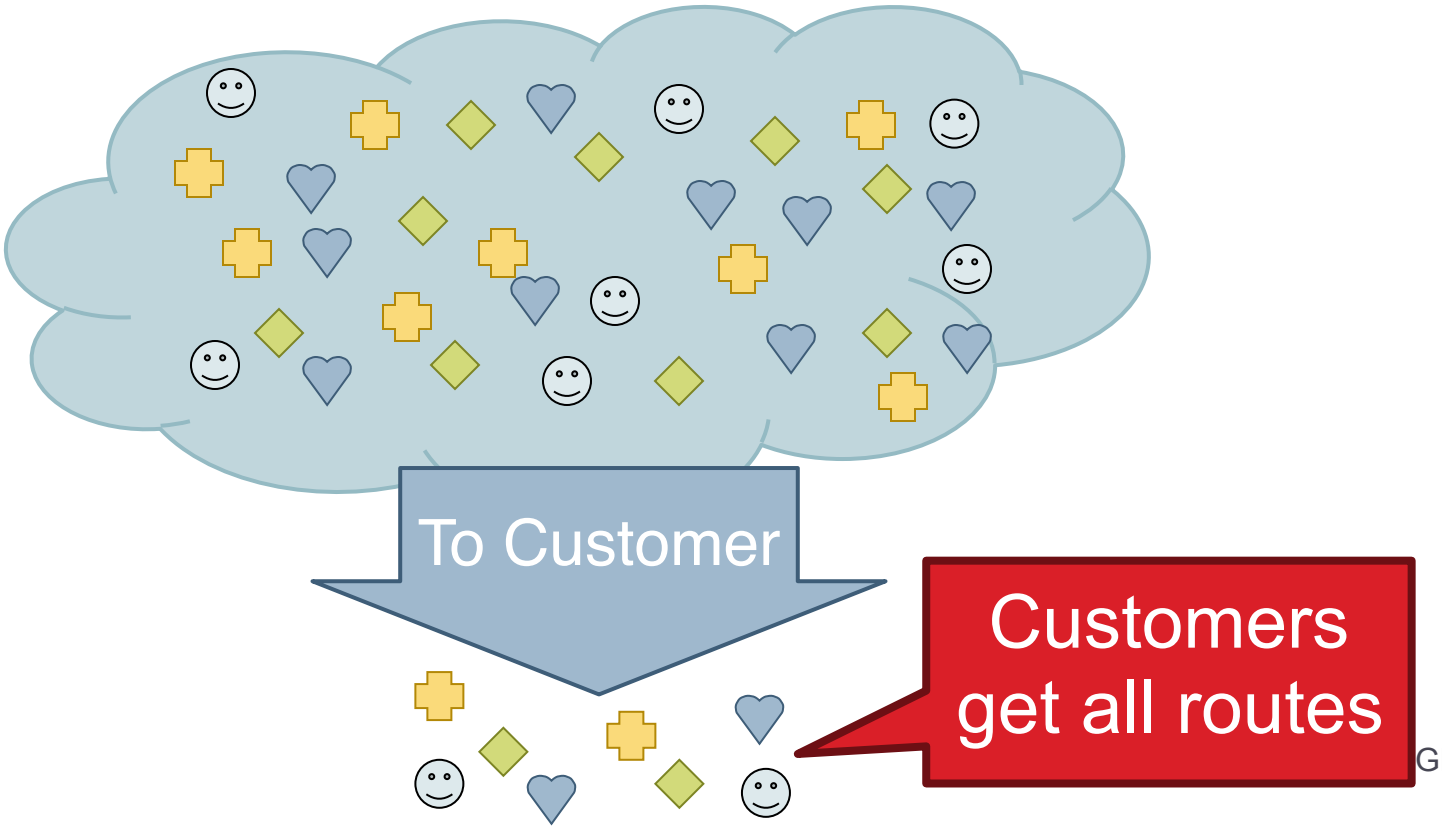
Importing Routes



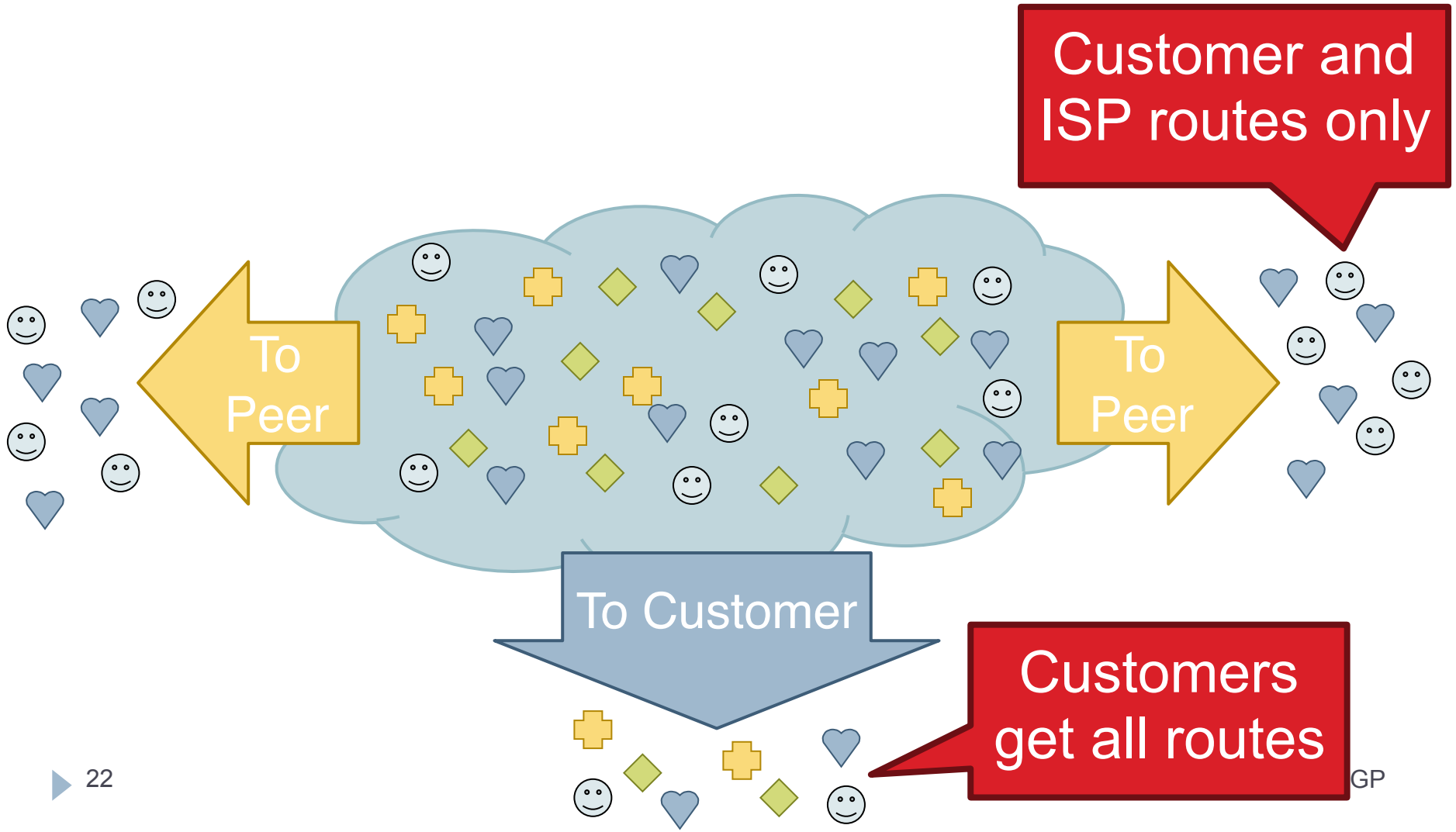
Exporting Routes



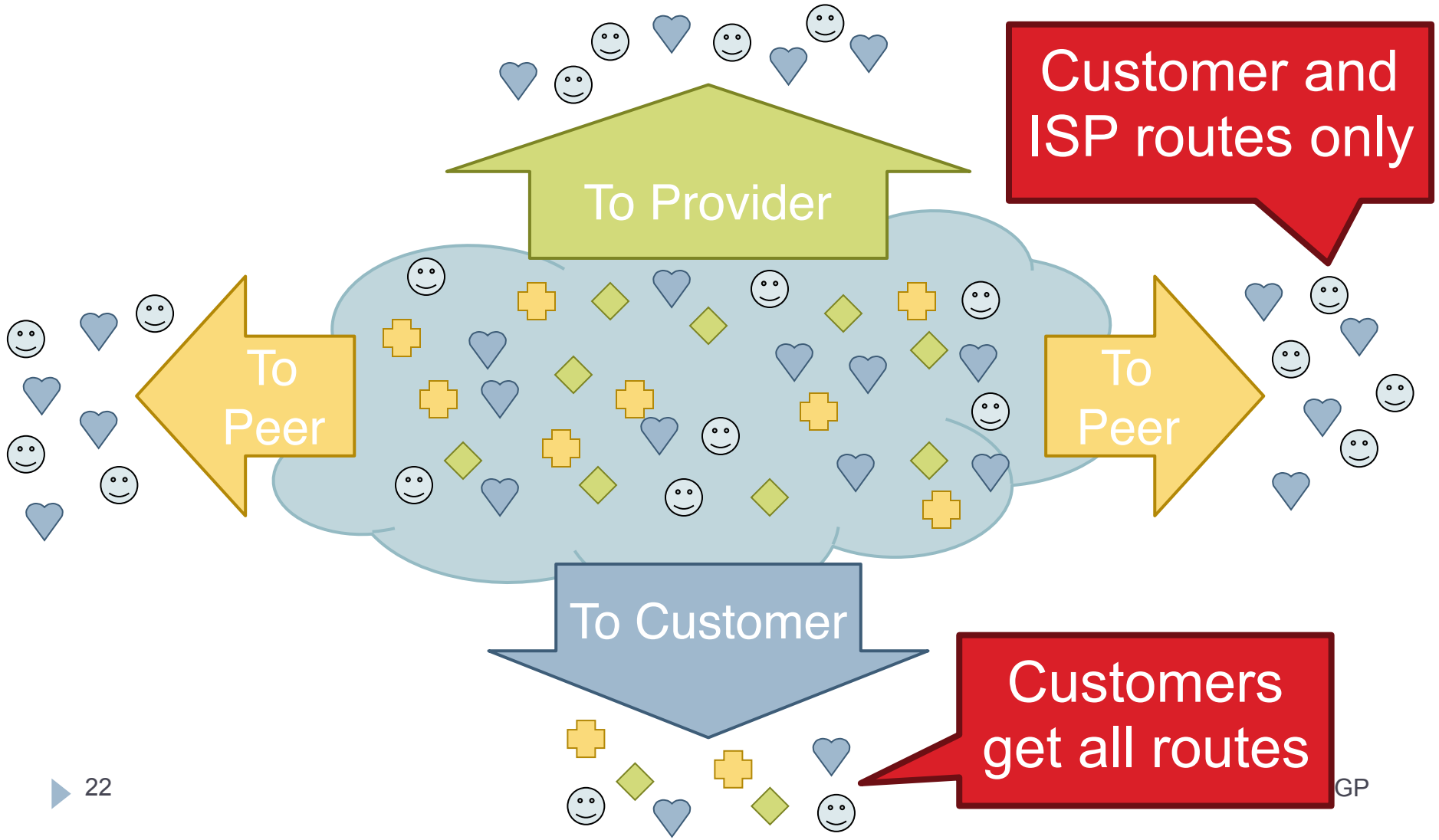
Exporting Routes



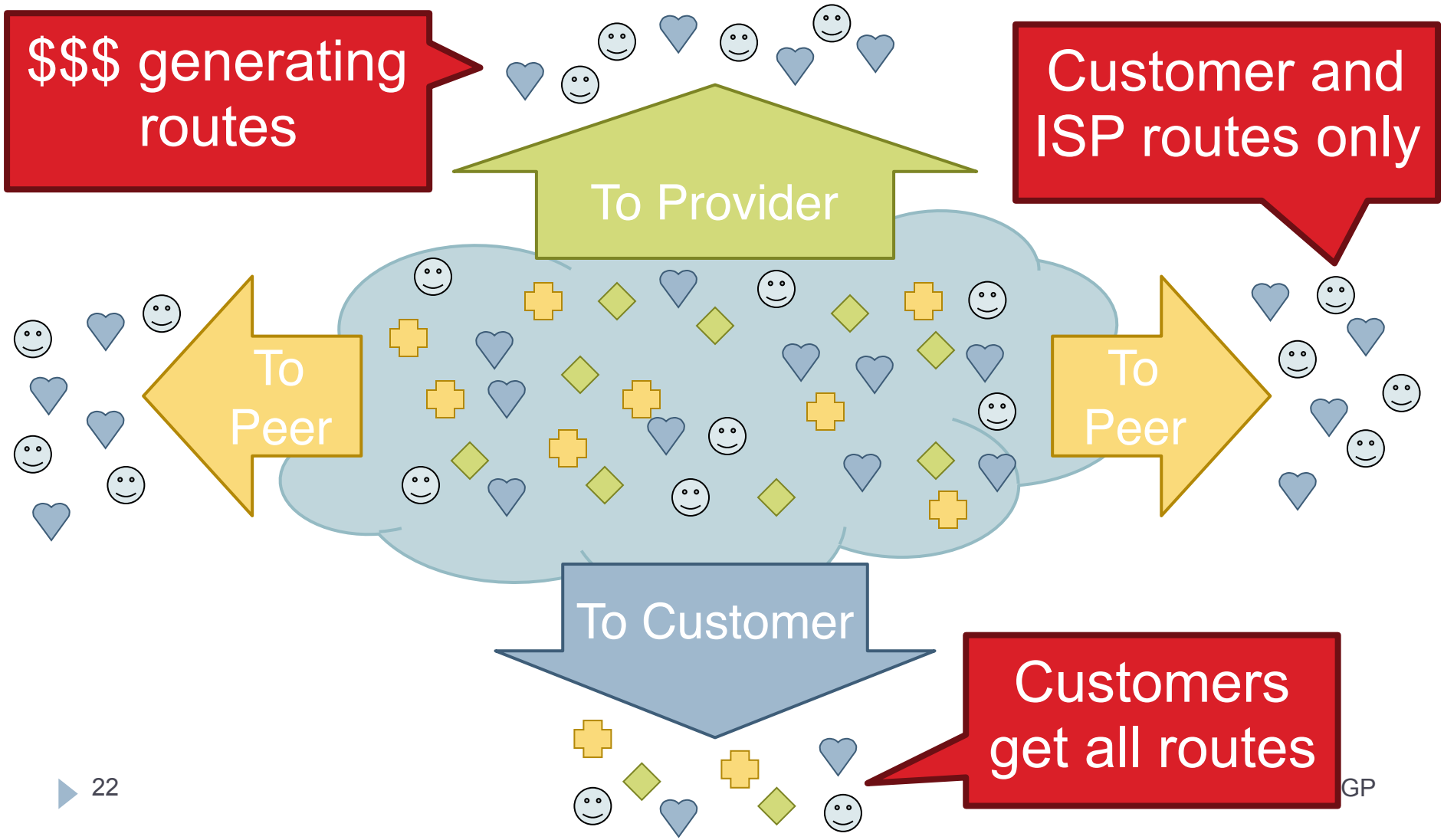
Exporting Routes



Exporting Routes



Exporting Routes

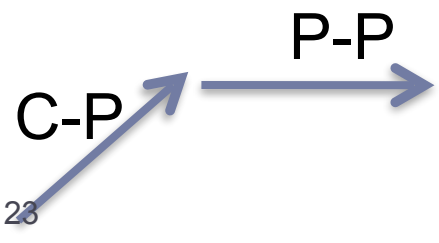


Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)

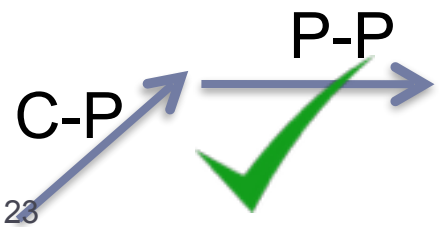
Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



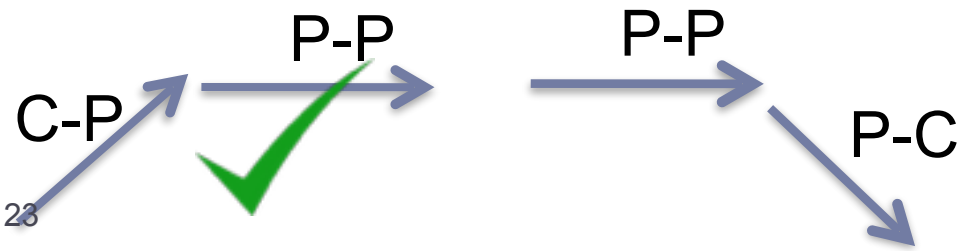
Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



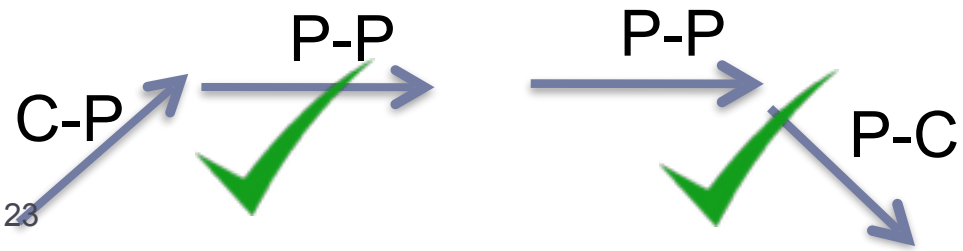
Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



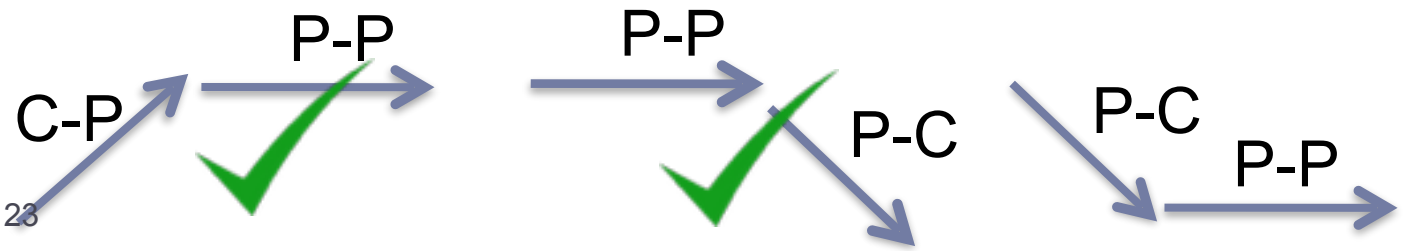
Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



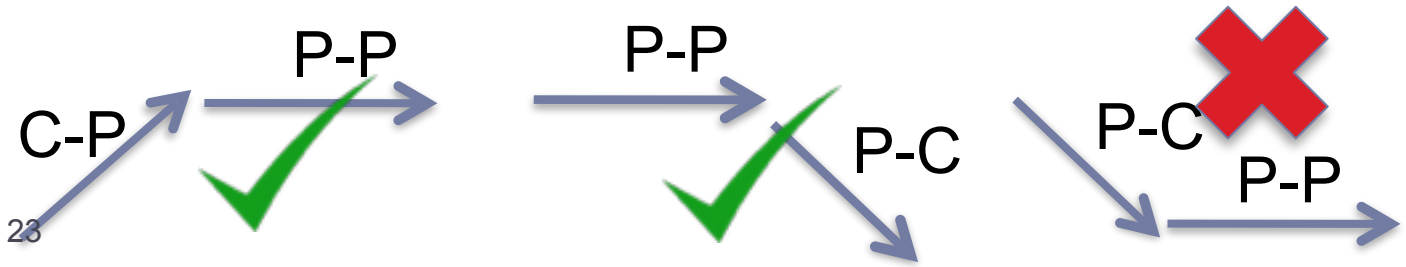
Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



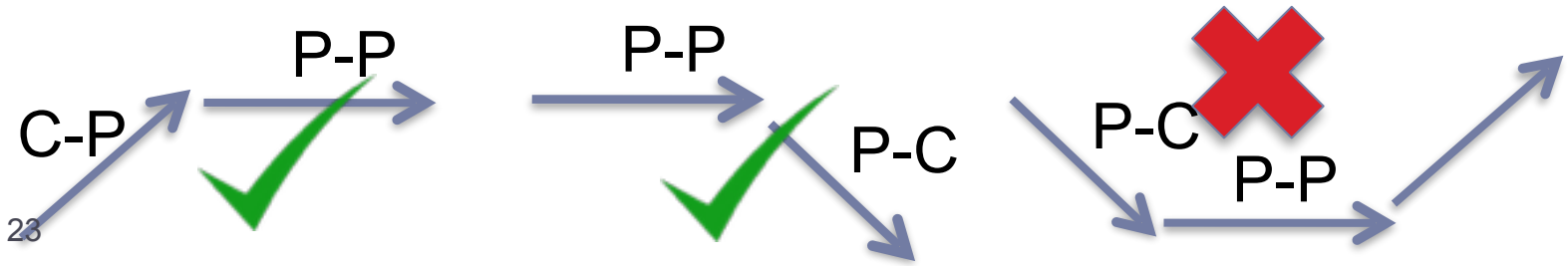
Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



Modeling BGP

- AS relationships
  - Customer/provider
  - Peer
  - Sibling, IXP
- Gao-Rexford model
  - AS prefers to use customer path, then peer, then provider
    - Follow the money!
  - Valley-free routing
  - Hierarchical view of routing (incorrect but frequently used)



AS Relationships: It's Complicated

- GR Model is strictly hierarchical
  - Each AS pair has exactly one relationship
  - Each relationship is the same for all prefixes
- In practice it's much more complicated
  - Rise of widespread peering
  - Regional, per-prefix peerings
  - Tier-1's being shoved out by "hypergiants"
  - IXPs dominating traffic volume
- Modeling is very hard, very prone to error
  - Huge potential impact for understanding Internet behavior

Other BGP Attributes

- AS_SET
  - Instead of a single AS appearing at a slot, it's a set of Ases
  - Why?
- Communities
  - Arbitrary number that is used by neighbors for routing decisions
    - Export this route only in Europe
    - Do not export to your peers
  - Usually stripped after first interdomain hop
  - Why?
- Prepending
  - Lengthening the route by adding multiple instances of ASN
  - Why?

2: Stable path problem.

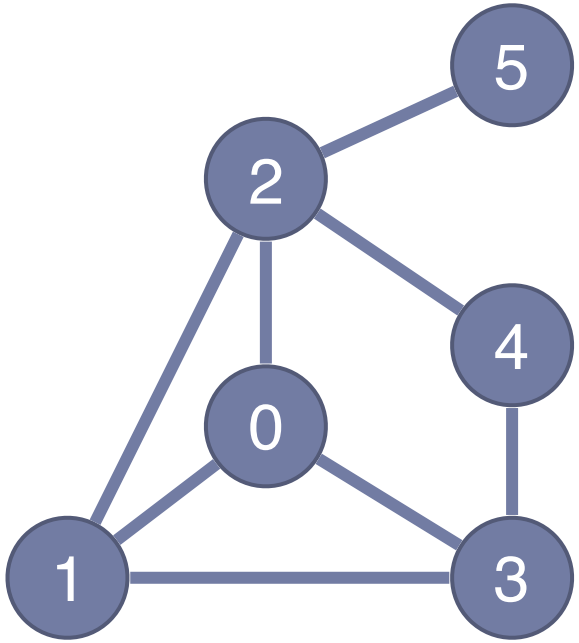
What Problem is BGP Solving?

Underlying Problem	Distributed Solution
Shortest Paths	RIP, OSPF, IS-IS, etc.
???	BGP

- Knowing ??? can:
  - Aid in the analysis of BGP policy
  - Aid in the design of BGP extensions
  - Help explain BGP routing anomalies
  - Give us a deeper understanding of the protocol

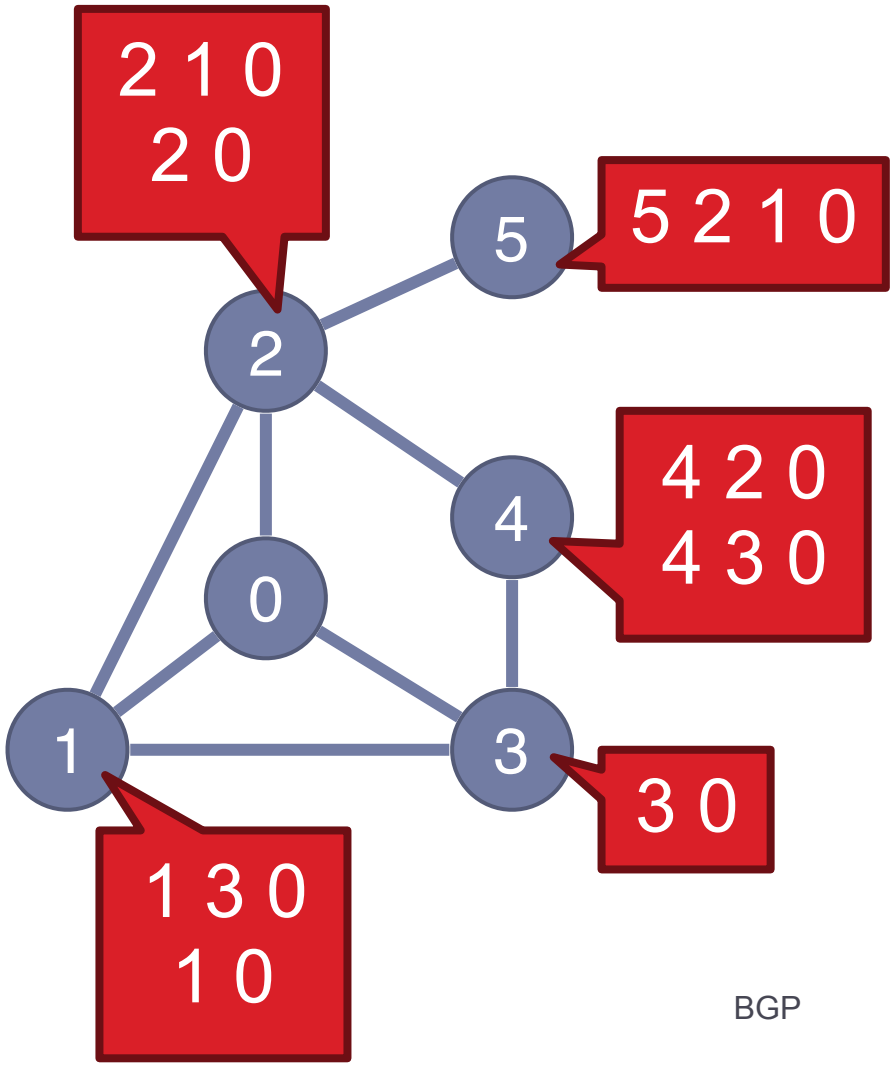
The Stable Paths Problem

- An instance of the SPP:
  - Graph of nodes and edges
  - Node 0, called the origin



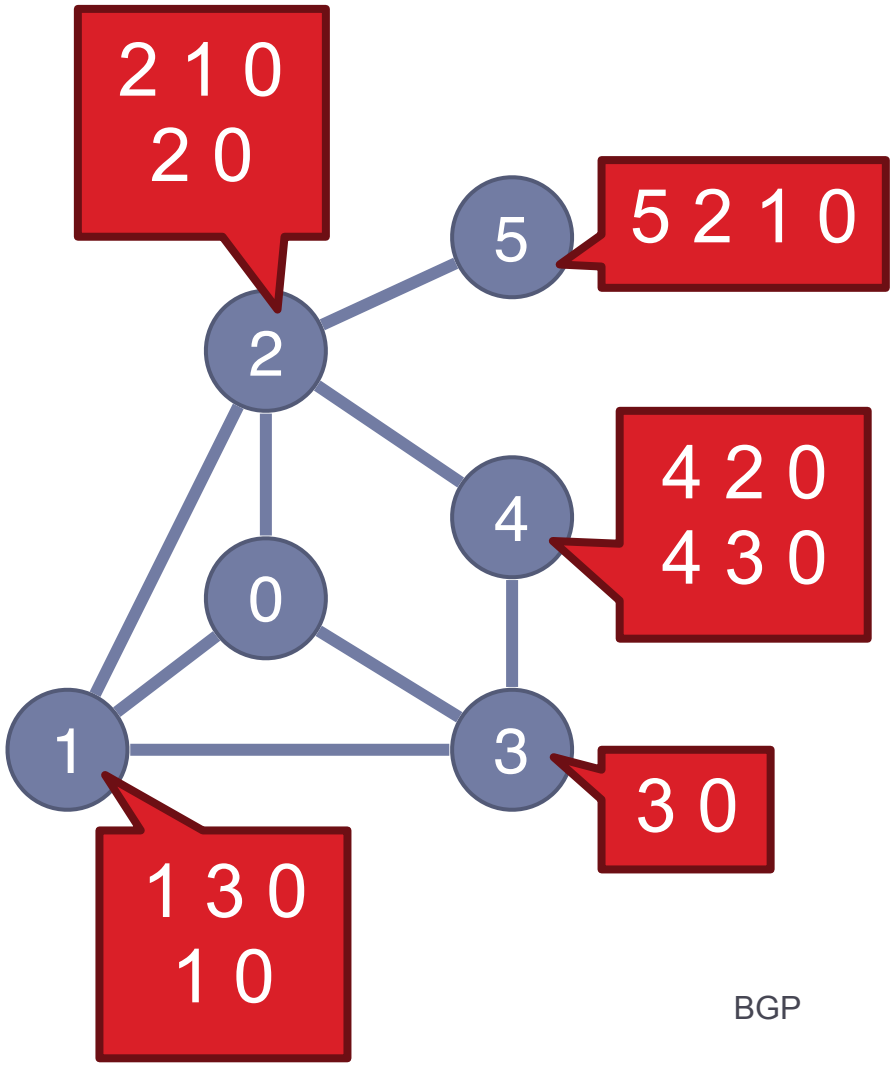
The Stable Paths Problem

- An instance of the SPP:
  - Graph of nodes and edges
  - Node 0, called the origin
  - A set of permitted paths from each node to the origin
    - Each set contains the null path



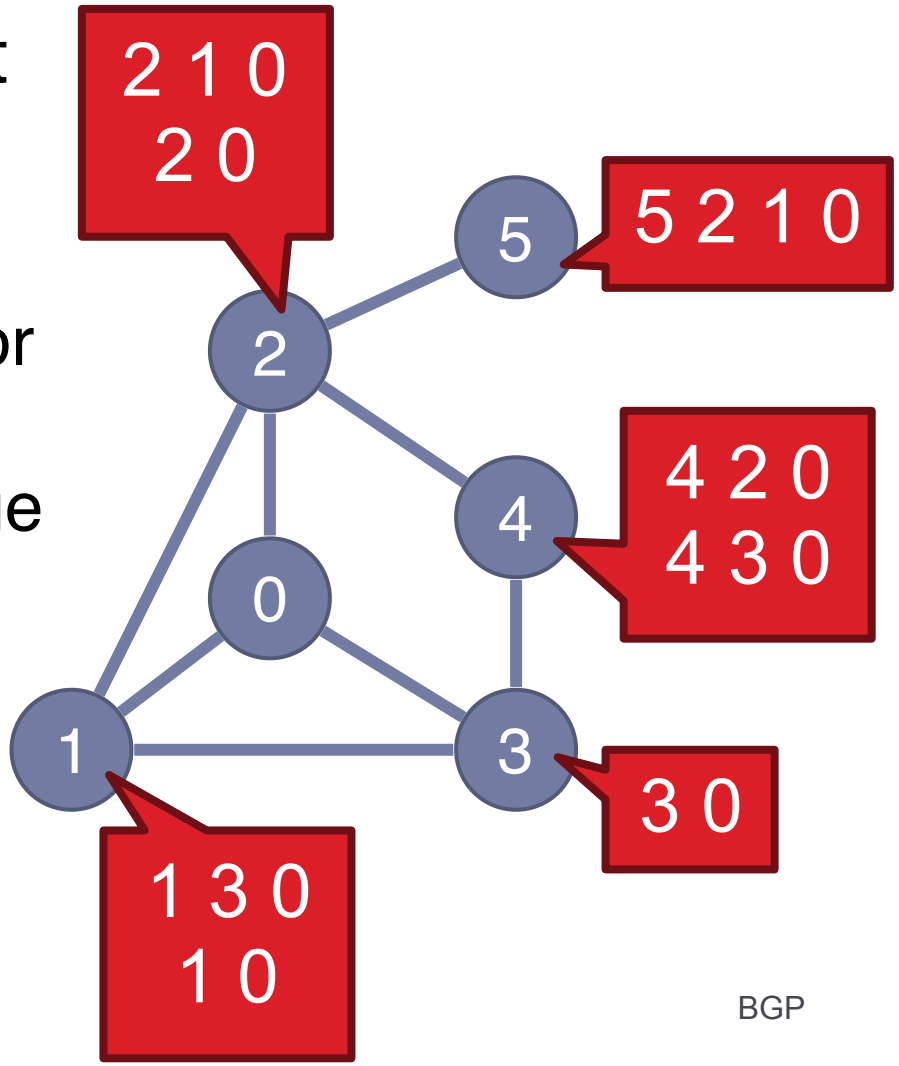
The Stable Paths Problem

- An instance of the SPP:
  - Graph of nodes and edges
  - Node 0, called the origin
  - A set of permitted paths from each node to the origin
    - Each set contains the null path
  - Each set of paths is ranked
    - Null path is always least preferred



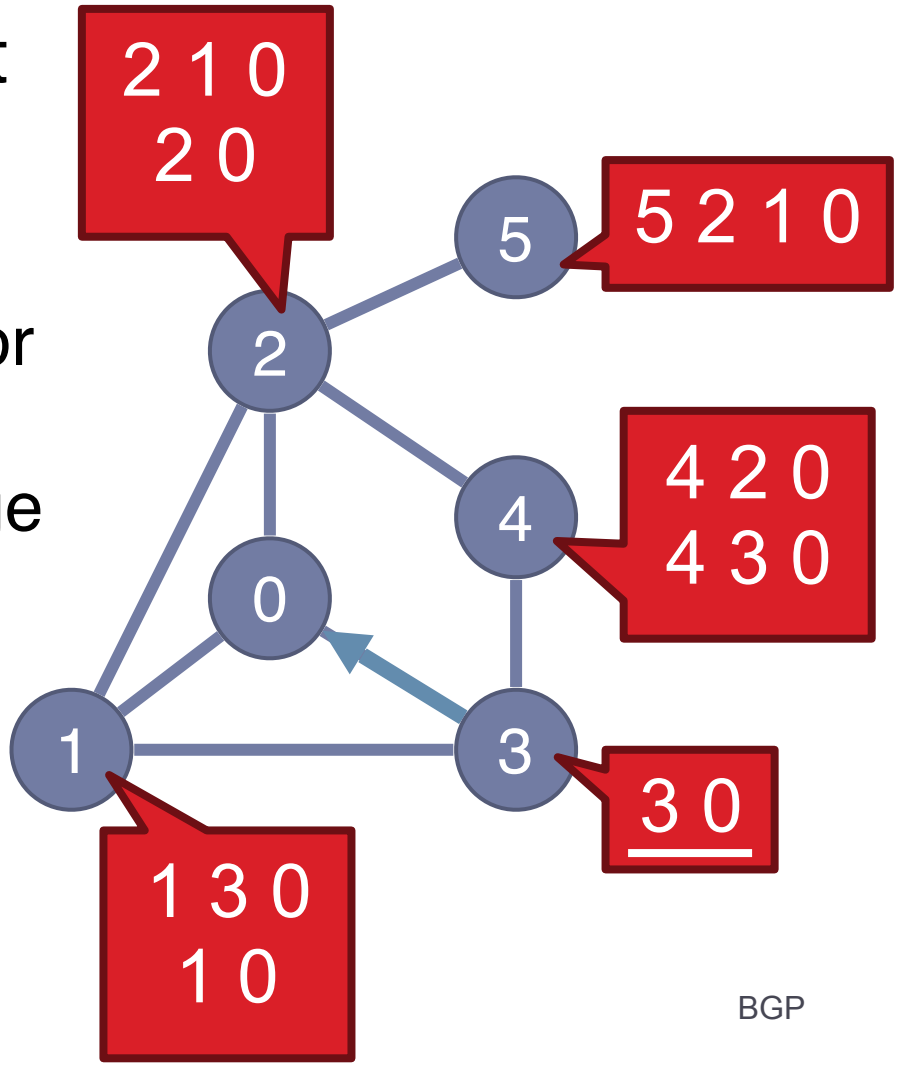
A Solution to the SPP

- A solution is an assignment of permitted paths to each node such that:
  - Node *u*'s path is either null or *uwP*, where path *wP* is assigned to node *w* and edge *u* → *w* exists
  - Each node is assigned the higest ranked path that is consistent with their neighbors



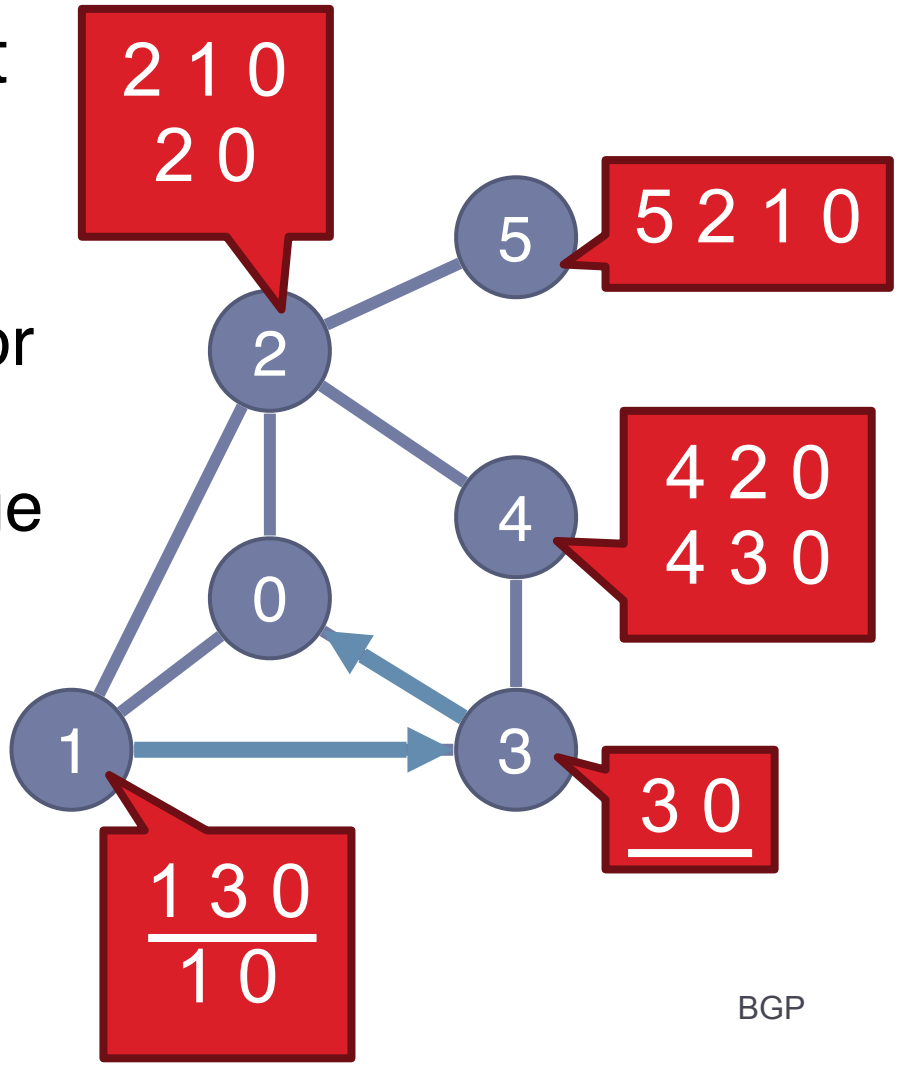
A Solution to the SPP

- A solution is an assignment of permitted paths to each node such that:
  - Node *u*'s path is either null or *uwP*, where path *wP* is assigned to node *w* and edge *u* → *w* exists
  - Each node is assigned the higest ranked path that is consistent with their neighbors



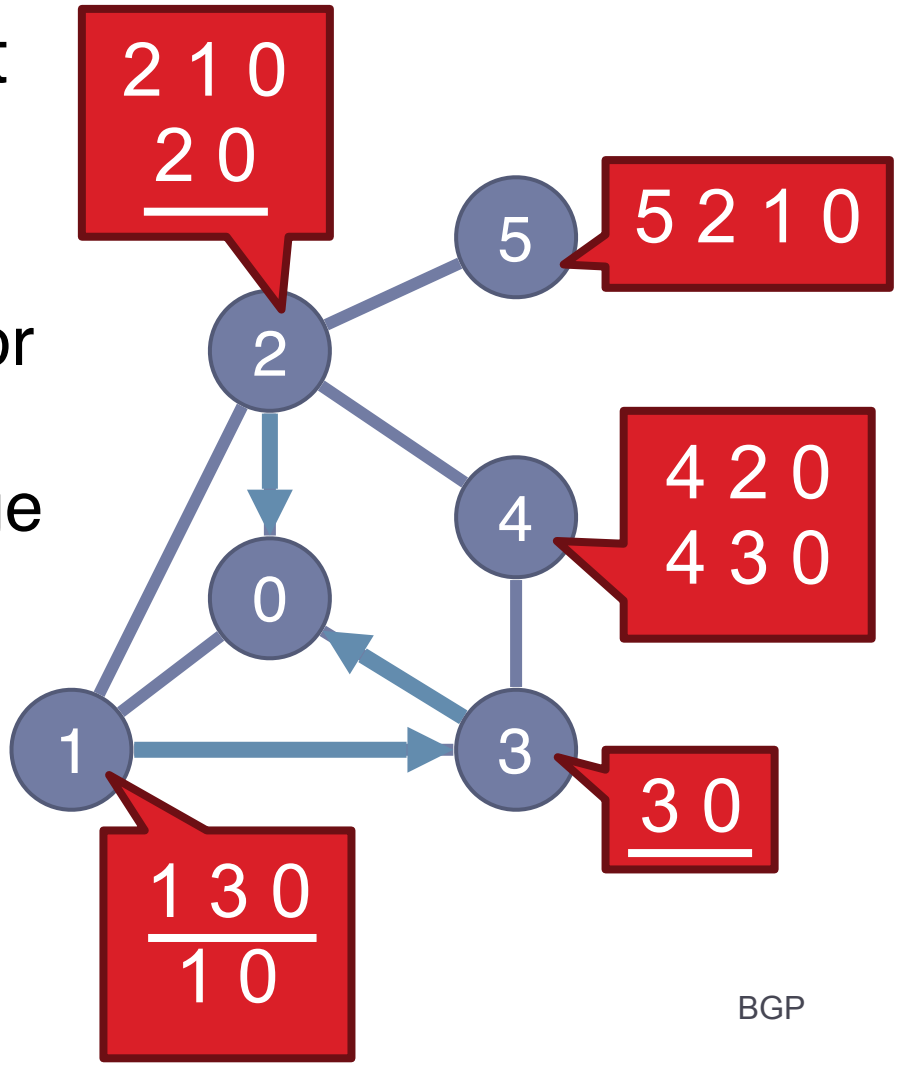
A Solution to the SPP

- A solution is an assignment of permitted paths to each node such that:
  - Node *u*'s path is either null or *uwP*, where path *wP* is assigned to node *w* and edge *u* → *w* exists
  - Each node is assigned the higest ranked path that is consistent with their neighbors



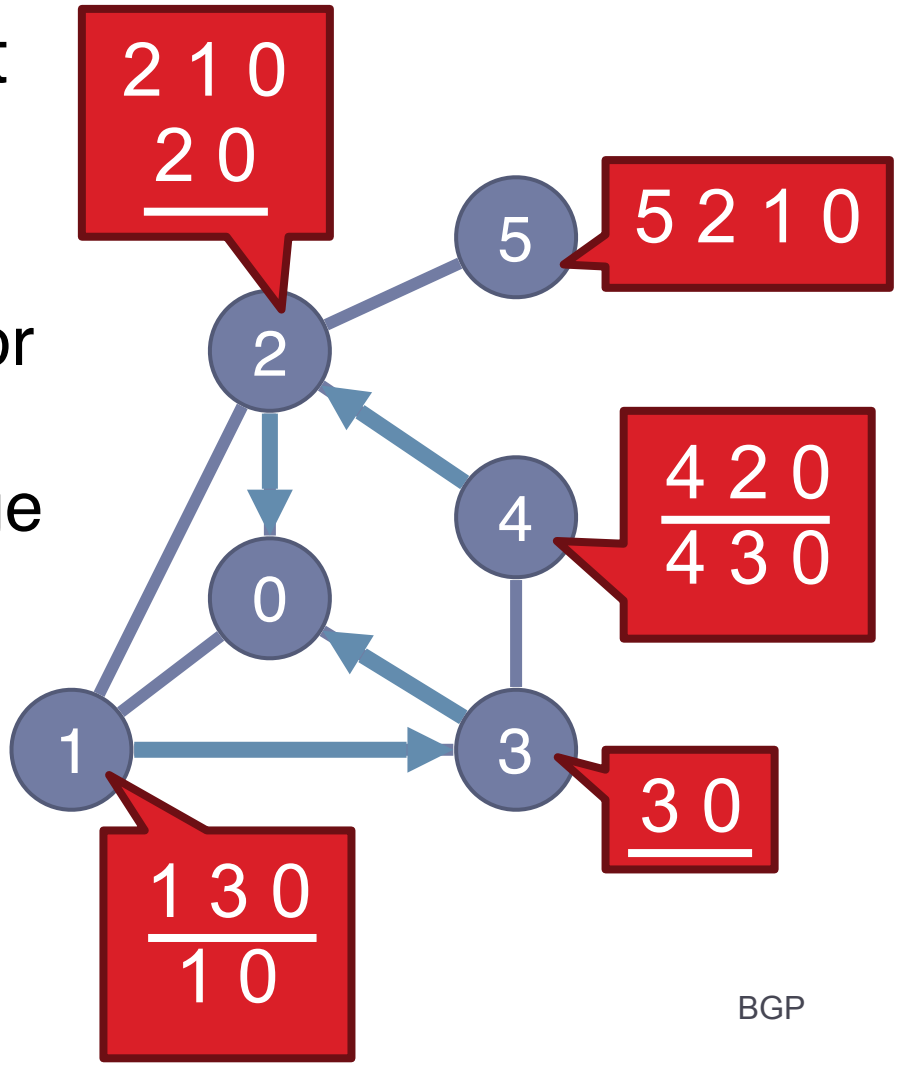
A Solution to the SPP

- A solution is an assignment of permitted paths to each node such that:
  - Node *u*'s path is either null or *uwP*, where path *wP* is assigned to node *w* and edge *u* → *w* exists
  - Each node is assigned the higest ranked path that is consistent with their neighbors



A Solution to the SPP

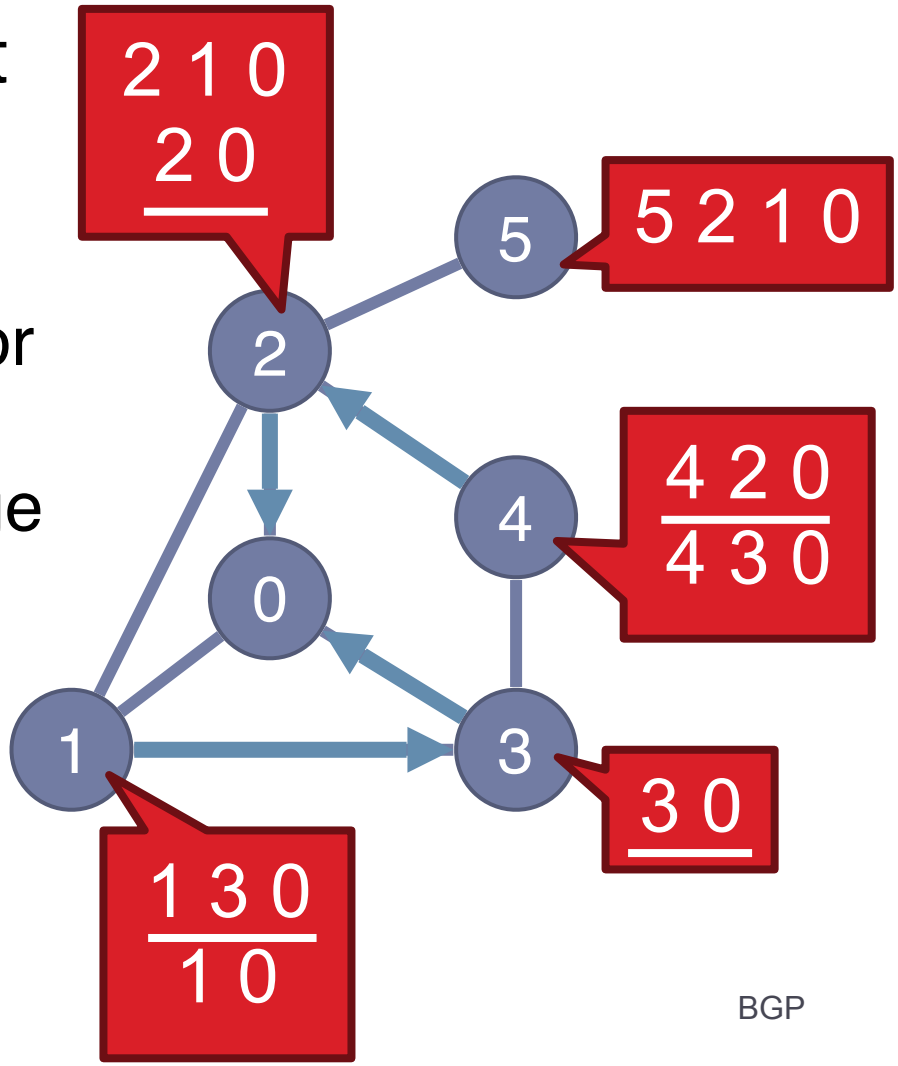
- A solution is an assignment of permitted paths to each node such that:
  - Node *u*'s path is either null or *uwP*, where path *wP* is assigned to node *w* and edge *u* → *w* exists
  - Each node is assigned the higest ranked path that is consistent with their neighbors



