Inter-Domain Routing: BGP

Richard T. B. Ma

School of Computing

National University of Singapore

CS 3103: Compute Networks and Protocols

Inter-Domain Routing

- Internet is a “network of networks”
- Hierarchy of Autonomous Systems
  - large, tier-1 provider with a nationwide backbone and international connections
  - medium-sized regional provider with smaller backbone
  - small network run by a single company or university
- Interaction between Autonomous Systems

Challenges for Inter-domain Routing

- Scale
  - millions of routers and 200,000+ prefixes
  - 35,000+ self-operated networks and 40K+ ASes
- Privacy
  - ASes don't want to expose internal topologies or their business relationships with neighbors
- Policy
  - no Internet-wide notion of a link cost metric
  - need control over where you send traffic and who can send traffic through you

Limitation of Link-State Routing

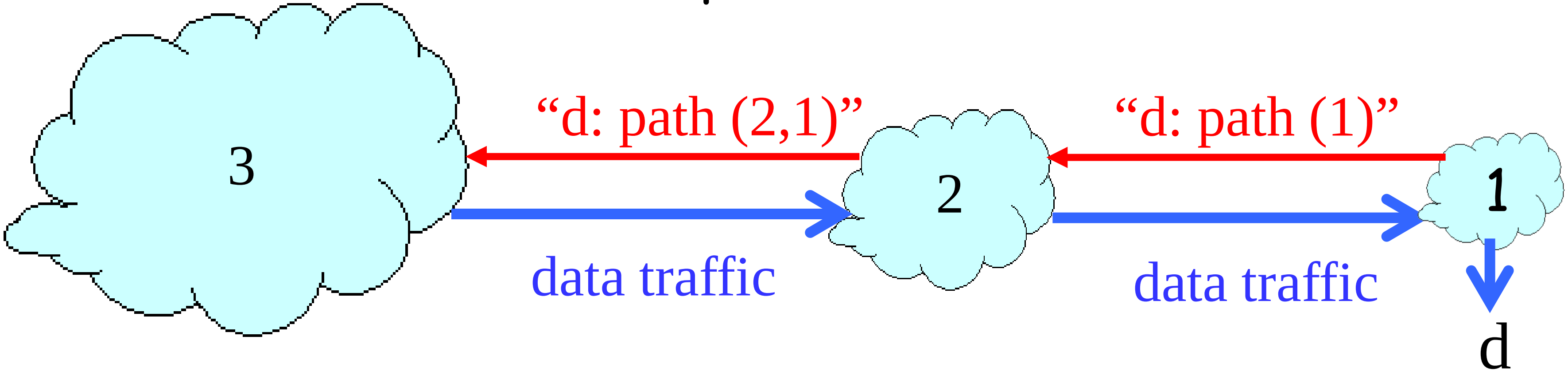
- Topology information is flooded
  - high bandwidth and storage overhead
  - nodes divulge sensitive information
- Entire path computed locally per node
  - high processing overhead in a large network
- Minimize some notion of total distance
  - works only if policy is shared and uniform
- Typically used only inside an AS
  - OSPF for instance

Cons and pros of DV approach

- advantages
  - hide details of the network topology
  - only next hop is determined per node
- disadvantages
  - minimizes some notion of total distance, which is difficult in an inter-domain setting
  - slow convergence due to the counting-to-infinity problem
- solution: extend the notion of a DV