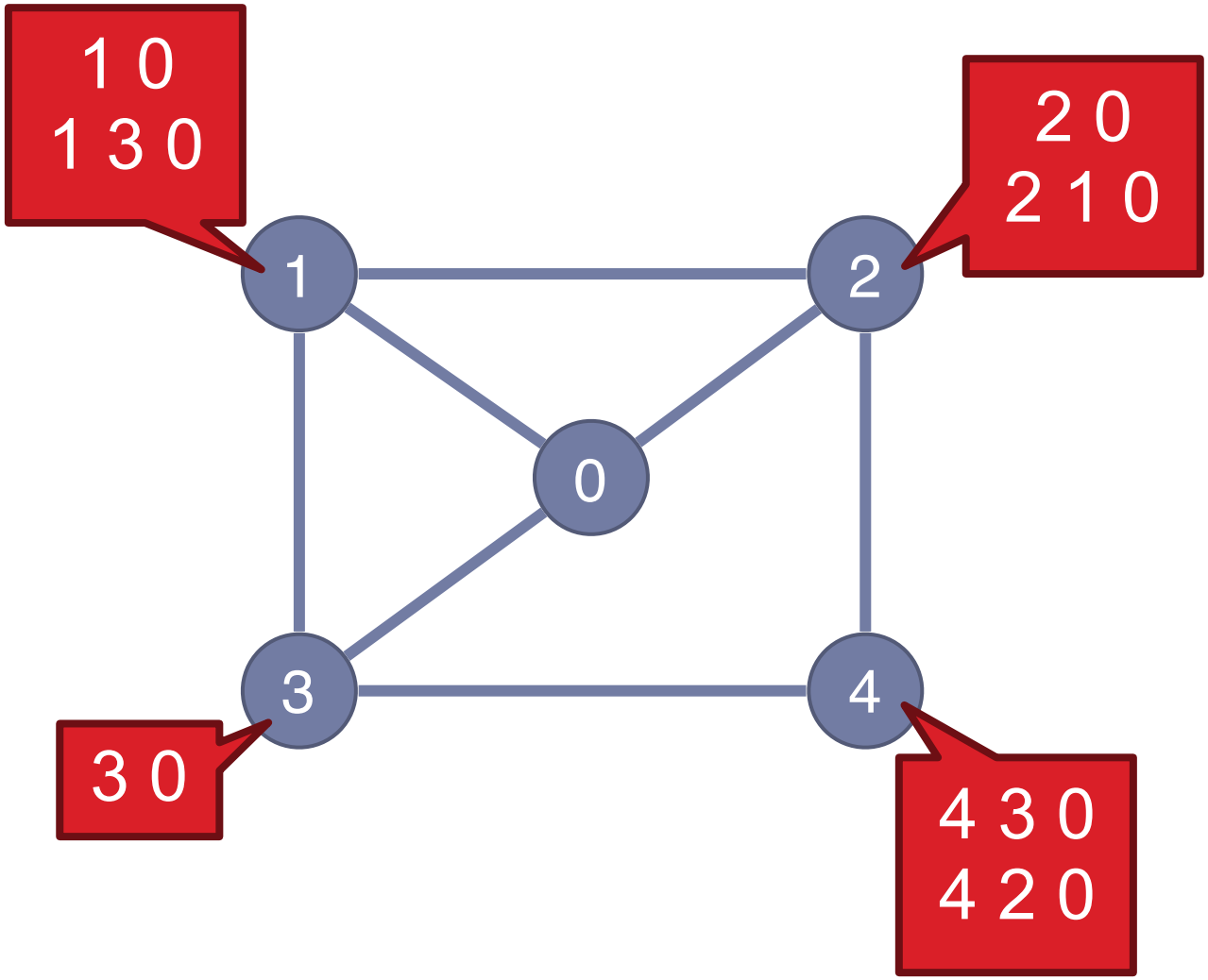
A Solution to the SPP

- A solution is an assignment of permitted paths to each
  ll or
  edge
  e
  nigest ranked path that is consistent
  with their neighbors

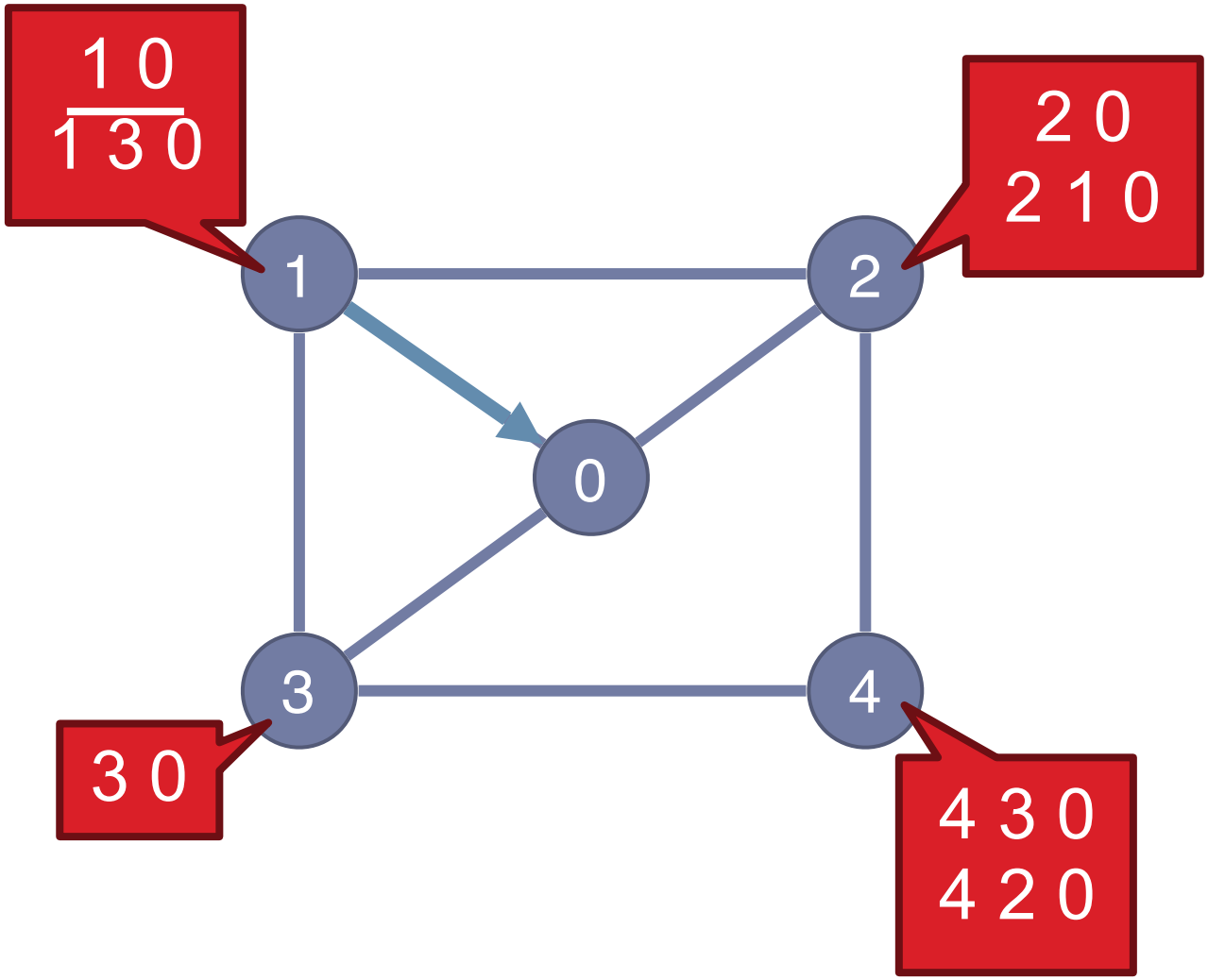
Solutions need not use the shortest paths, or form a spanning tree



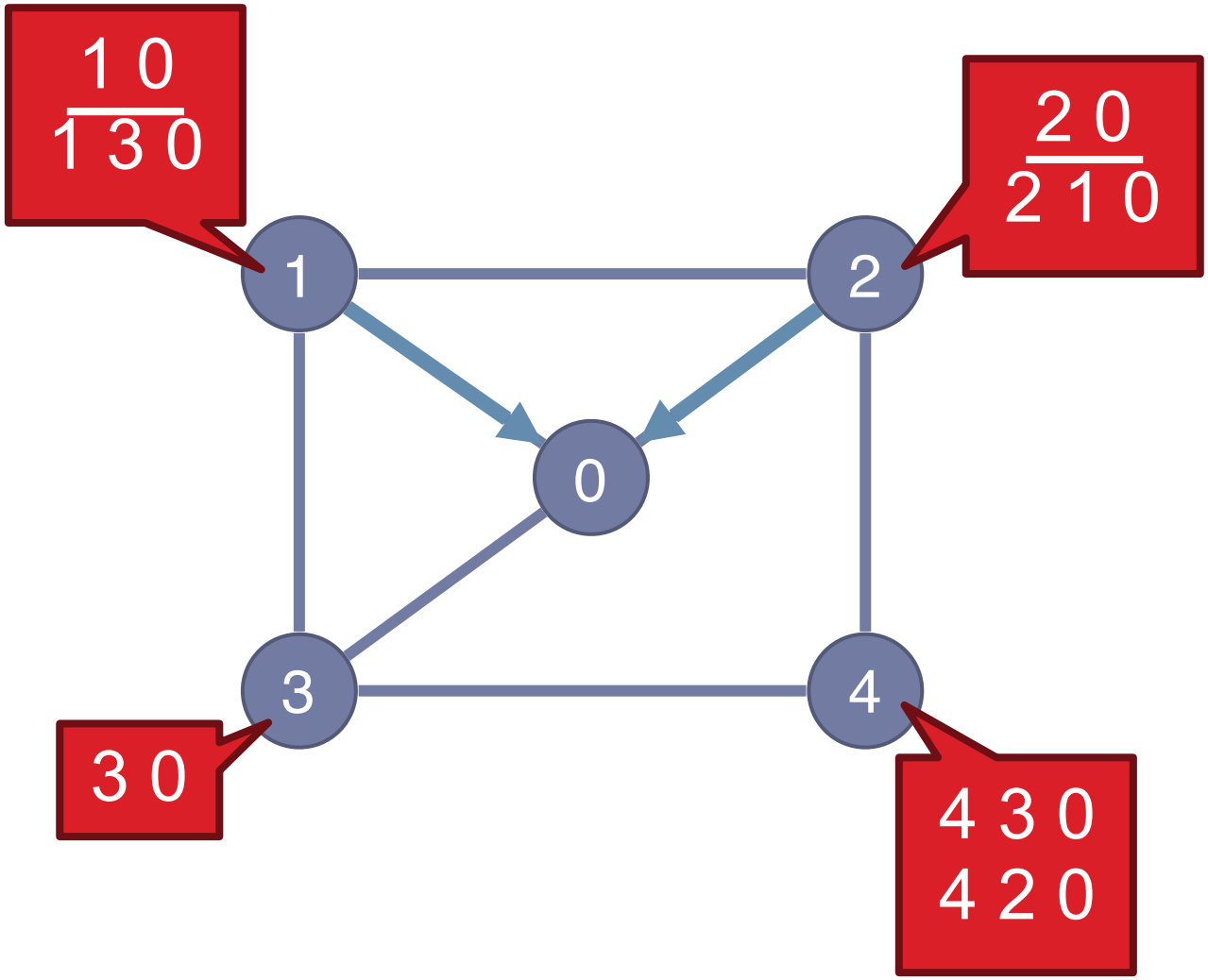
Simple SPP Example



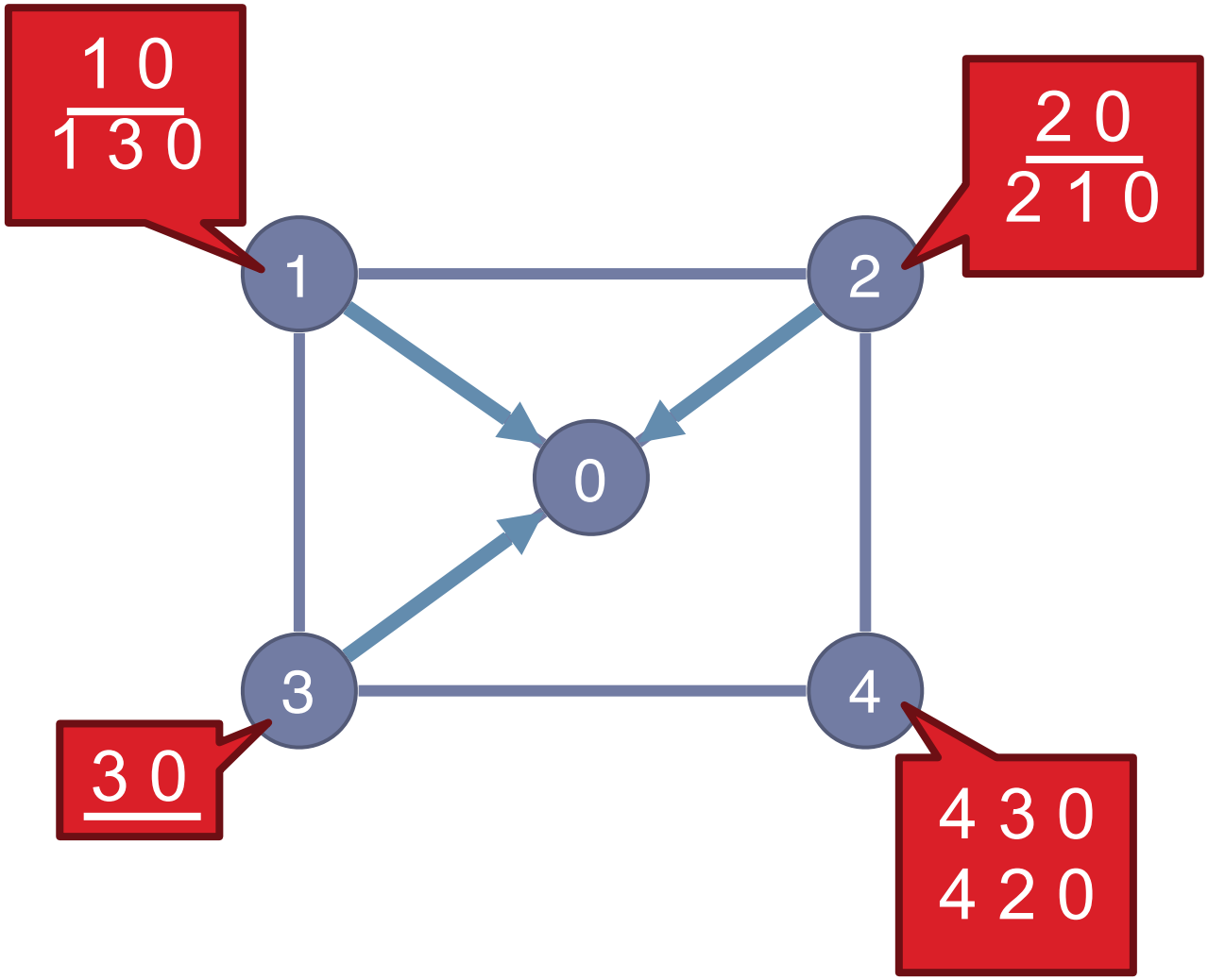
Simple SPP Example



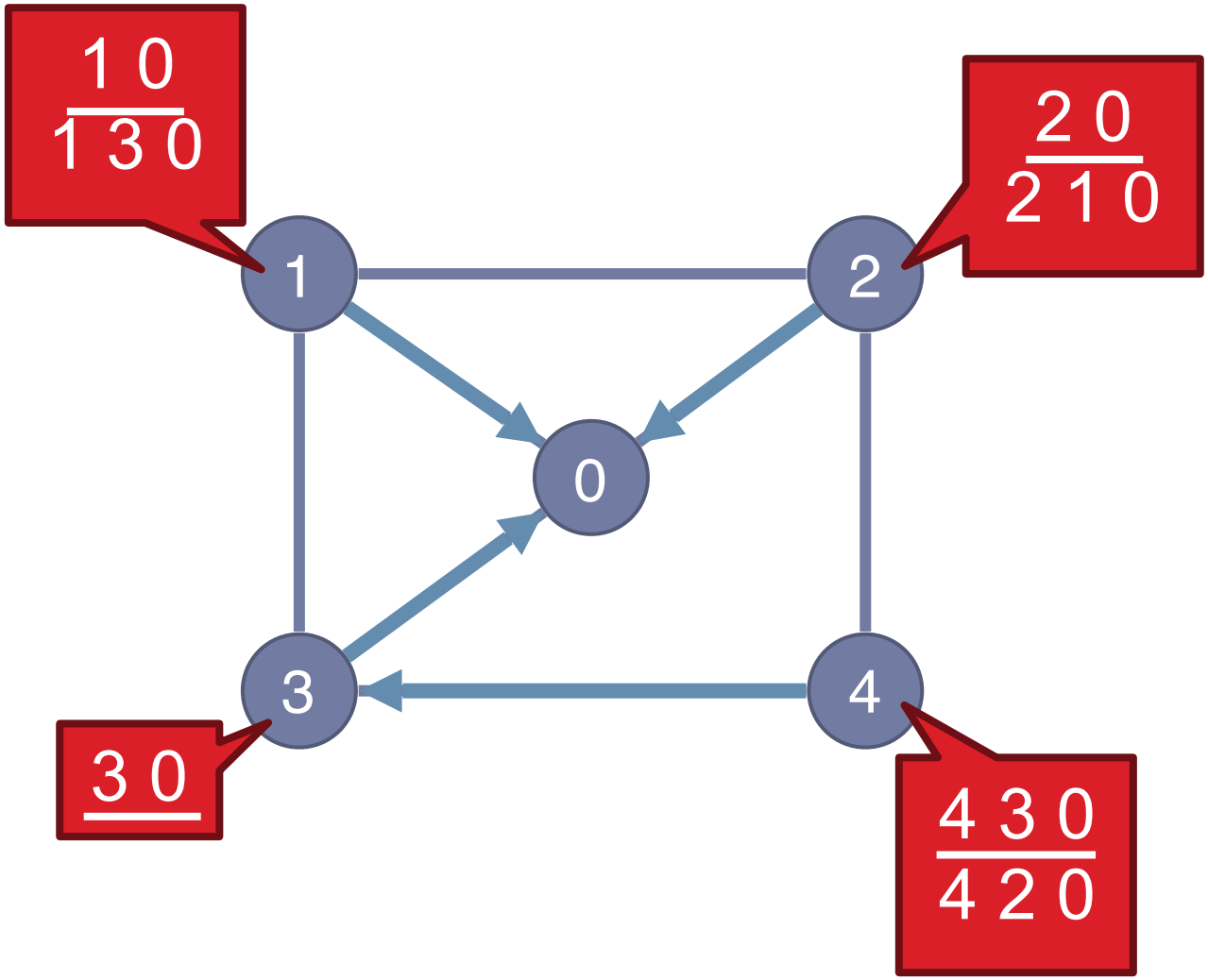
Simple SPP Example



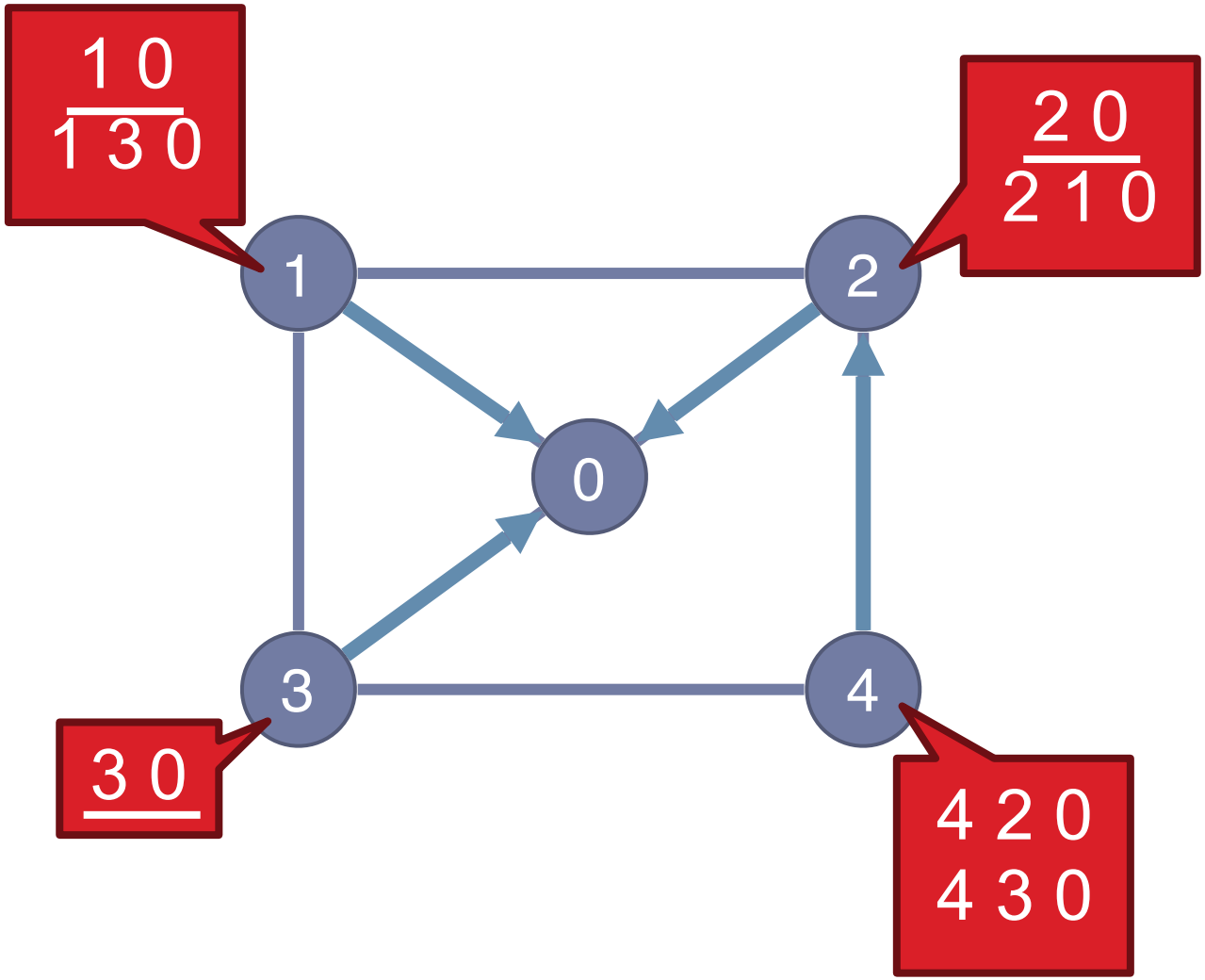
Simple SPP Example



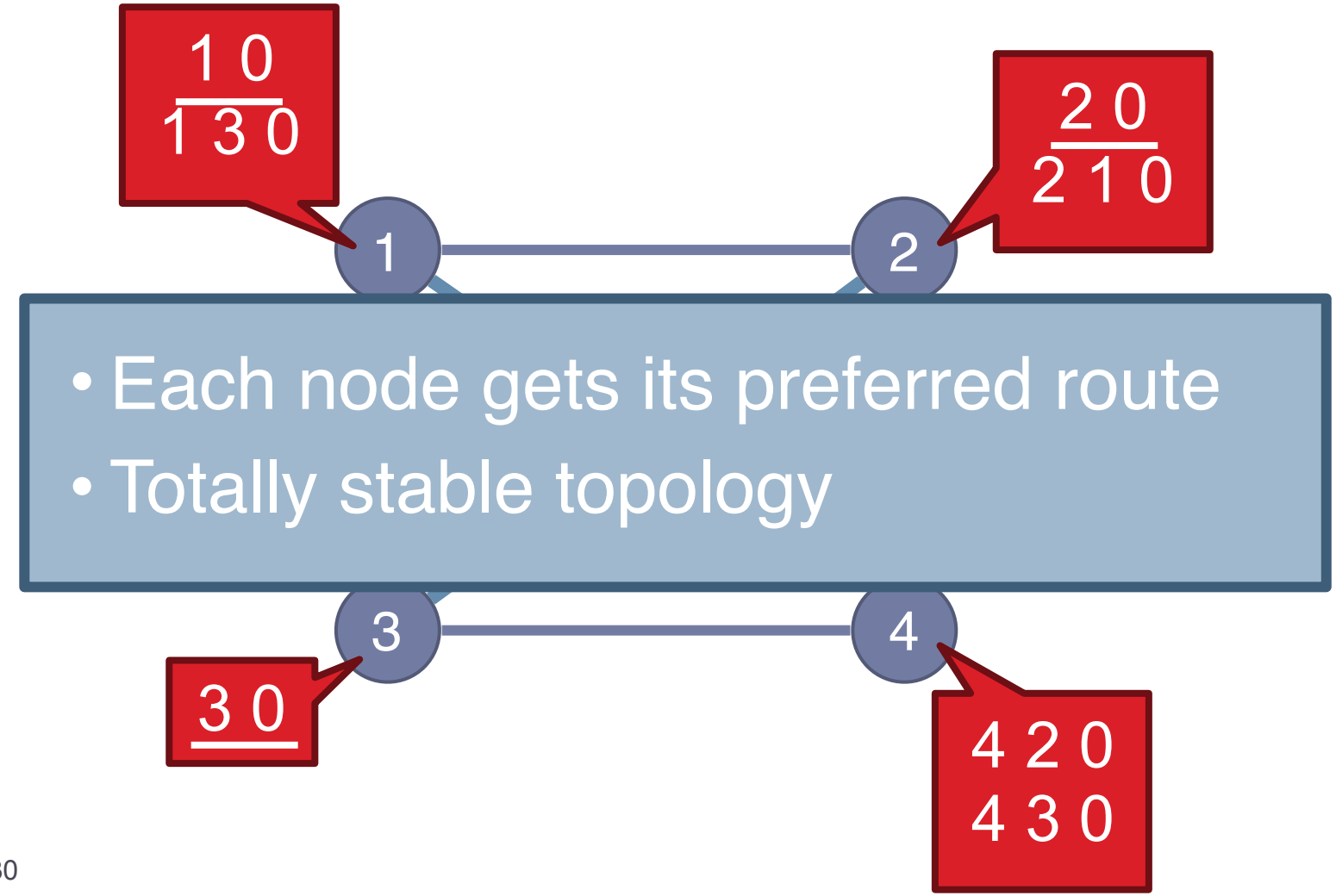
Simple SPP Example



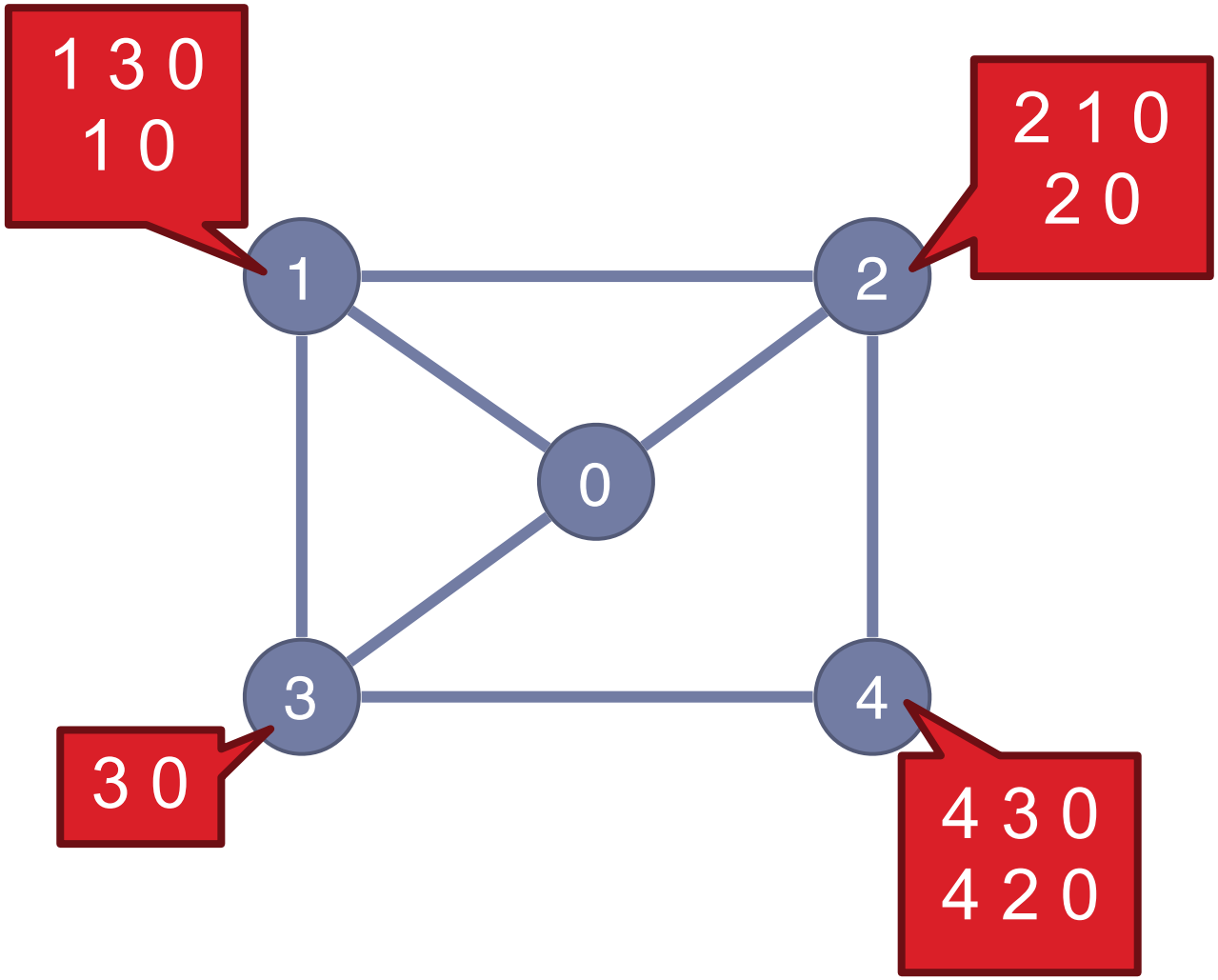
Simple SPP Example



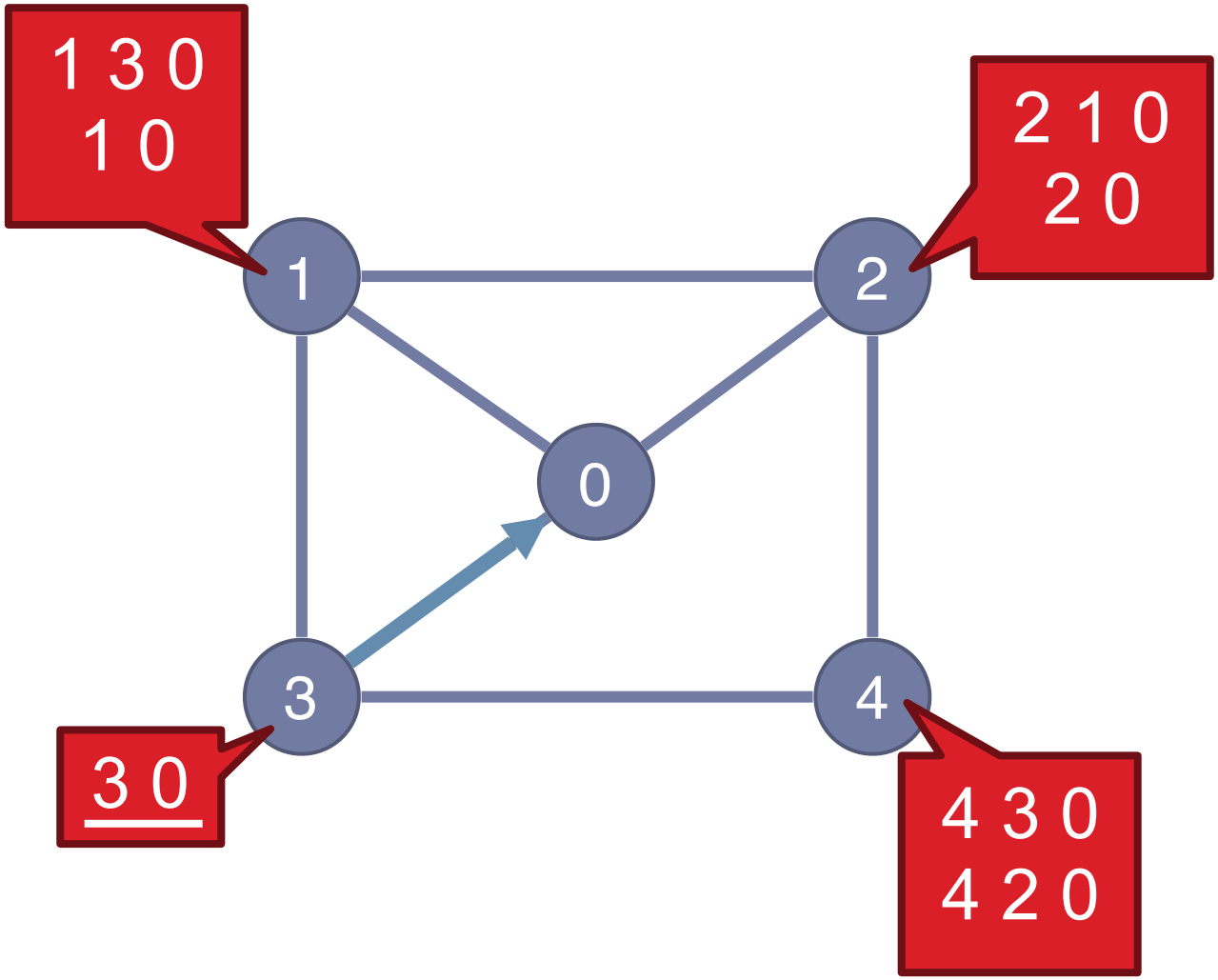
Simple SPP Example



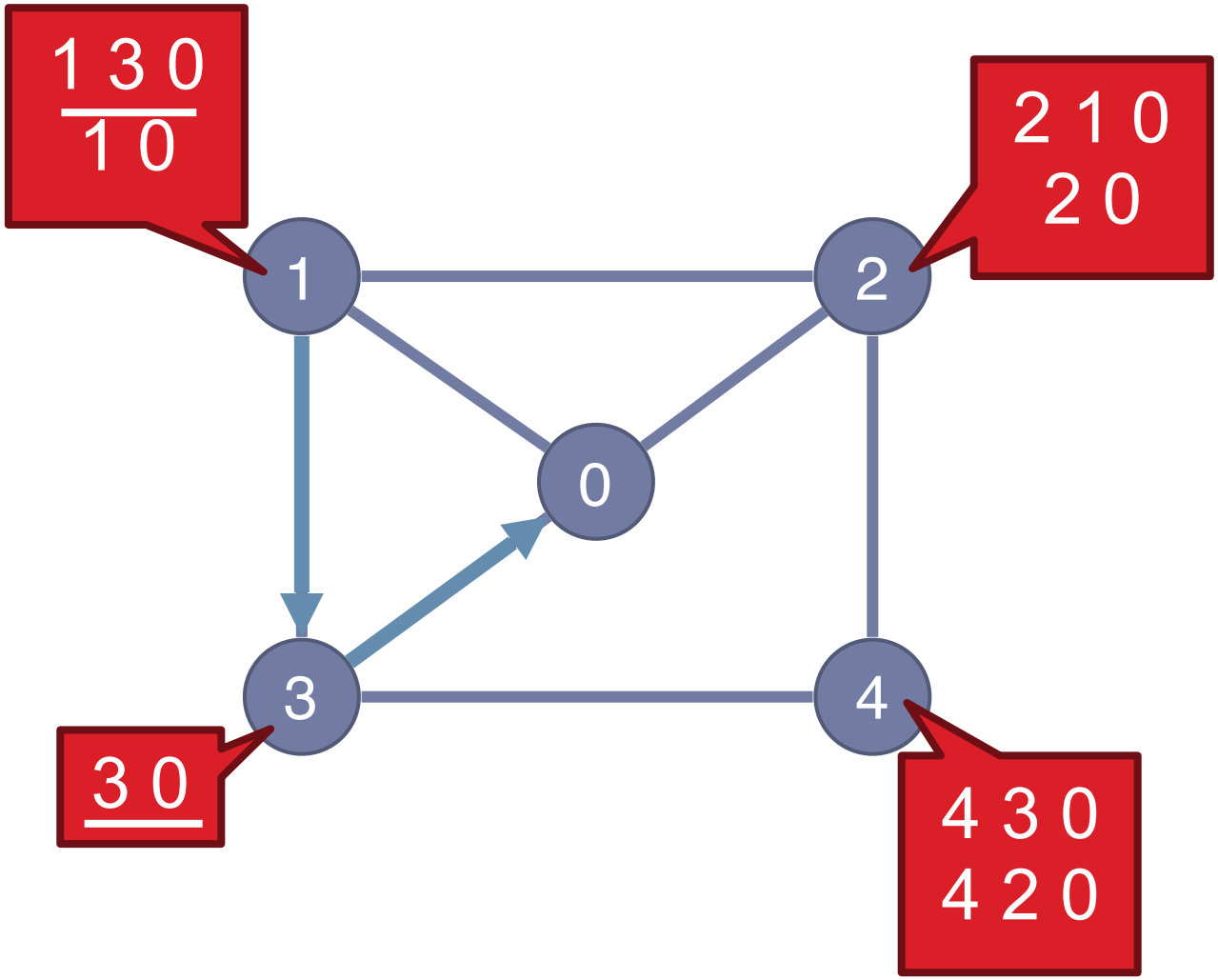
Good Gadget



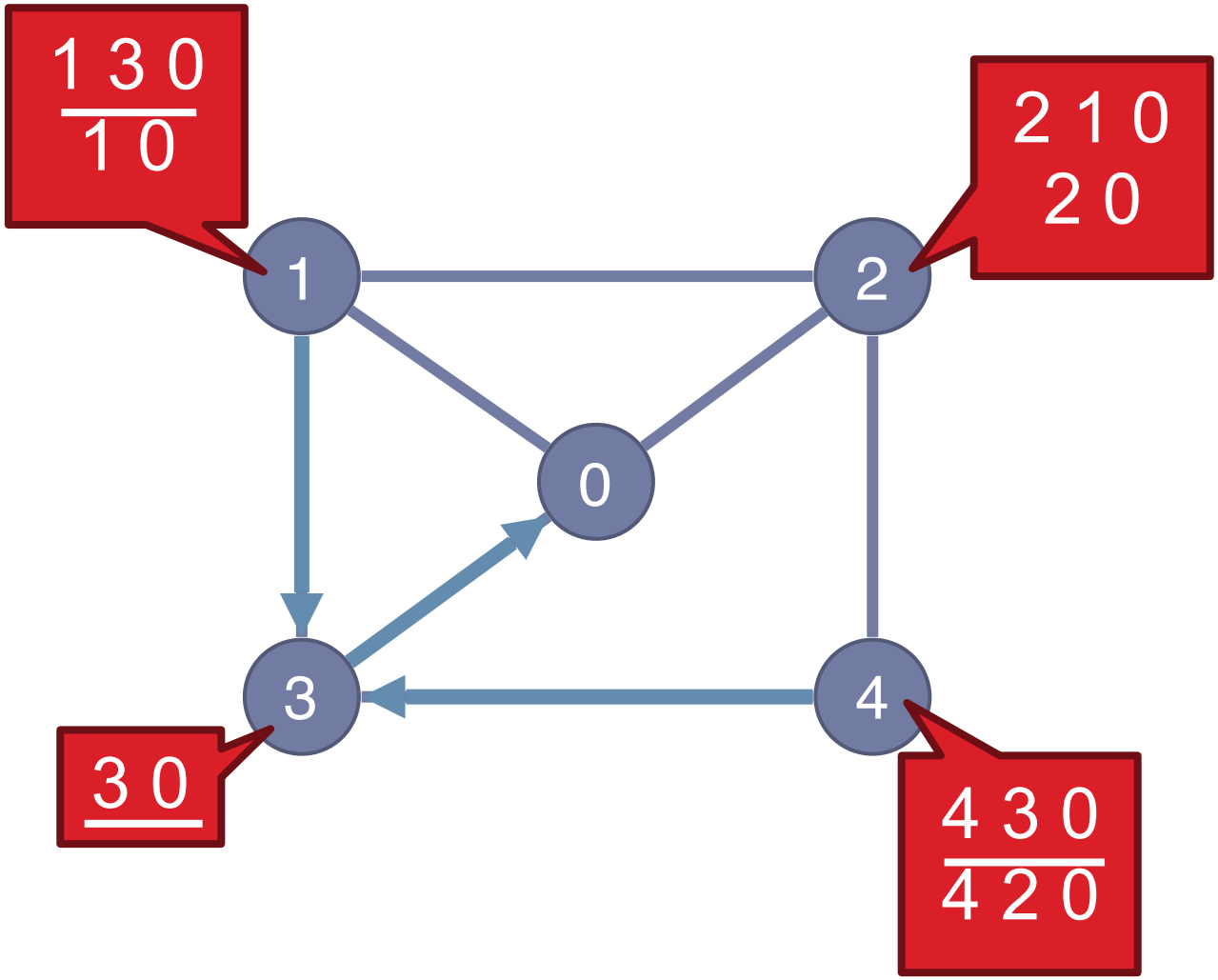
Good Gadget



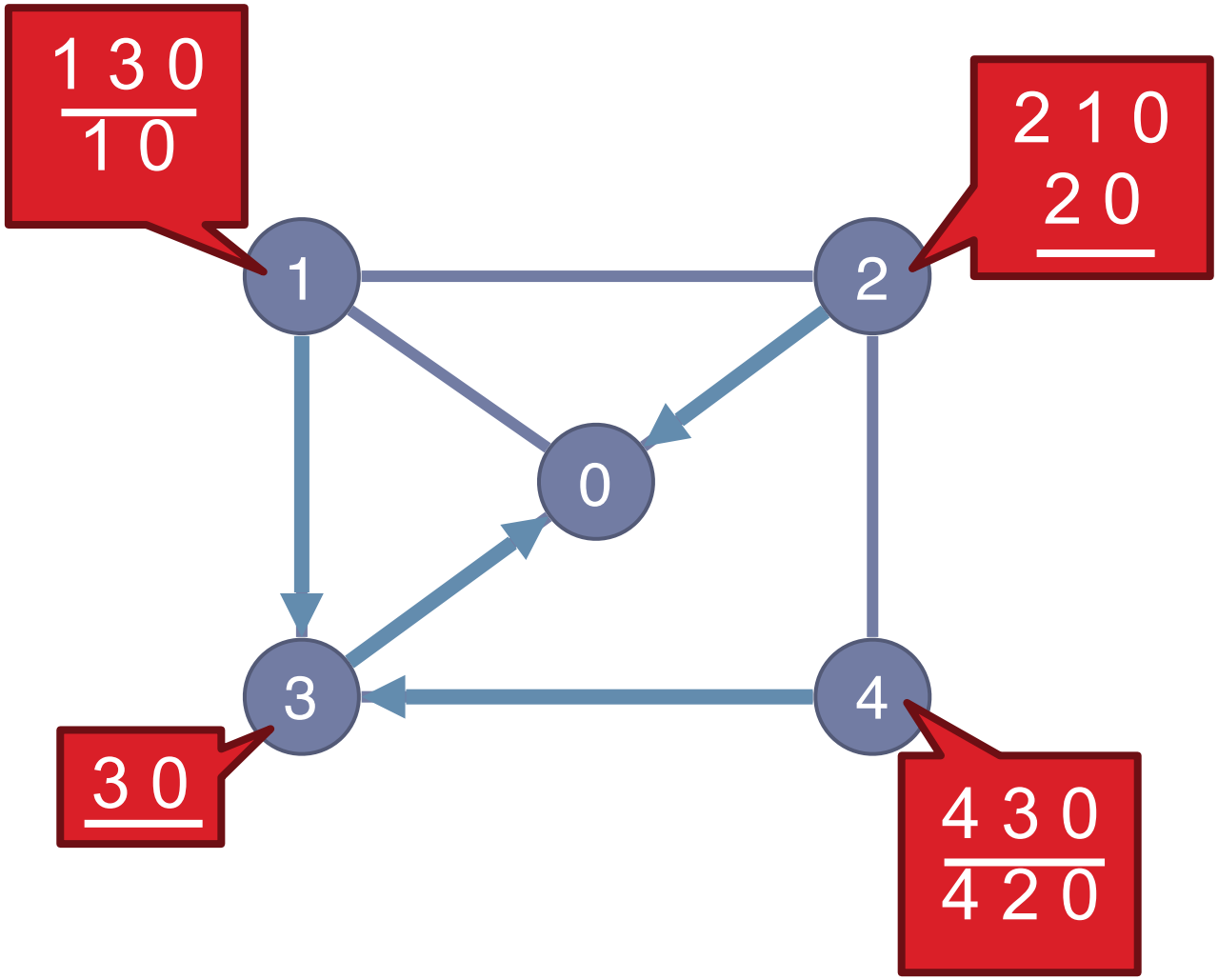
Good Gadget



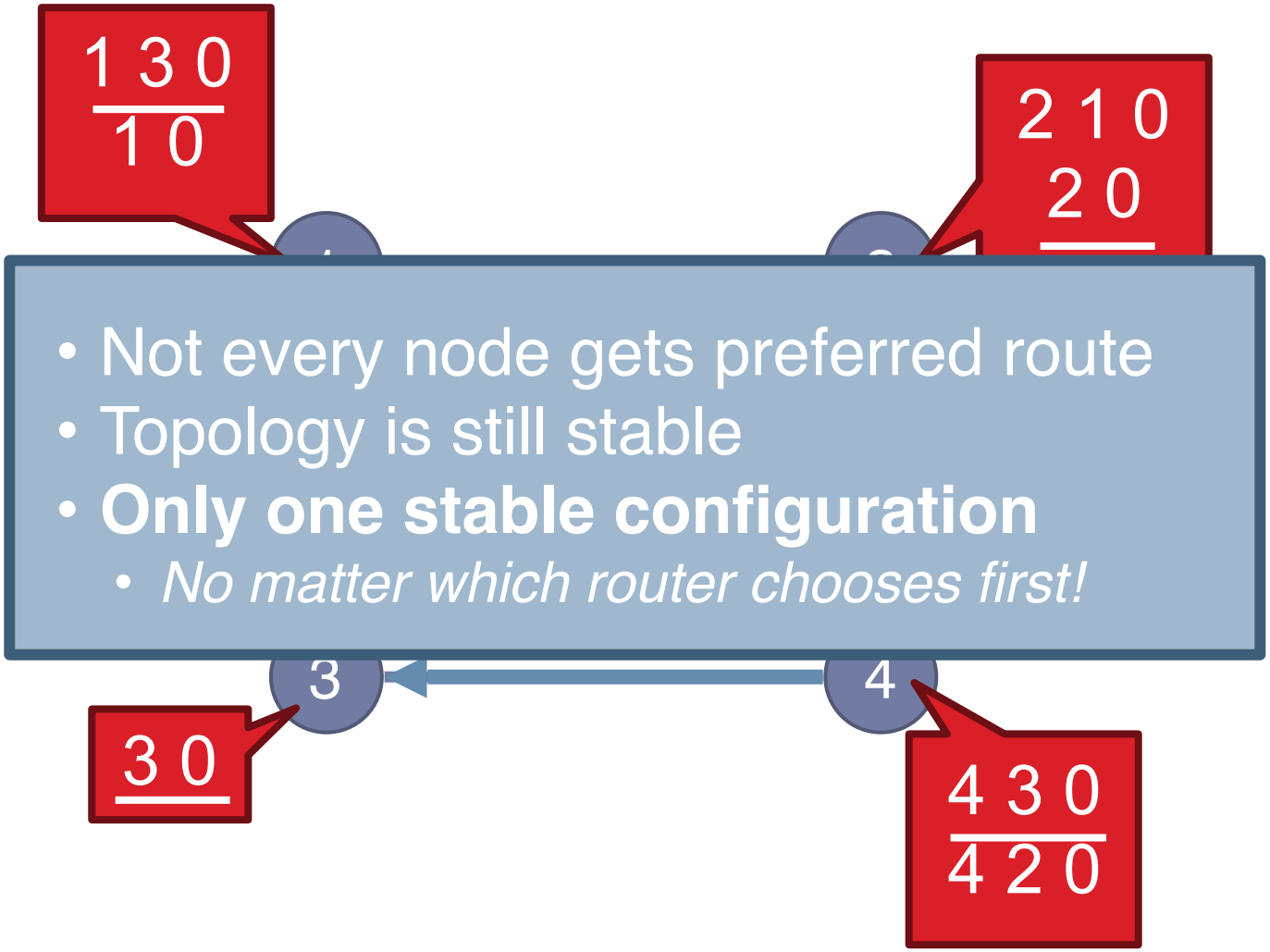
Good Gadget



Good Gadget

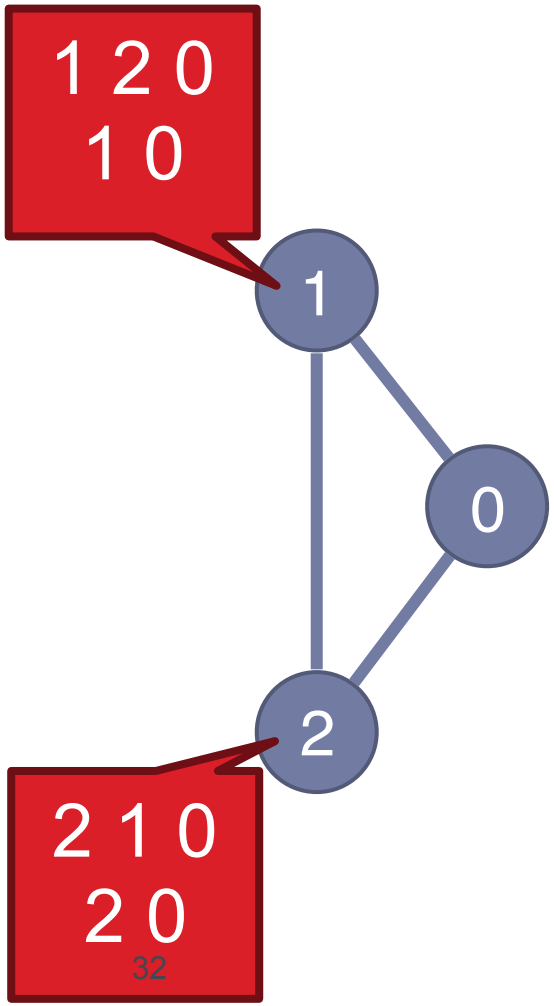


Good Gadget

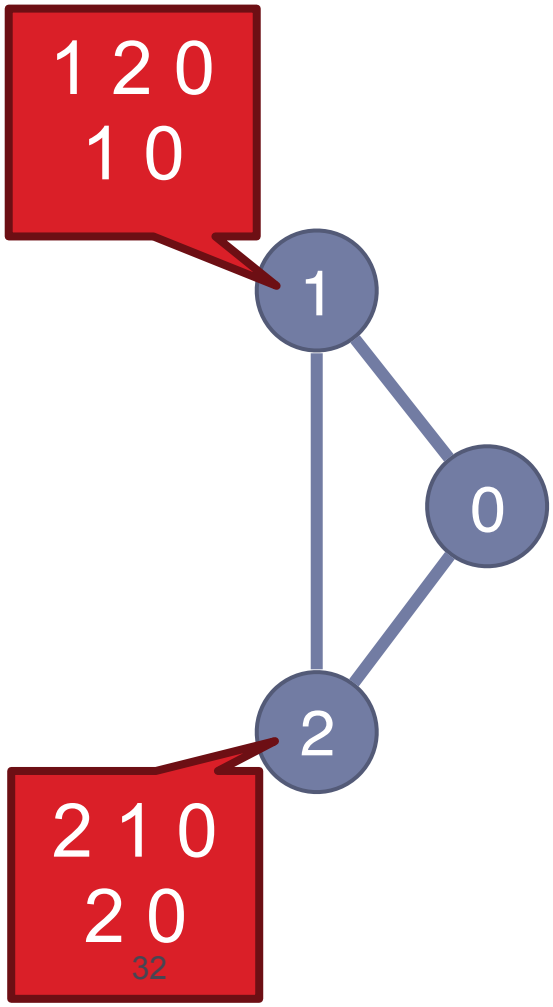


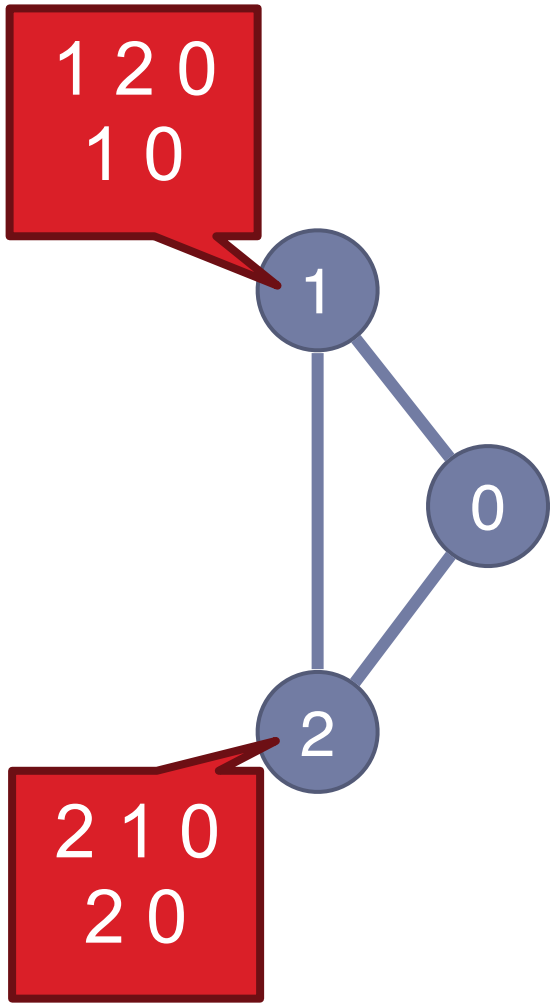
- Not every node gets preferred route
- Topology is still stable
- **Only one stable configuration**
  - *No matter which router chooses first!*

SPP May Have Multiple Solutions

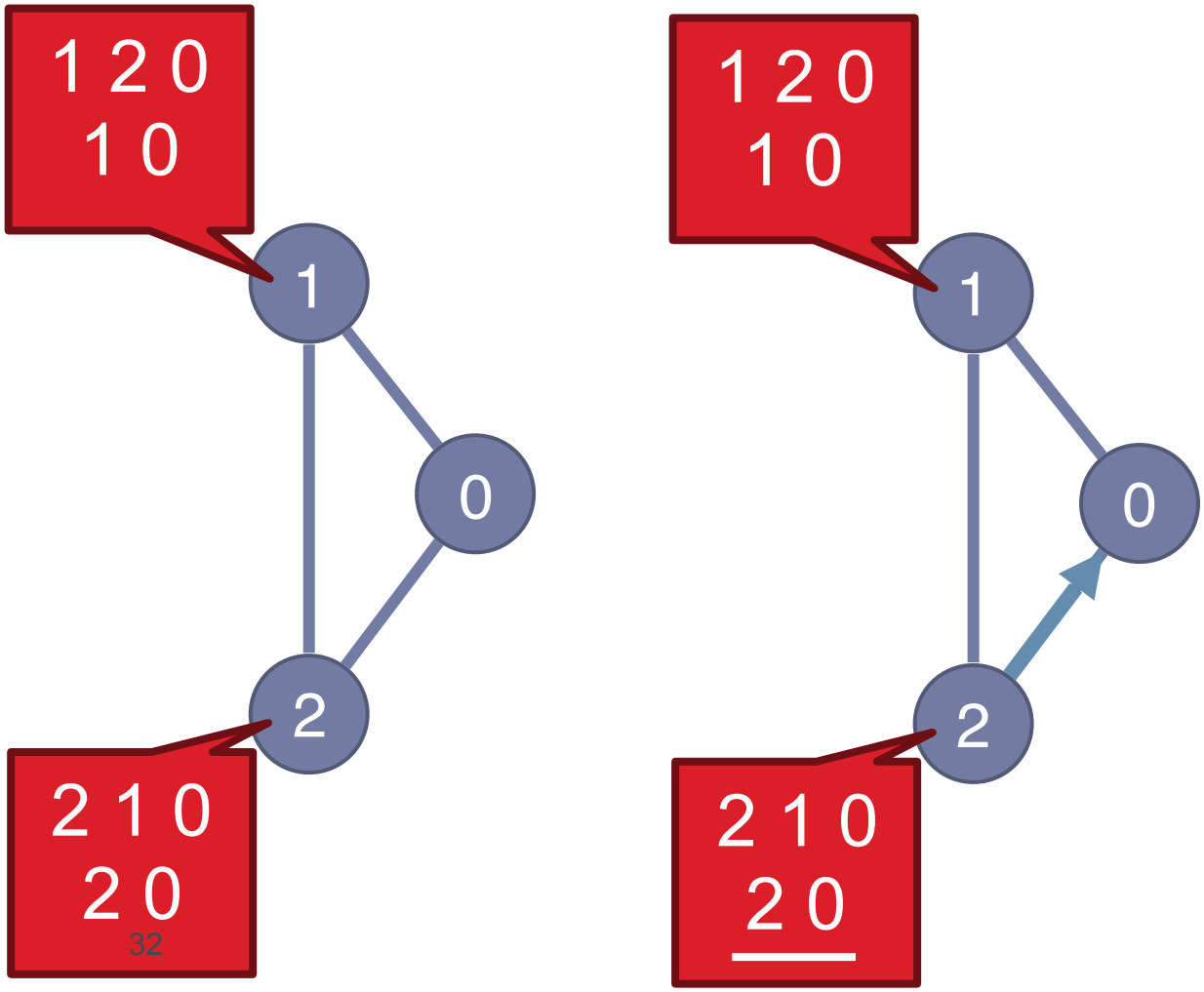


SPP May Have Multiple Solutions

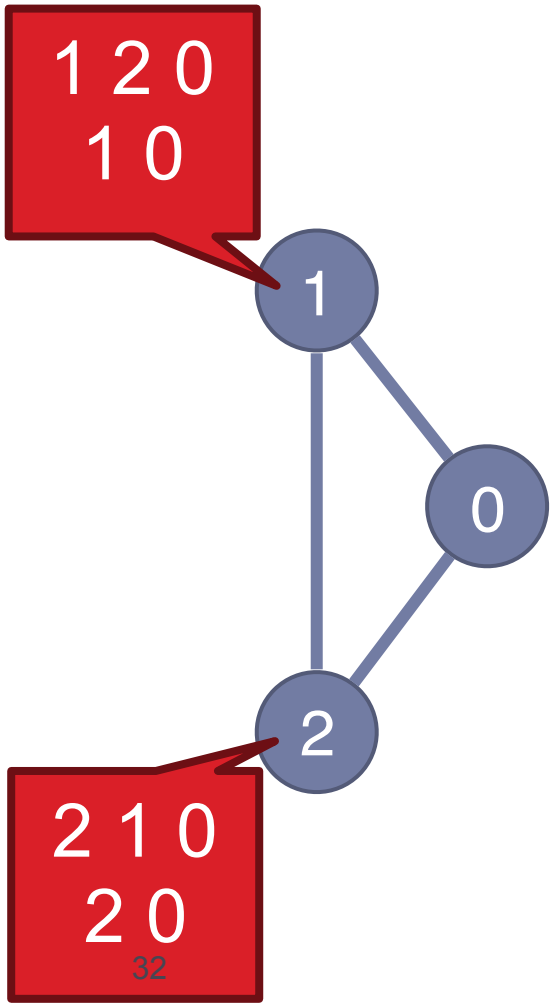


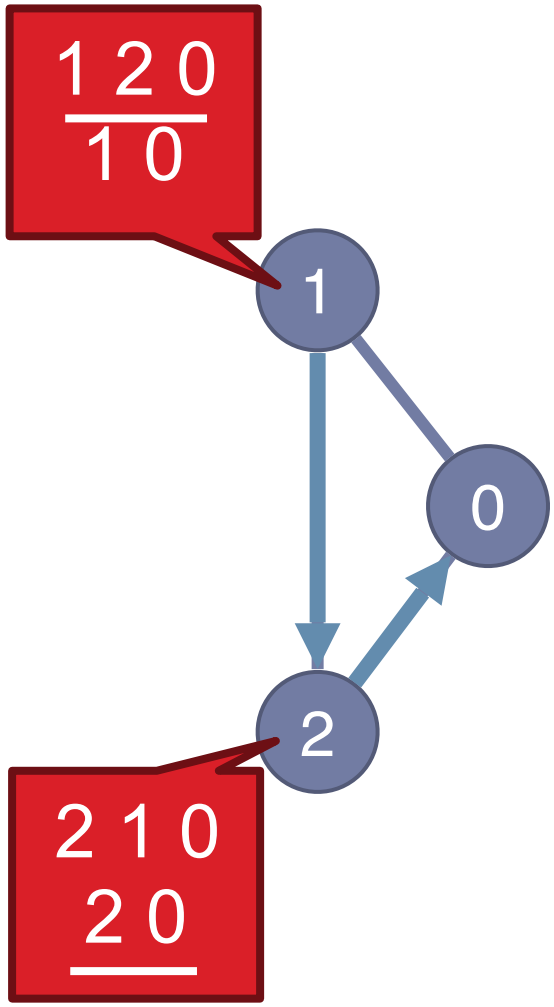


SPP May Have Multiple Solutions

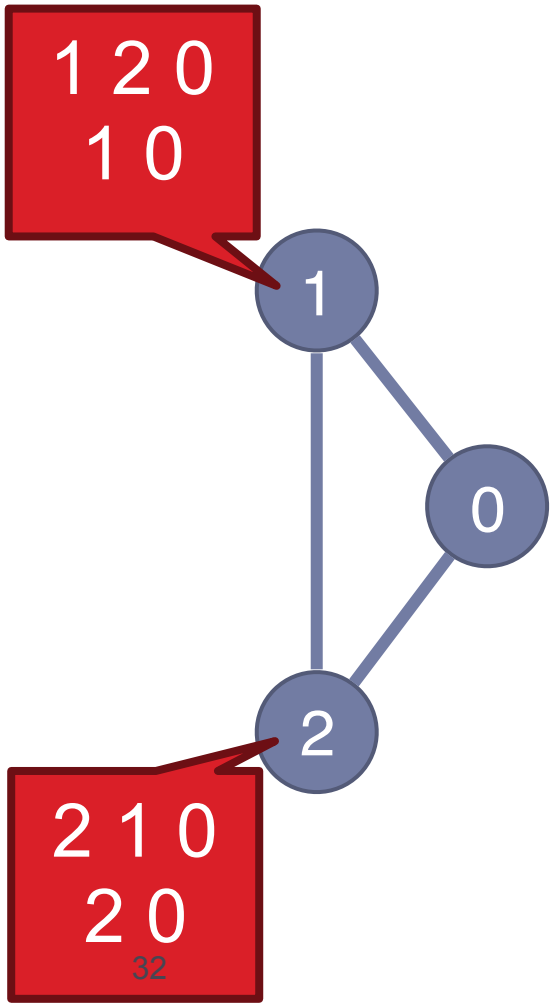


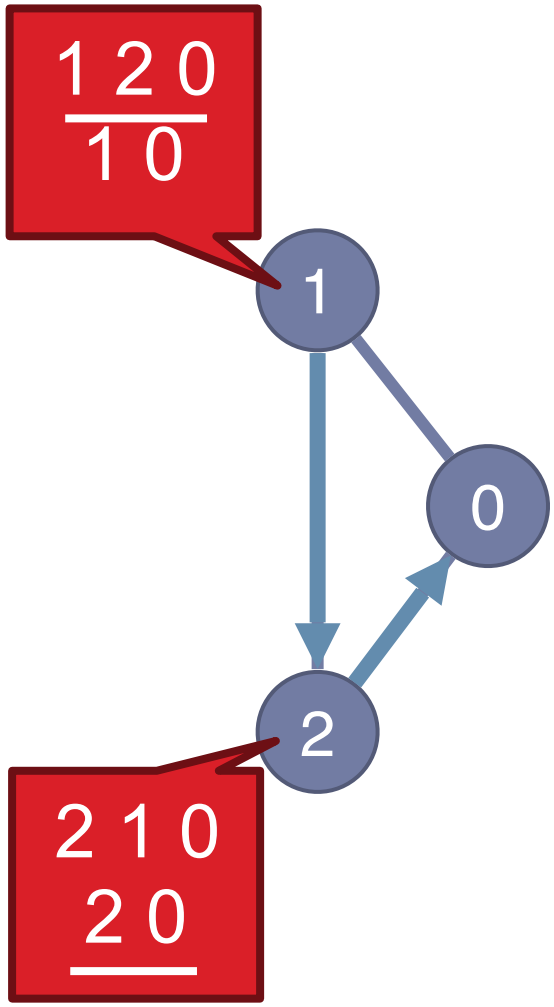
SPP May Have Multiple Solutions

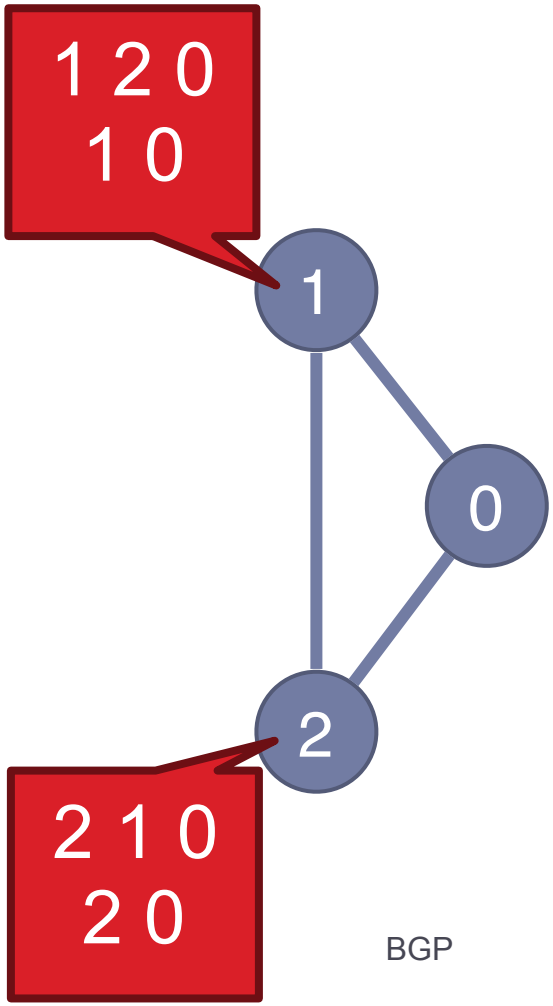




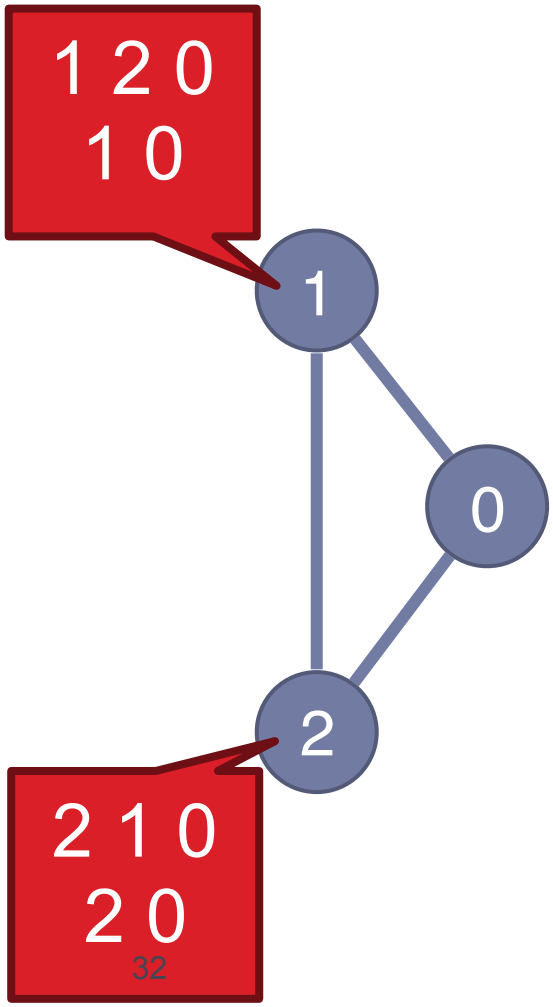
SPP May Have Multiple Solutions

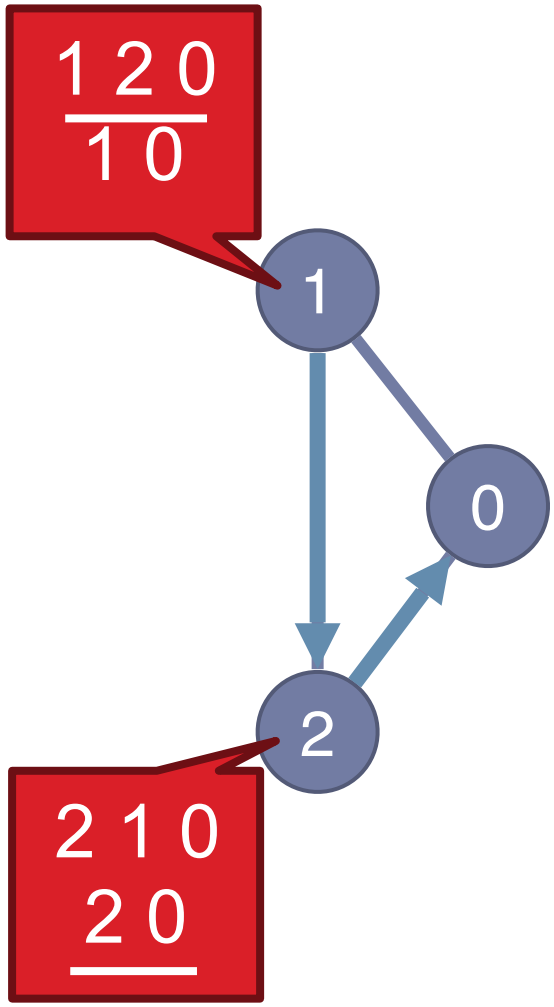


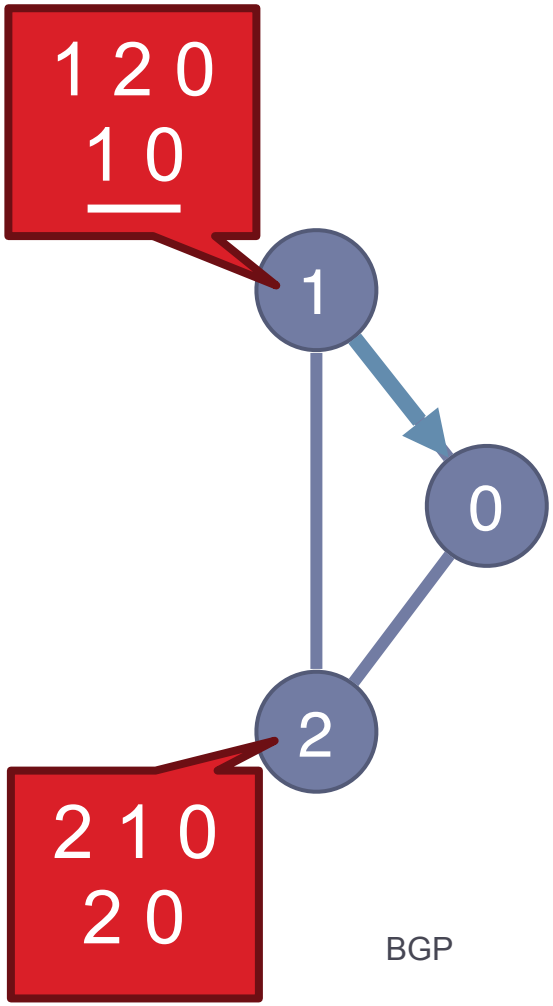




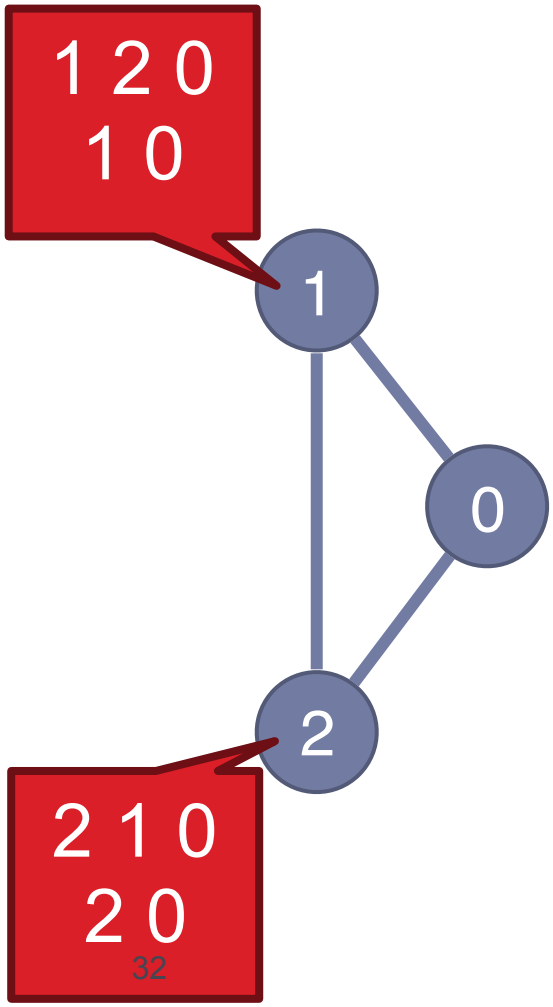
SPP May Have Multiple Solutions

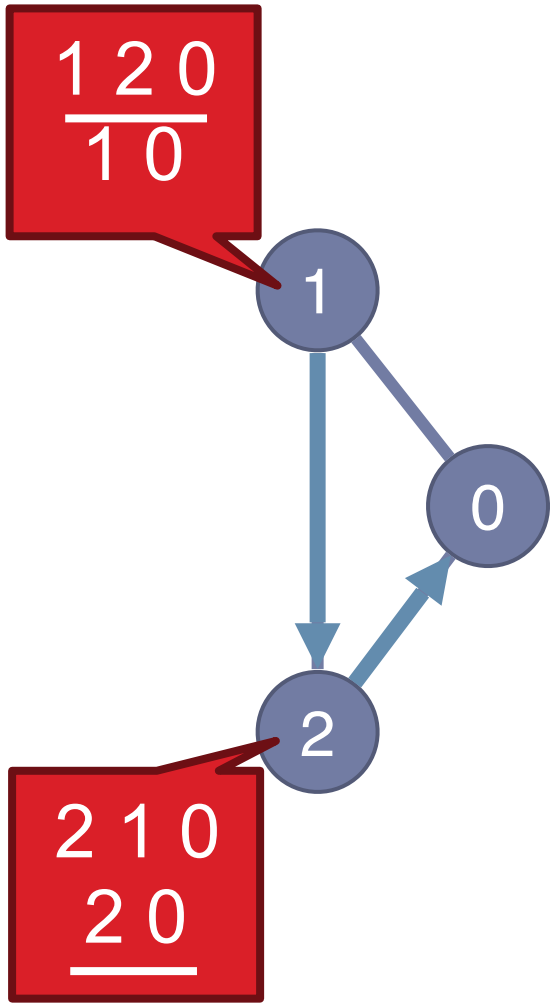


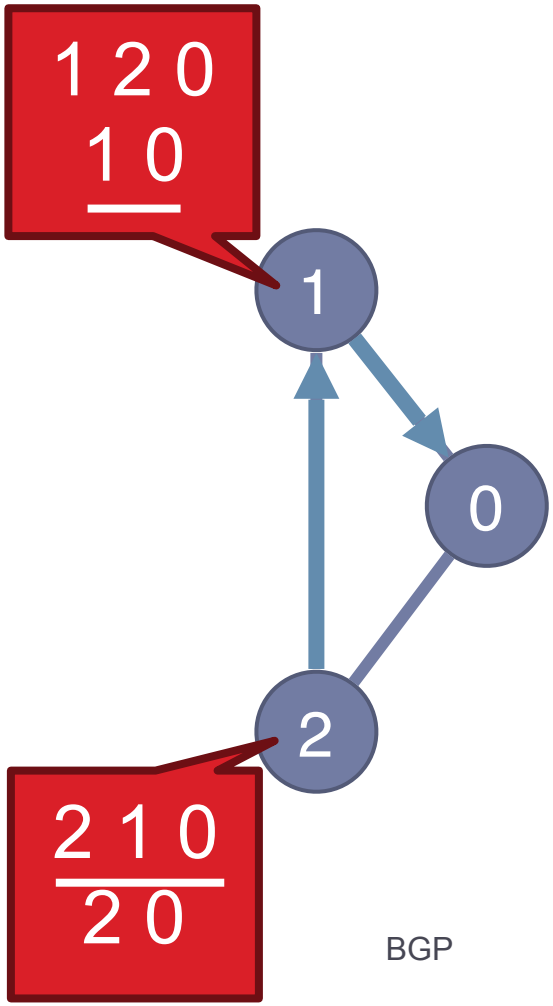




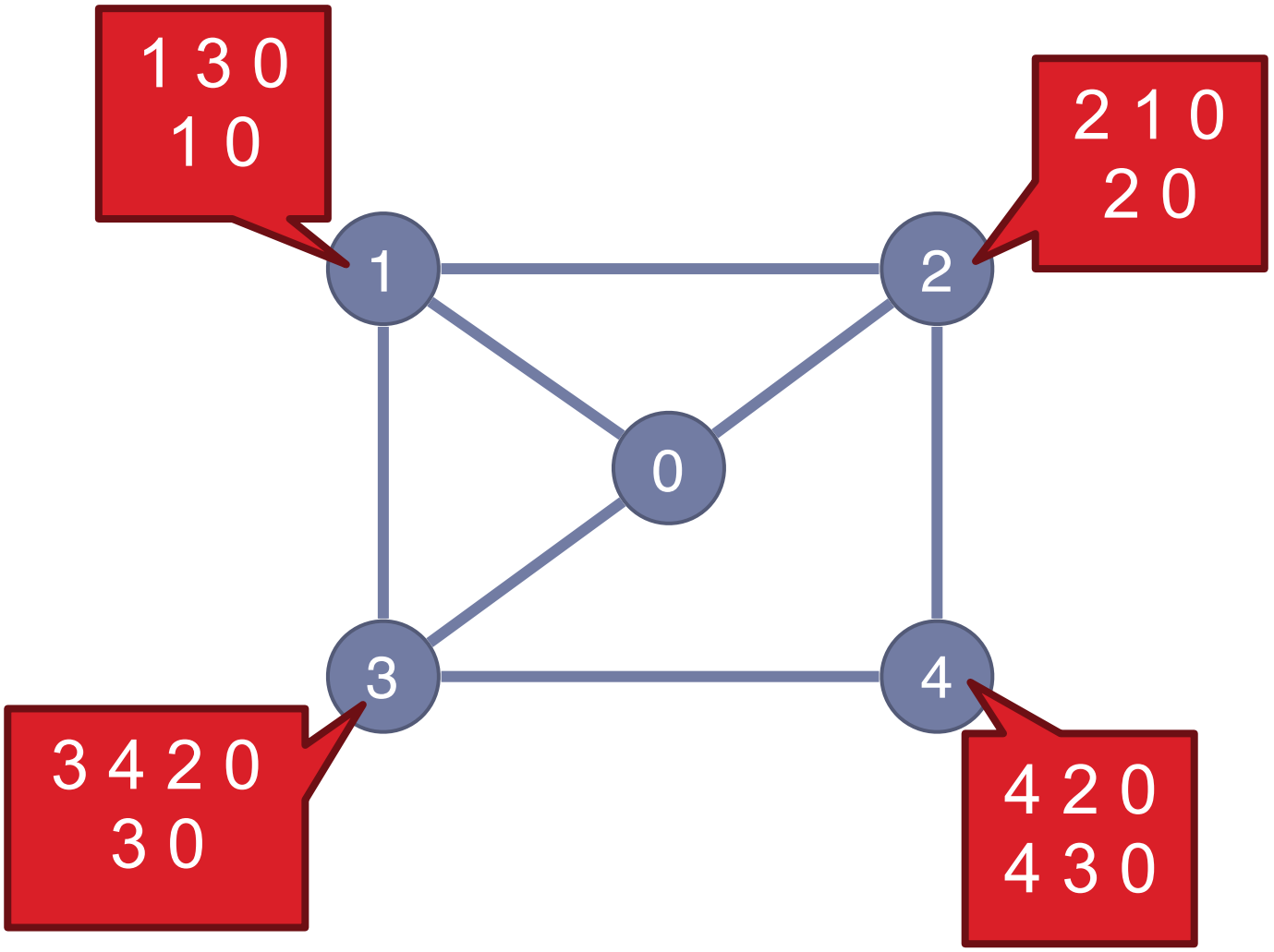
SPP May Have Multiple Solutions



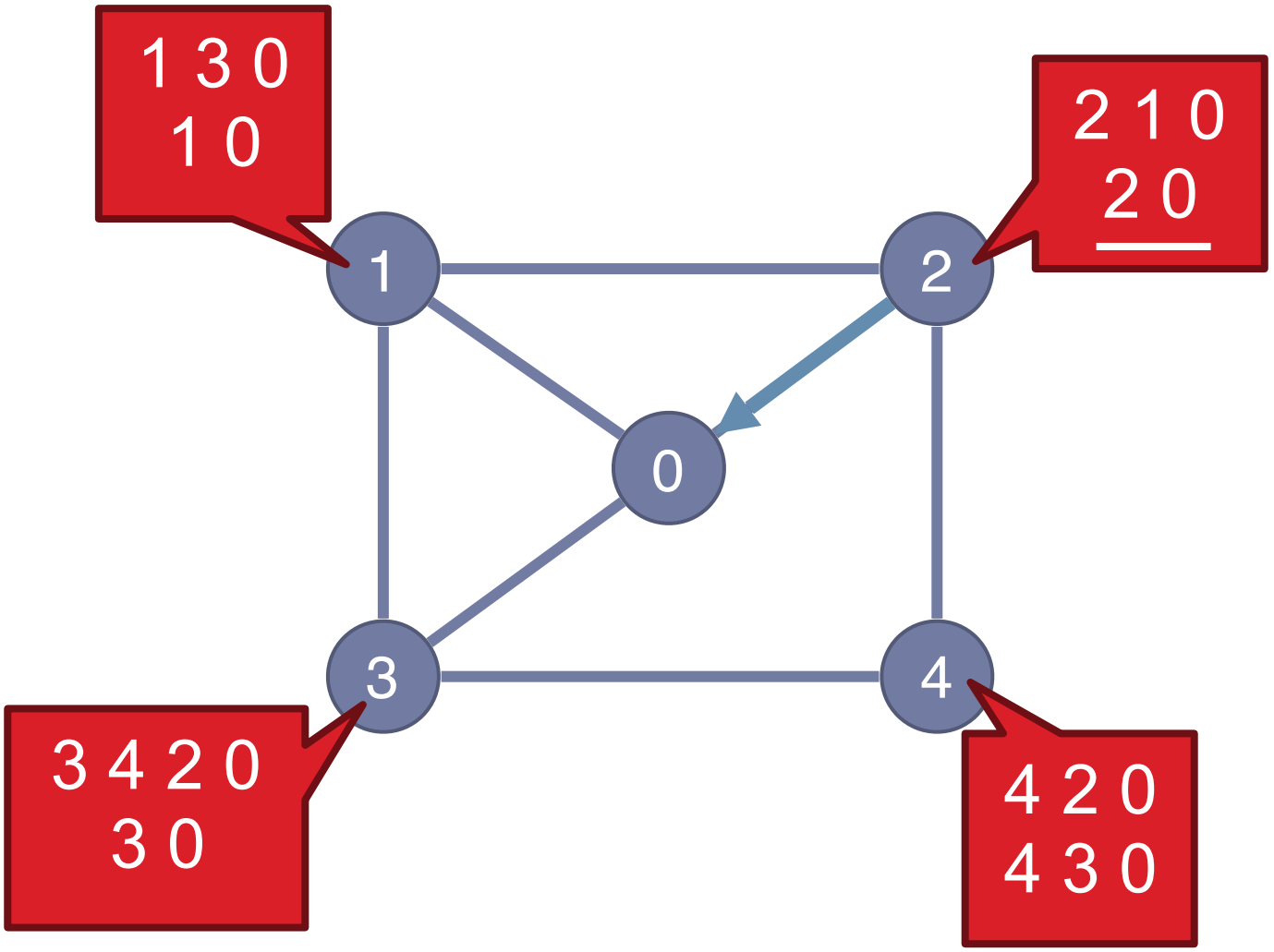




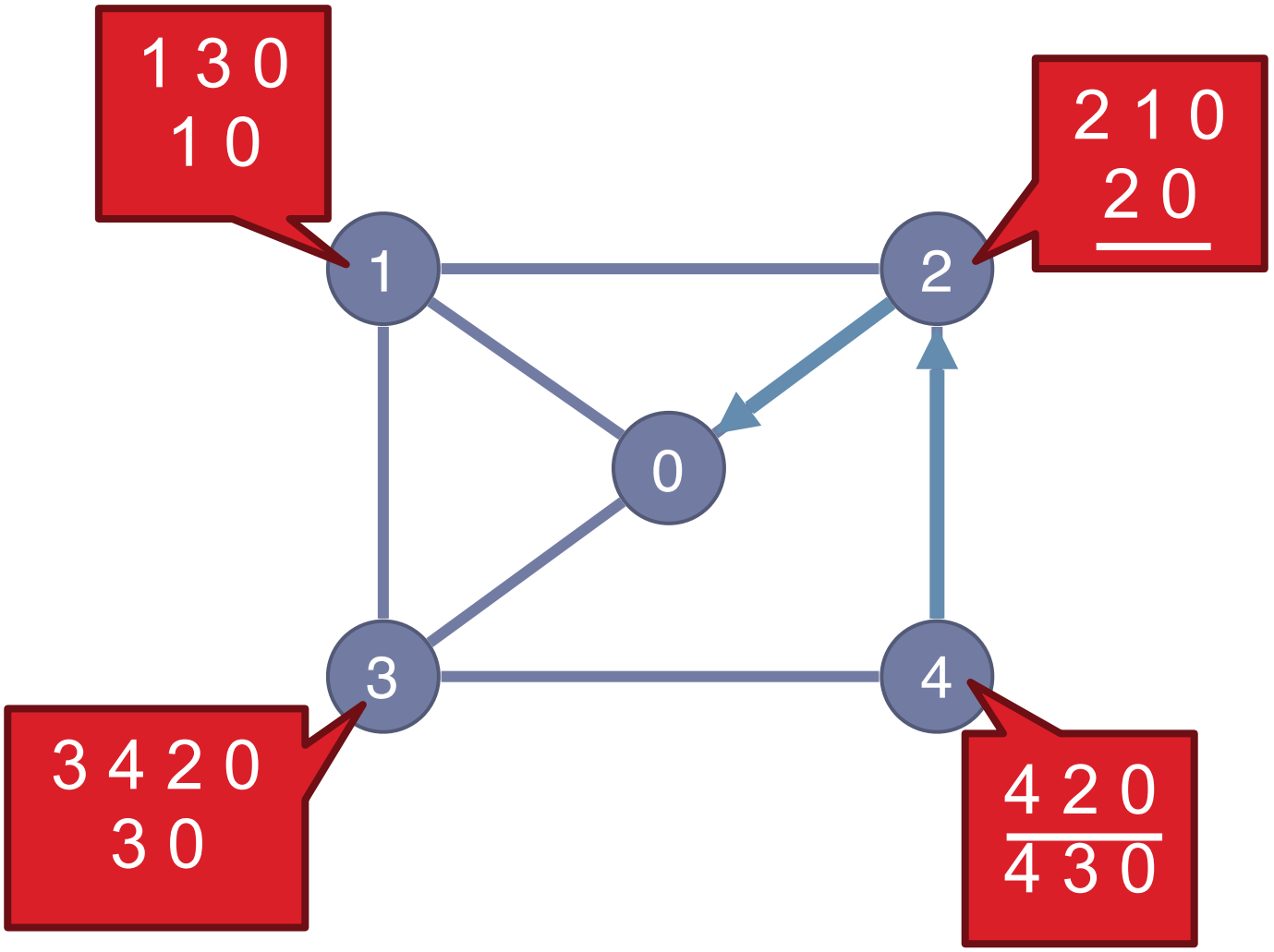
Bad Gadget



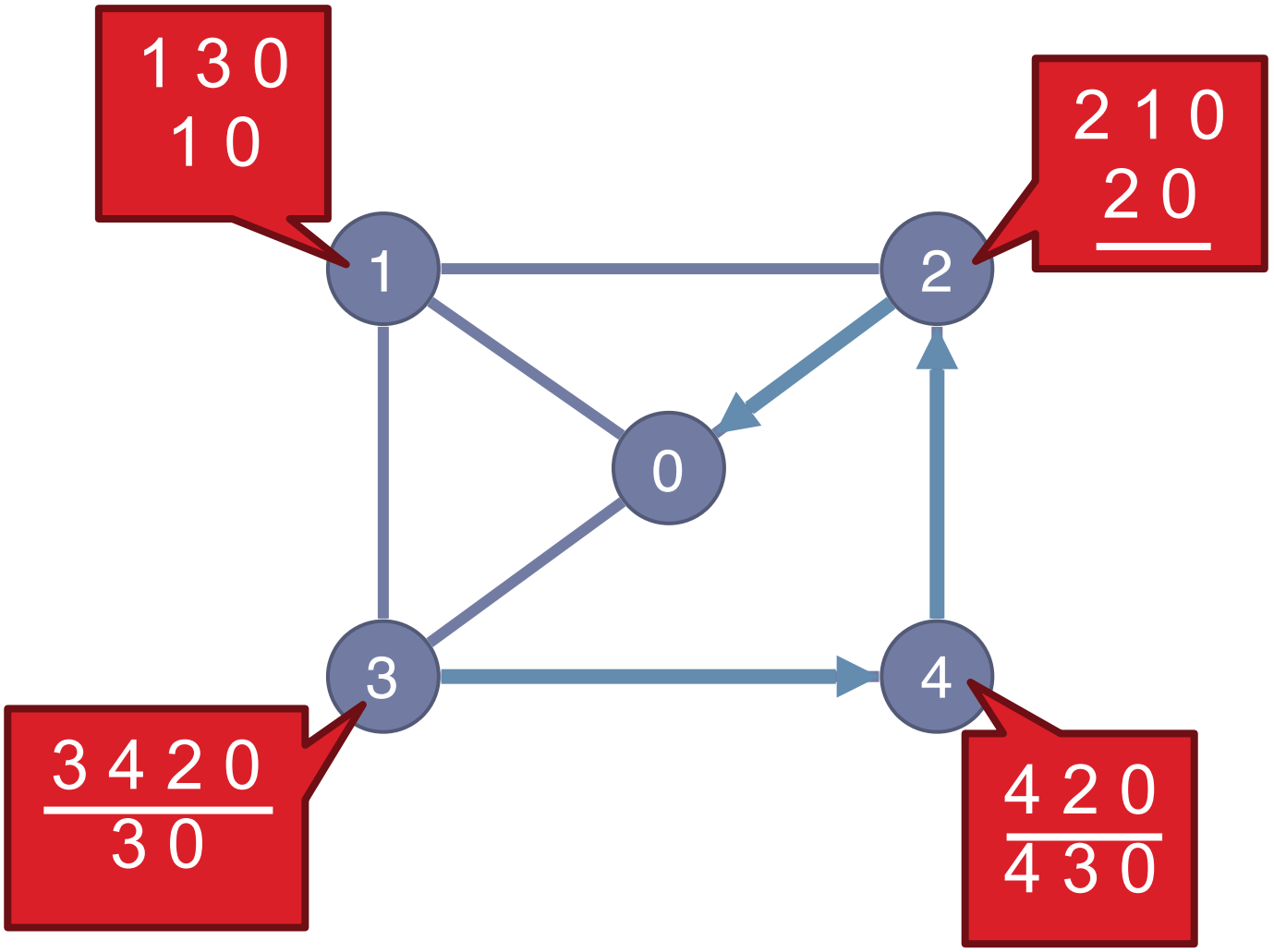
Bad Gadget



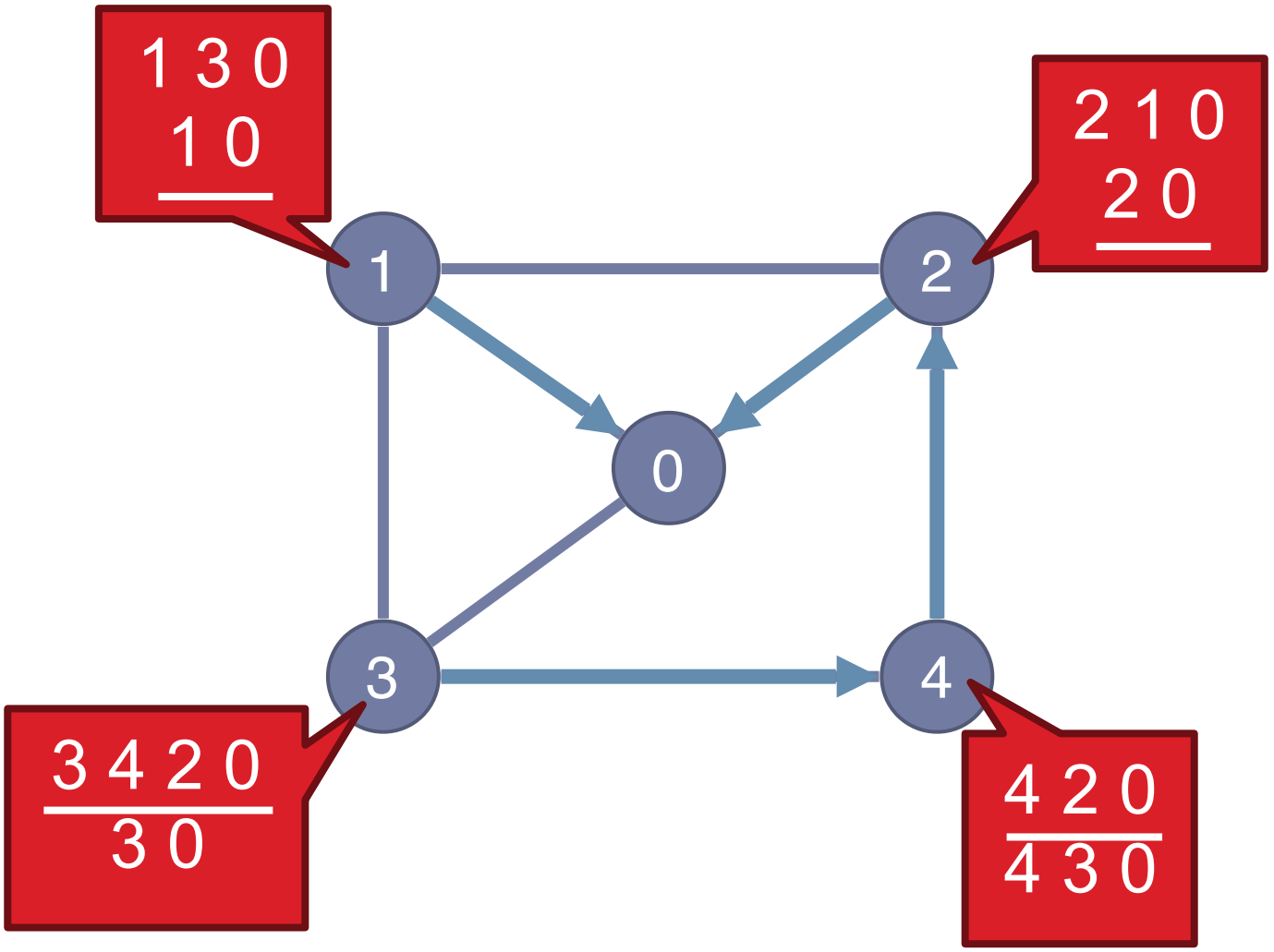
Bad Gadget



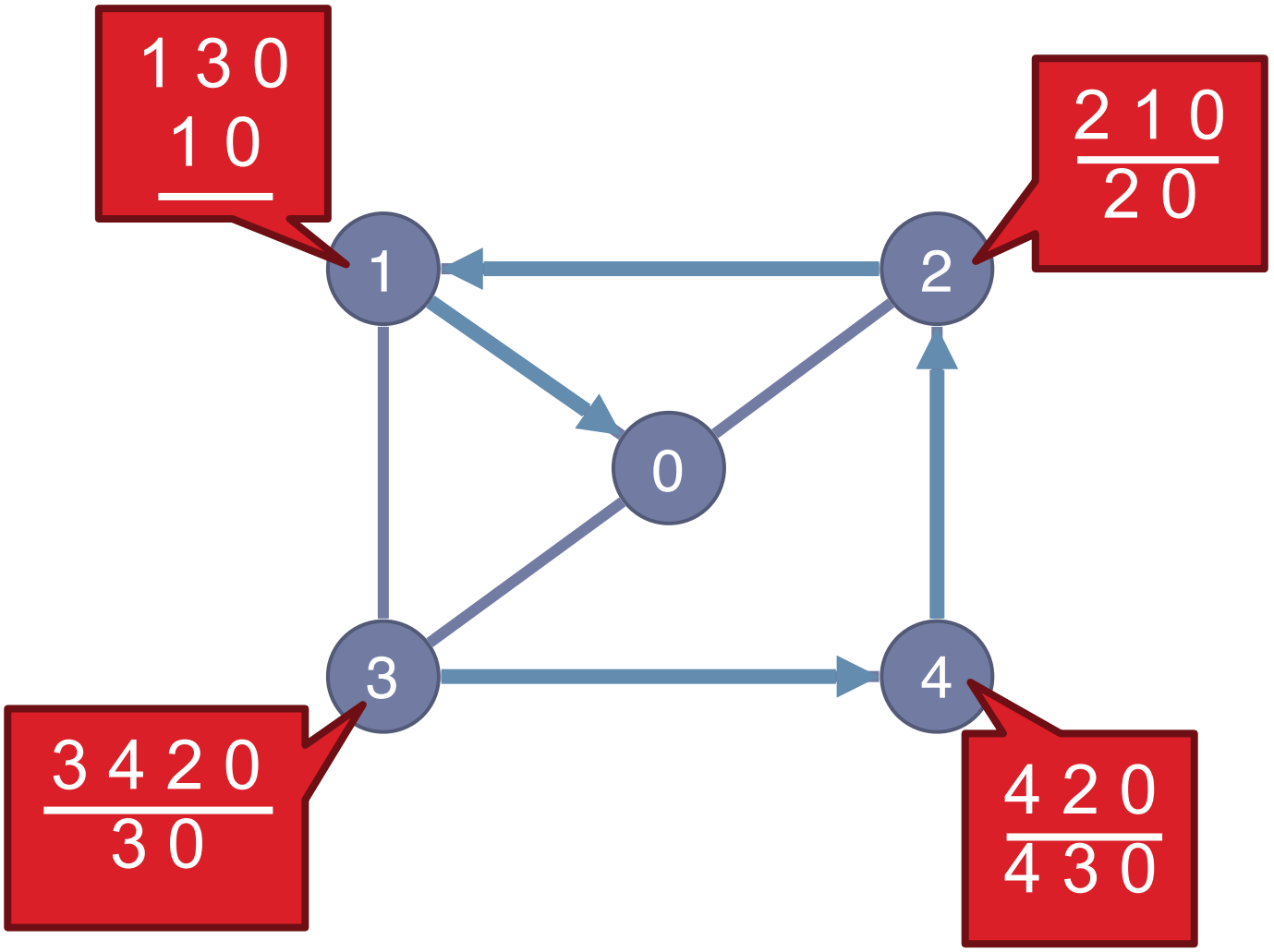
Bad Gadget