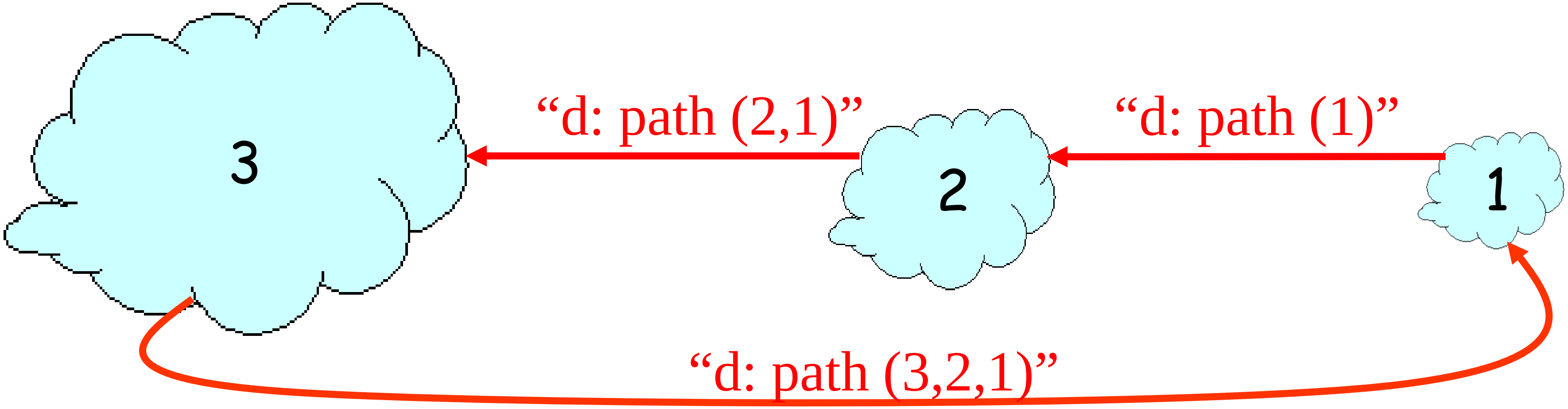
Path-Vector Routing

- Extension of distance-vector routing
  - support flexible routing policies
  - avoid count-to-infinity problem
- Key ides: advertise the entire path
  - DV: send distance metric per destination d
  - PV: send the entire path for each destination d



Faster Loop Detection

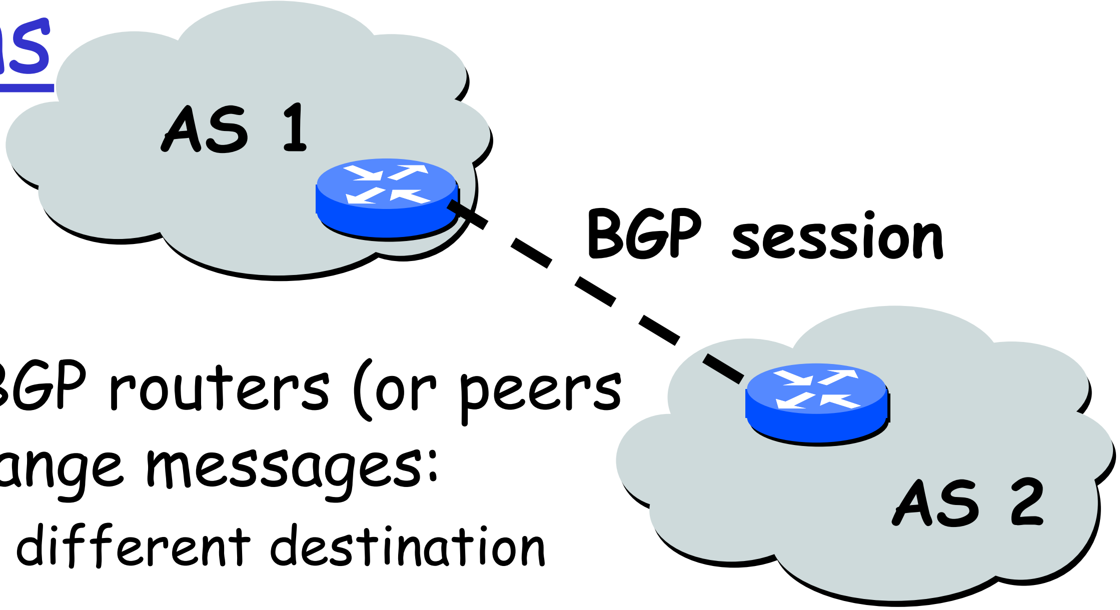
- Node can easily detect a loop
  - check if itself is in the path
- Node can simply discard paths with loops
  - e.g., node 1 simply discards the advertisement