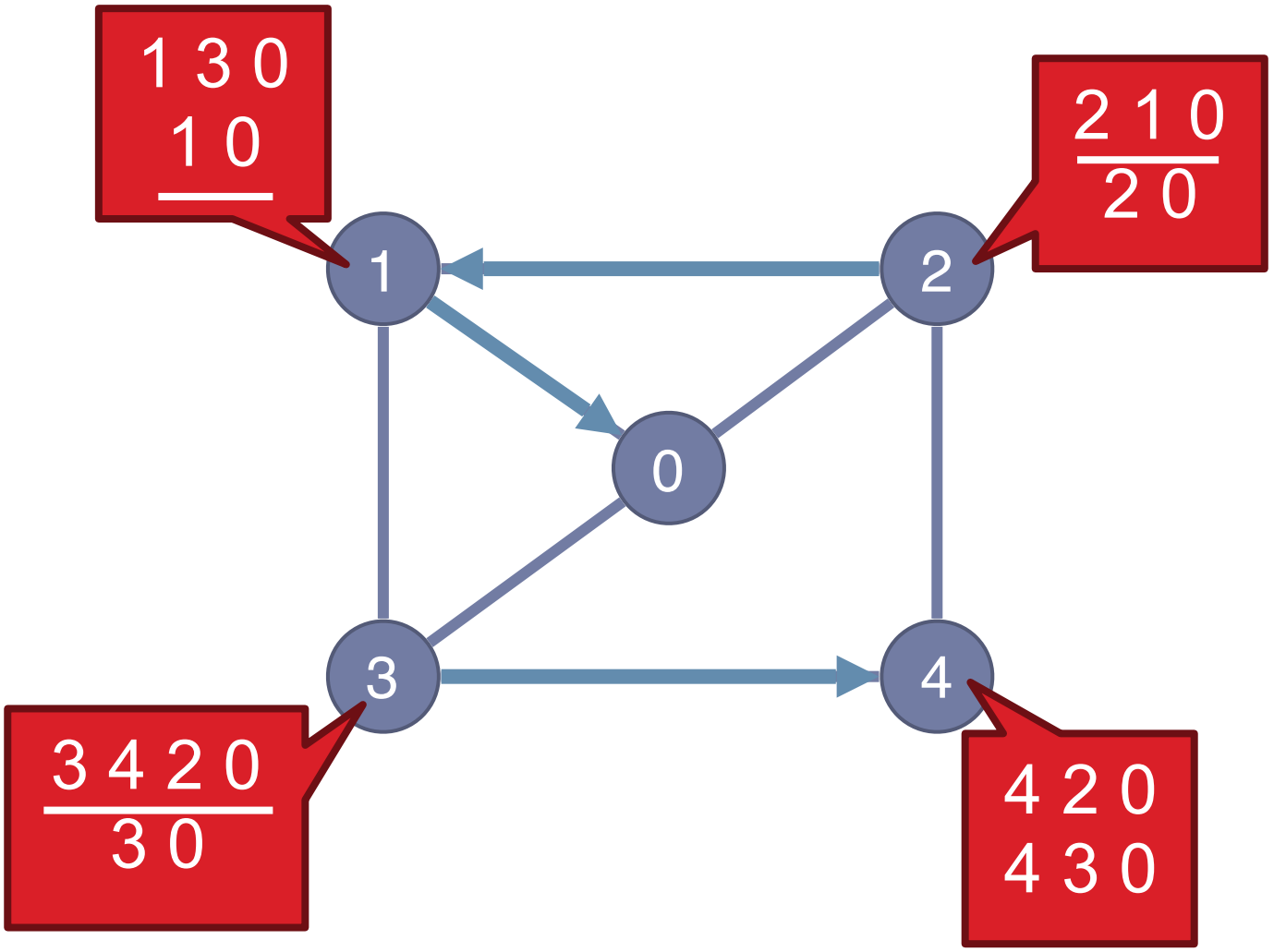
Bad Gadget



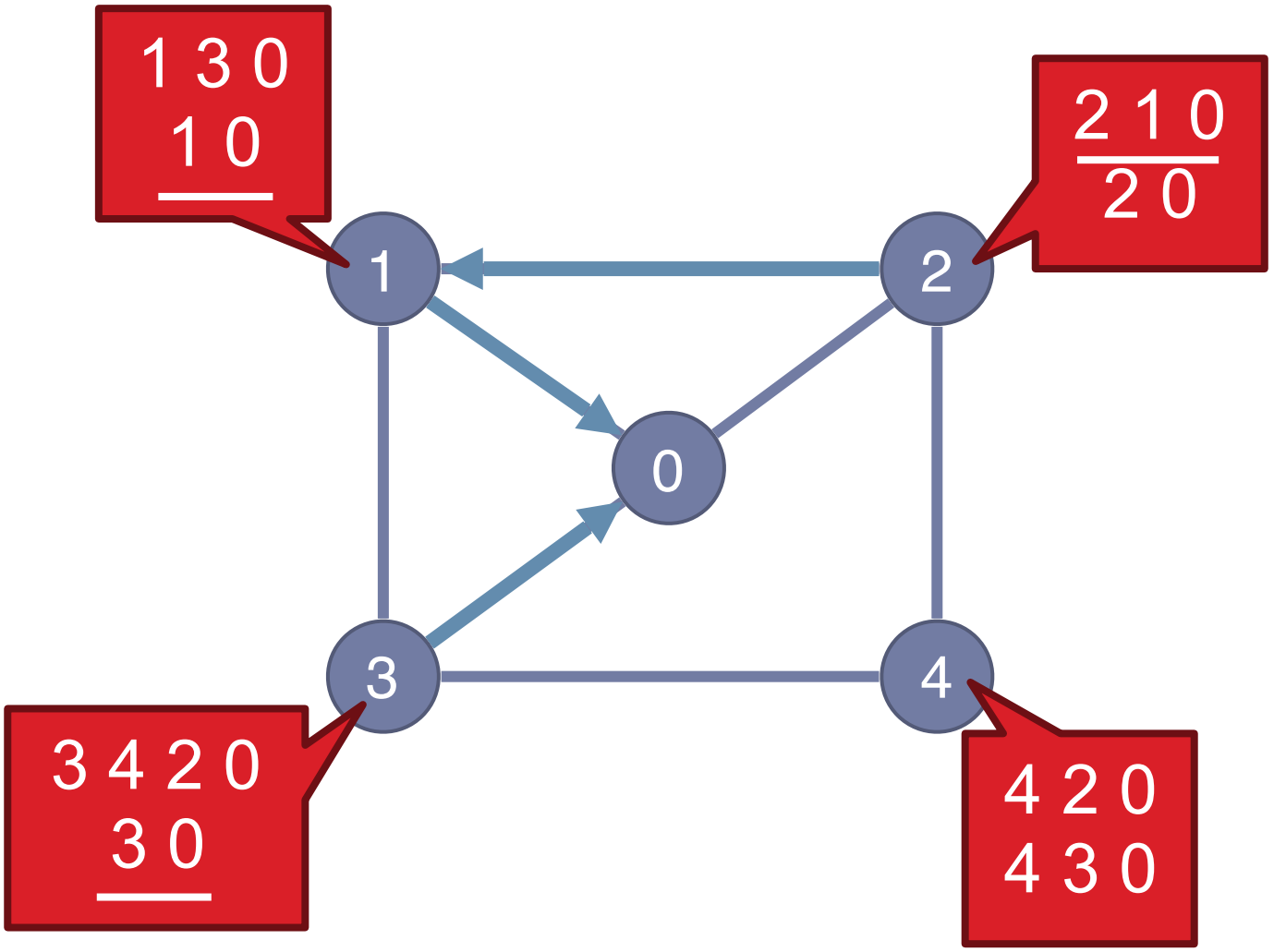
Bad Gadget



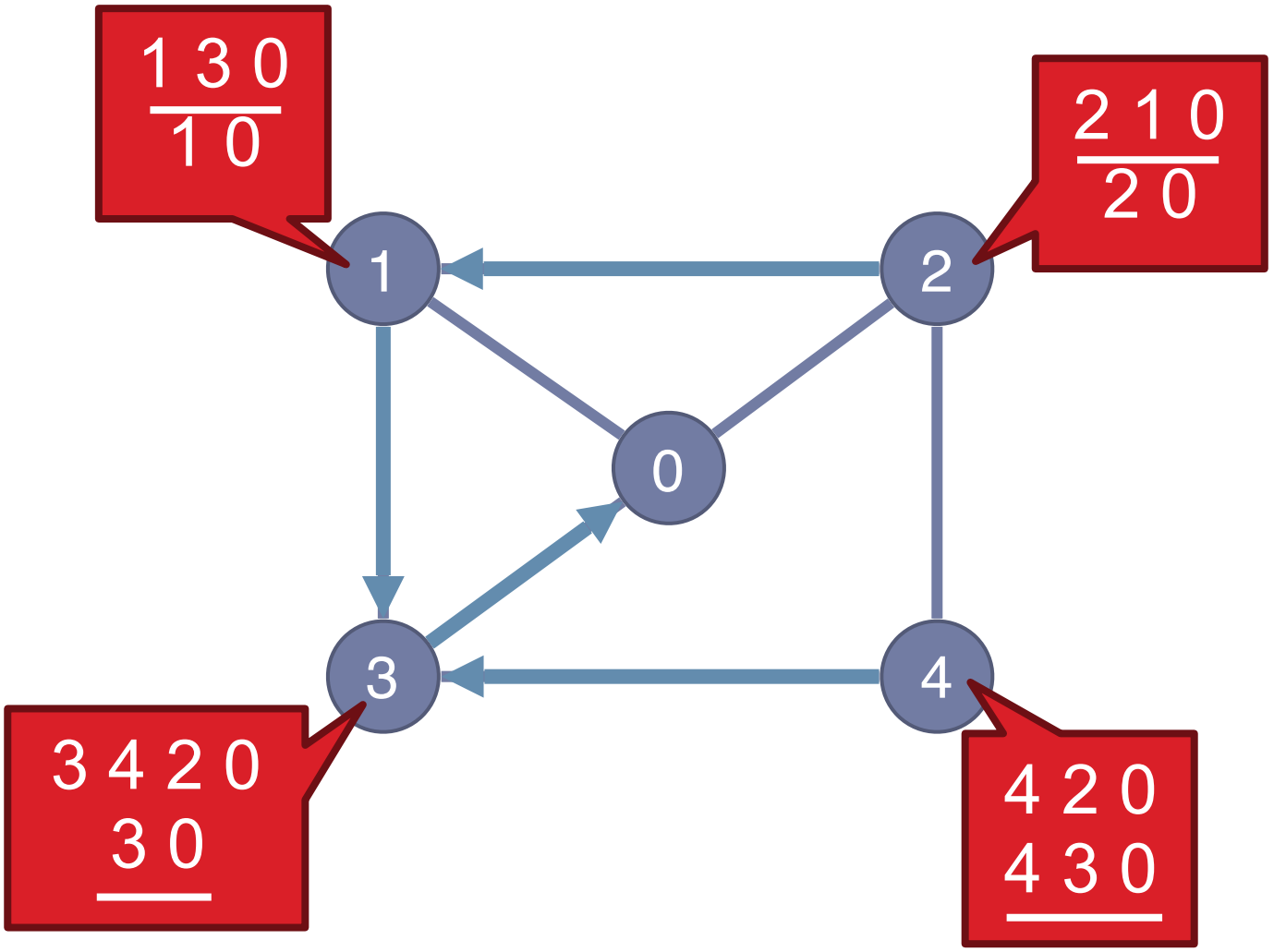
Bad Gadget



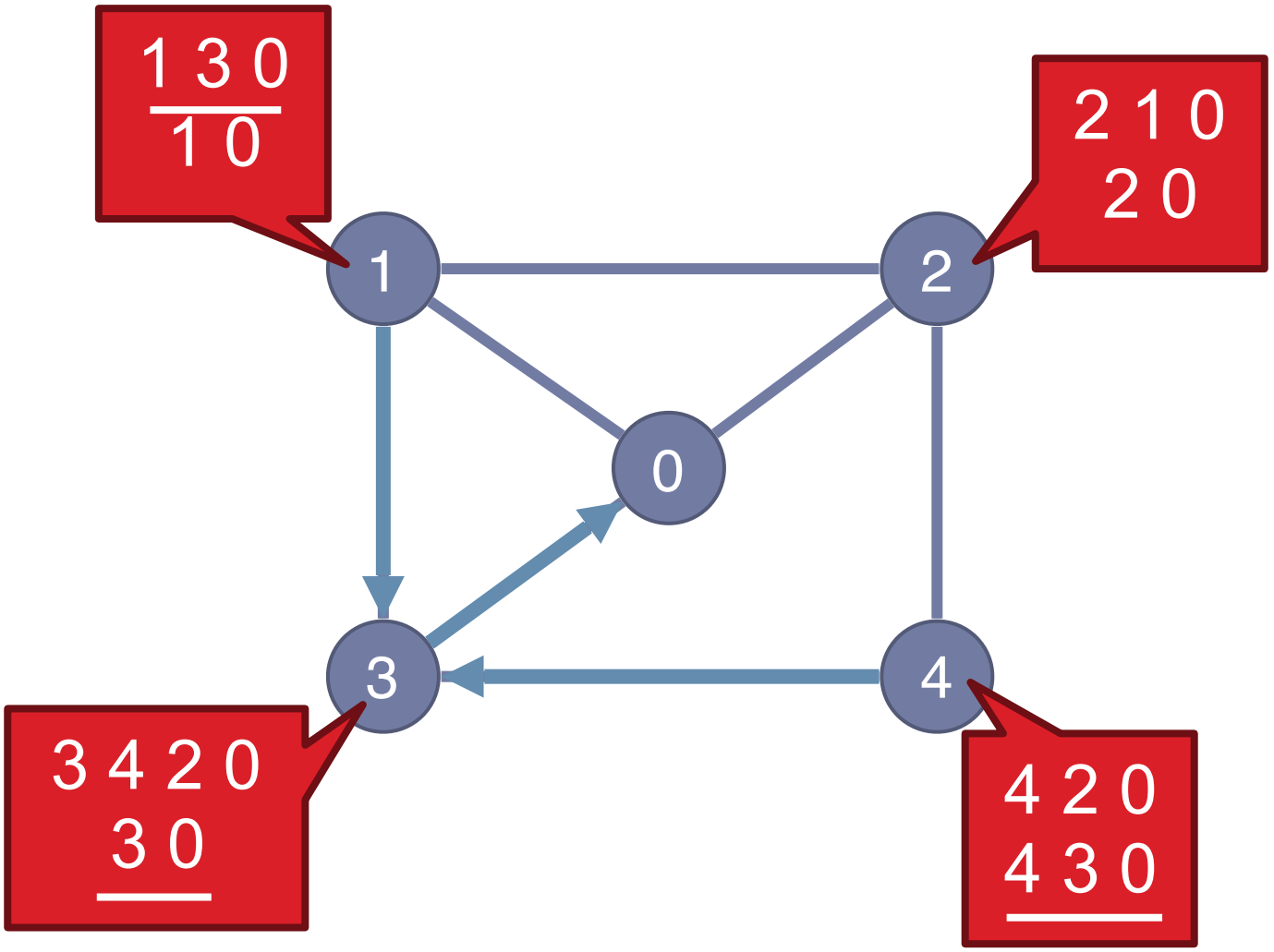
Bad Gadget



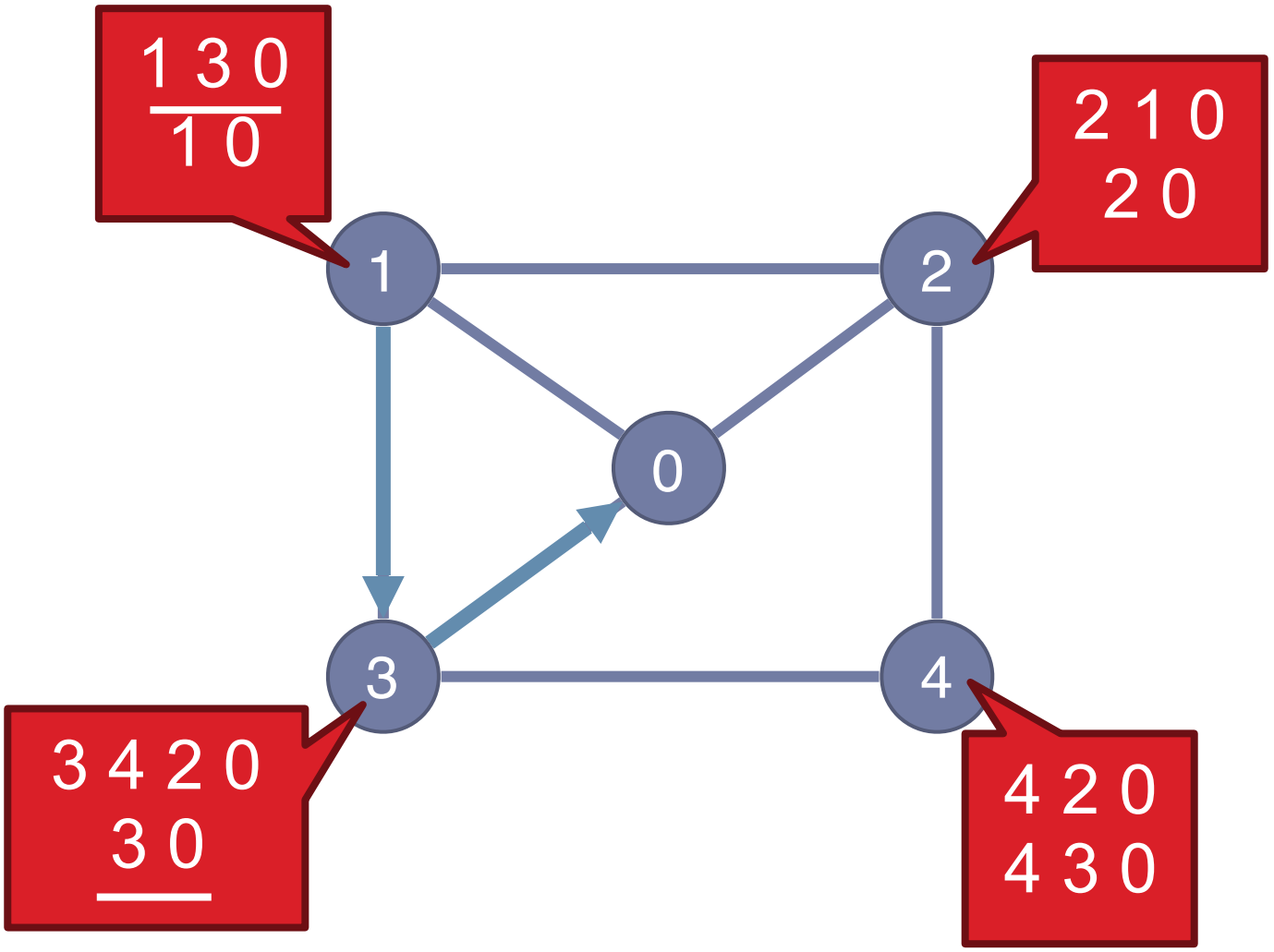
Bad Gadget



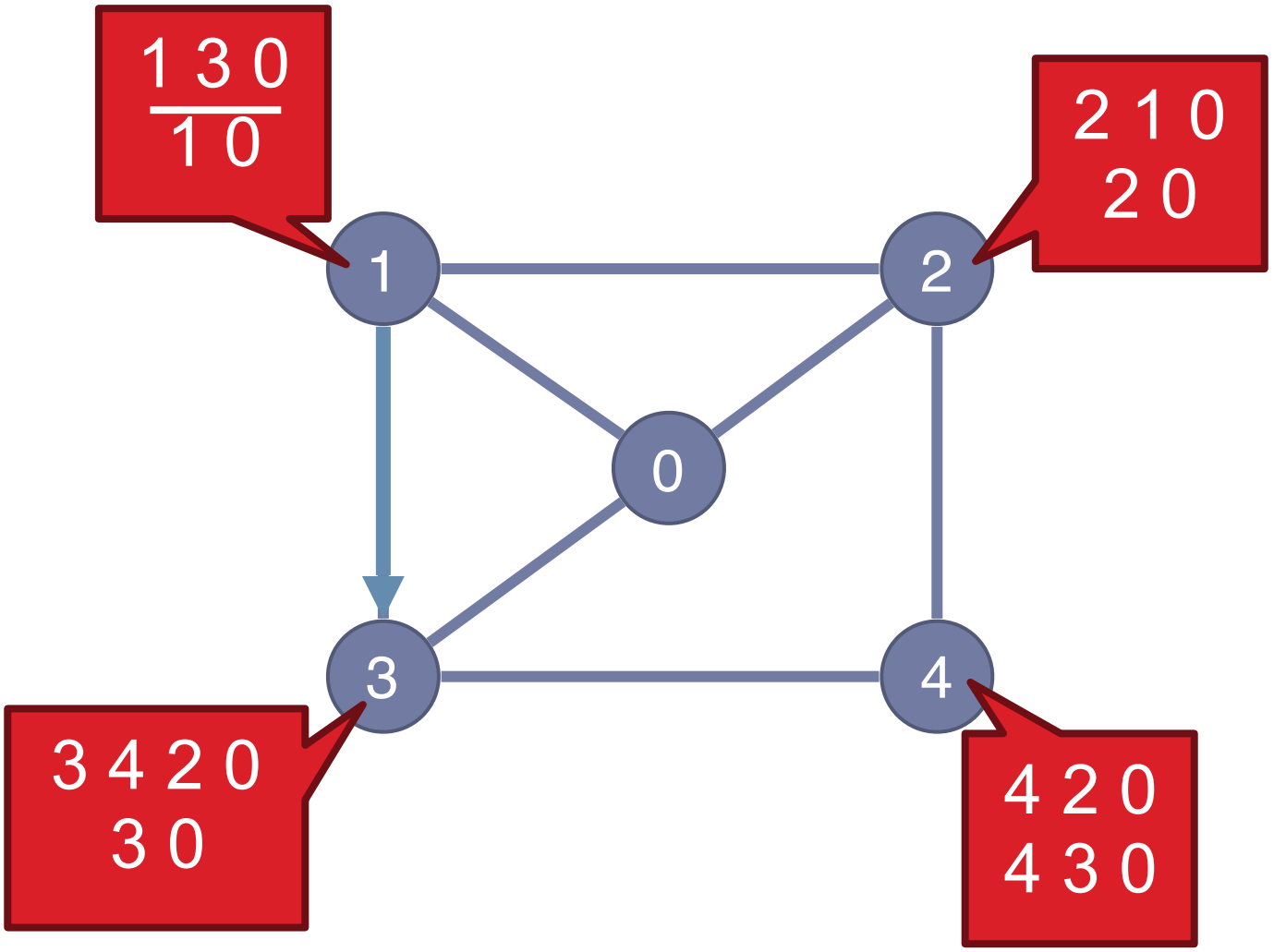
Bad Gadget



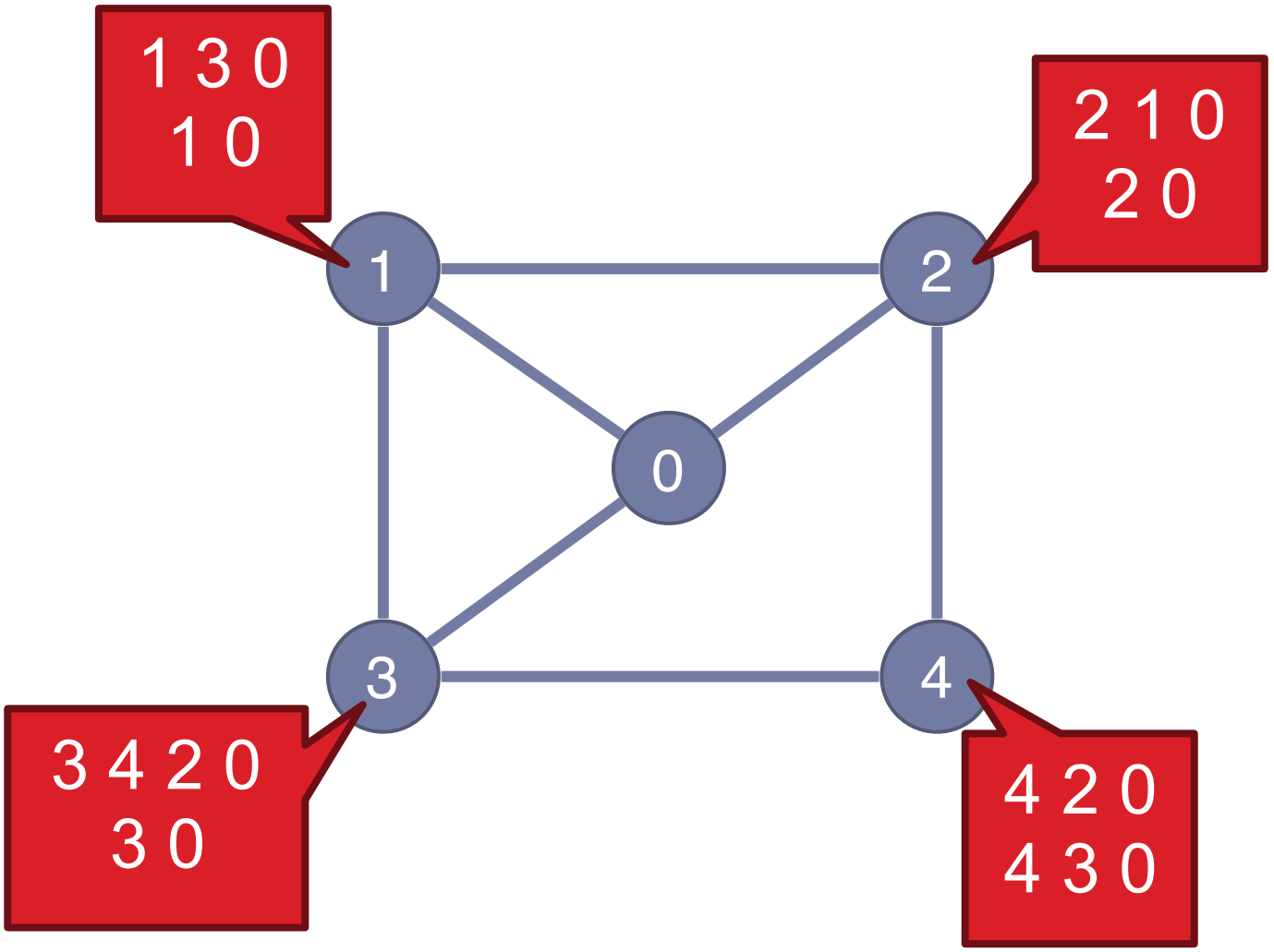
Bad Gadget



Bad Gadget



Bad Gadget

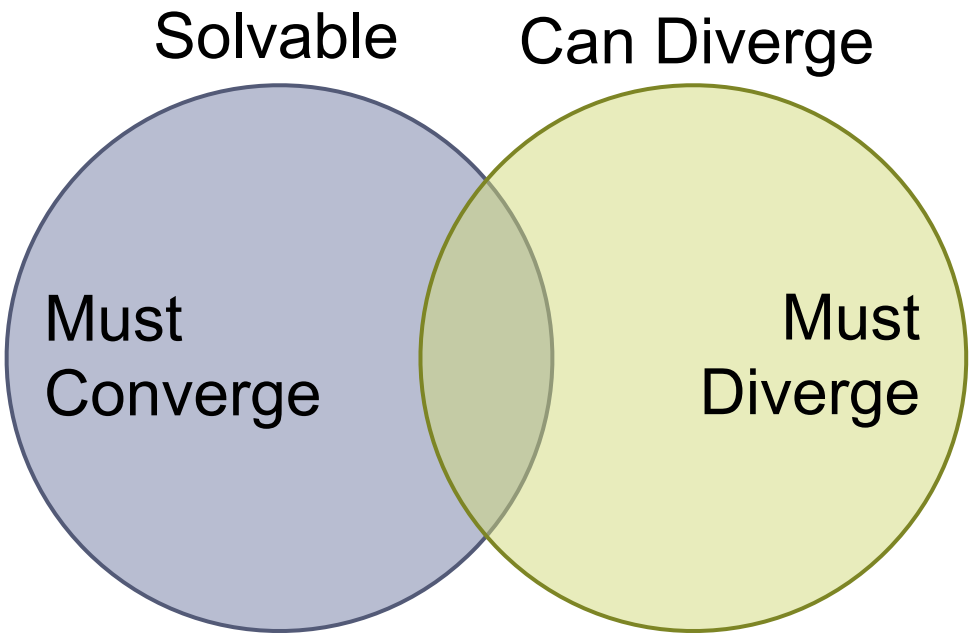


Bad Gadget

- That was only one round of oscillation!
- This keeps going, infinitely
- Problem stems from:
  - Local (not global) decisions
  - Ability of one node to improve its path selection

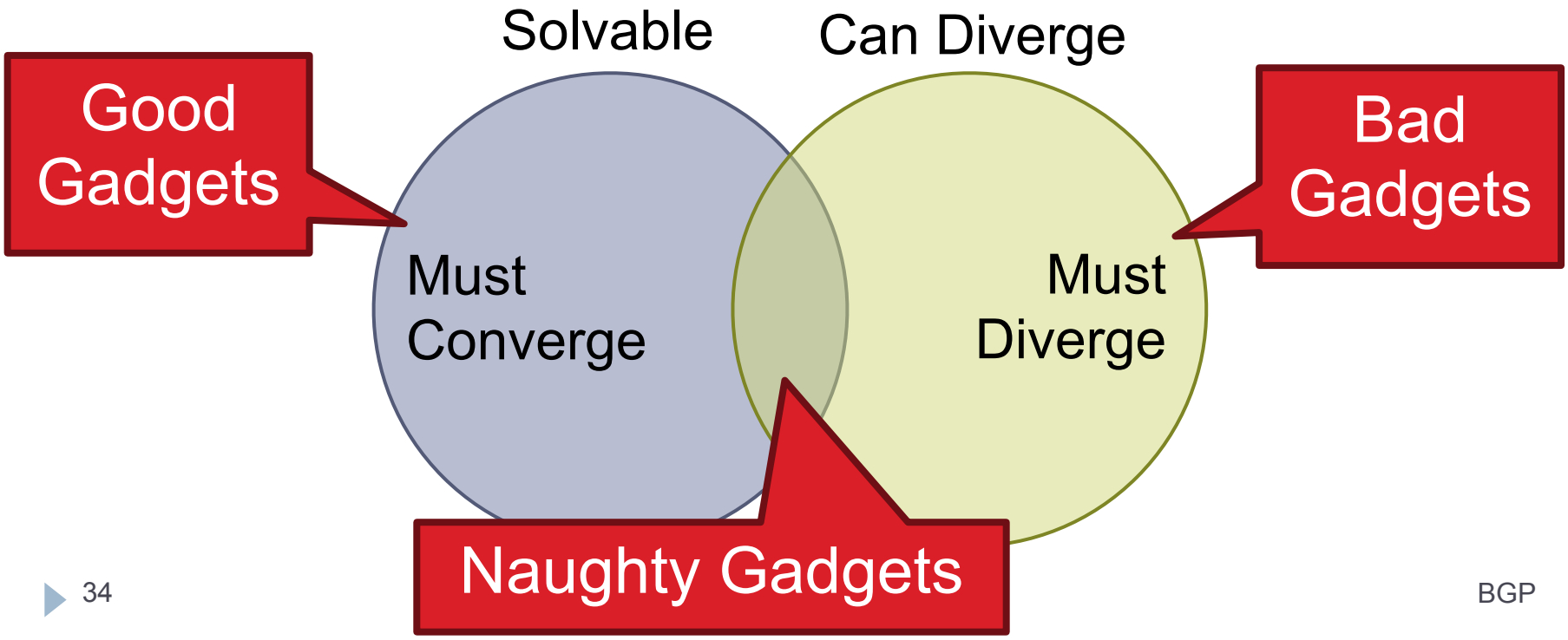
SPP Explains BGP Divergence

- BGP is not guaranteed to converge to stable routing
  - Policy inconsistencies may lead to “livelock”
  - Protocol oscillation

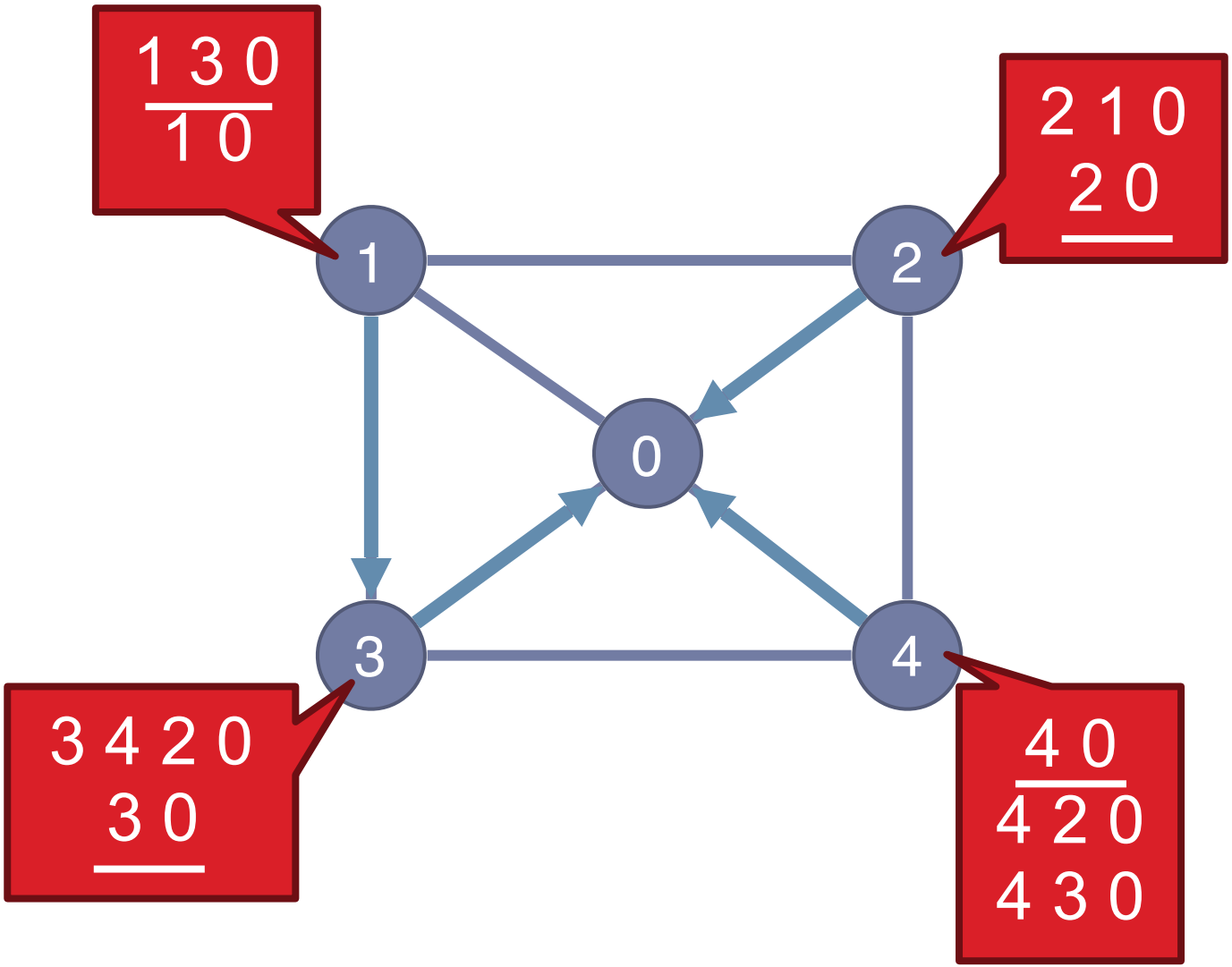


SPP Explains BGP Divergence

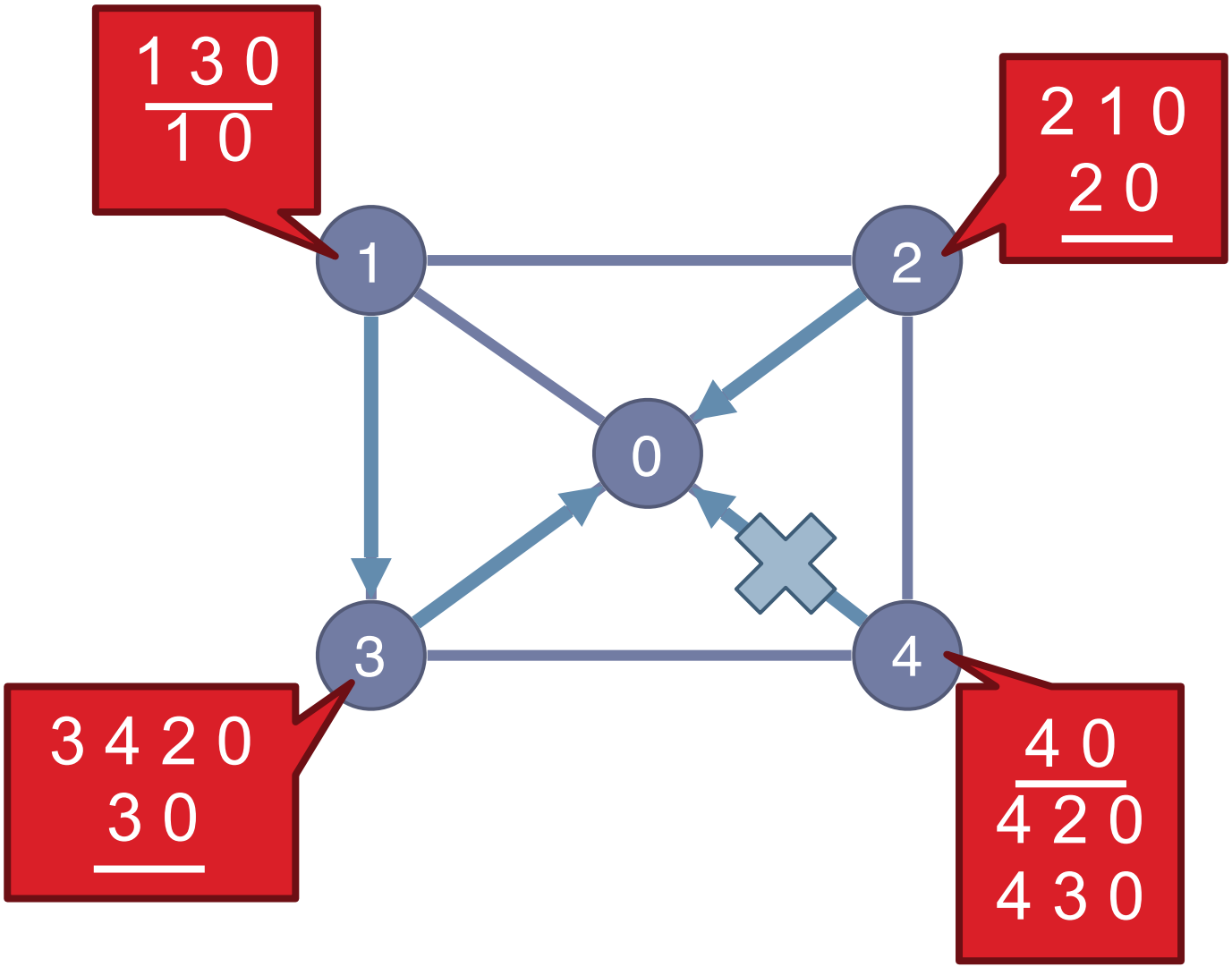
- BGP is not guaranteed to converge to stable routing
  - Policy inconsistencies may lead to "livelock"
  - Protocol oscillation