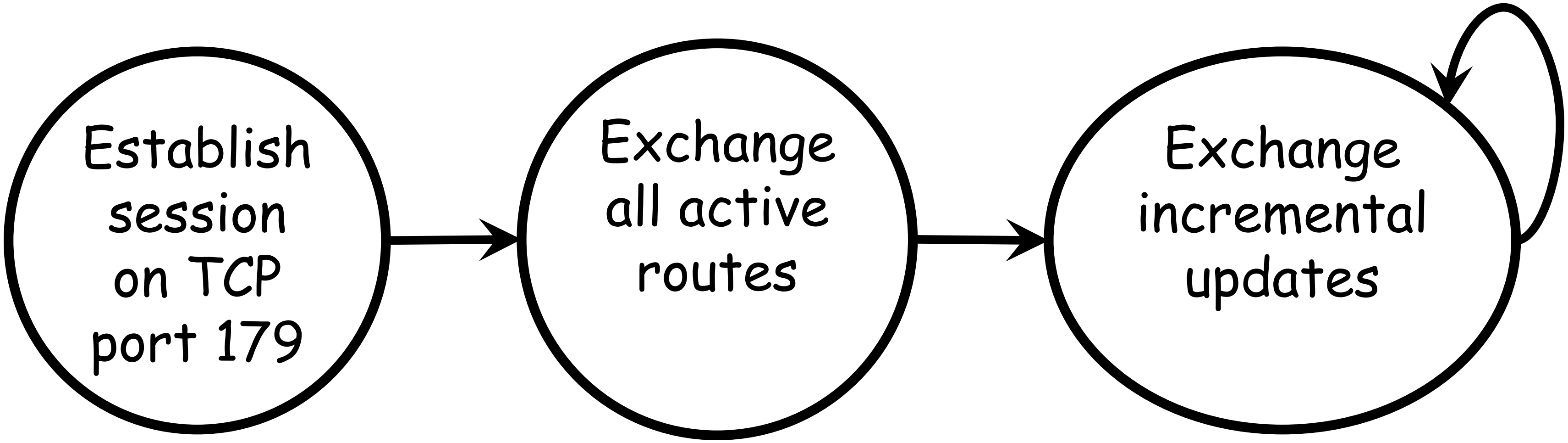
Border Gateway Protocol (BGP)

- BGP: *the* de facto inter-domain routing protocol
  - prefix-based path-vector protocol
  - BGP4 described in RFC 4271 (104 pages)
  - RFC 4276 gives an implementation report on BGP
  - RFC 4277 describes operational experiences using BGP
  - enable policy-based routing based on AS Paths
- allows subnet to advertise its existence to rest of Internet: *"I am here"*
- allows ASes to determine "good" routes to other networks based on reachability info and policy

BGP operations



- BGP session: two BGP routers (or peers or speakers) exchange messages:
  - advertise *paths* to different destination network prefixes