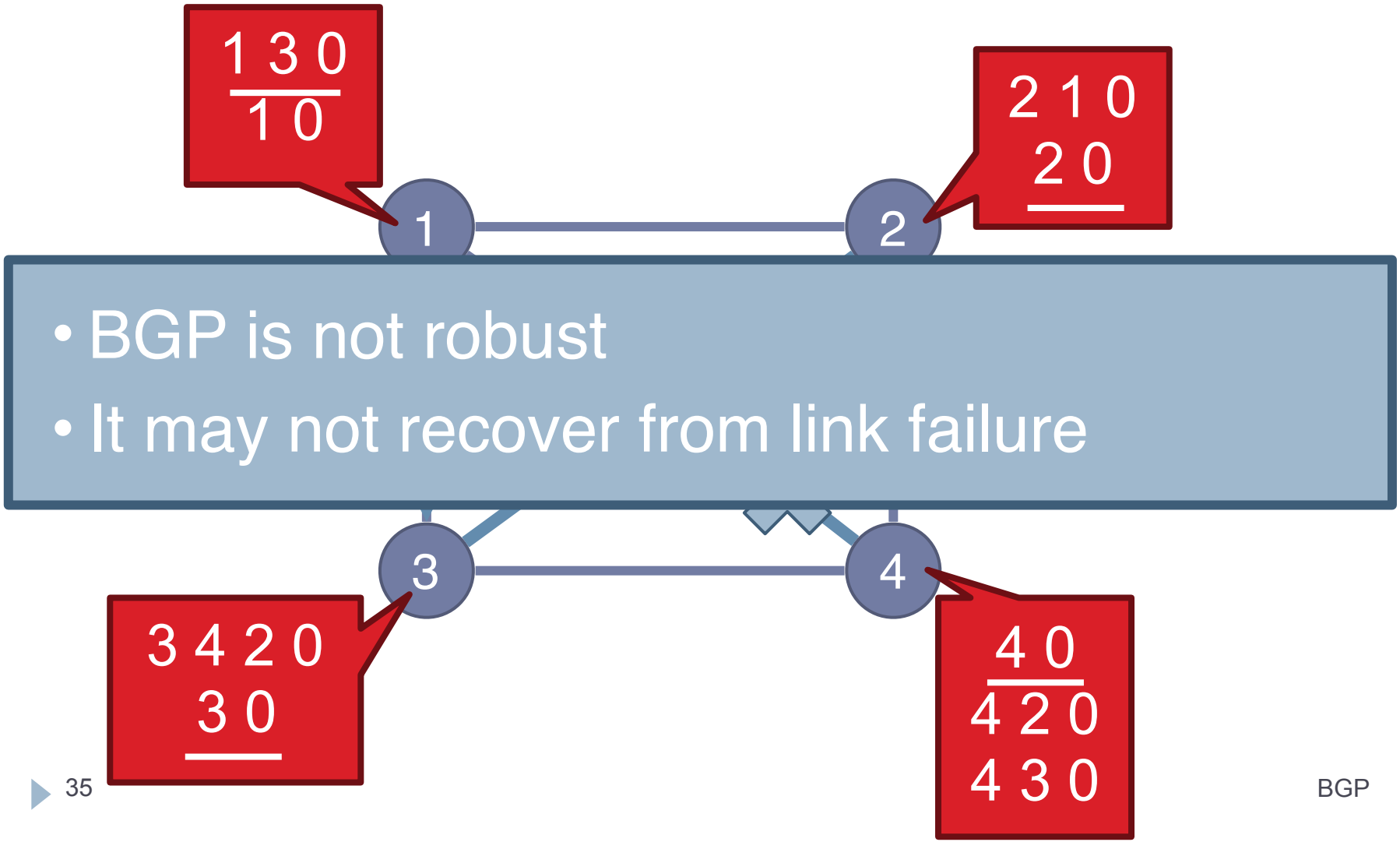
Beware of Backup Policies



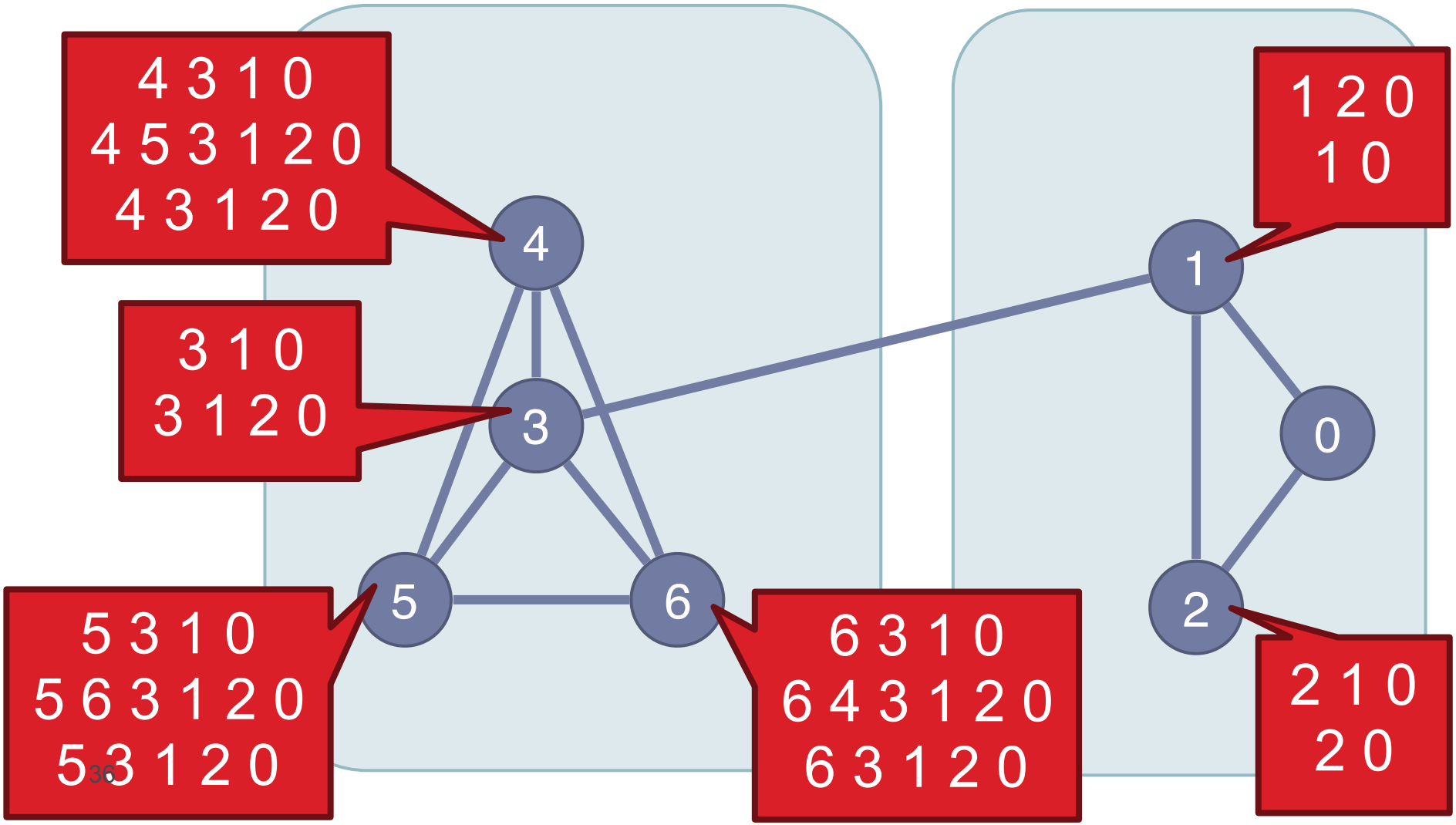
Beware of Backup Policies



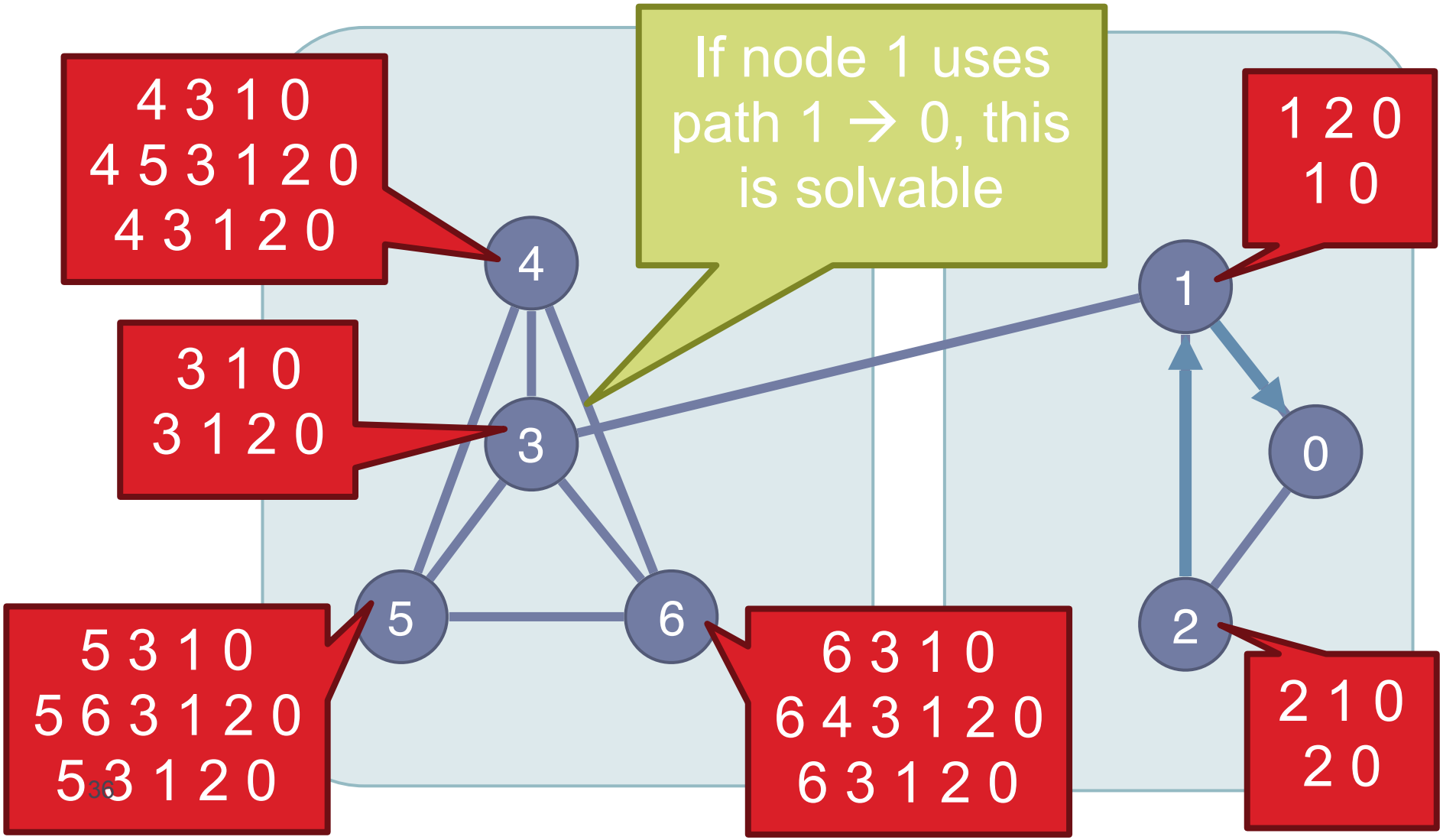
Beware of Backup Policies



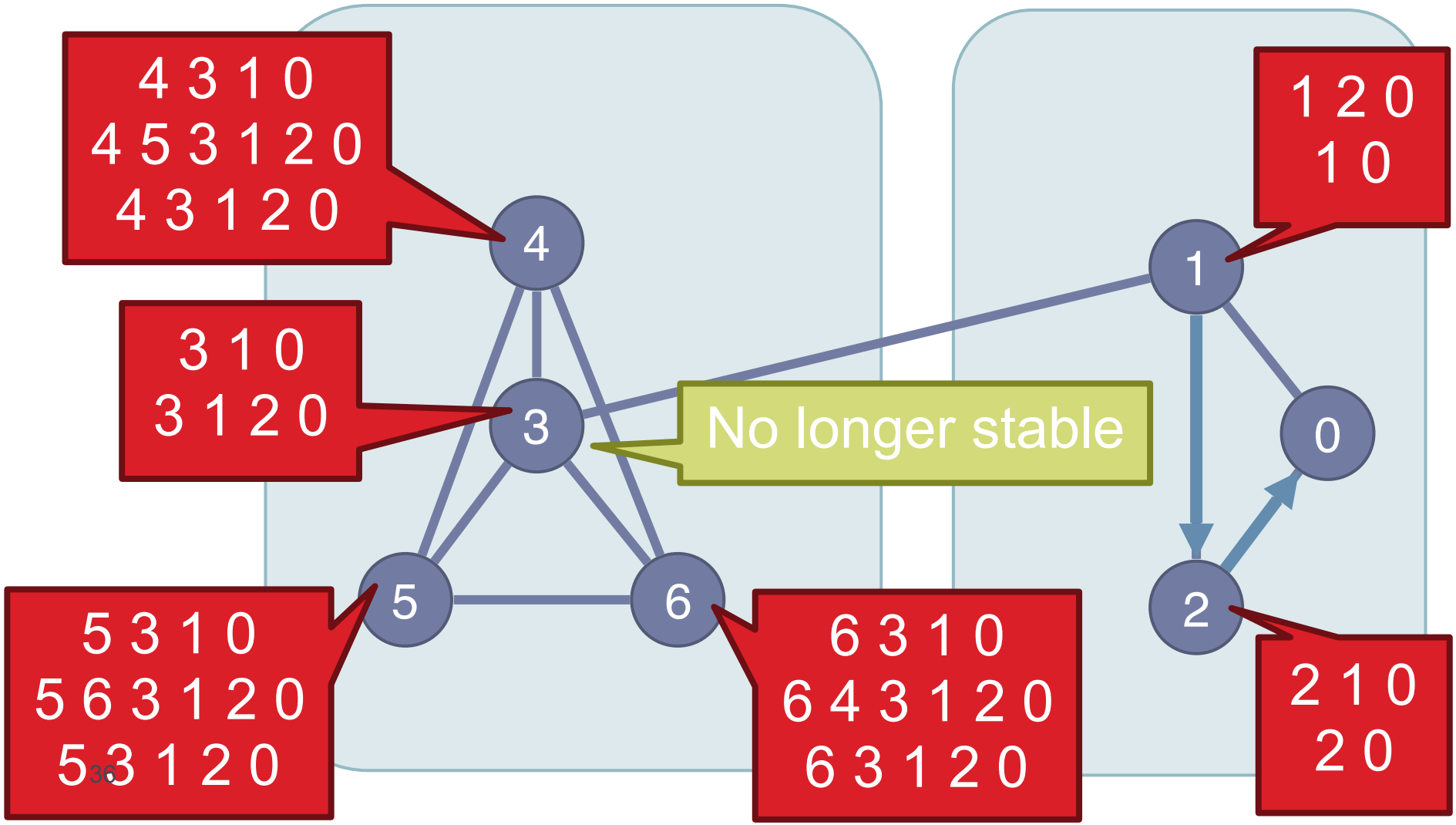
BGP is Precarious



BGP is Precarious



BGP is Precarious

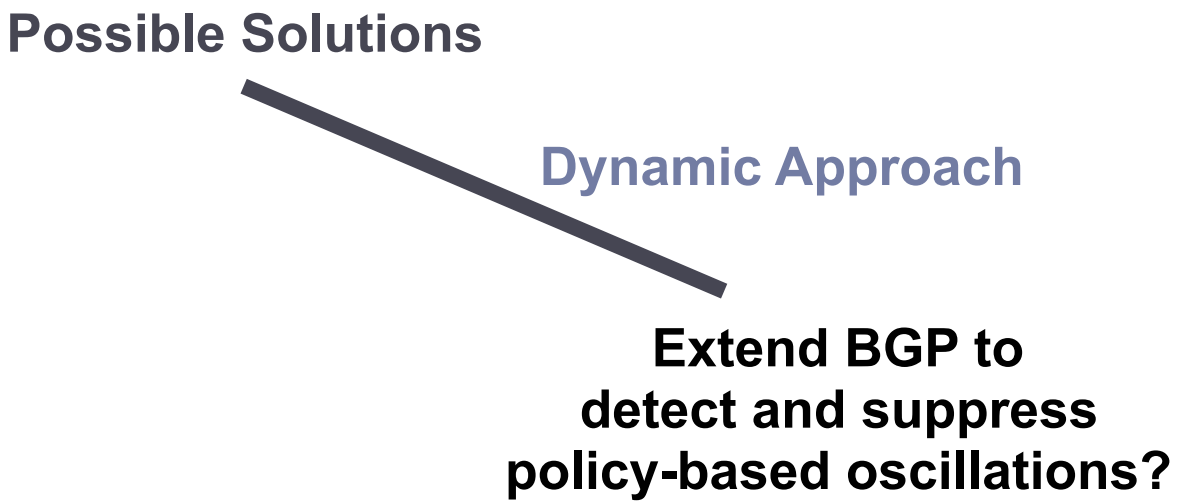


Can BGP Be Fixed?

- Unfortunately, SPP is NP-complete

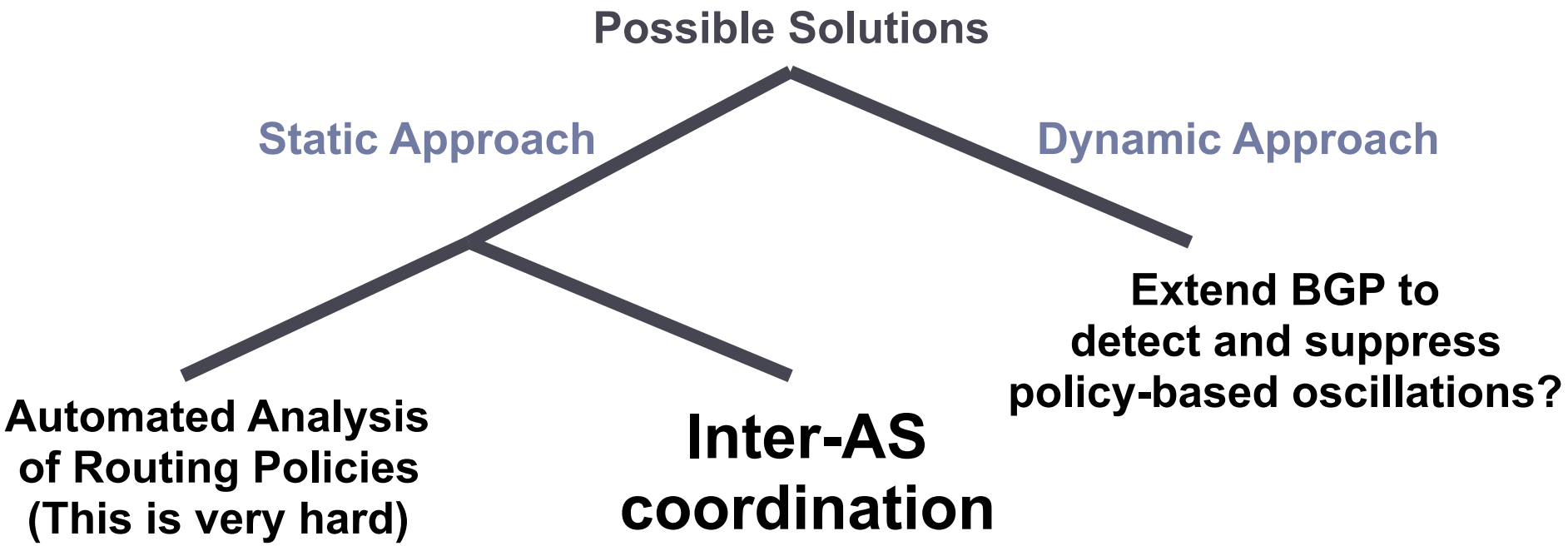
Can BGP Be Fixed?

- Unfortunately, SPP is NP-complete



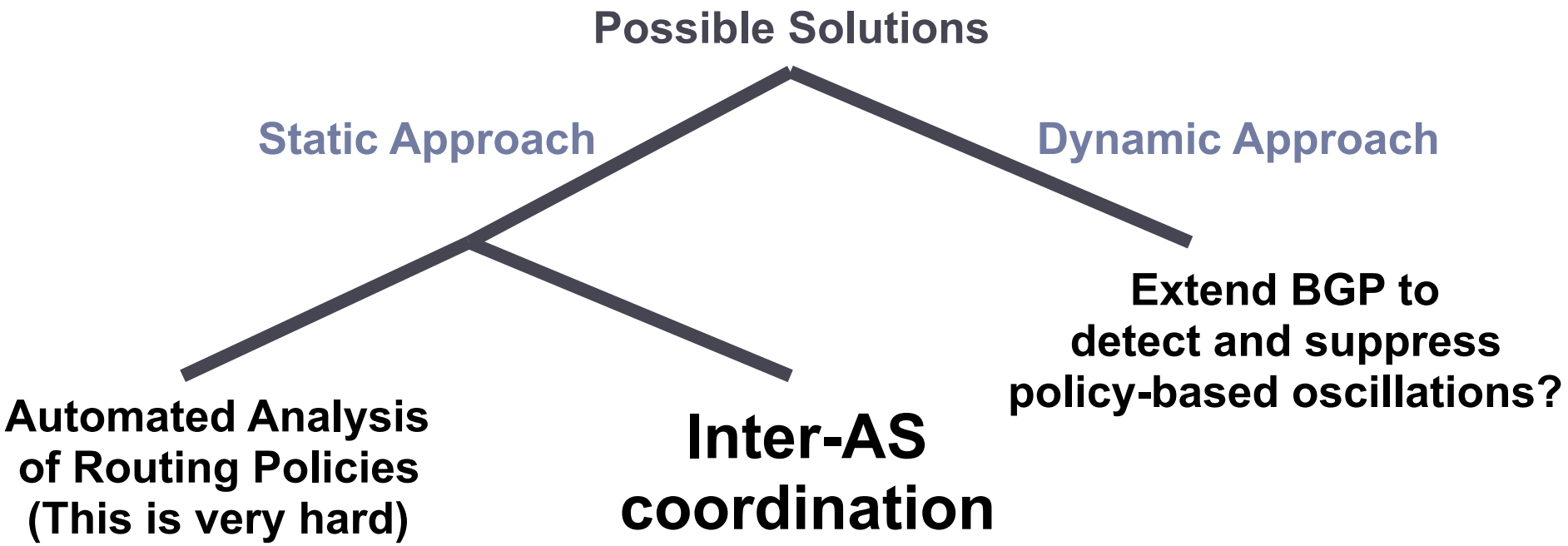
Can BGP Be Fixed?

- Unfortunately, SPP is NP-complete



Can BGP Be Fixed?

- Unfortunately, SPP is NP-complete



These approaches are complementary

3: Debugging BGP path problems

Control plane vs. Data Plane

- Control:
  - Make sure that if there's a path available, data is forwarded over it
  - BGP sets up such paths at the AS-level
- Data:
  - For a destination, send packet to most-preferred next hop
  - Routers forward data along IP paths

Control plane vs. Data Plane

- Control:
  - Make sure that if there's a path available, data is forwarded over it
  - BGP sets up such paths at the AS-level
- Data:
  - For a destination, send packet to most-preferred next hop
  - Routers forward data along IP paths

- How does the control plane know if a data path is broken?
  - Direct-neighbor connectivity
  - What if the outage isn't in the direct neighbor?

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

Improving Internet Availability

- New Internet design
  - Monitoring everywhere in the network
  - Visibility into all available routes
  - Any operator can impact routes affecting her traffic
- Challenges
  - What should we monitor?
  - What do we do with additional visibility?
  - How to use additional control?

A Practical Approach

- We can do this already in today's Internet
  - Crowdsourcing monitoring
  - Use existing protocols/systems in unintended ways
- Allows us to address problems today
  - Also informs future Internet designs

Operators Struggle to Locate Failures

"Traffic attempting to pass through Level3's network in the Washington, DC area is getting lost in the abyss. Here's a trace

from Verizon residential to Level3." *Outages mailing list, Dec. 2010*

Operators Struggle to Locate Failures

"Traffic attempting to pass through Level3's network in the Washington, DC area is getting lost in the abyss. Here's a trace from Verizon residential to Level3." *Outages mailing list, Dec. 2010*

Mailing List User 1

1 Home router

2 Verizon in Baltimore

3 Verizon in Philly

4 Alter.net in DC

5 Level3 in DC

6 * * *

7 * * *

Operators Struggle to Locate Failures

"Traffic attempting to pass through Level3's network in the Washington, DC area is getting lost in the abyss. Here's a trace from Verizon residential to Level3." *Outages mailing list, Dec. 2010*

Mailing List User 1

1 Home router
2 Verizon in Baltimore
3 Verizon in Philly
4 Alter.net in DC
5 Level3 in DC
6 * * *
7 * * *

Mailing List User 2

1 Home router
2 Verizon in DC
3 Alter.net in DC
4 Level3 in DC
5 Level3 in Chicago
6 Level3 in Denver
7 * * *
8 * * *

Reasons for Long-Lasting Outages

Long-term outages are:

- Repaired over slow, human timescales
- Not well understood
- Caused by routers advertising paths that do not work
  - E.g., corrupted memory on line card causes black hole
  - E.g., bad cross-layer interactions cause failed MPLS tunnel

Key Challenges for Internet Repair

- Lack of visibility
  - Where is the outage?
  - Which networks are (un)affected?
  - Who caused the outage?
- Lack of control
  - Reverse paths determined by possibly distant ASes
  - Limited means to affect such paths

Goals and Approach

Improve availability through:

- Failure isolation and remediation
- Identifying the AS(es) responsible for path changes

Key techniques:

- Visibility
  - Active measurements from distributed vantage points
  - Passive collection of BGP feeds
- Control
  - On-demand BGP prepending to route around outages
  - Active BGP measurements to identify alternative paths

LIFEGUARD: **L**ocating **I**nternet **F**ailures **E**ffectively and **G**enerating **U**sable **A**lternate **R**outes **D**ynamically

- Locate the ISP / link causing the problem
  - Building blocks
  - Example
  - Description of technique

- Suggest that other ISPs reroute around the problem

LIFEGUARD: **L**ocating **I**nternet **F**ailures **E**ffectively and **G**enerating **U**sable **A**lternate **R**outes **D**ynamically

- Locate the ISP / link causing the problem
  - Building blocks
  - Example
  - Description of technique

- Suggest that other ISPs reroute around the problem

Building blocks for failure isolation

LIFEGUARD can use:

- Ping to test reachability
- Traceroute to measure forward path
- Distributed vantage points (VPs)
  - PlanetLab for our experiments
  - Some can source spoof
- Reverse traceroute to measure reverse path (NSDI '10)
  - I'll teach you about this during the security lecture
- Atlas of historical forward/reverse paths between VPs and targets

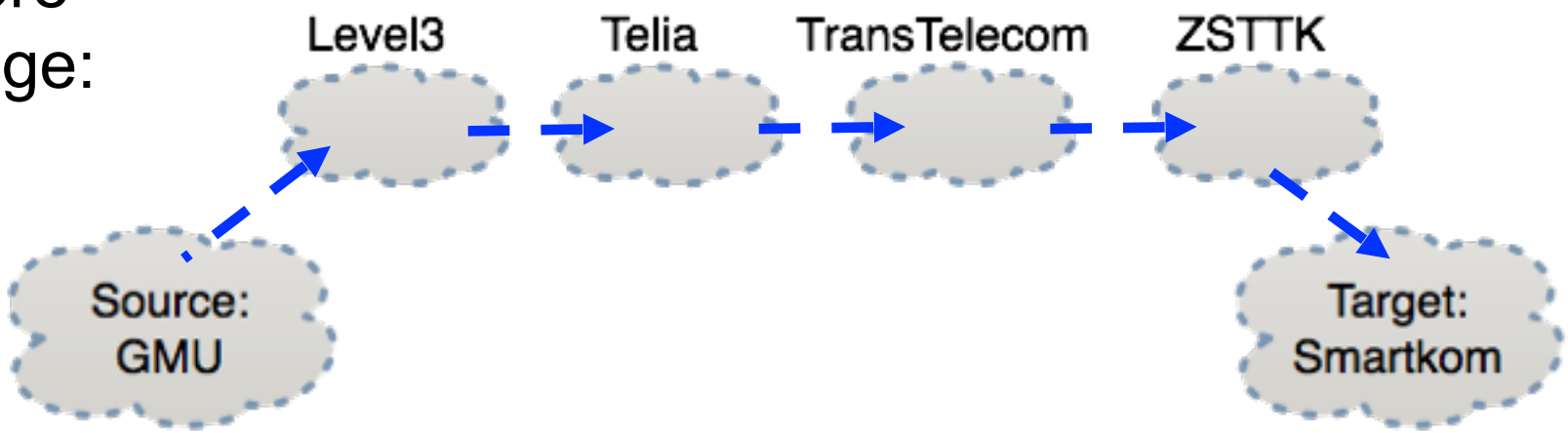
How Does **LIFEGUARD** Locate a Failure?

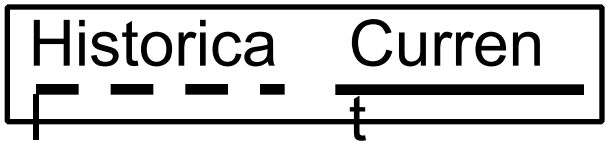
Before outage:

- Historical atlas enables reasoning about changes
- Traceroute yields only path from GMU to target
- Reverse traceroute reveals path asymmetry

How Does **LIFEGUARD** Locate a Failure?

Before outage:

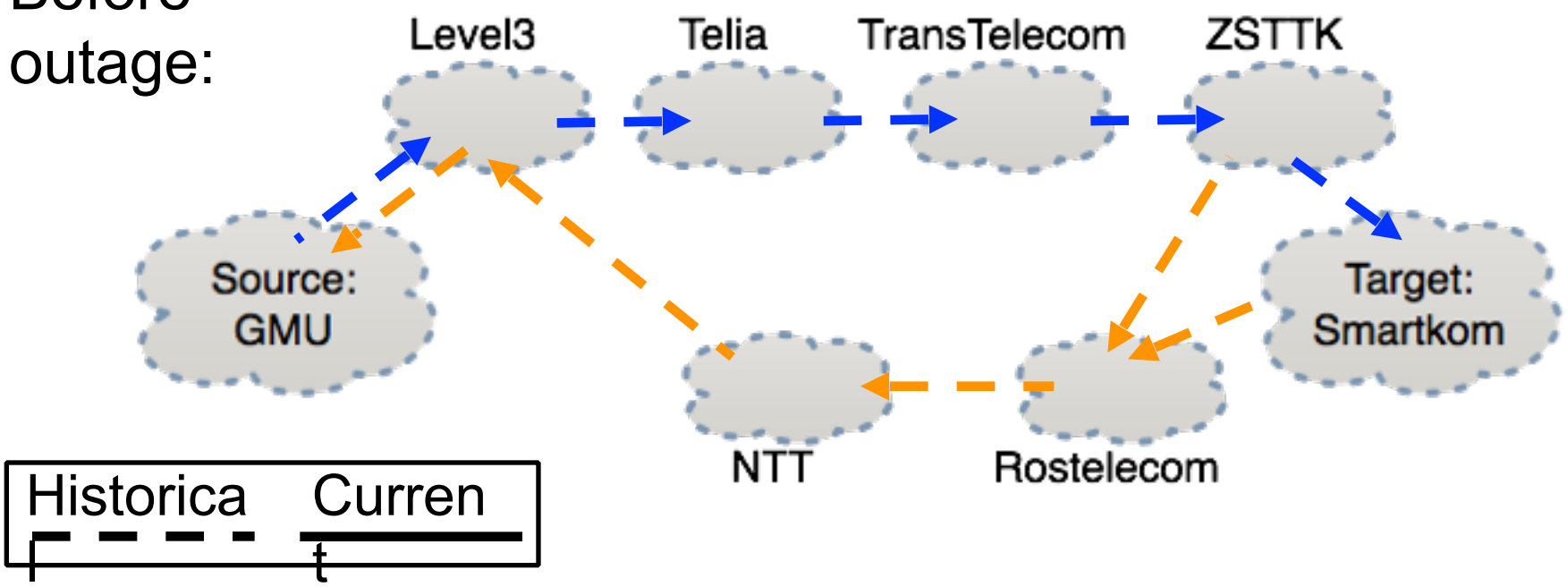




- Historical atlas enables reasoning about changes
- Traceroute yields only path from GMU to target
- Reverse traceroute reveals path asymmetry

How Does **LIFEGUARD** Locate a Failure?

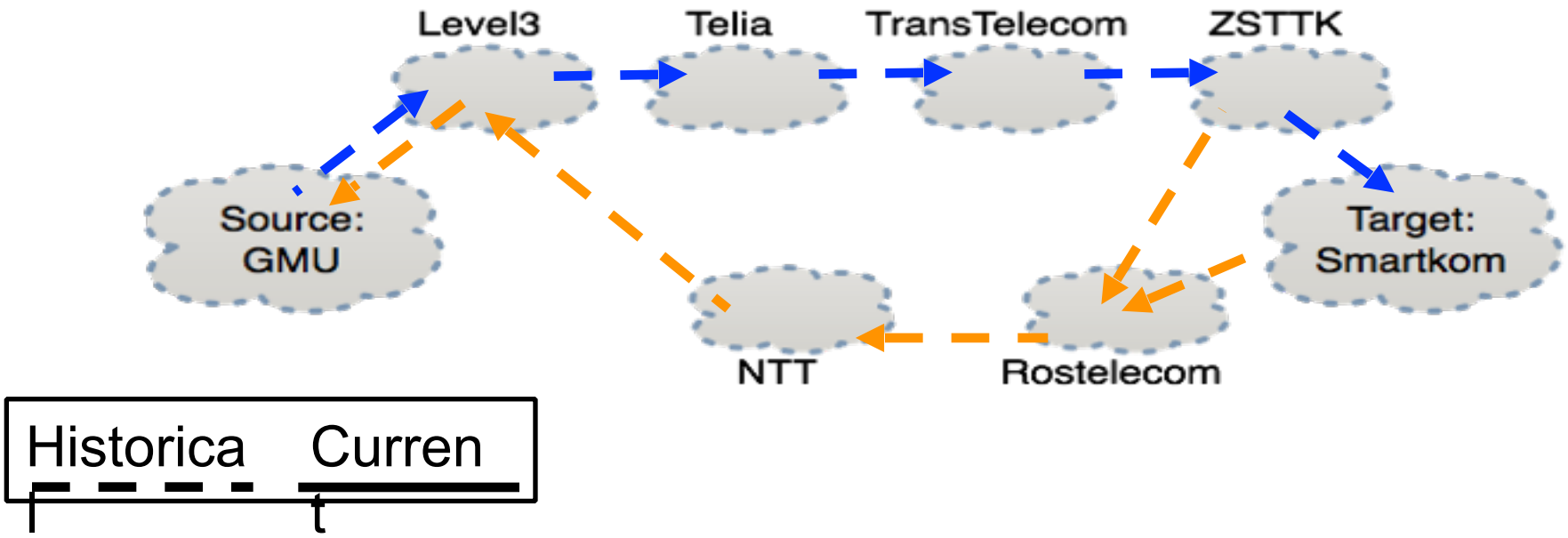
Before outage:



- Historical atlas enables reasoning about changes
- Traceroute yields only path from GMU to target
- Reverse traceroute reveals path asymmetry

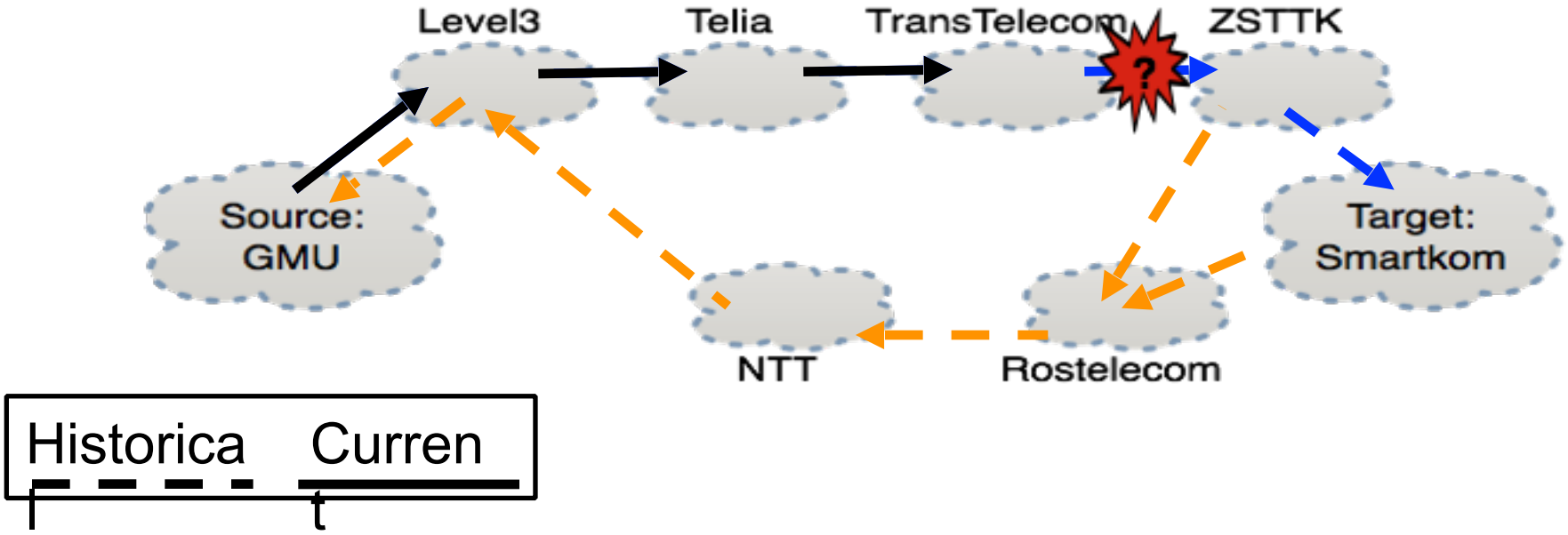
How Does **LIFEGUARD** Locate a Failure?

During outage:

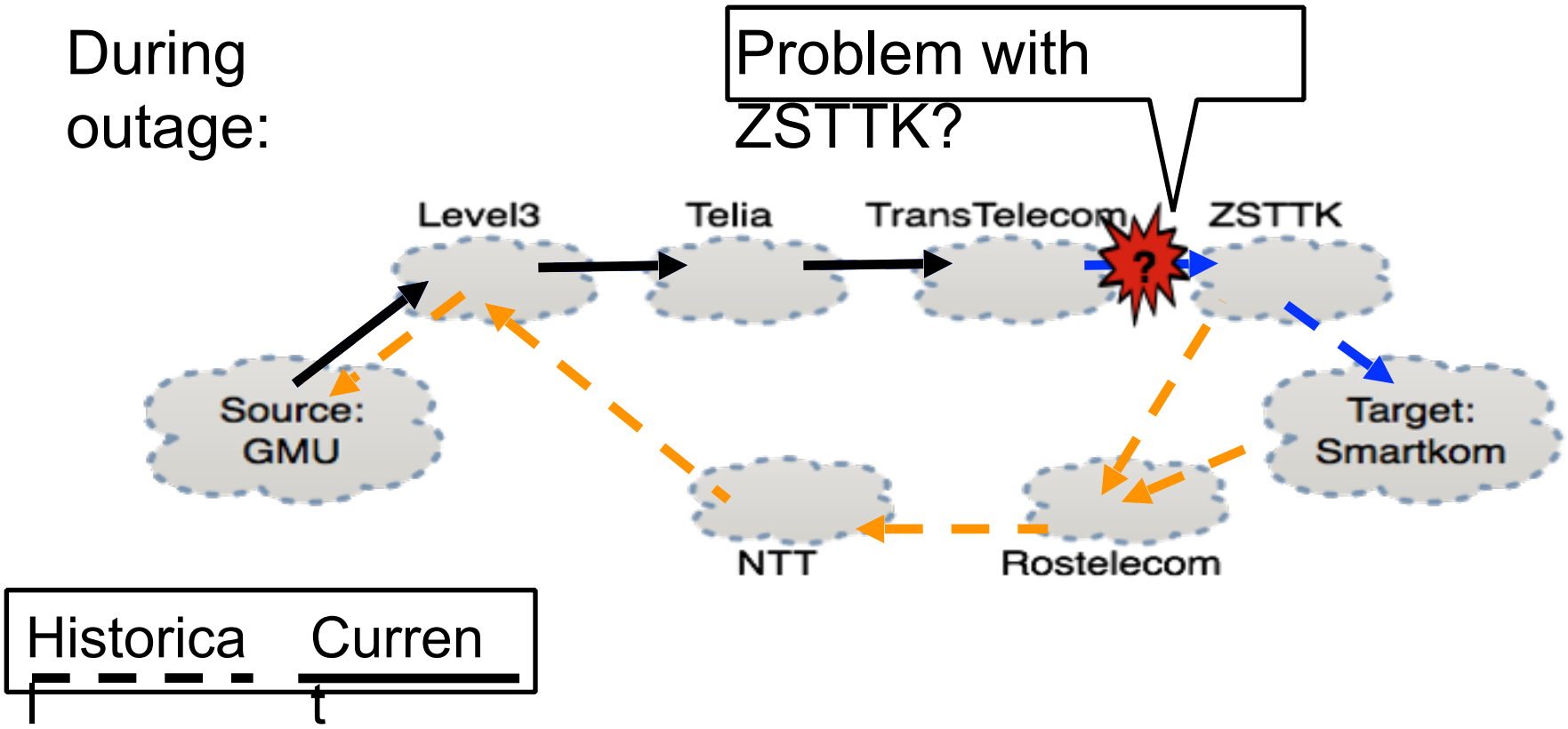


How Does **LIFEGUARD** Locate a Failure?

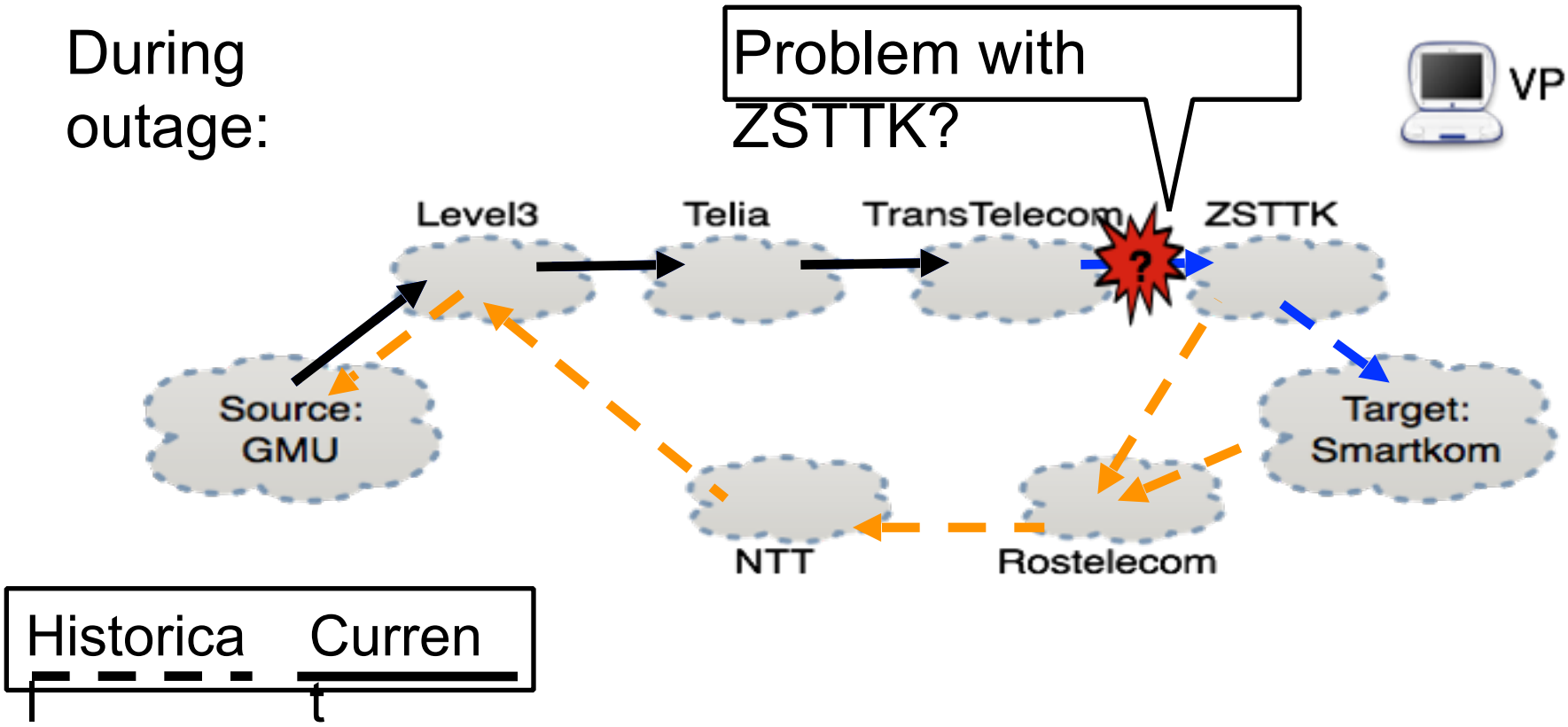
During outage:



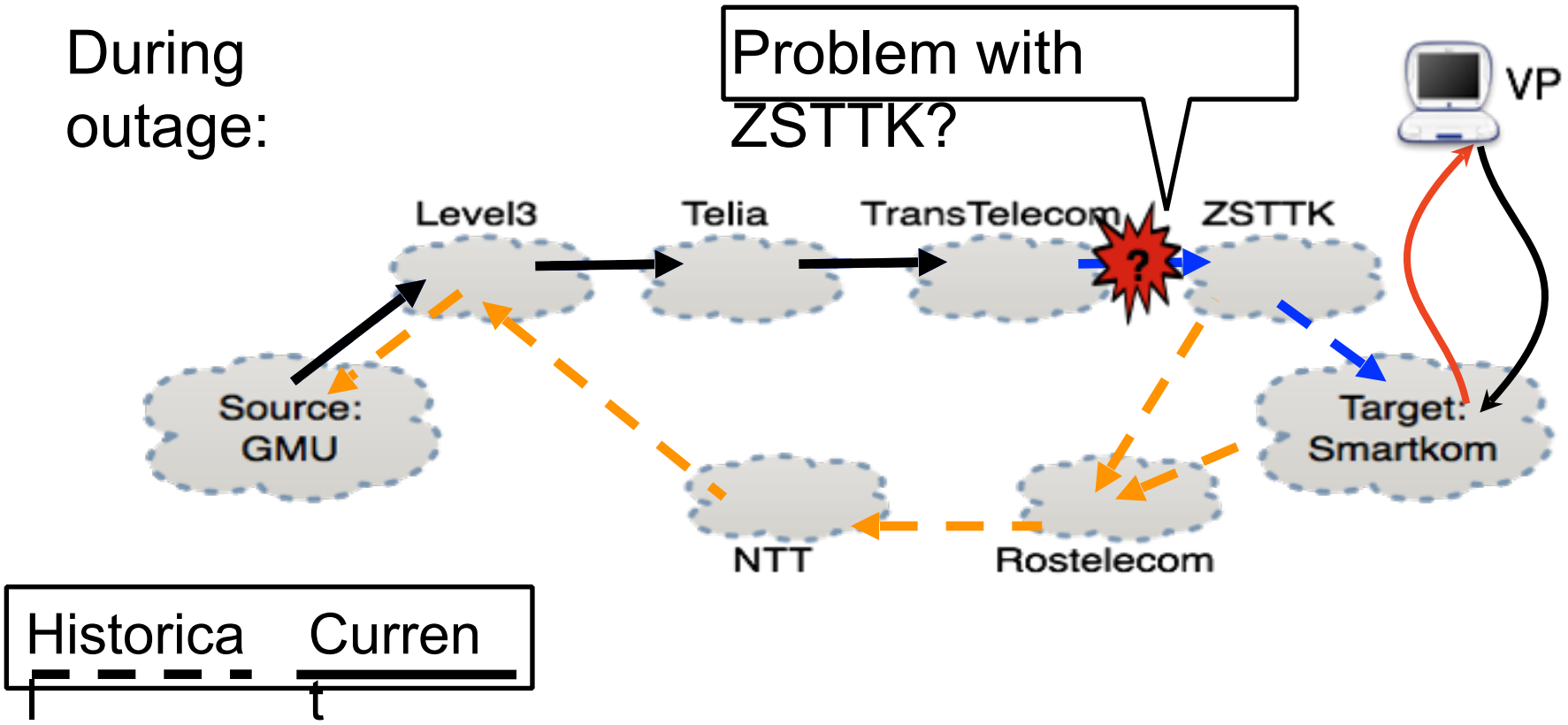
How Does **LIFEGUARD** Locate a Failure?



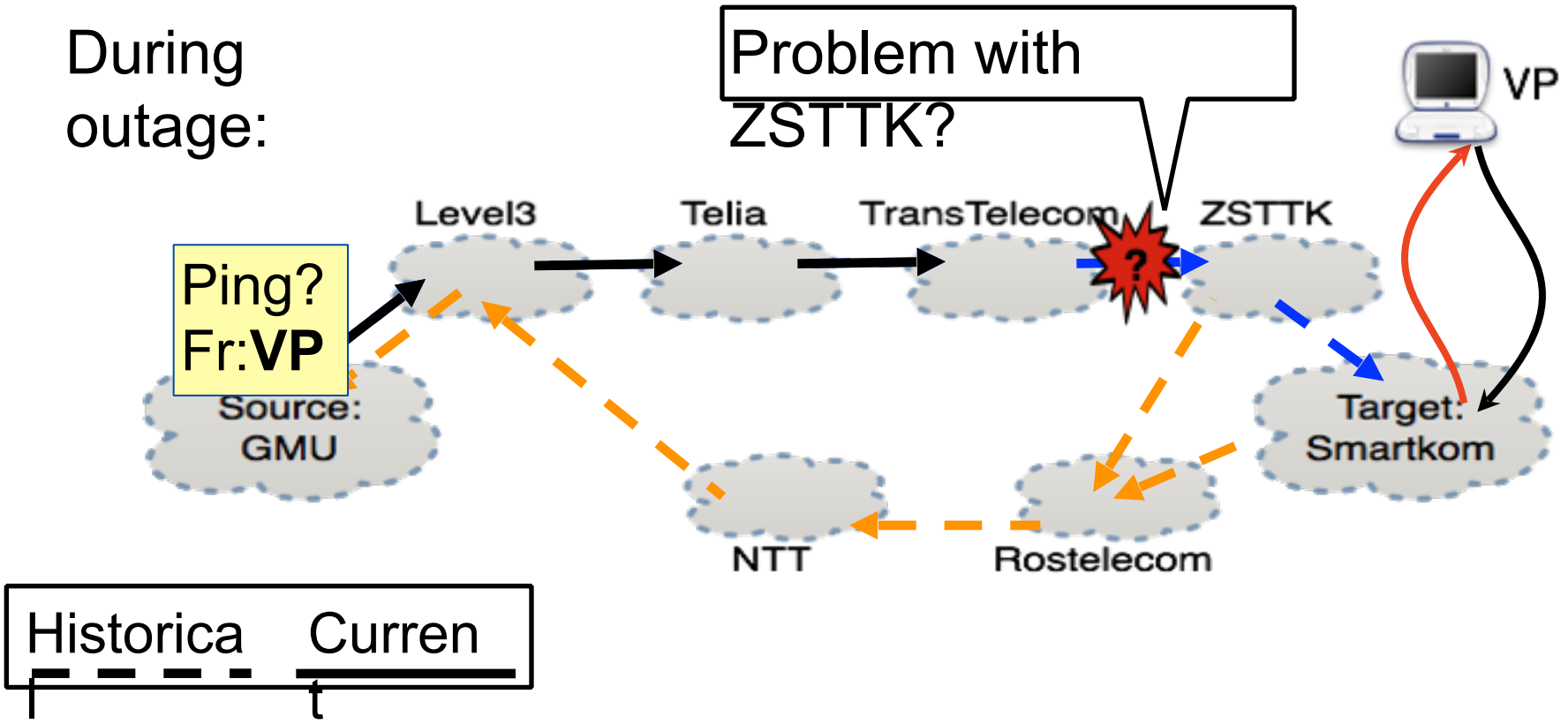
How Does **LIFEGUARD** Locate a Failure?



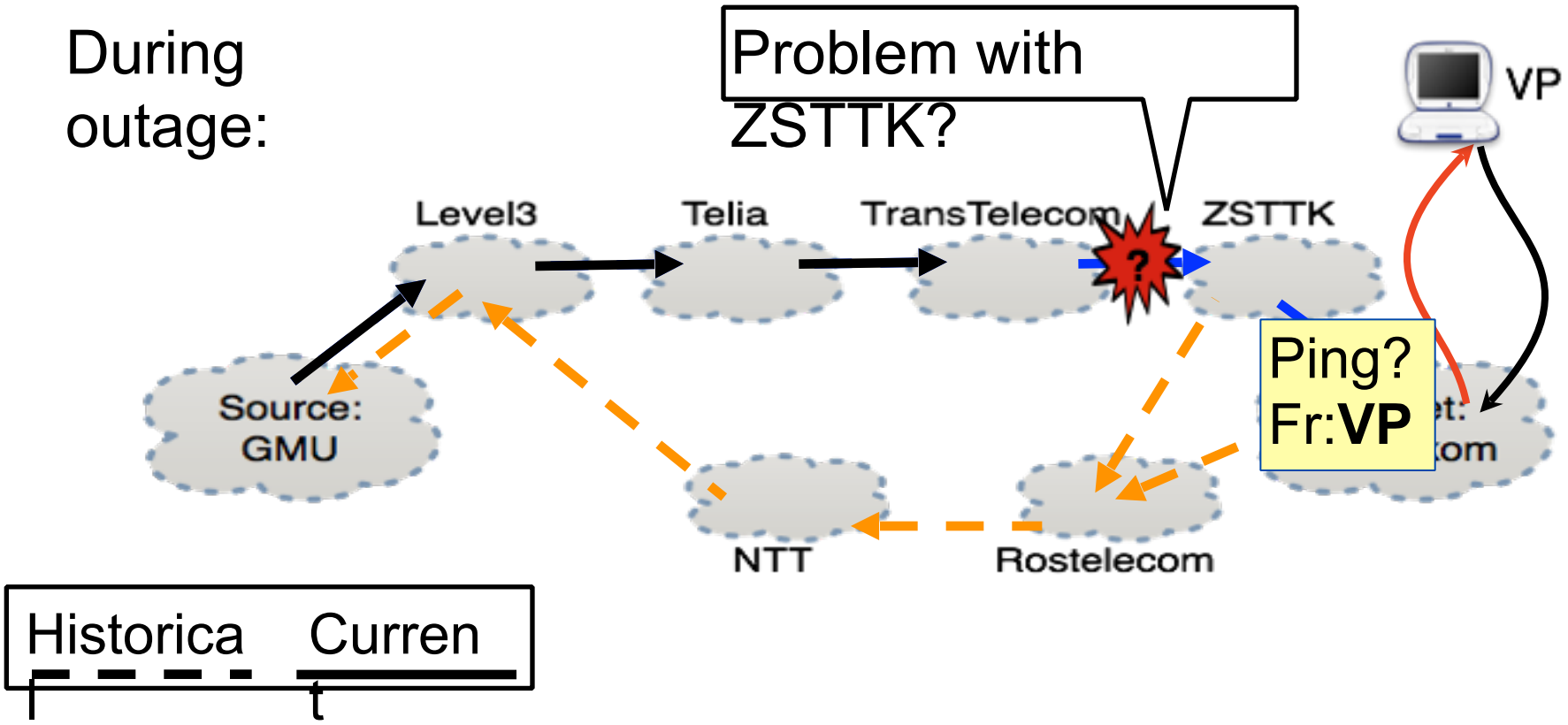
How Does **LIFEGUARD** Locate a Failure?



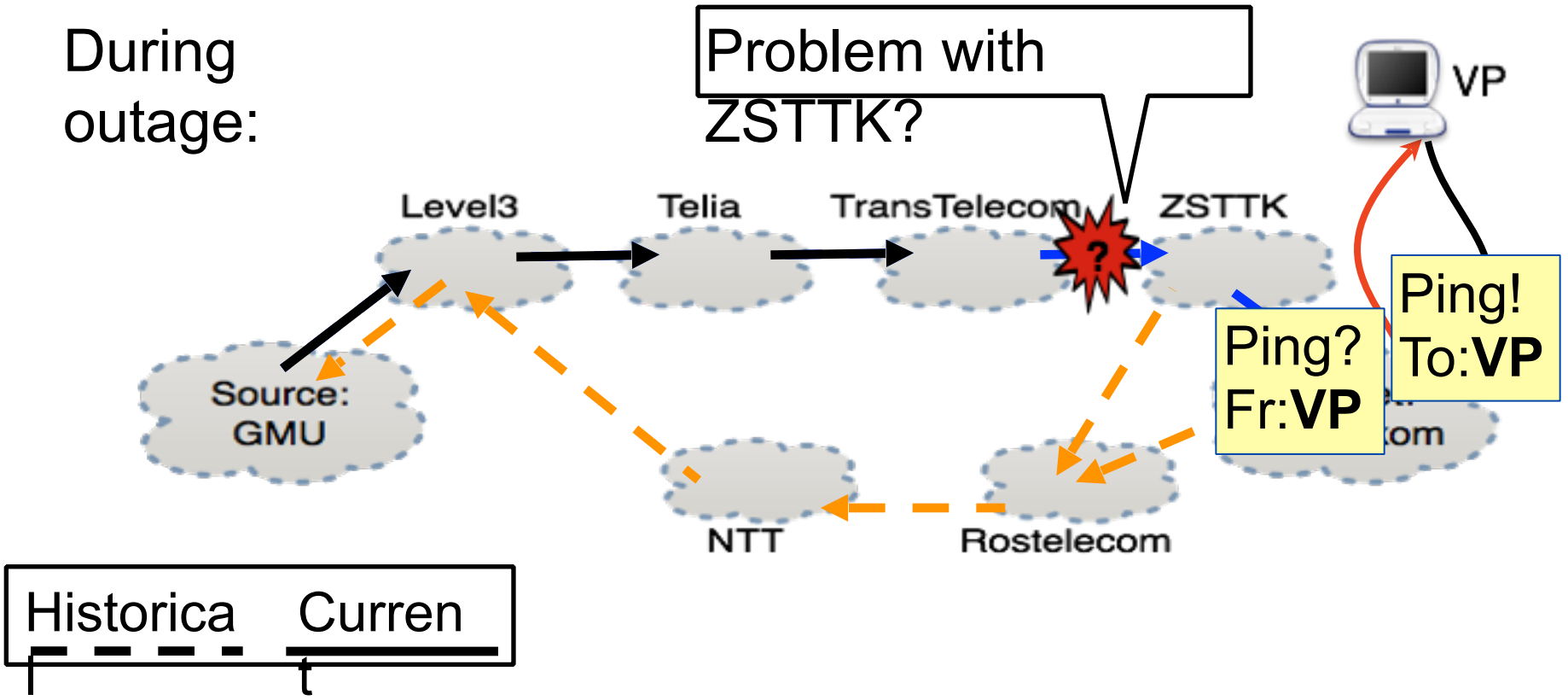
How Does **LIFEGUARD** Locate a Failure?



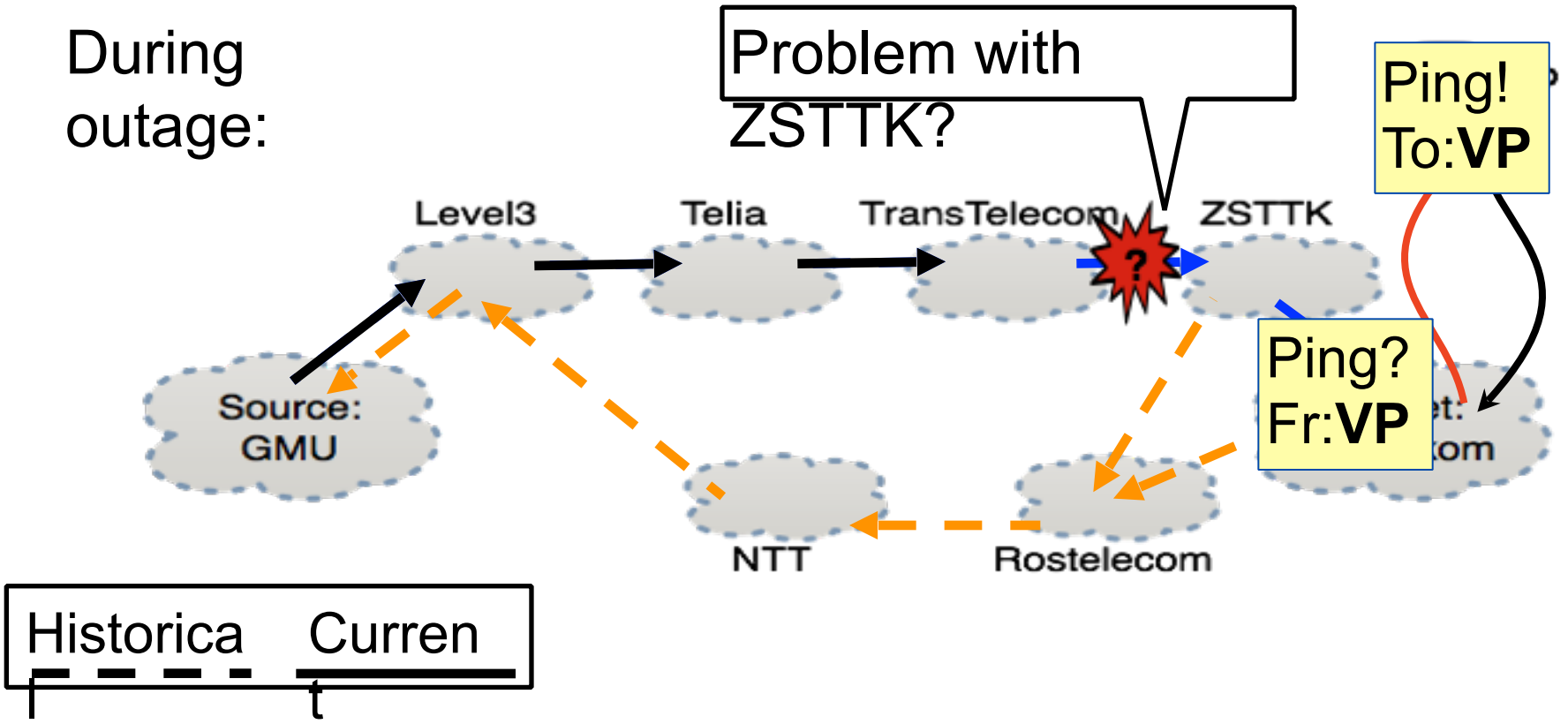
How Does **LIFEGUARD** Locate a Failure?



How Does **LIFEGUARD** Locate a Failure?

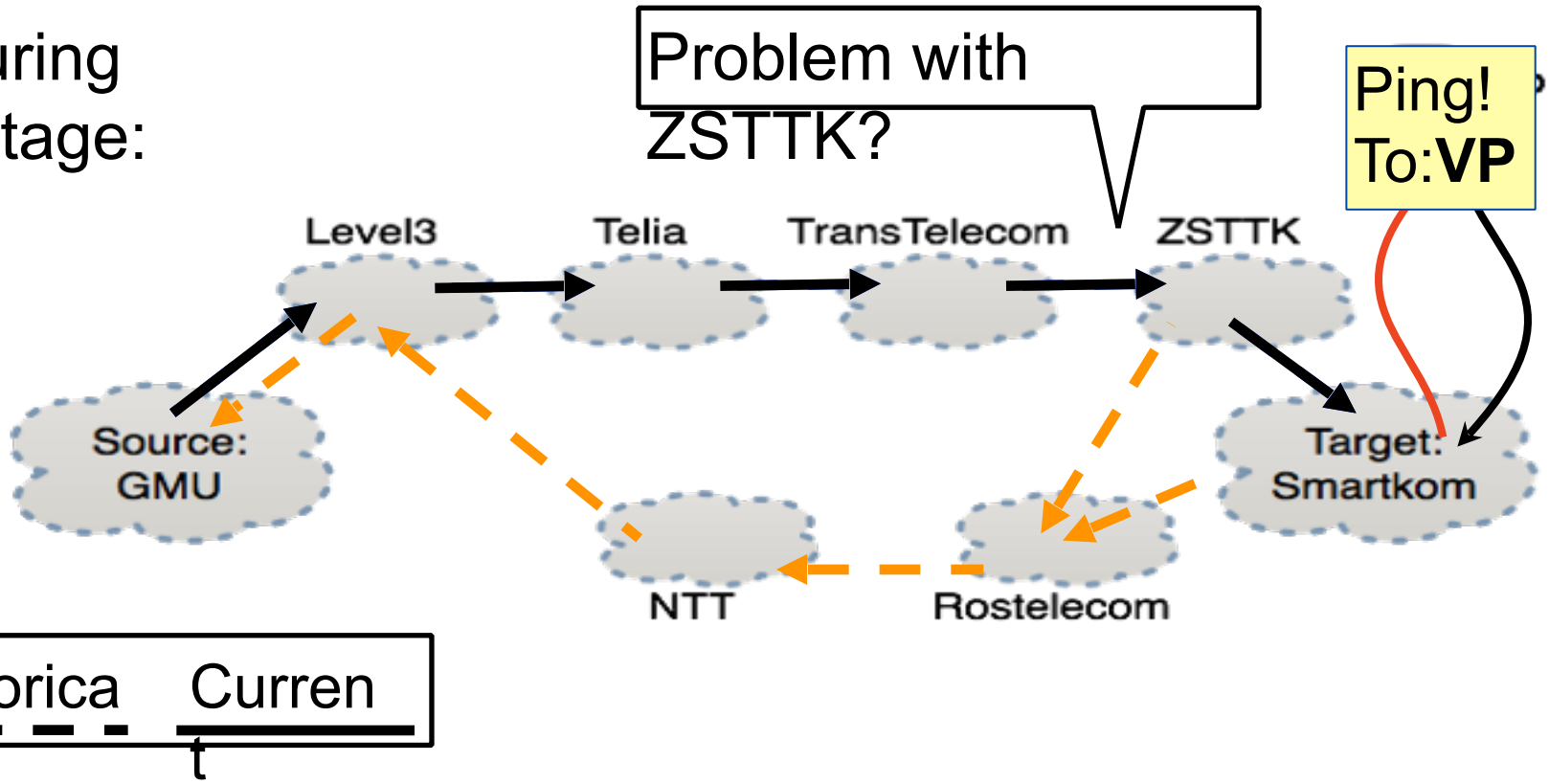


How Does **LIFEGUARD** Locate a Failure?



How Does **LIFEGUARD** Locate a Failure?

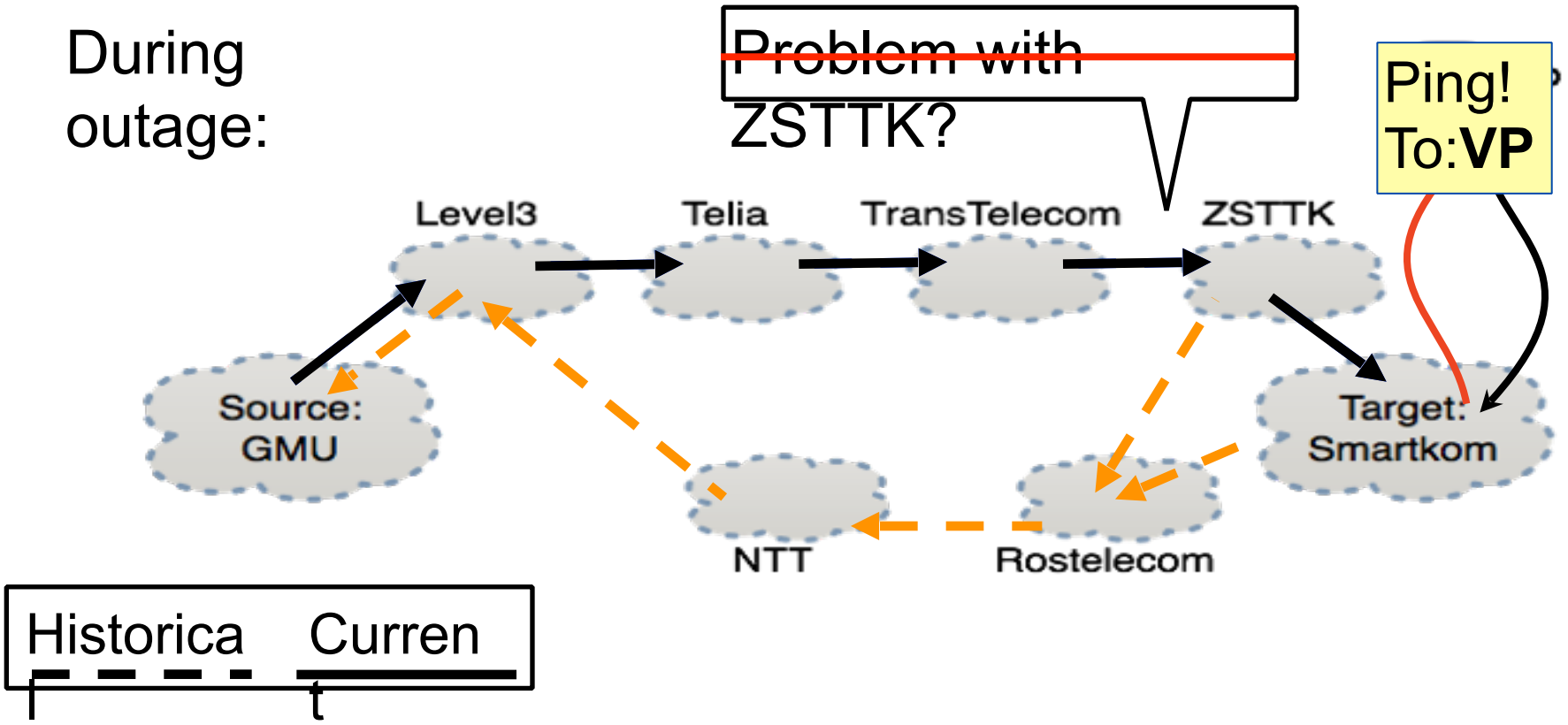
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

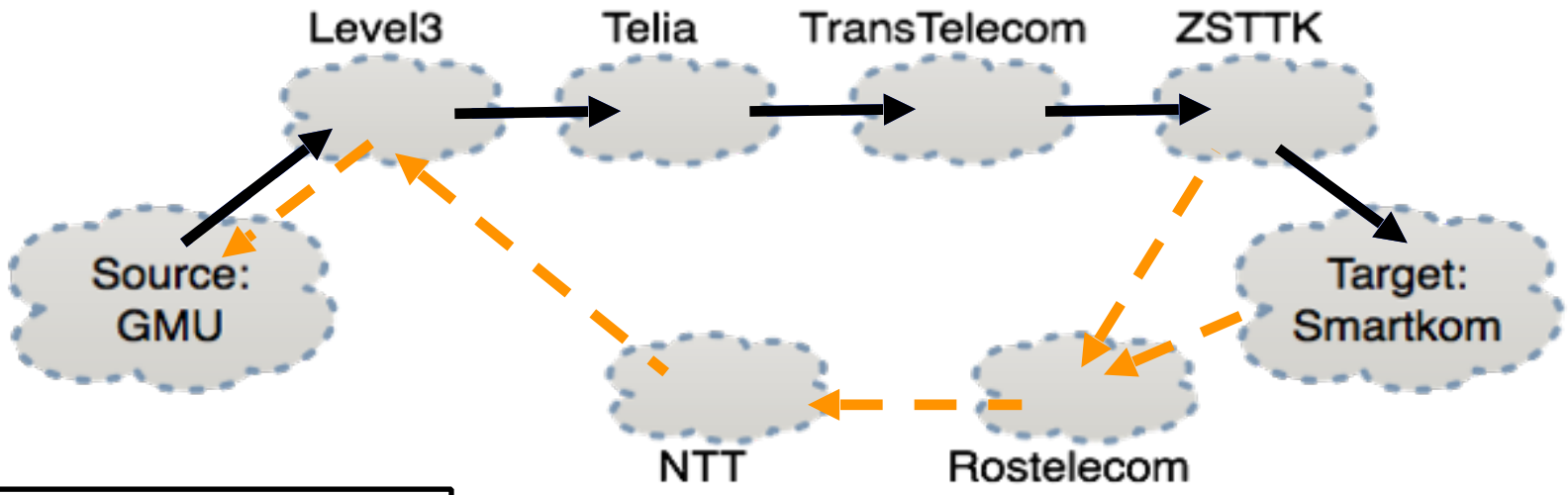
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

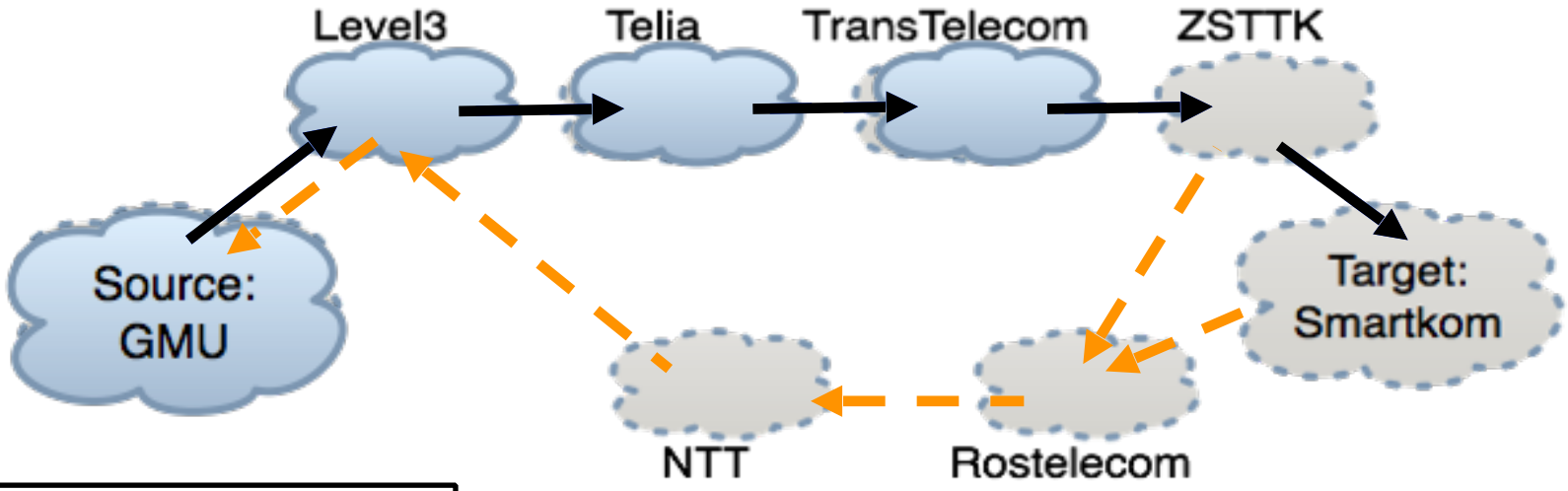
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

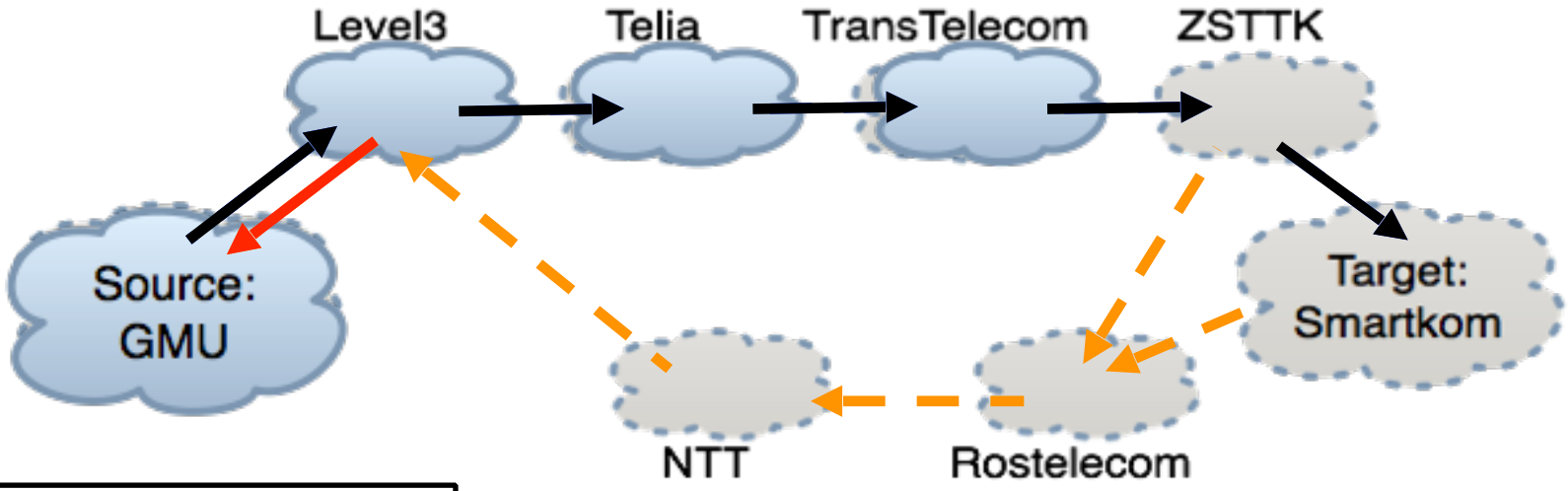
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

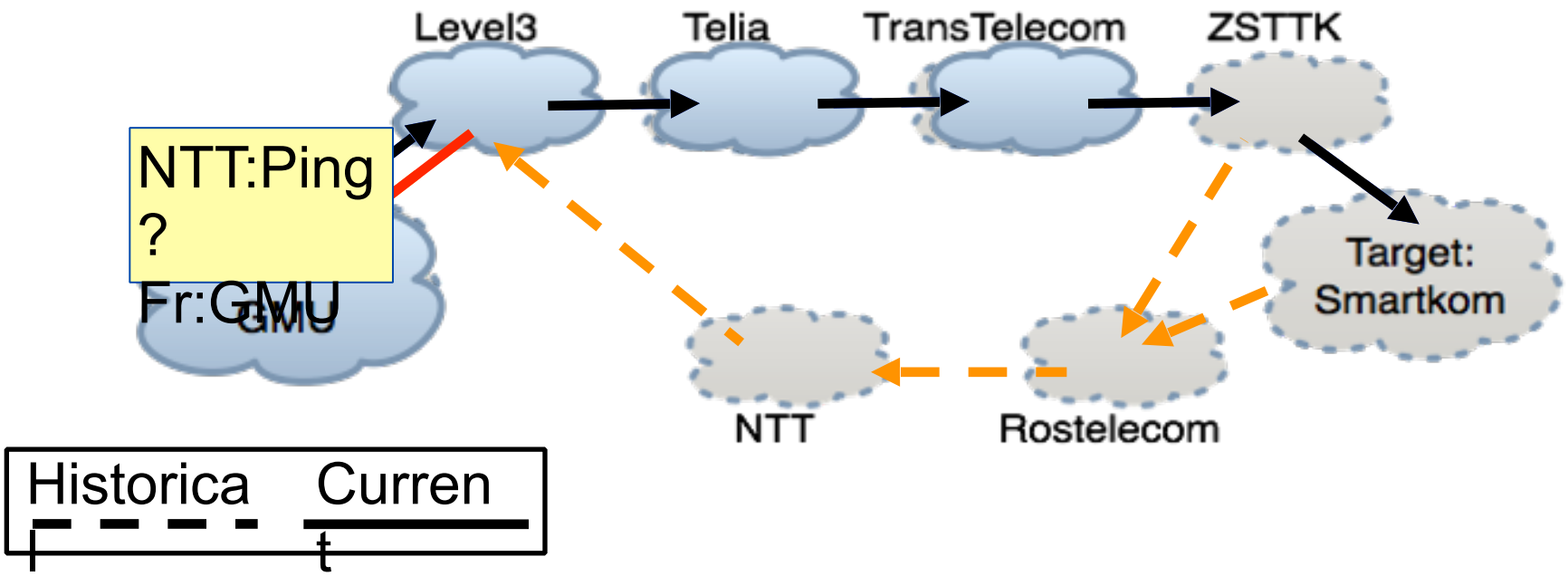
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

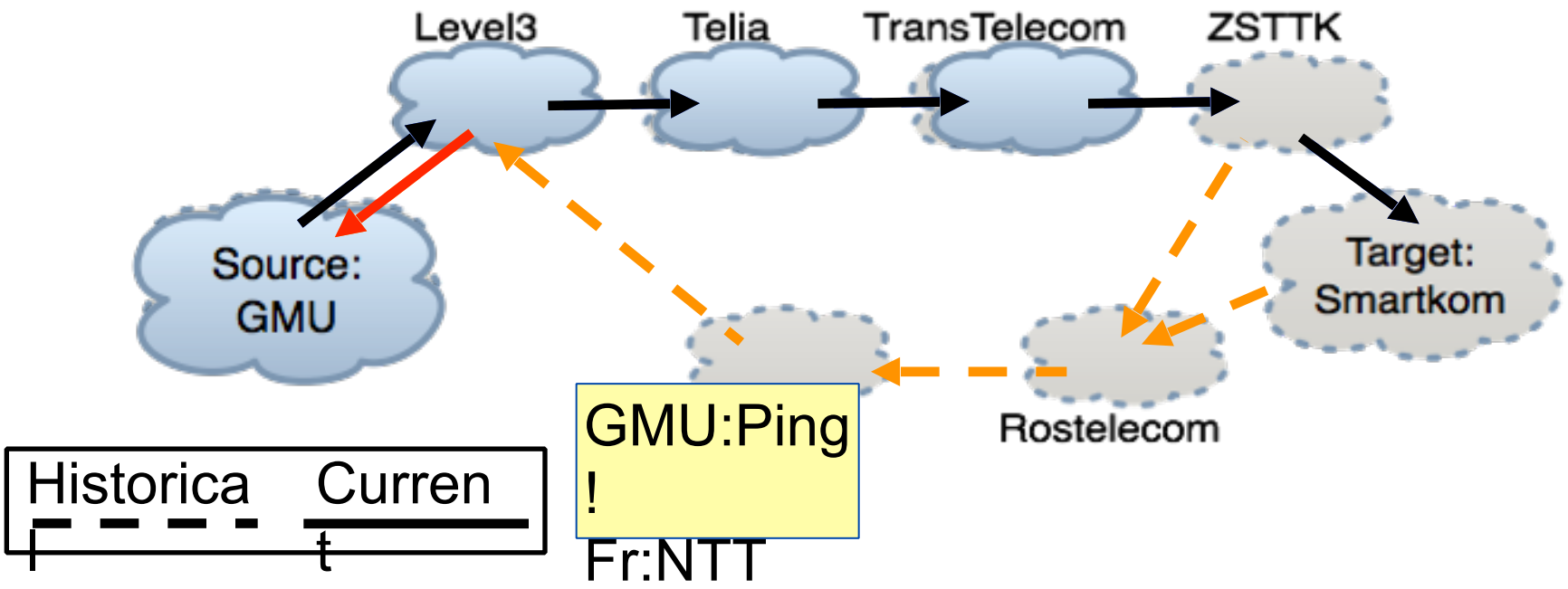
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

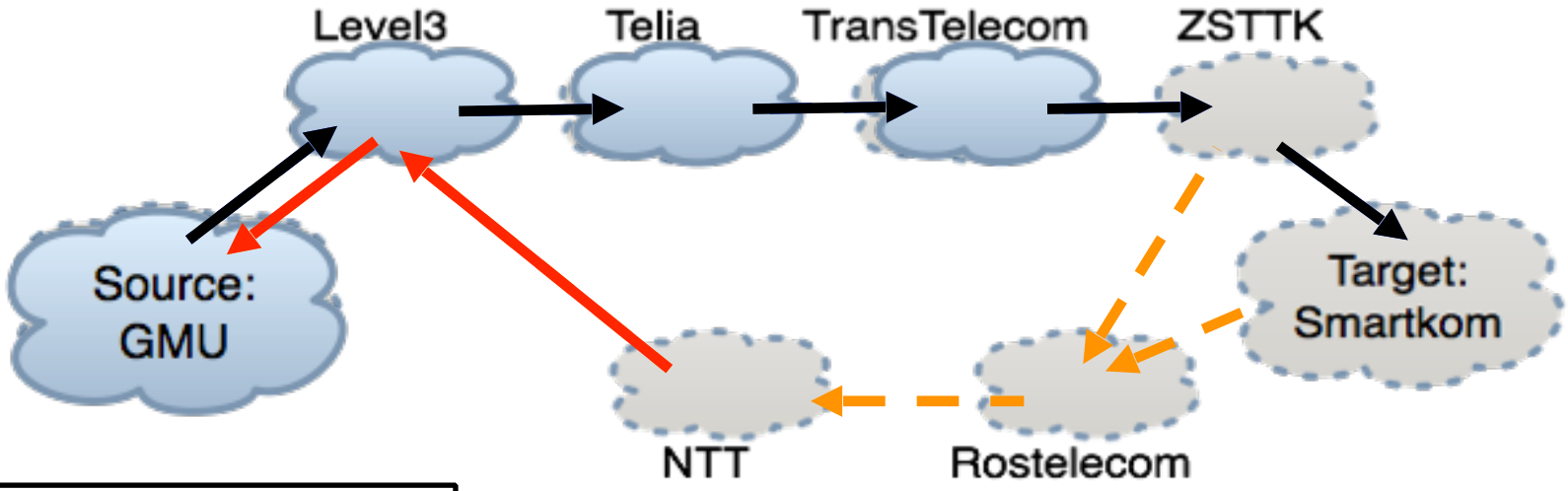
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

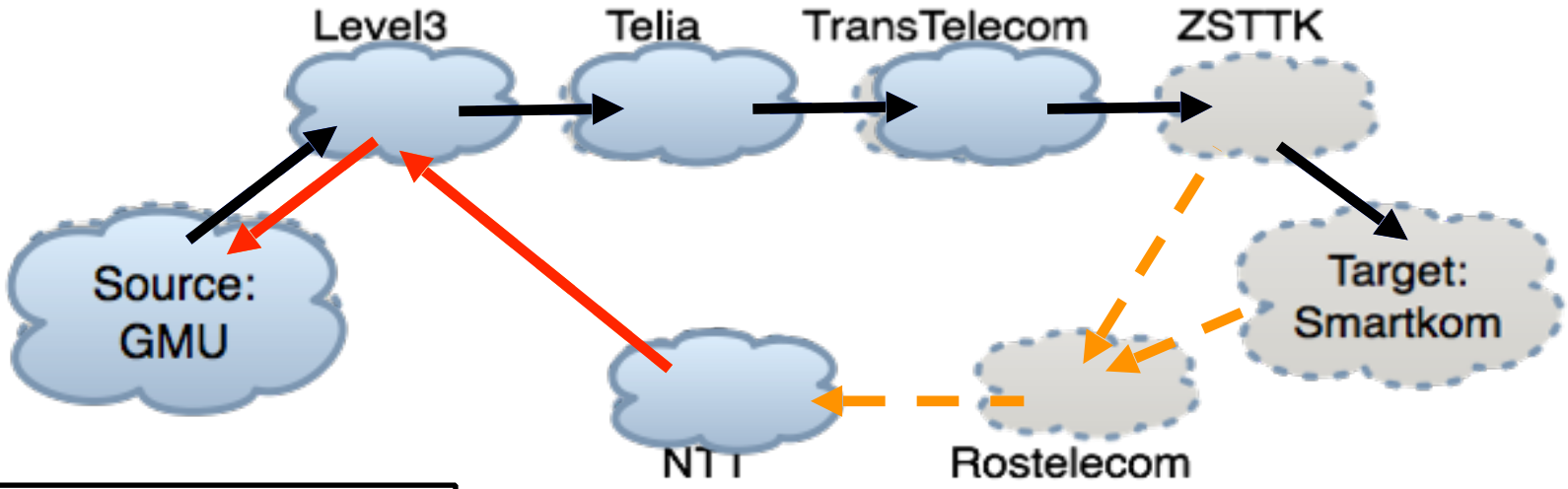
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

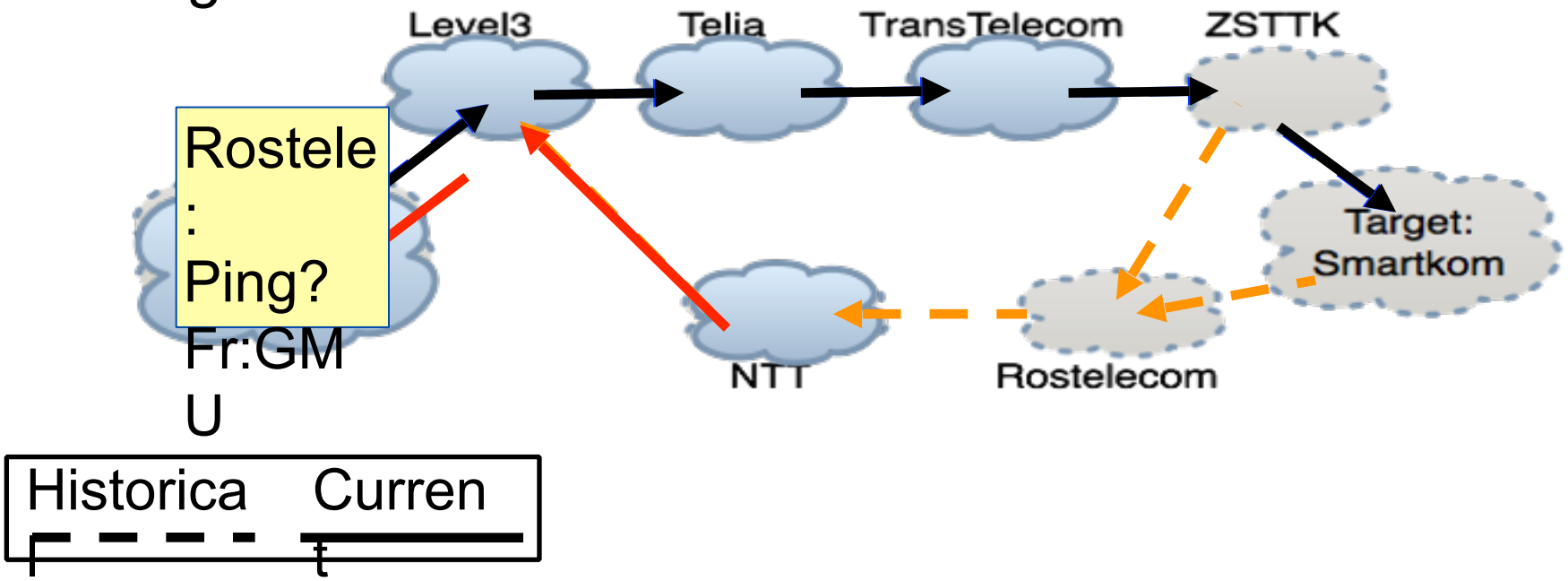
During outage:



- Forward path works

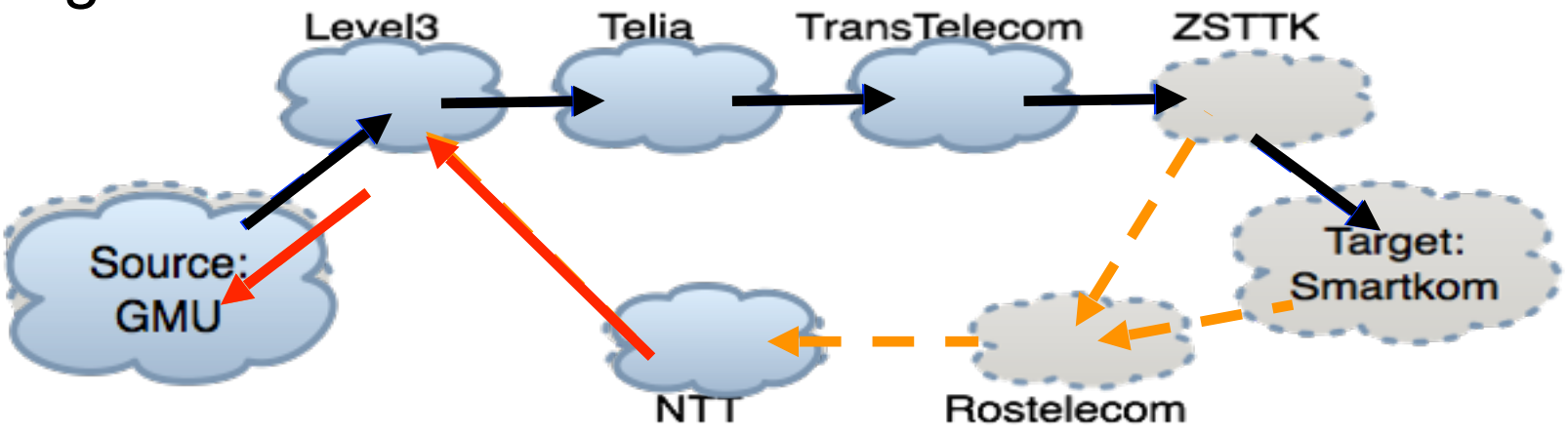
How Does **LIFEGUARD** Locate a Failure?

During outage:



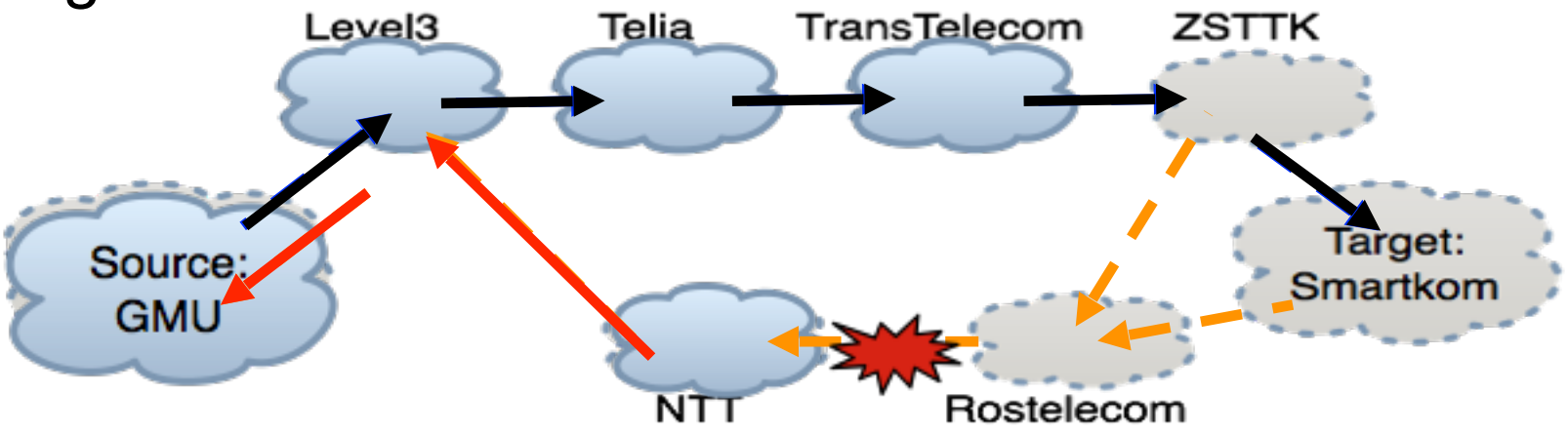
How Does **LIFEGUARD** Locate a Failure?

During outage:



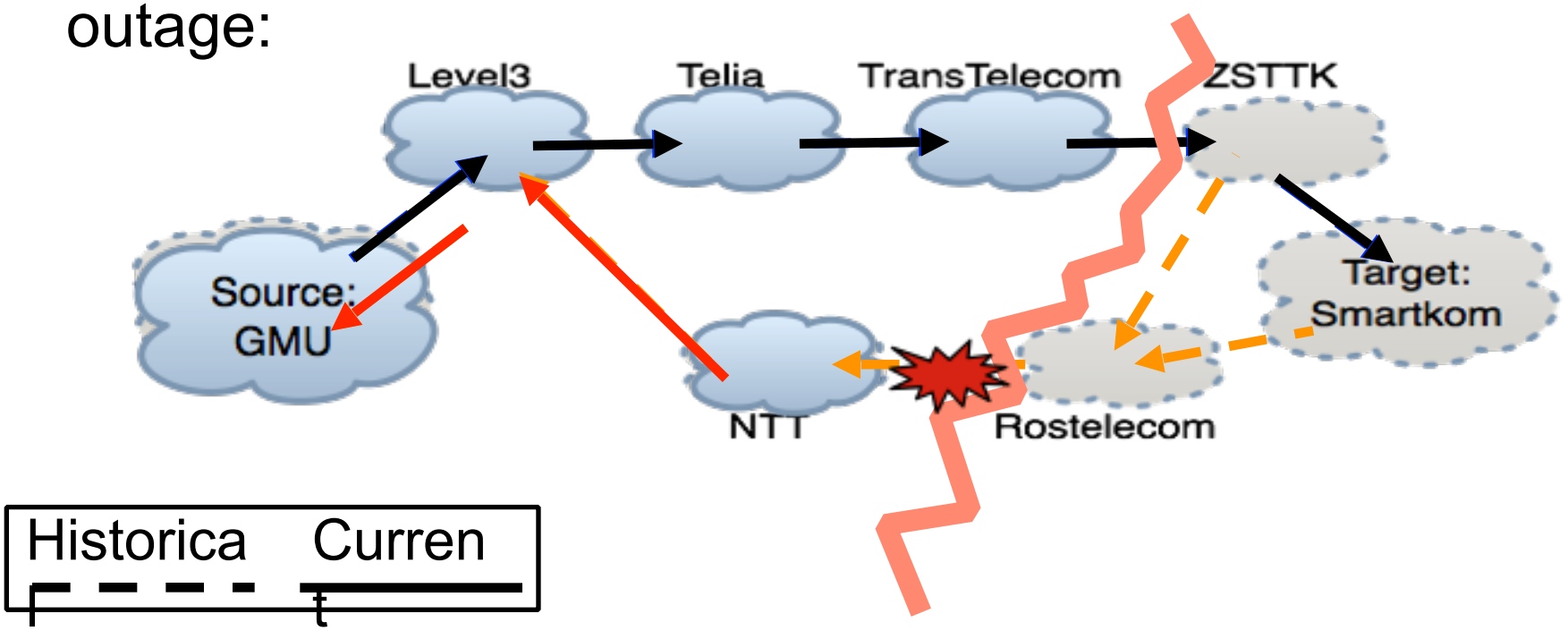
How Does **LIFEGUARD** Locate a Failure?

During outage:



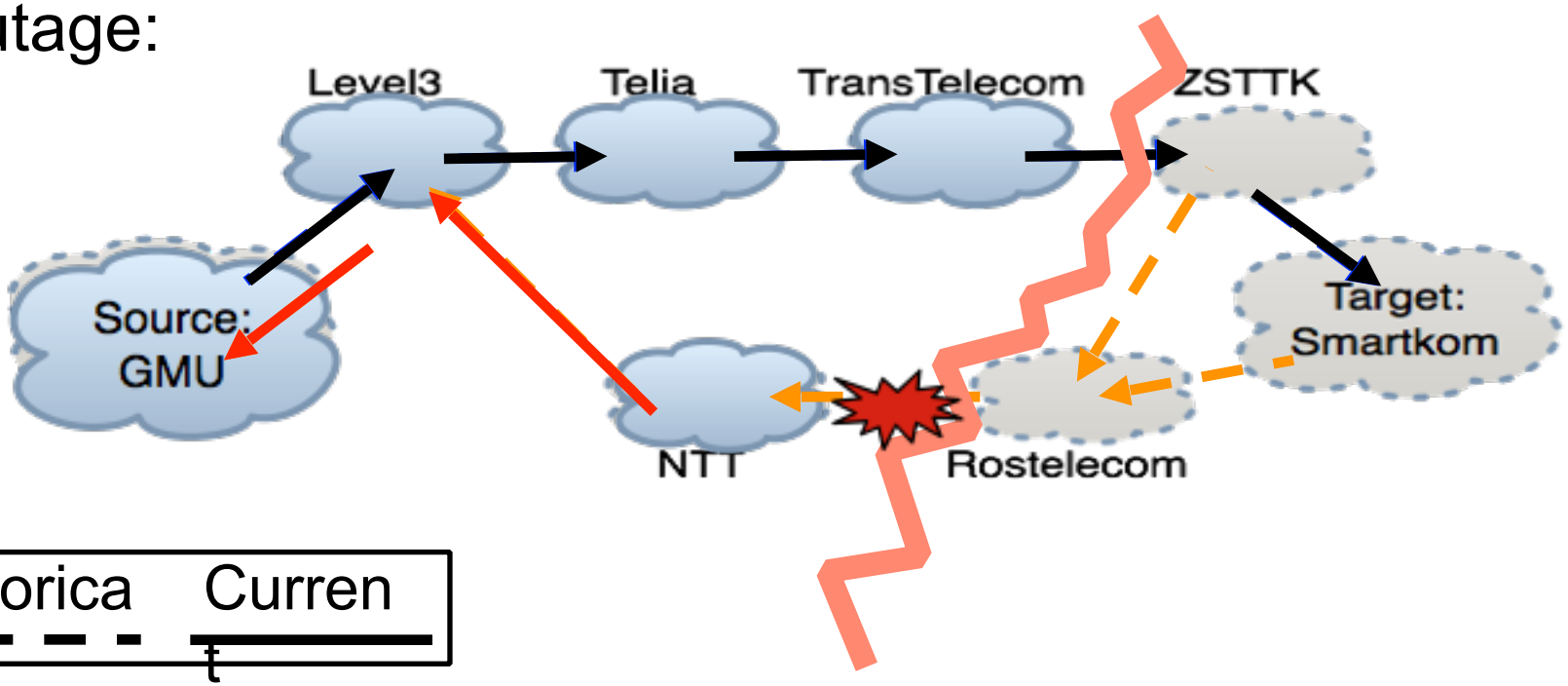
How Does **LIFEGUARD** Locate a Failure?

During outage:



How Does **LIFEGUARD** Locate a Failure?

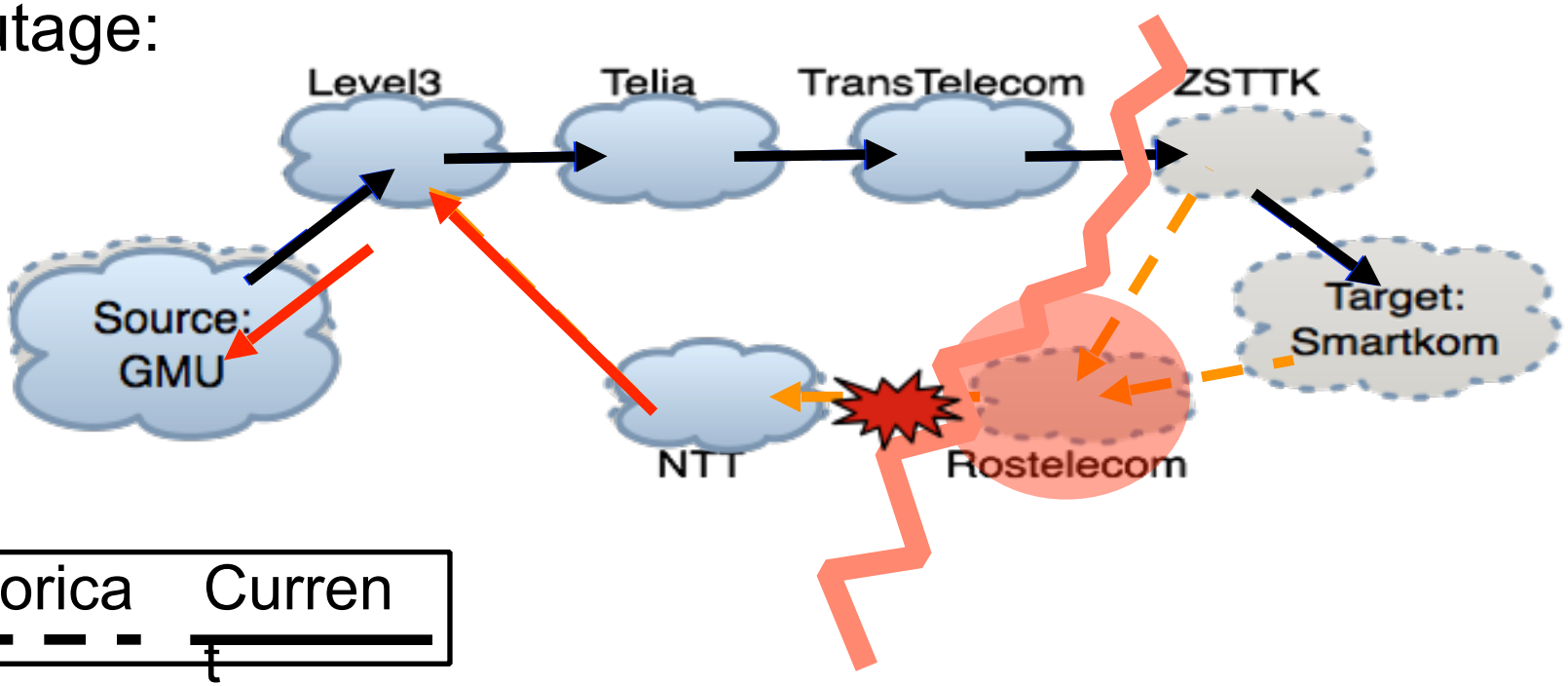
During outage:



- Forward path works

How Does **LIFEGUARD** Locate a Failure?

During outage:



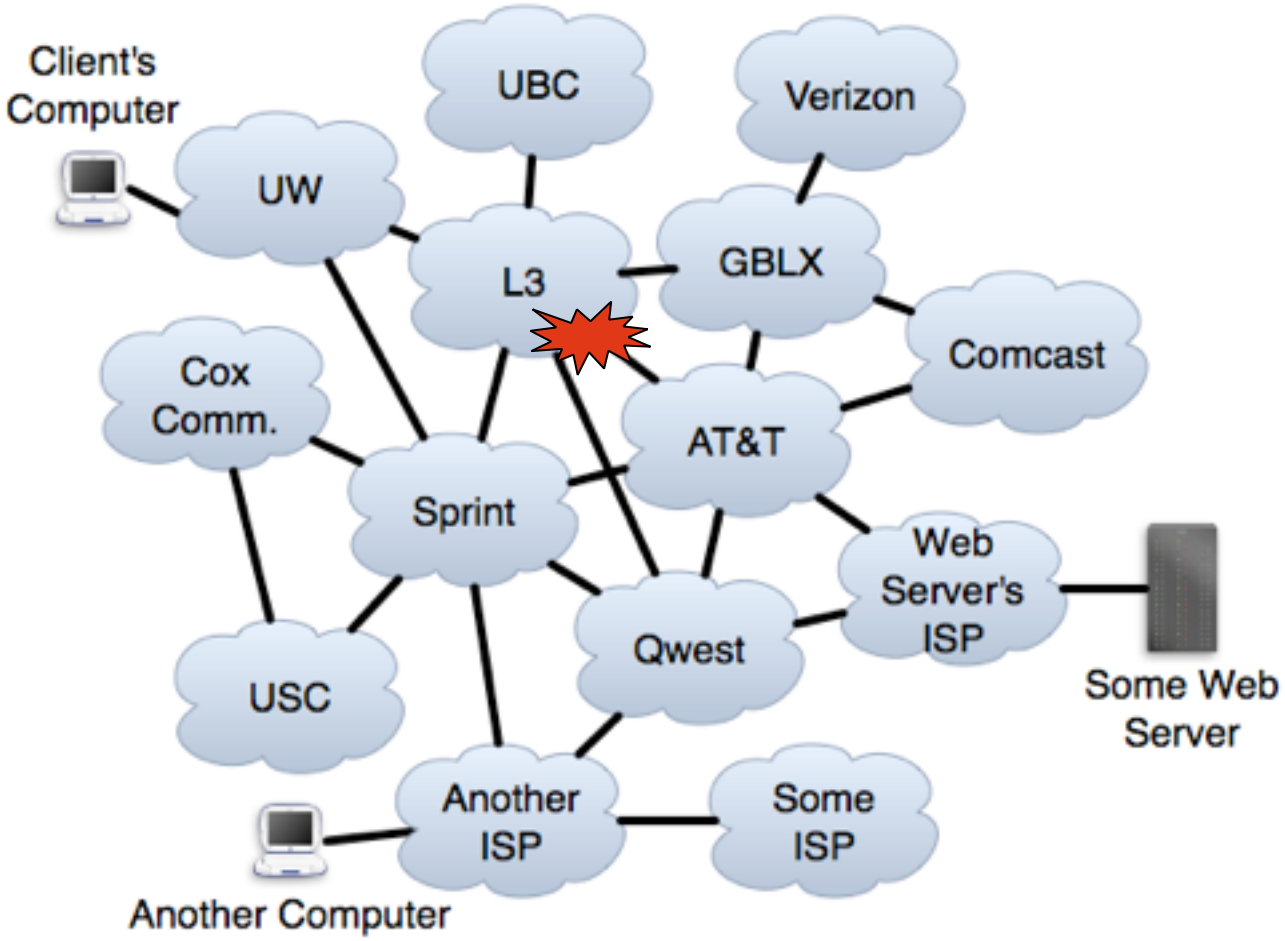
- Forward path works
- Rostelcom is not forwarding traffic towards GMU

How **LIFEGUARD** Locates Failures

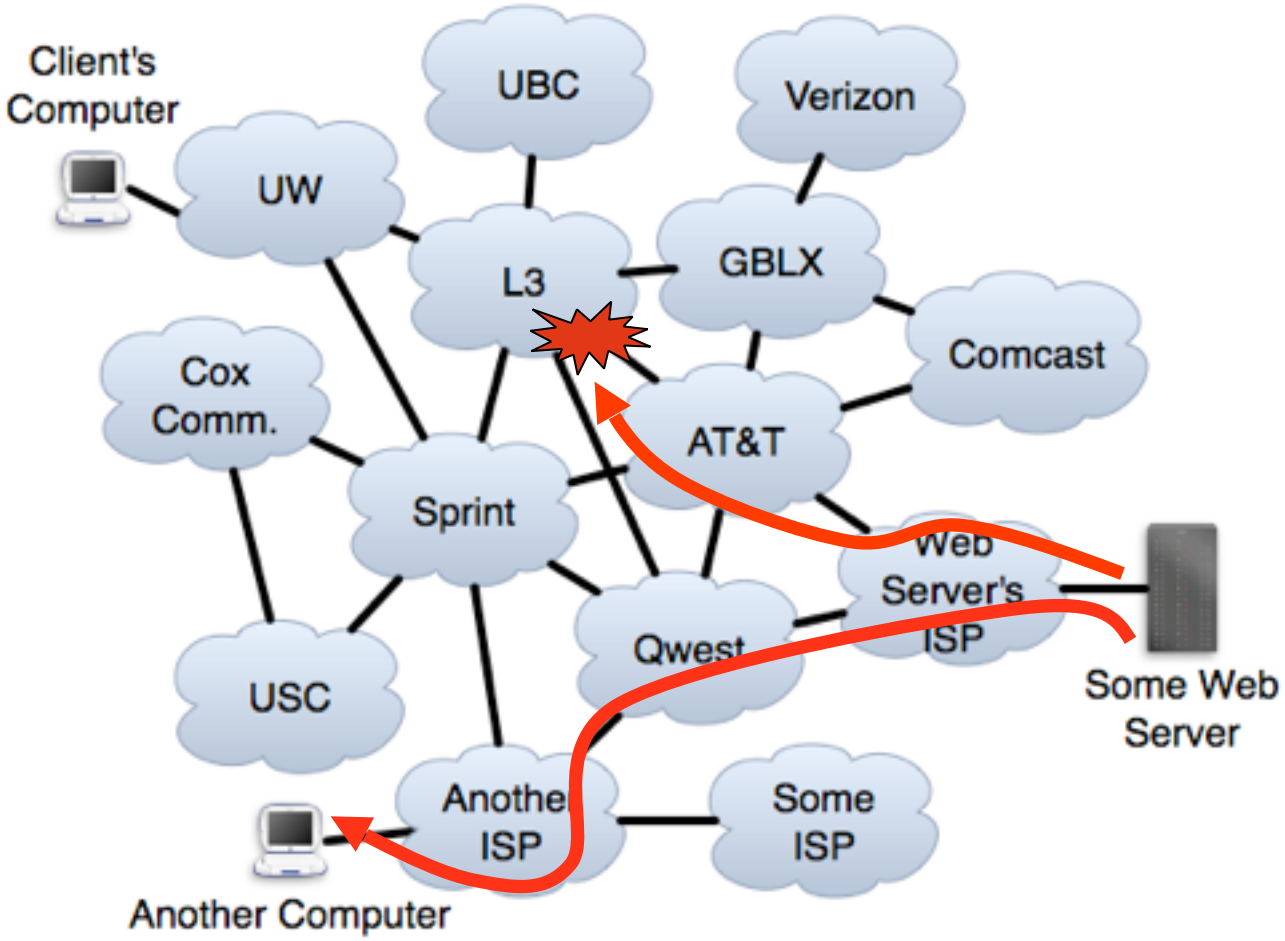
LIFEGUARD:

1. Maintains background historical atlas
2. Isolates direction of failure, measures working direction
3. Tests historical paths in failing direction in order to prune candidate failure locations
4. Locates failure as being at the horizon of reachability

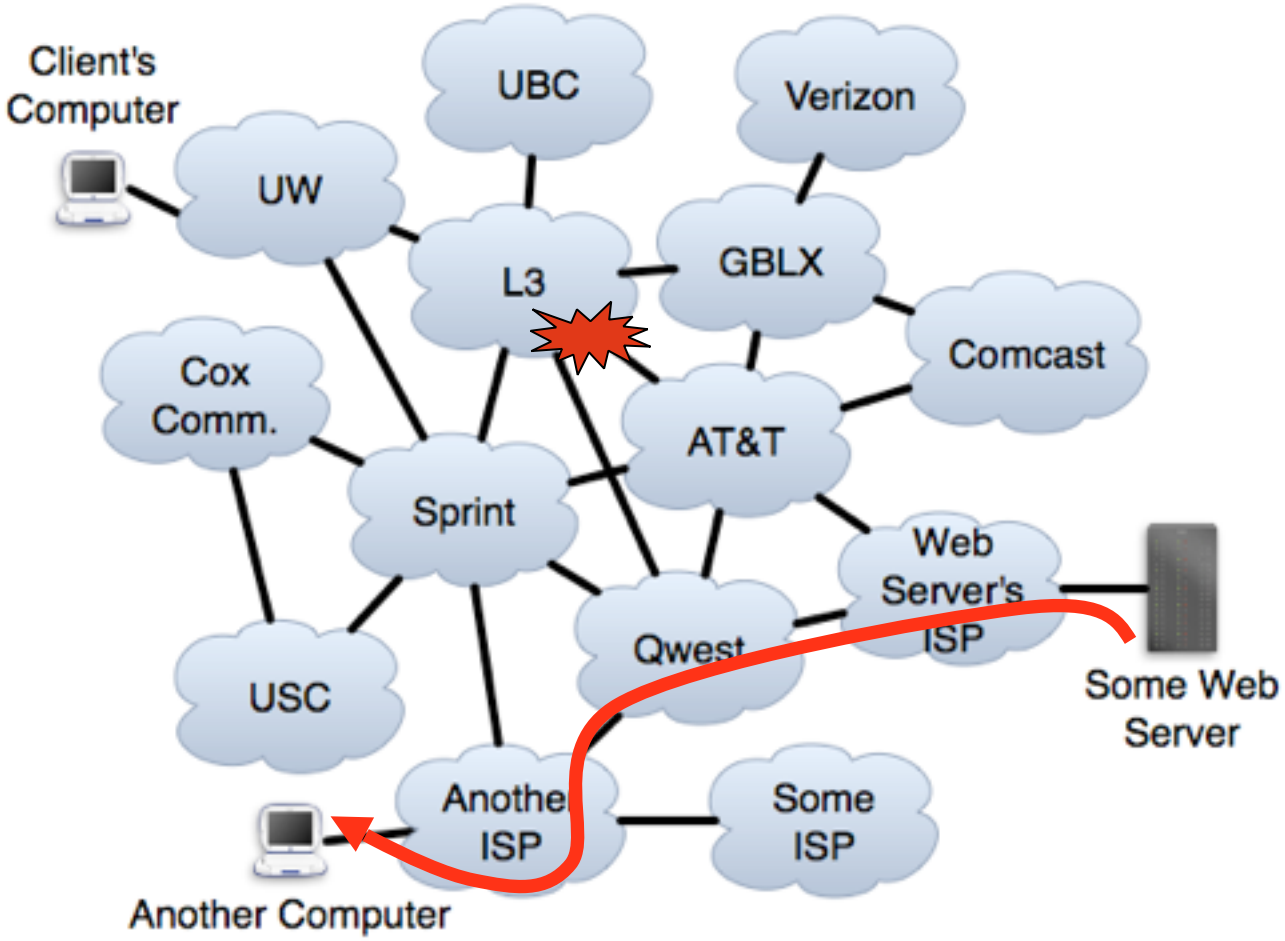
Self-Repair of Forward Paths



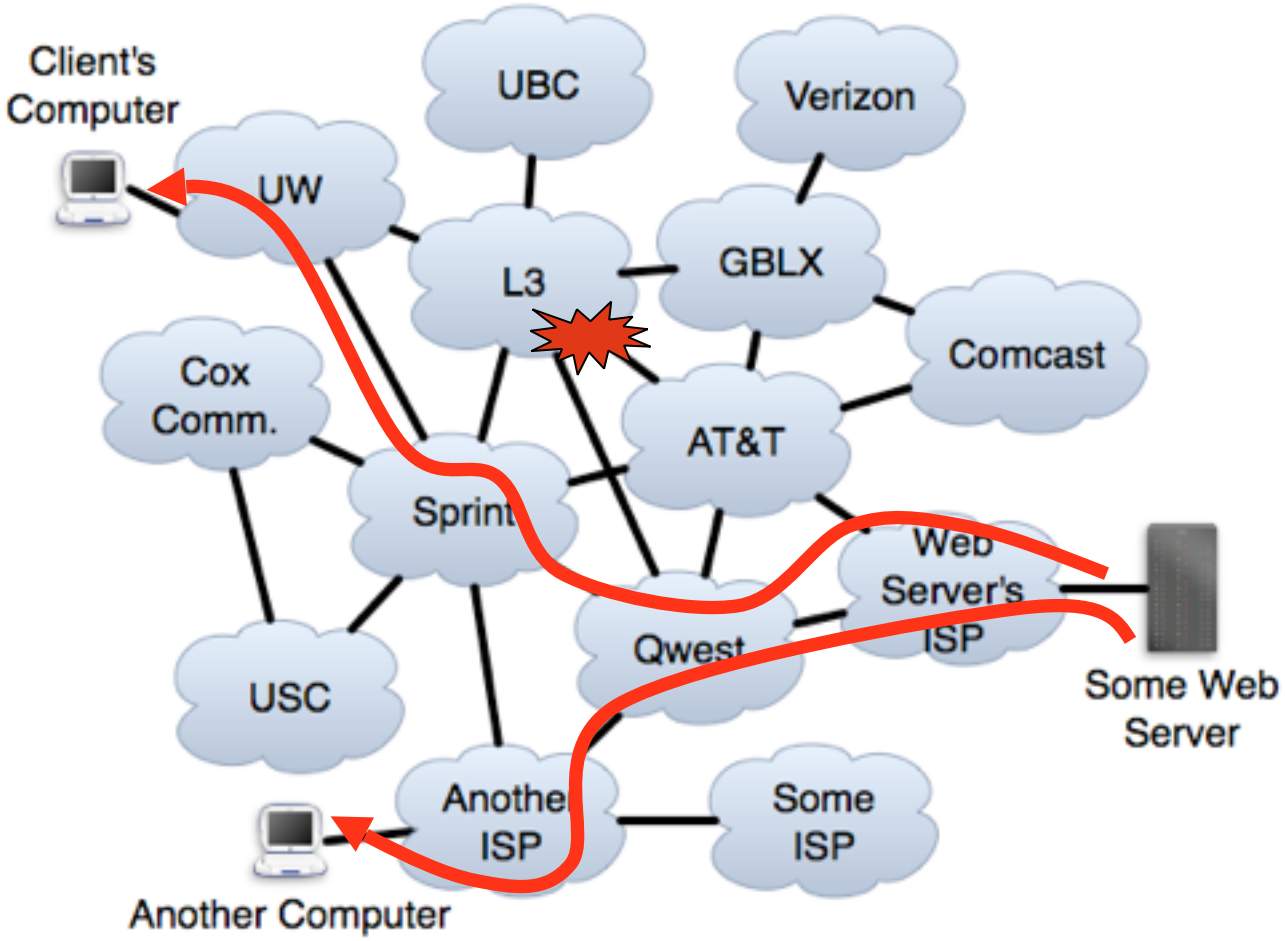
Self-Repair of Forward Paths



Self-Repair of Forward Paths



Self-Repair of Forward Paths



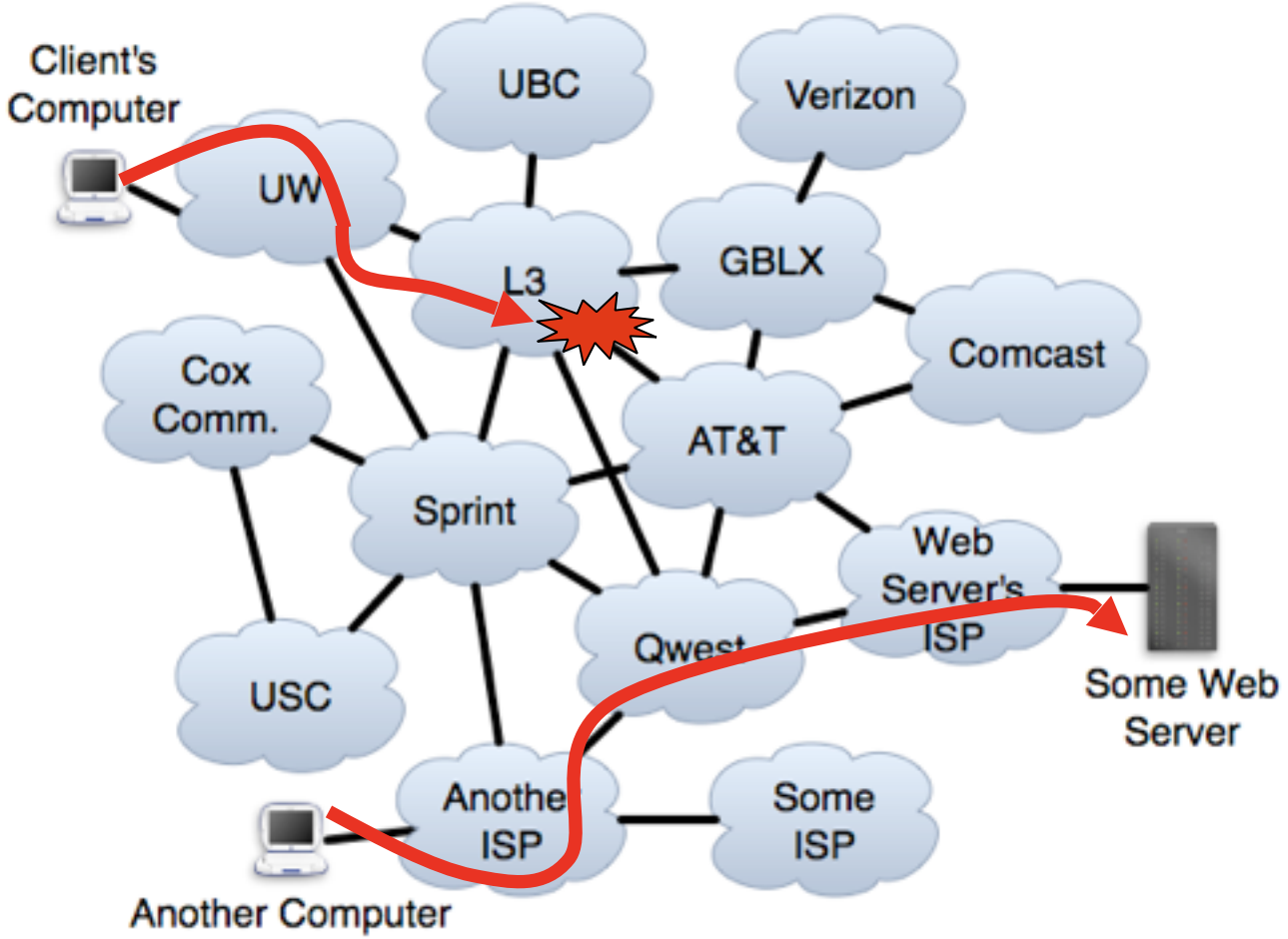
A Mechanism for Failure Avoidance

Forward path: Choose route that avoids ISP or ISP-ISP link

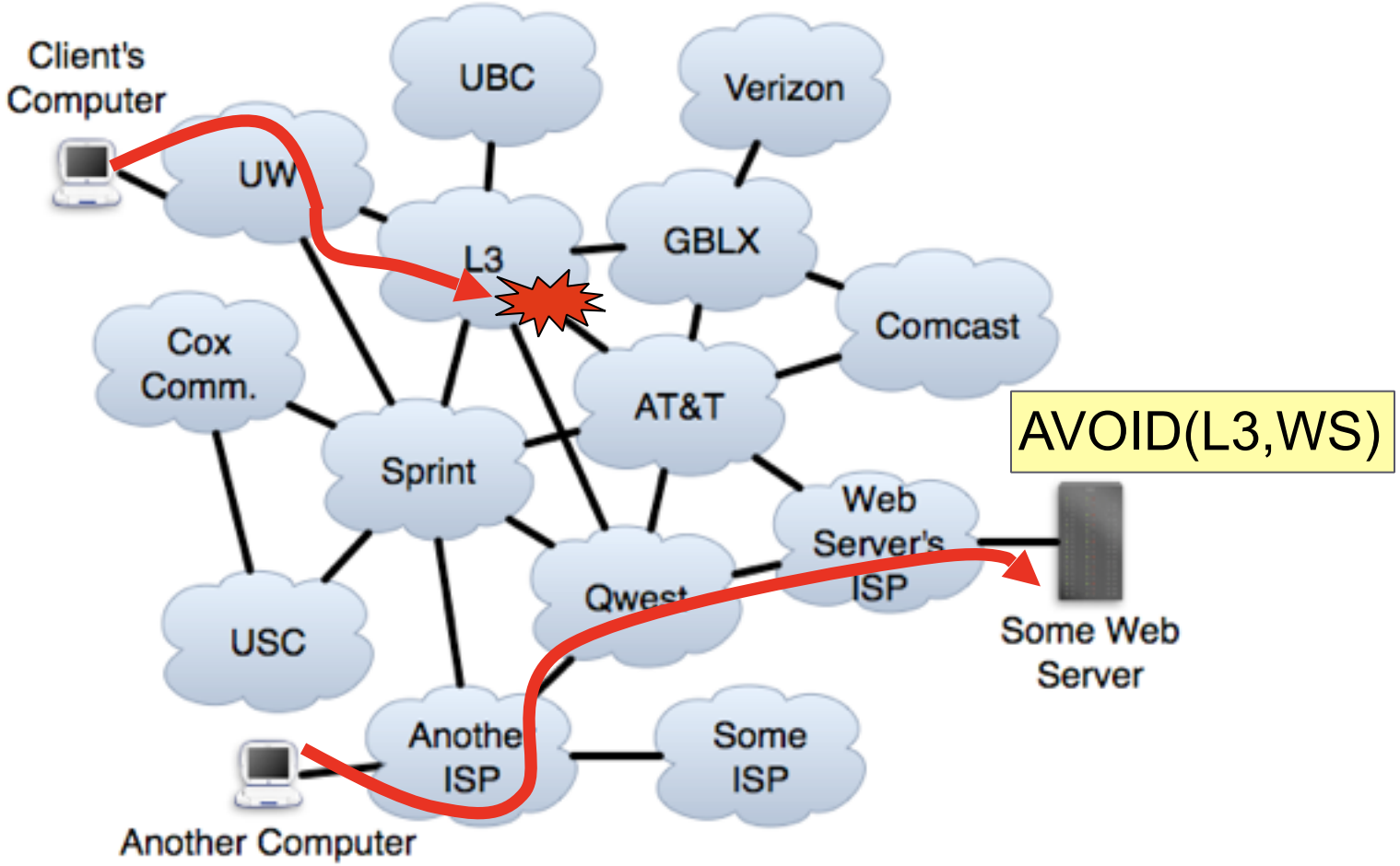
Reverse path: Want others to choose paths to prefix P that avoid ISP or ISP-ISP link X

- Want a BGP announcement AVOID(X,P):
  - Any ISP with a route to P that avoids X uses such a route
  - Any ISP not using X need only pass on the announcement

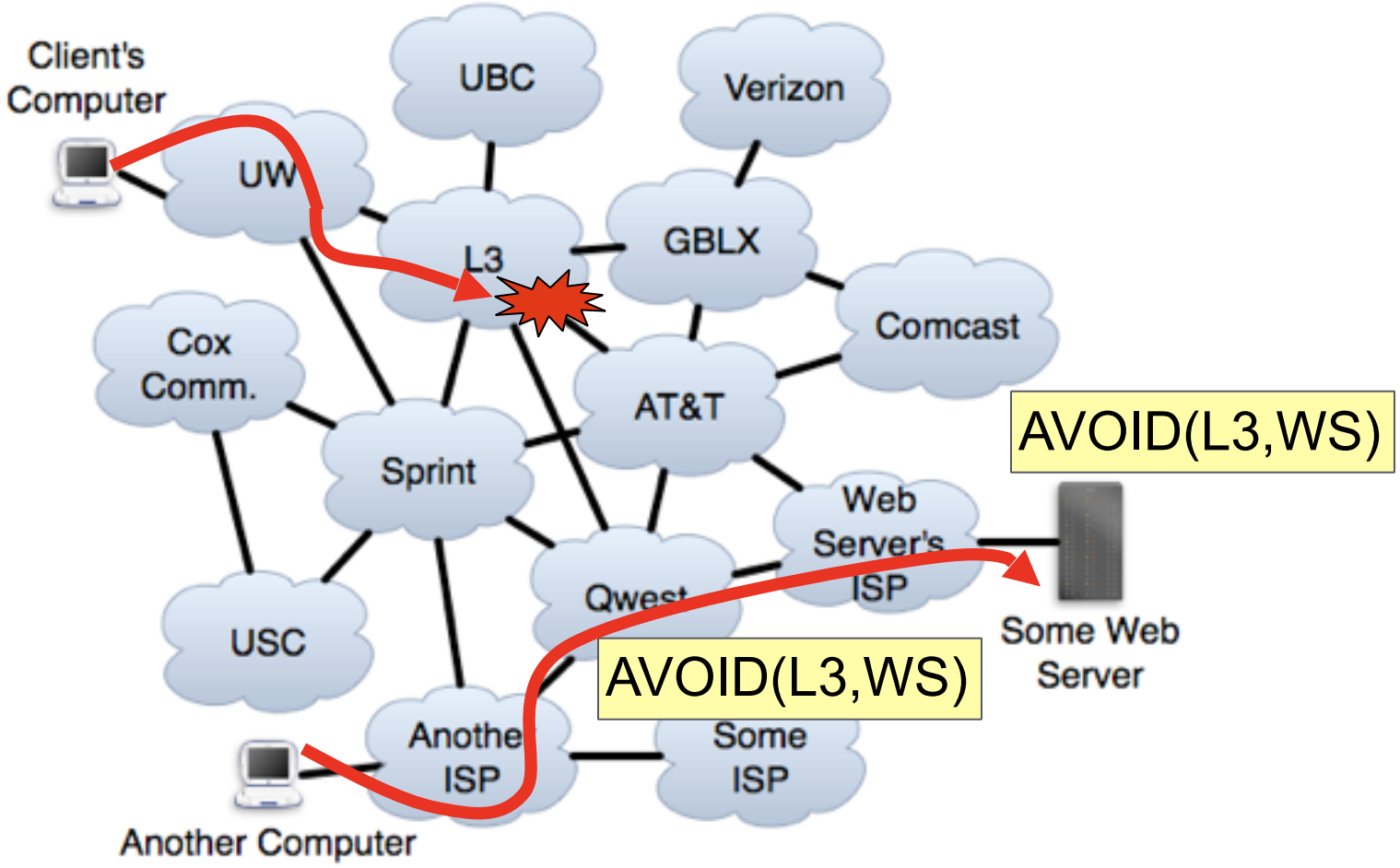
Ideal Self-Repair of Reverse Paths



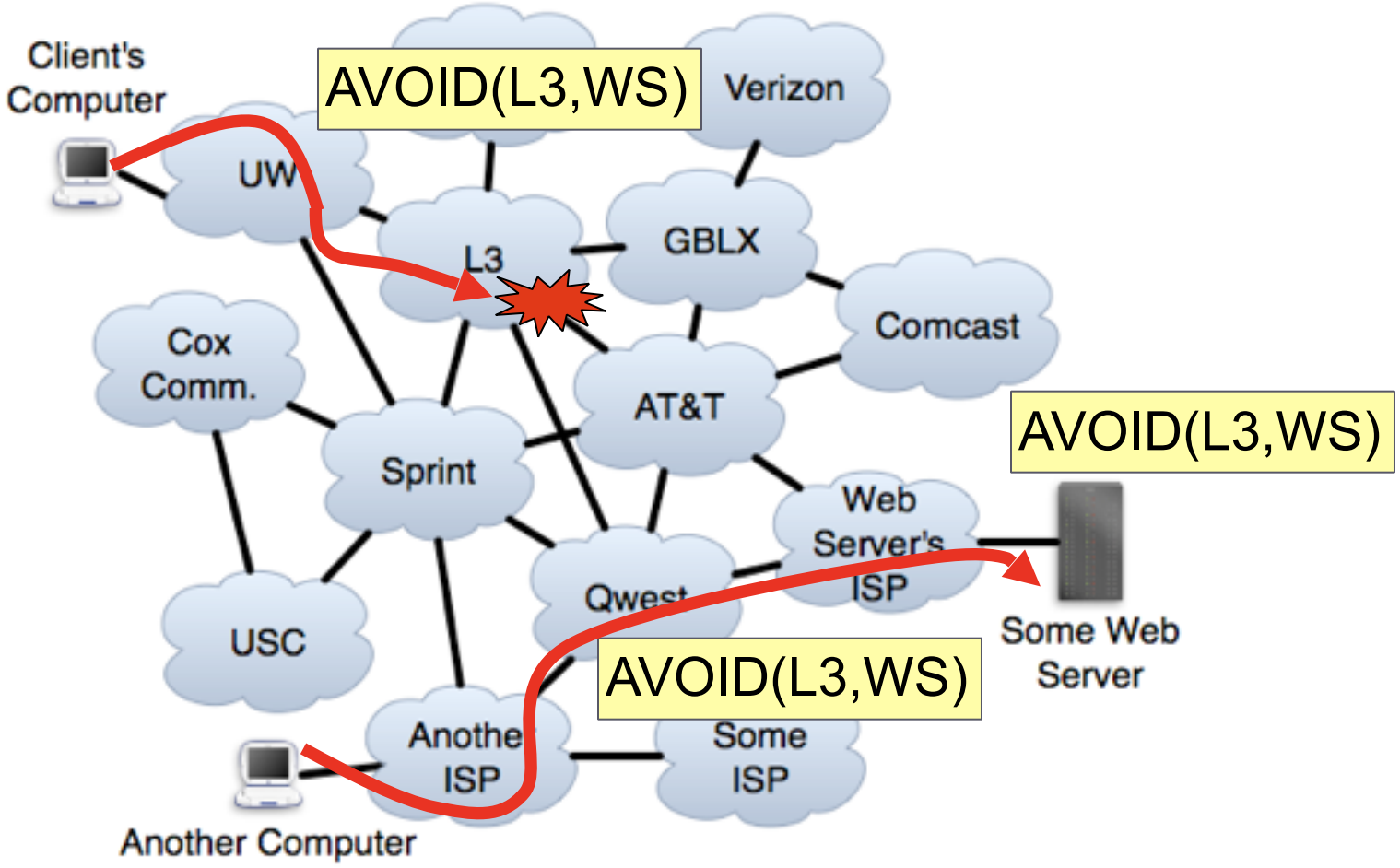
Ideal Self-Repair of Reverse Paths



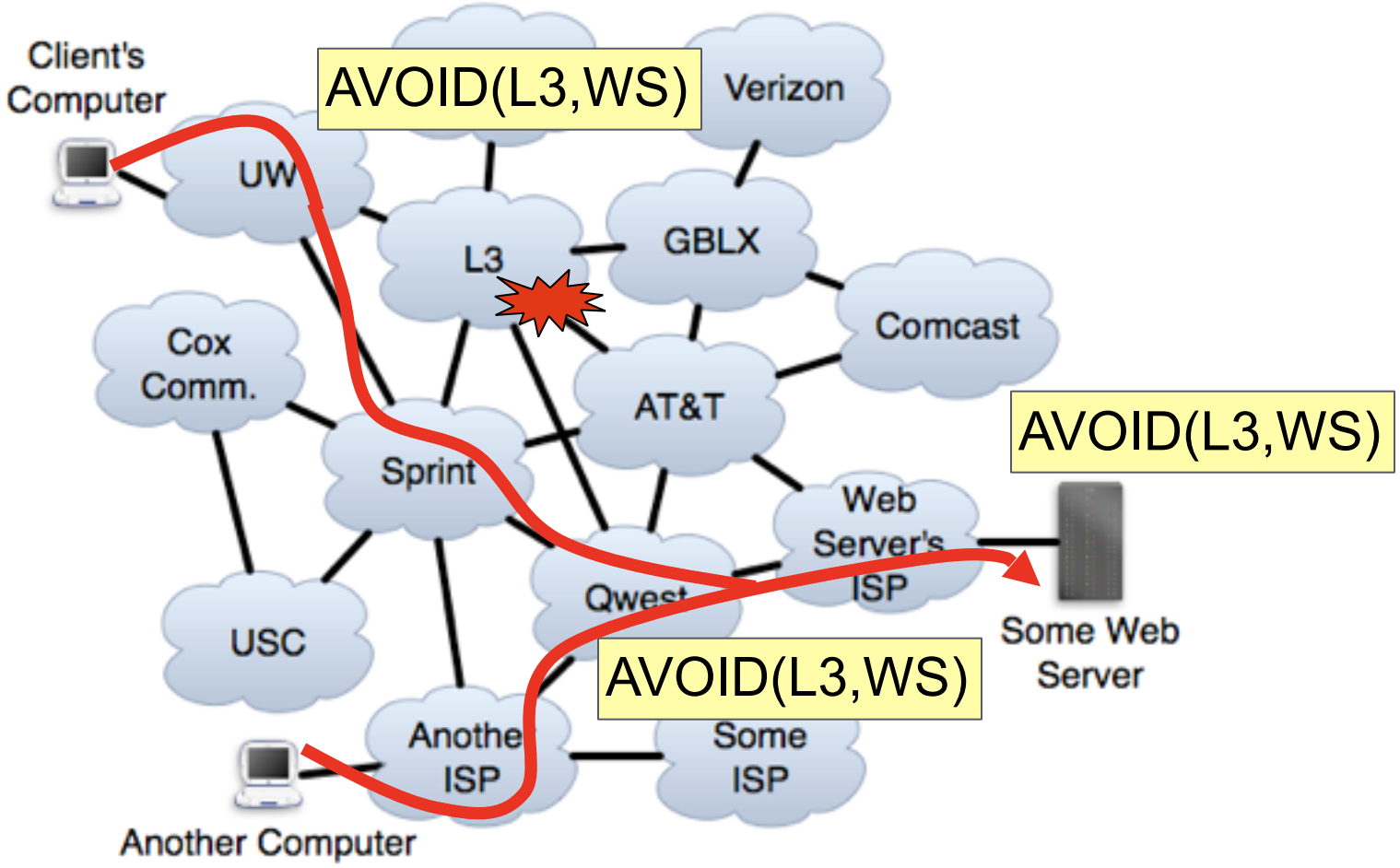
Ideal Self-Repair of Reverse Paths



Ideal Self-Repair of Reverse Paths



Ideal Self-Repair of Reverse Paths



Do paths exist that AVOID problem?

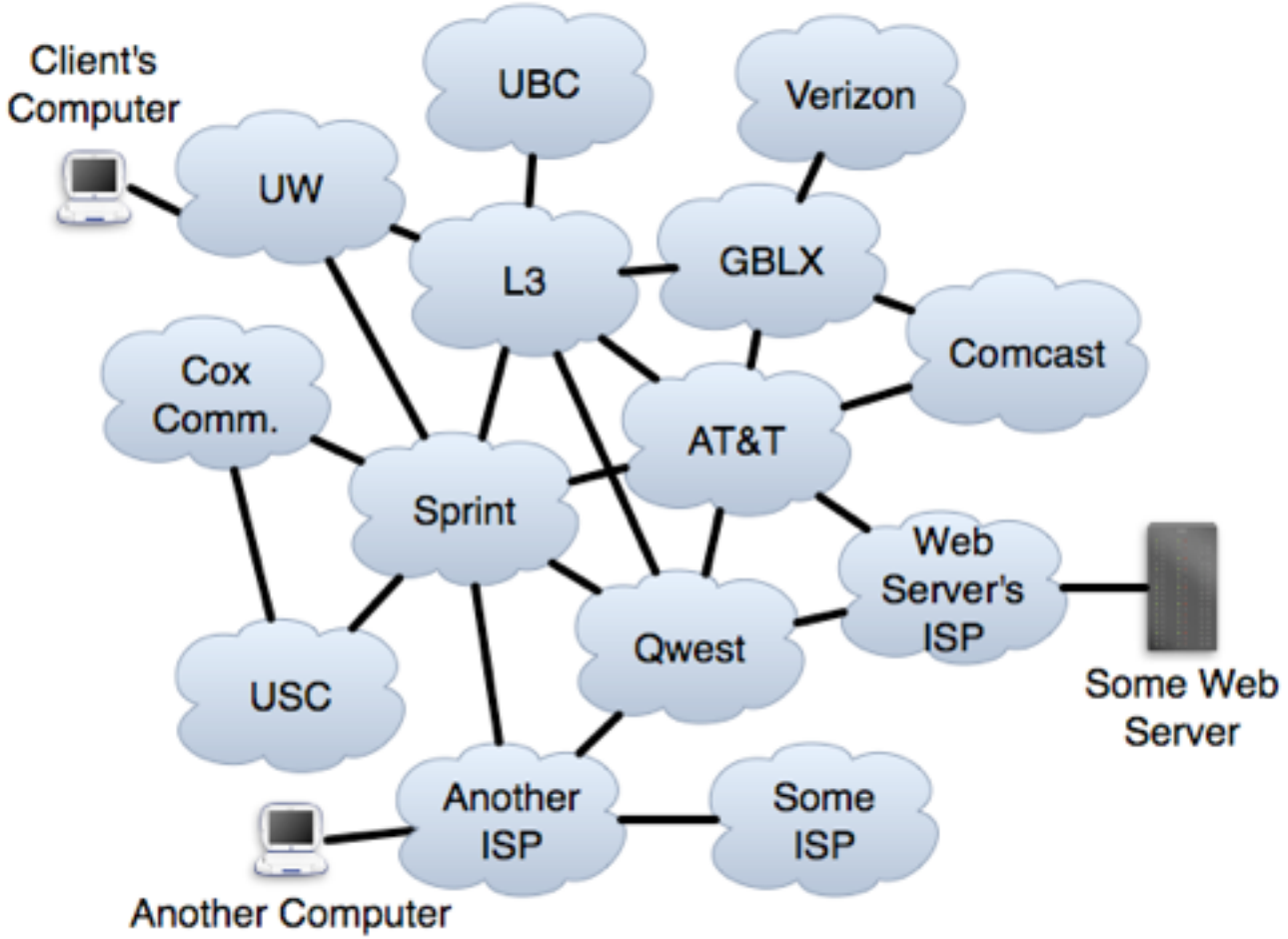
LIFEGUARD repairs outages by instructing others to avoid particular routes.

Q: Do alternative routes exist?

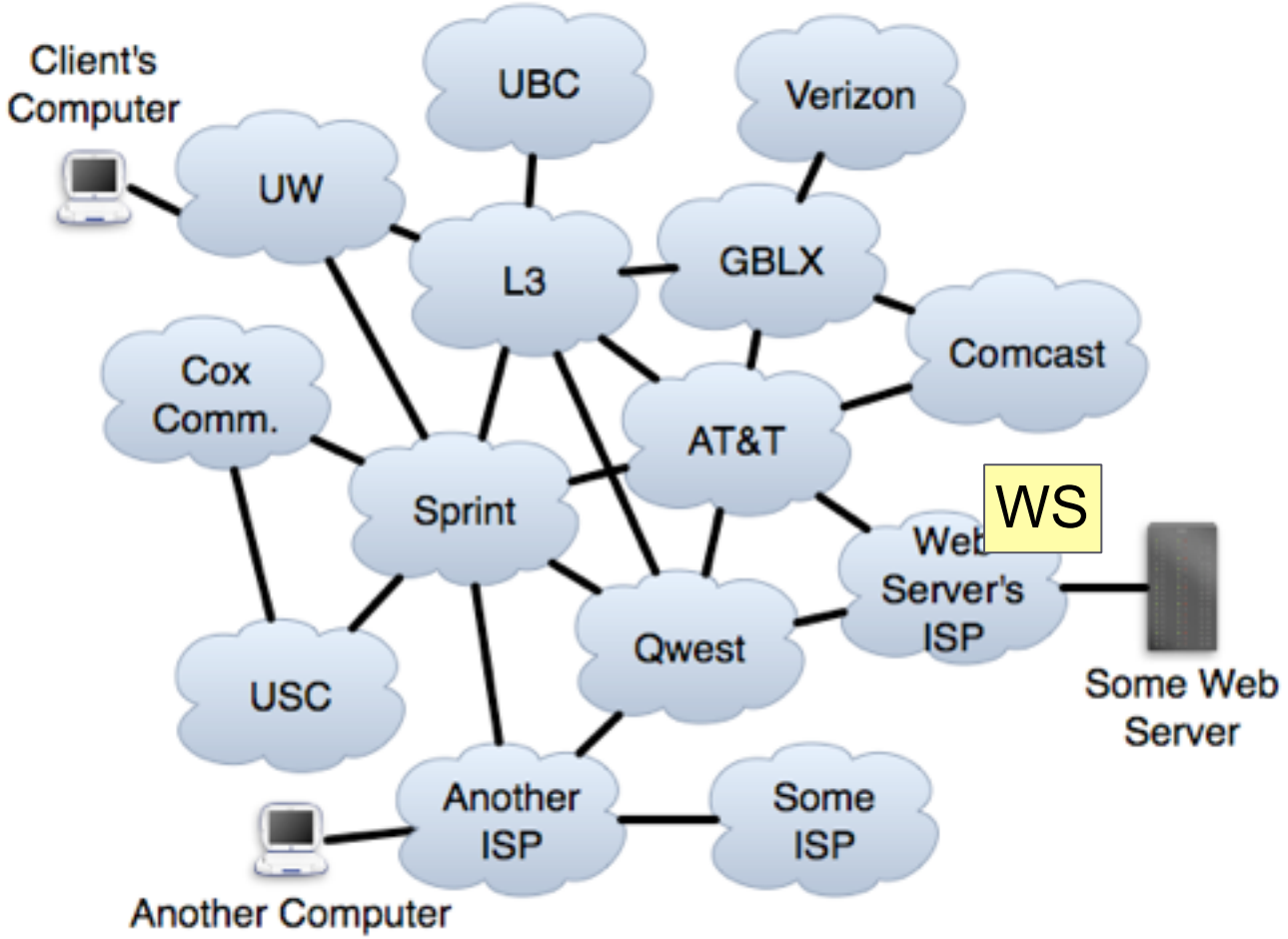
A: Alternate policy-compliant paths exist in 90% of simulated AVOID(X,P) *announcements.*

- Simulated 10 million AVOIDs on actual measured routes.

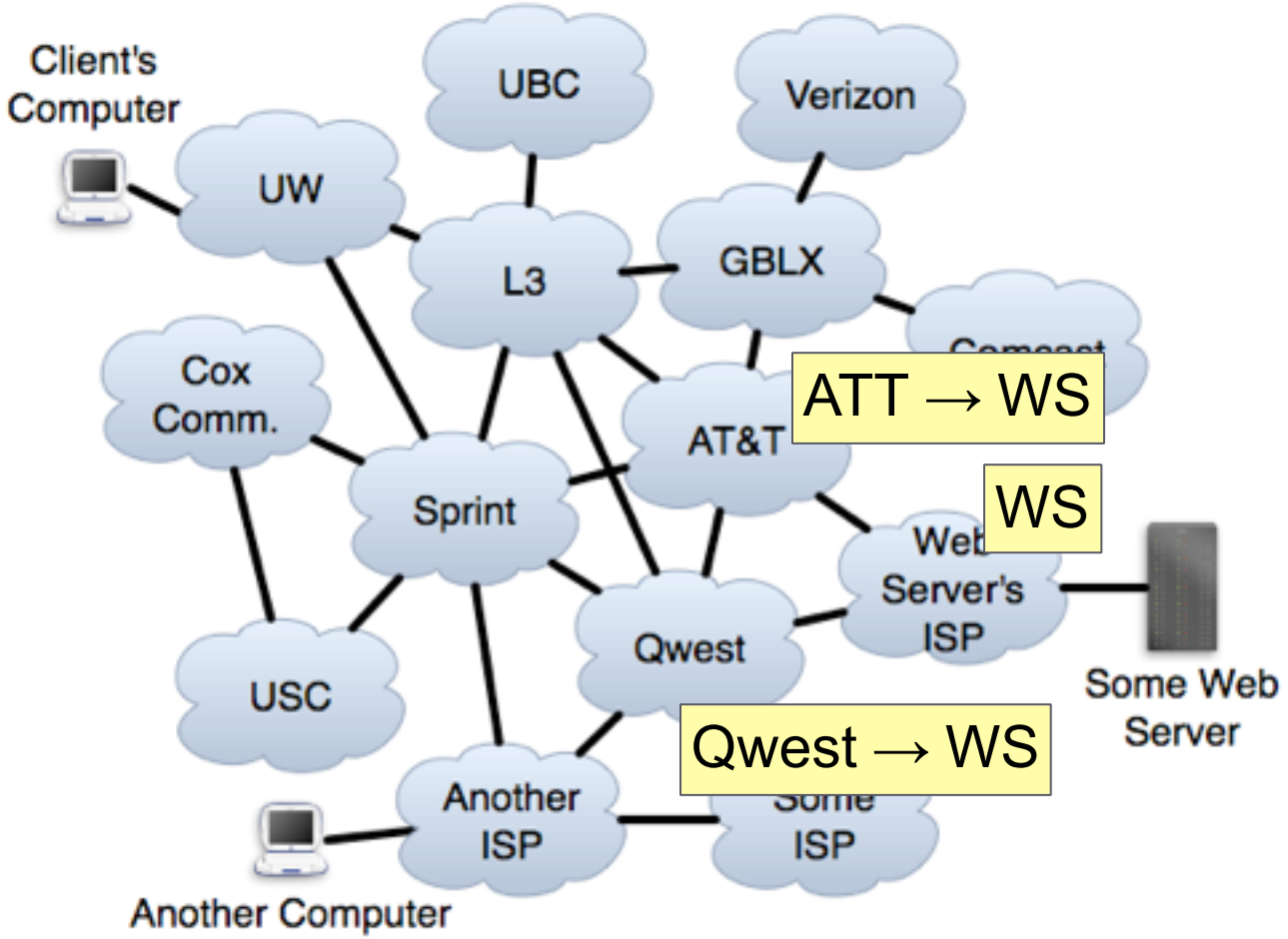
Practical Self-Repair of Reverse Paths



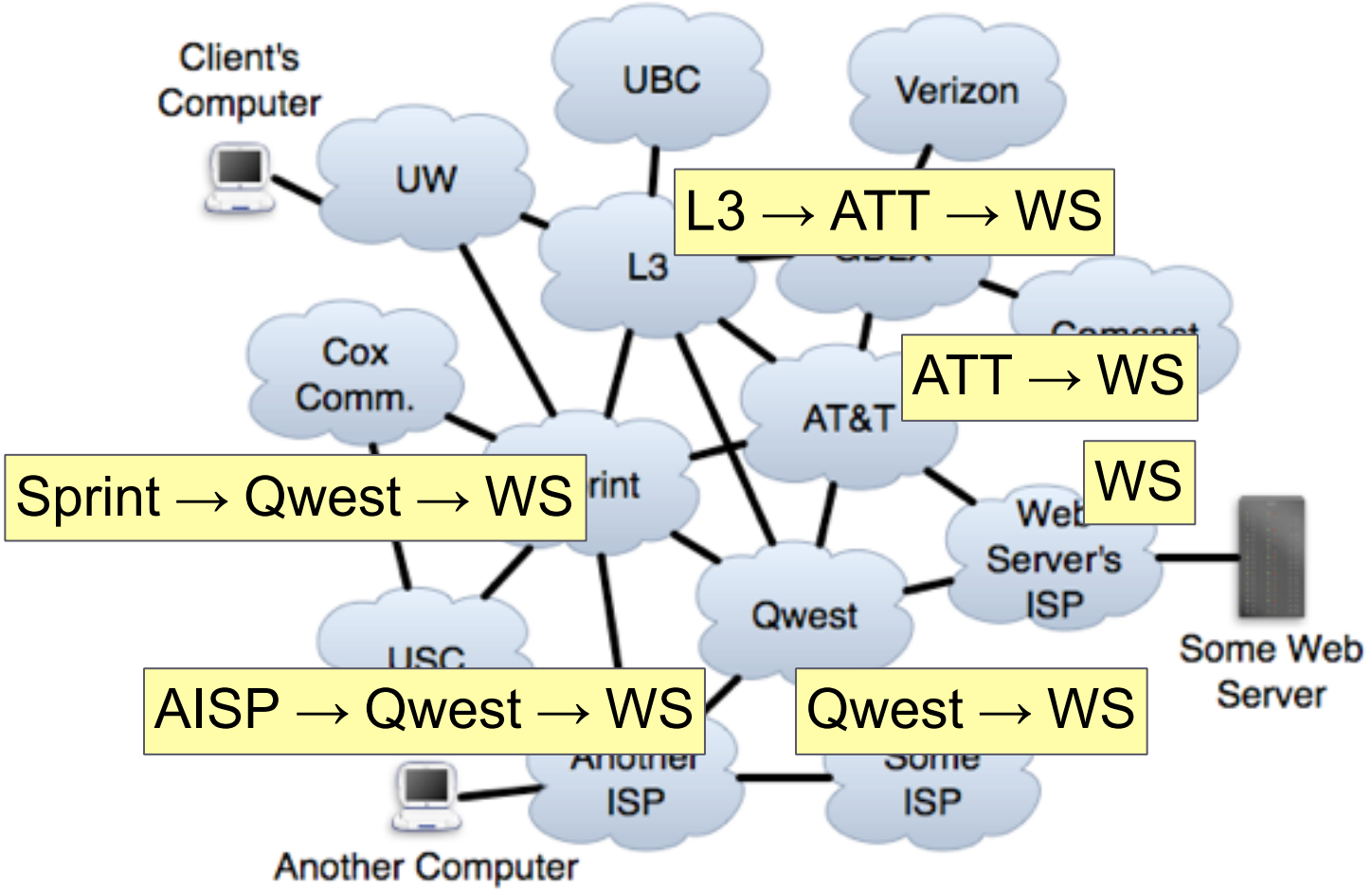
Practical Self-Repair of Reverse Paths



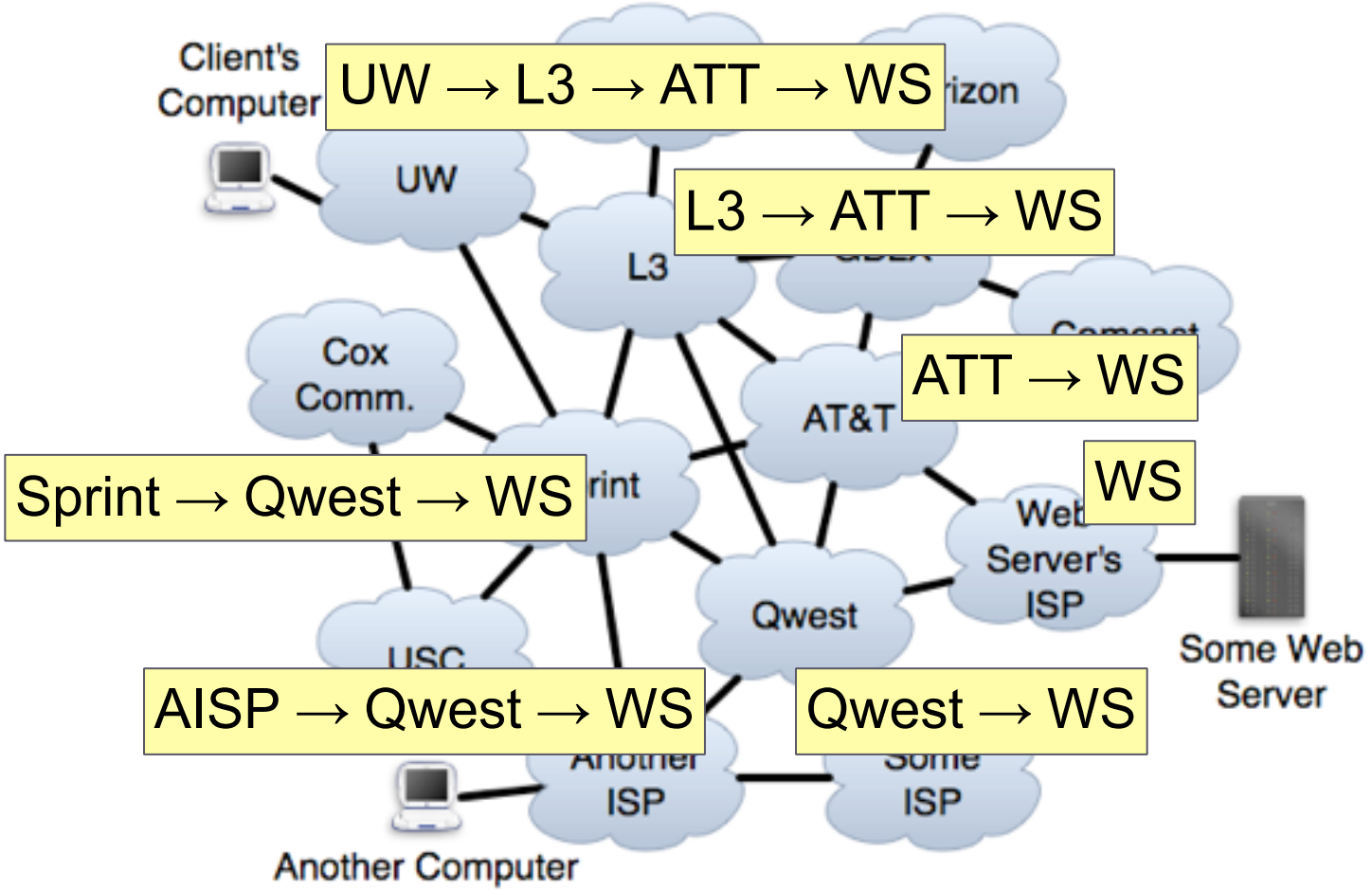
Practical Self-Repair of Reverse Paths



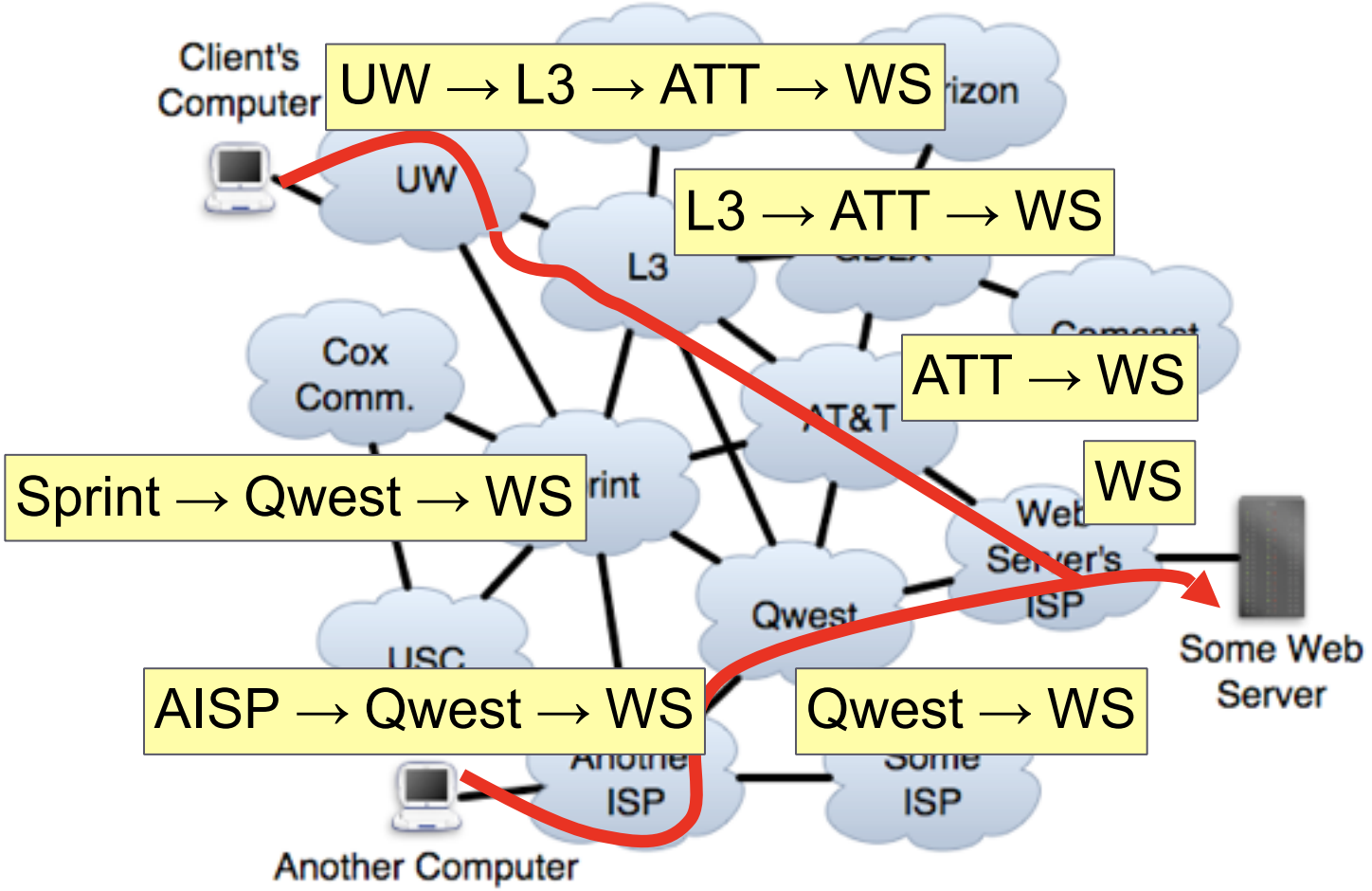
Practical Self-Repair of Reverse Paths



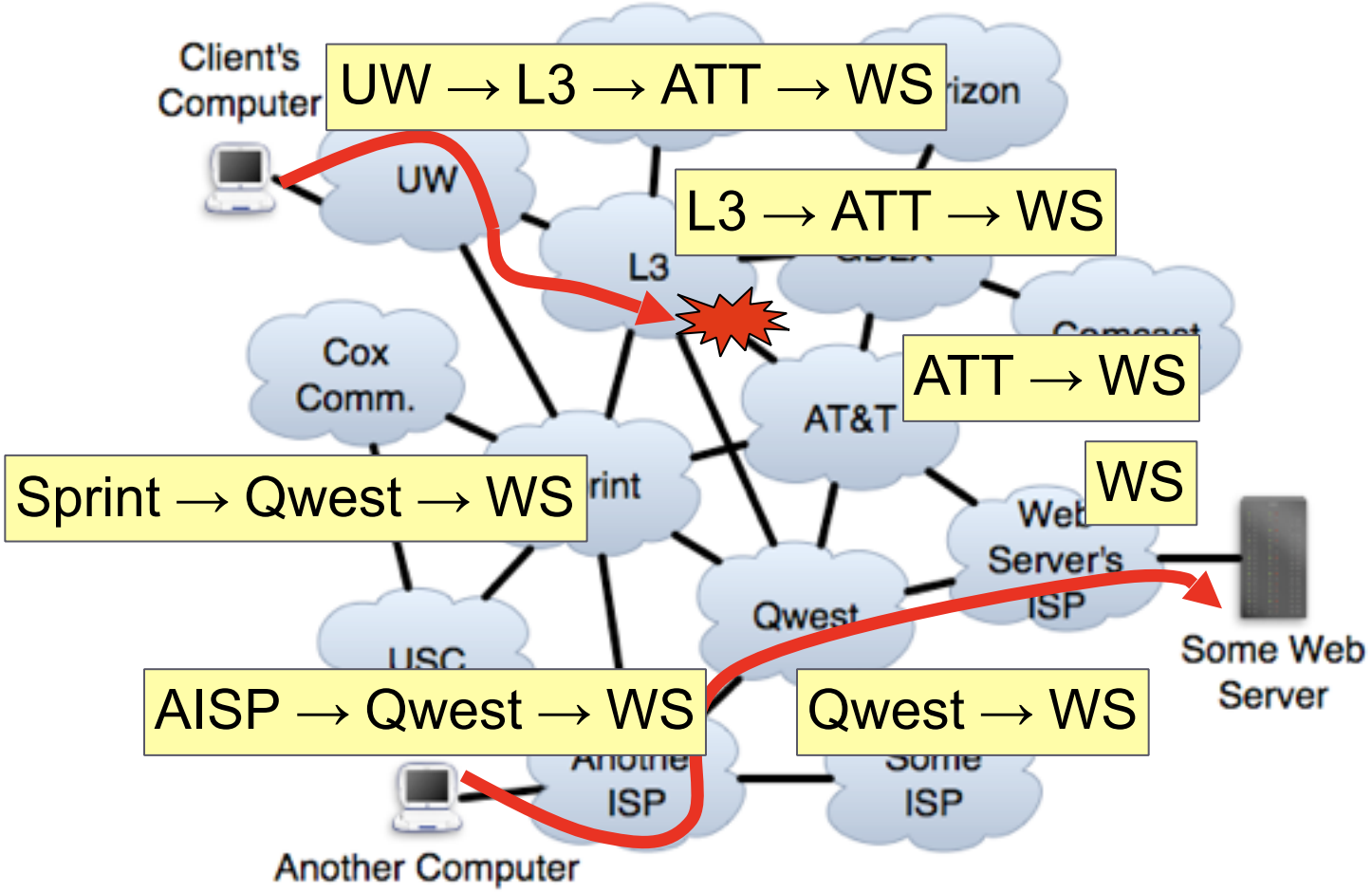
Practical Self-Repair of Reverse Paths



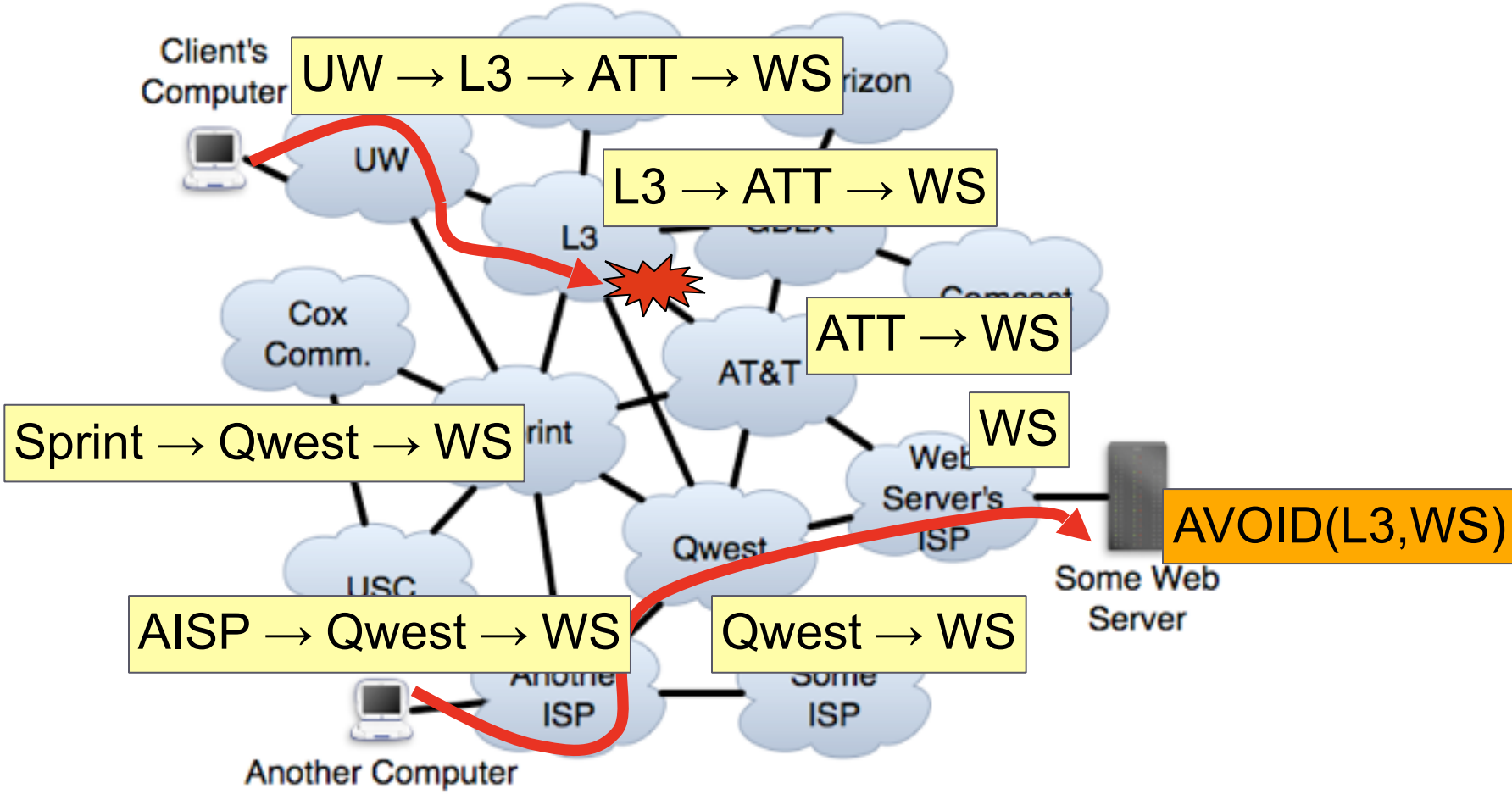
Practical Self-Repair of Reverse Paths



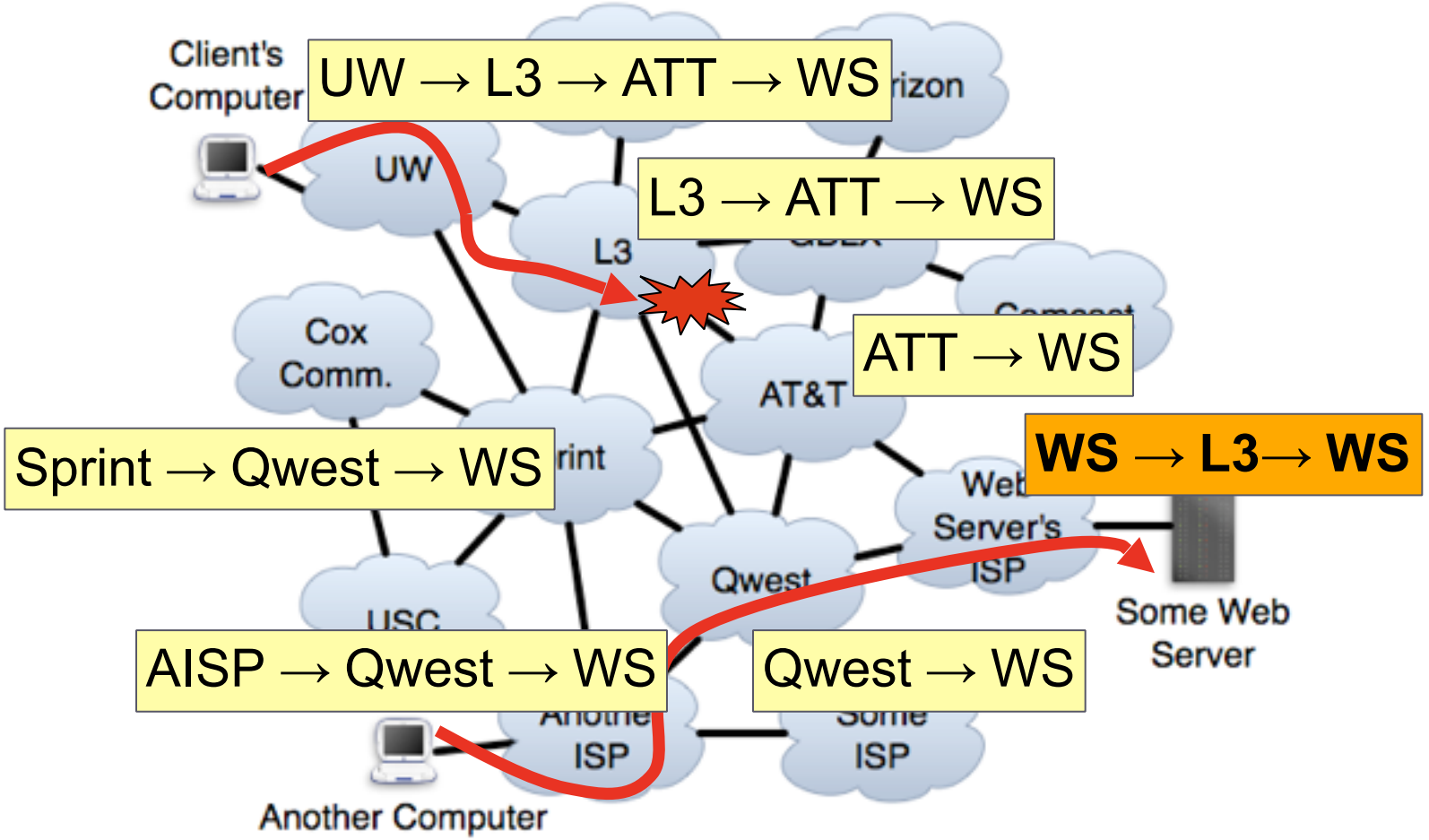
Practical Self-Repair of Reverse Paths



Practical Self-Repair of Reverse Paths

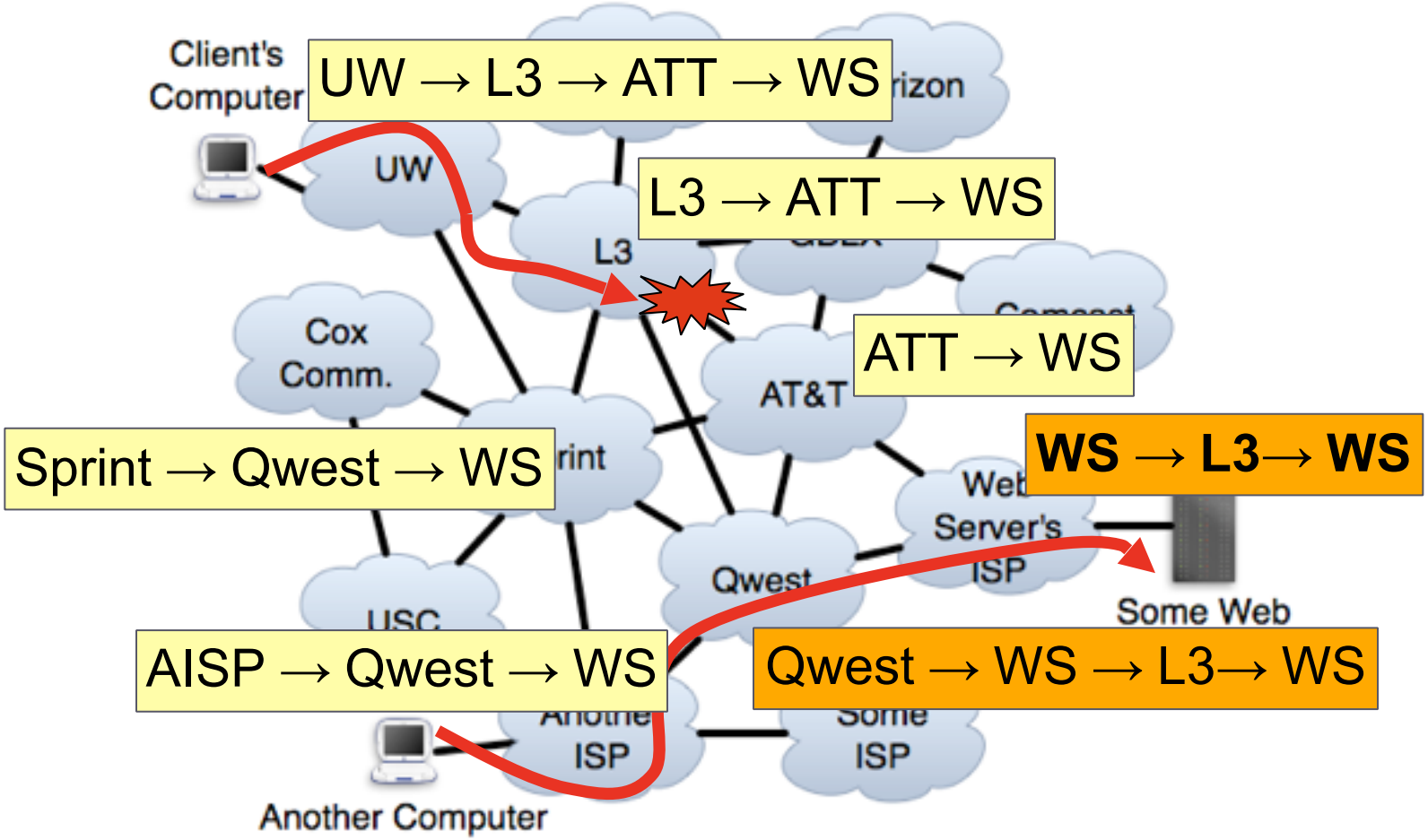


Practical Self-Repair of Reverse Paths



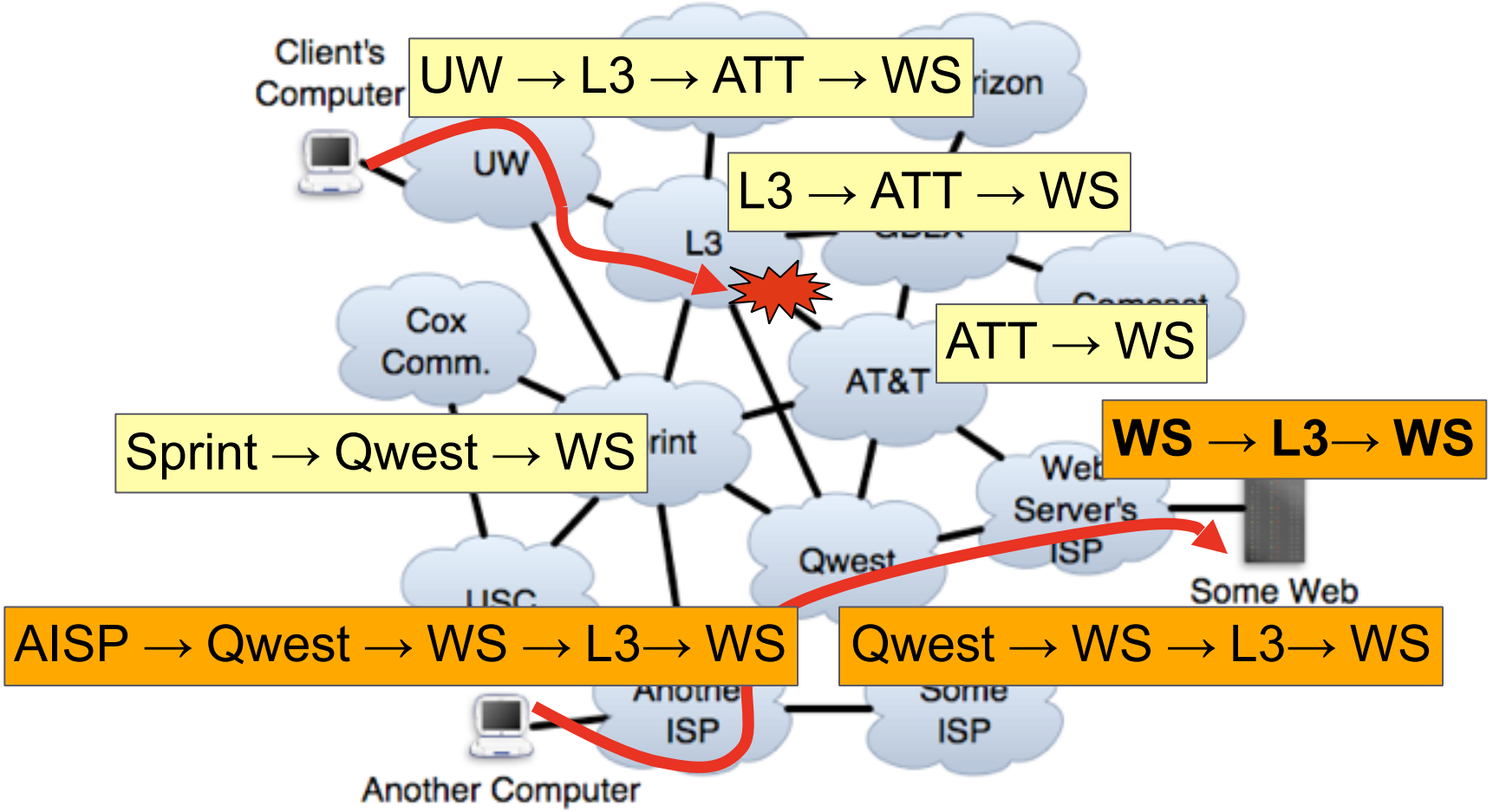
BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



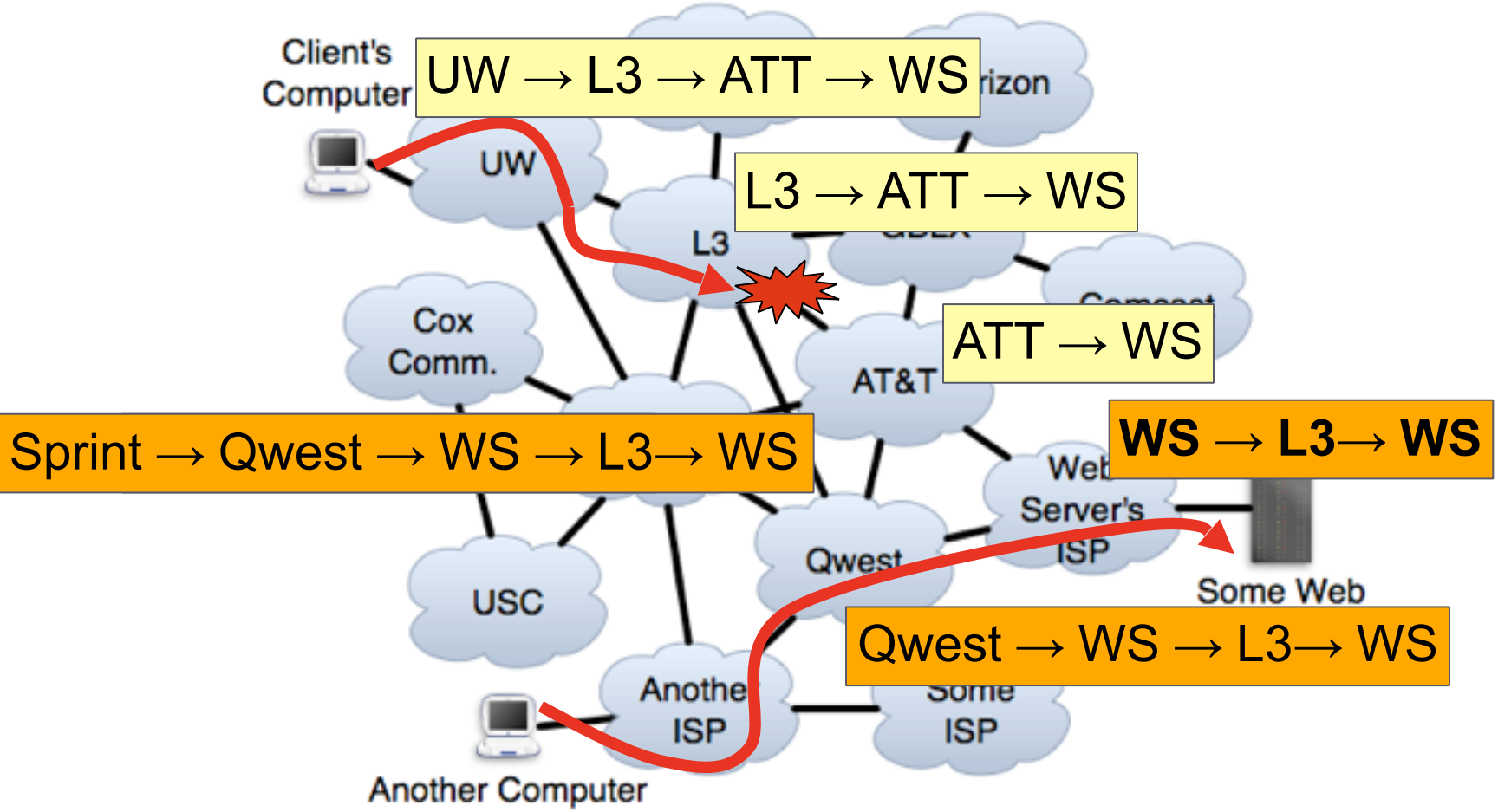
BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



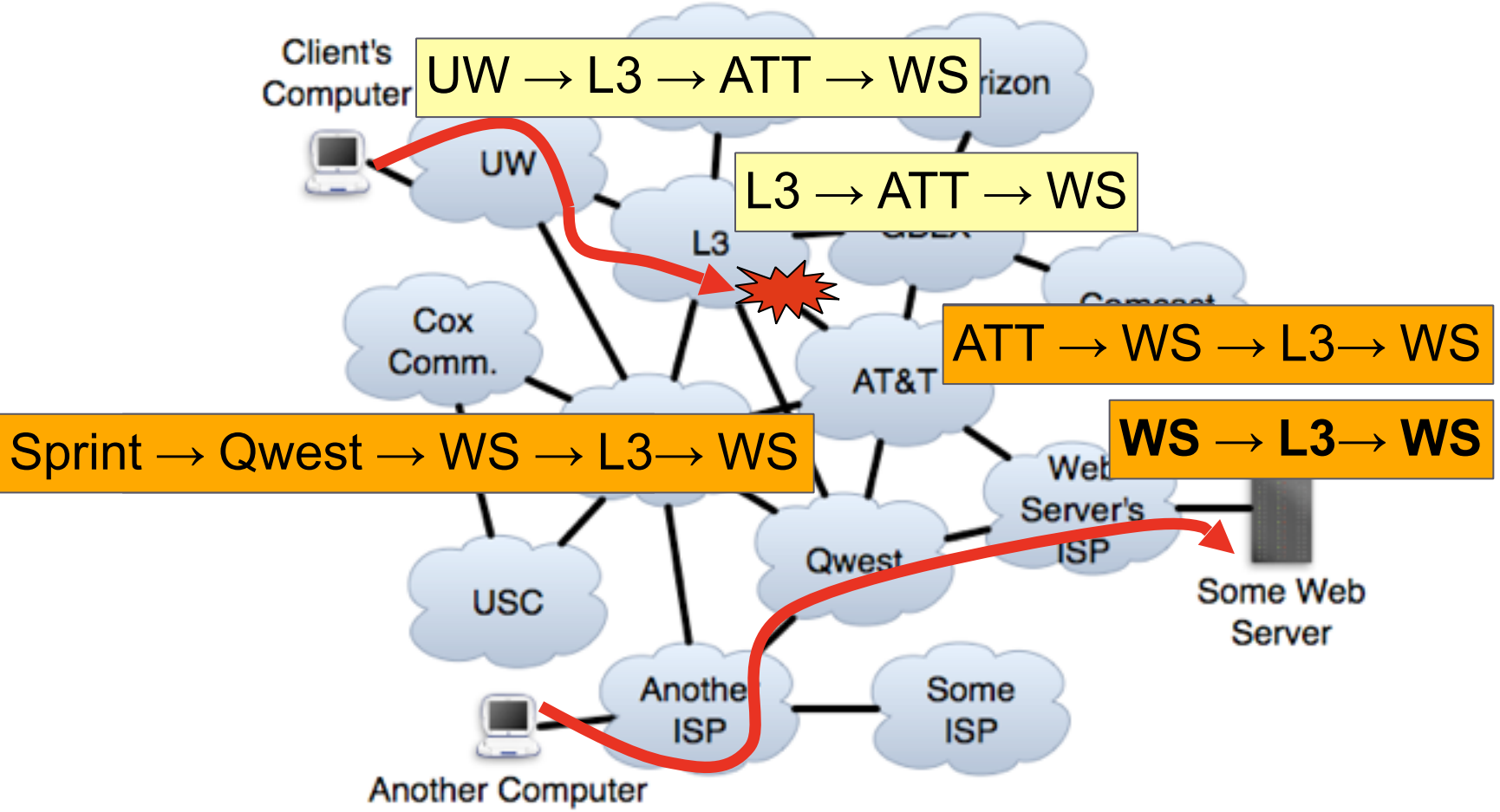
BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



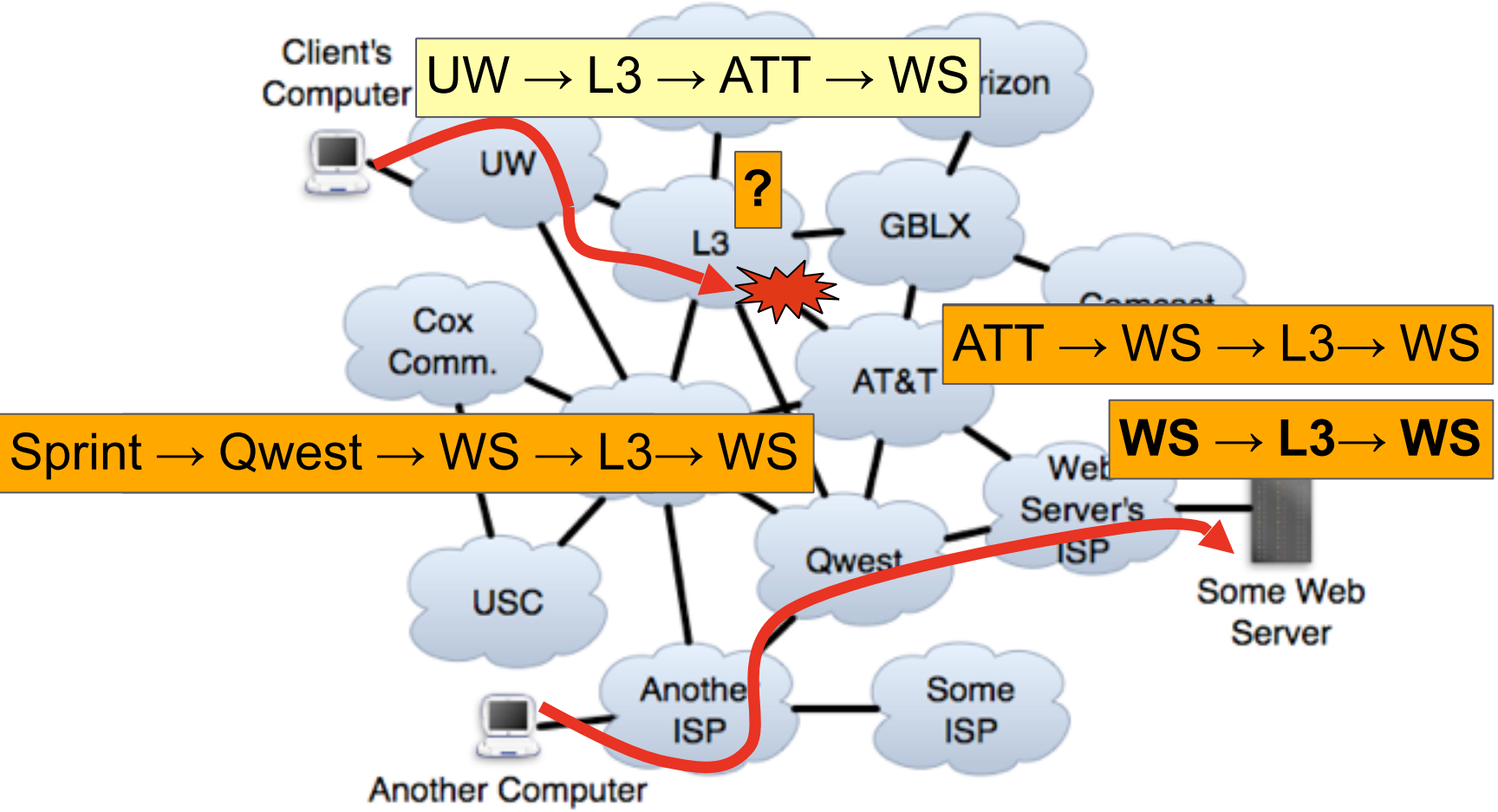
BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



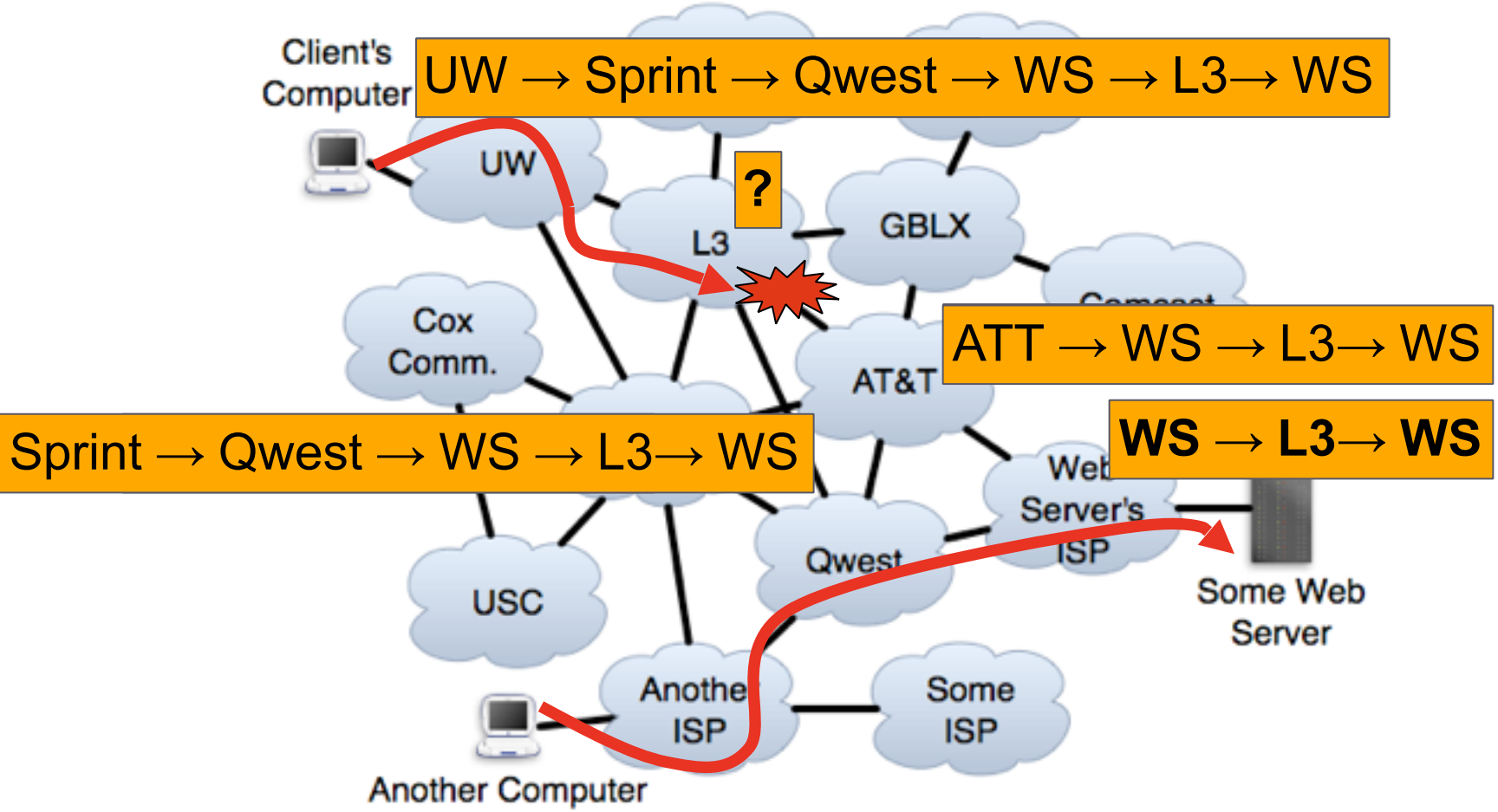
BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



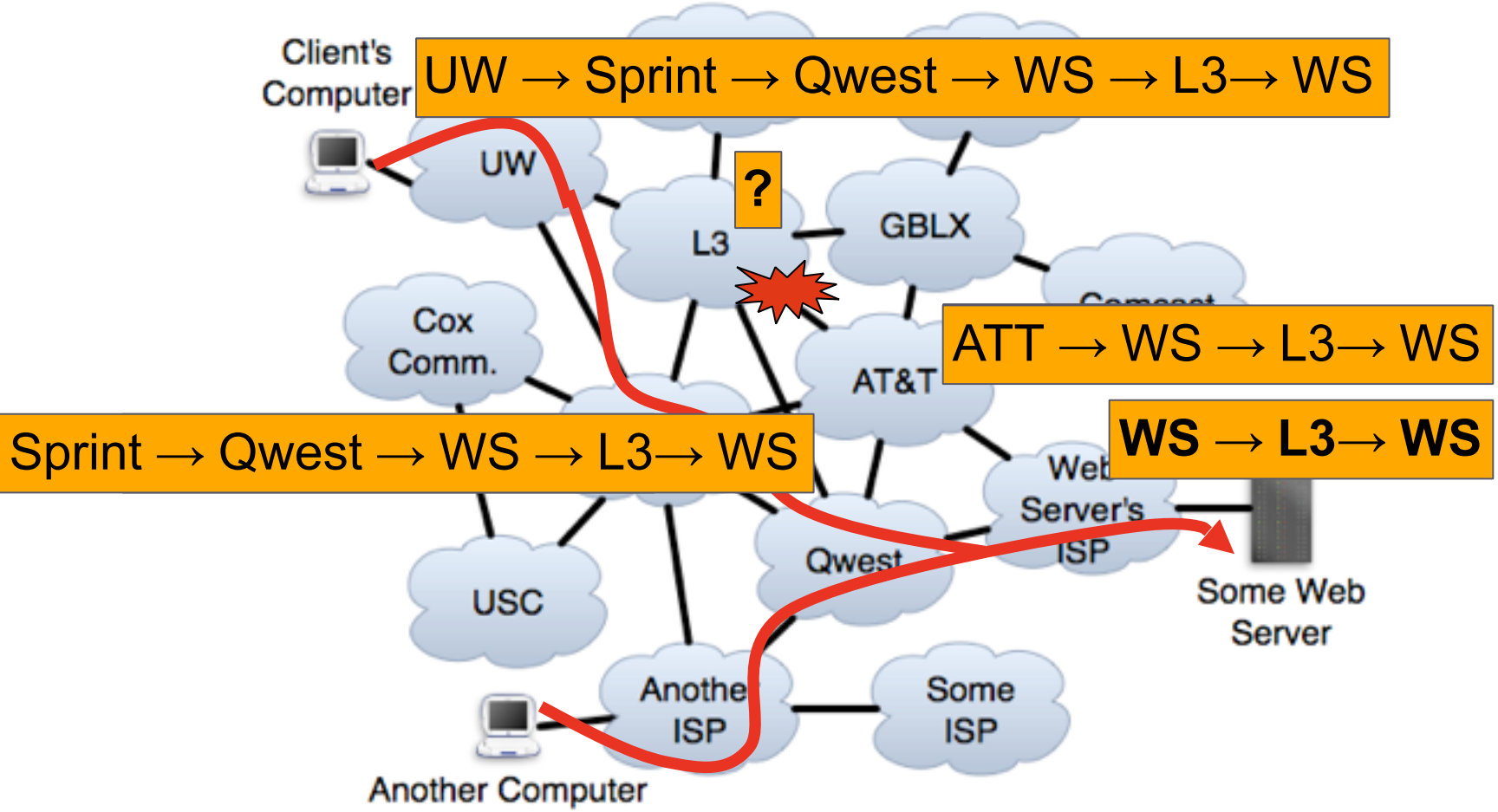
BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



BGP loop prevention encourages switch to working path.

Practical Self-Repair of Reverse Paths



BGP loop prevention encourages switch to working path.

Inter-Domain Routing Summary

- 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

Why are these still issues?

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

Very similar issues to IPv6 deployment