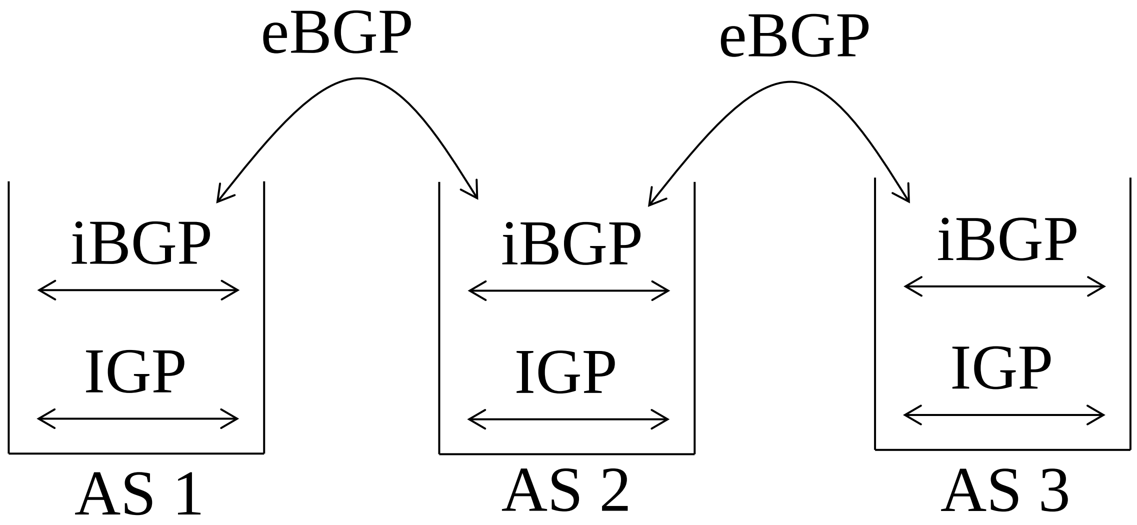
Establish session on TCP port 179

Exchange all active routes

Exchange incremental updates

While connection ALIVE, exchange route UPDATE messages

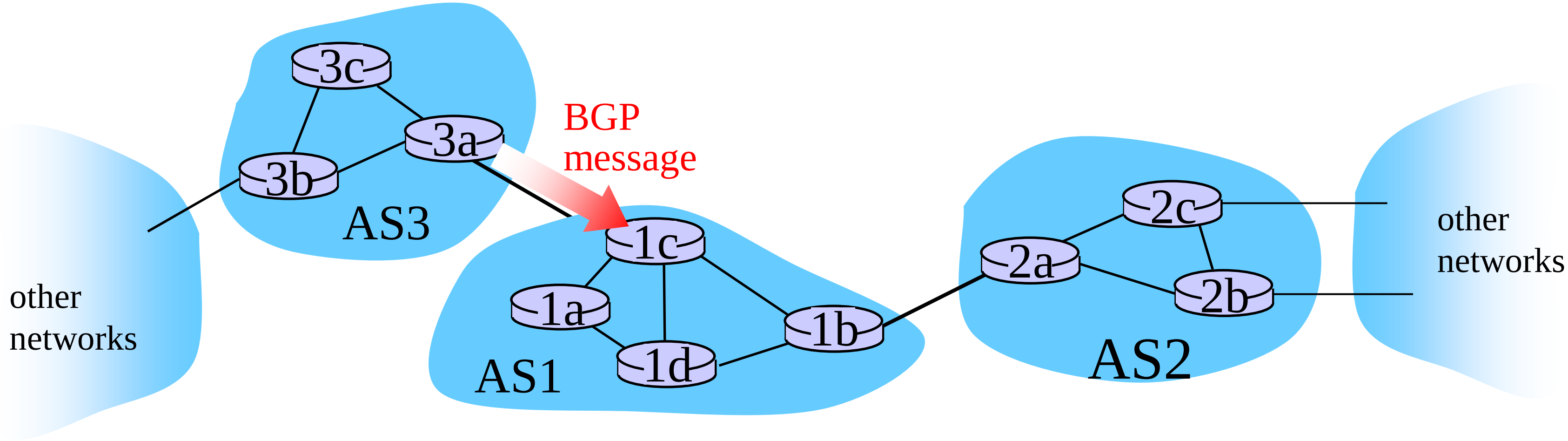
BGP/IGP model used in ISPs



- eBGP: exchange reachability info from neighbor ASes; implement routing policy
- iBGP: propagate reachability info across backbone; carry ISP's own customer prefixes

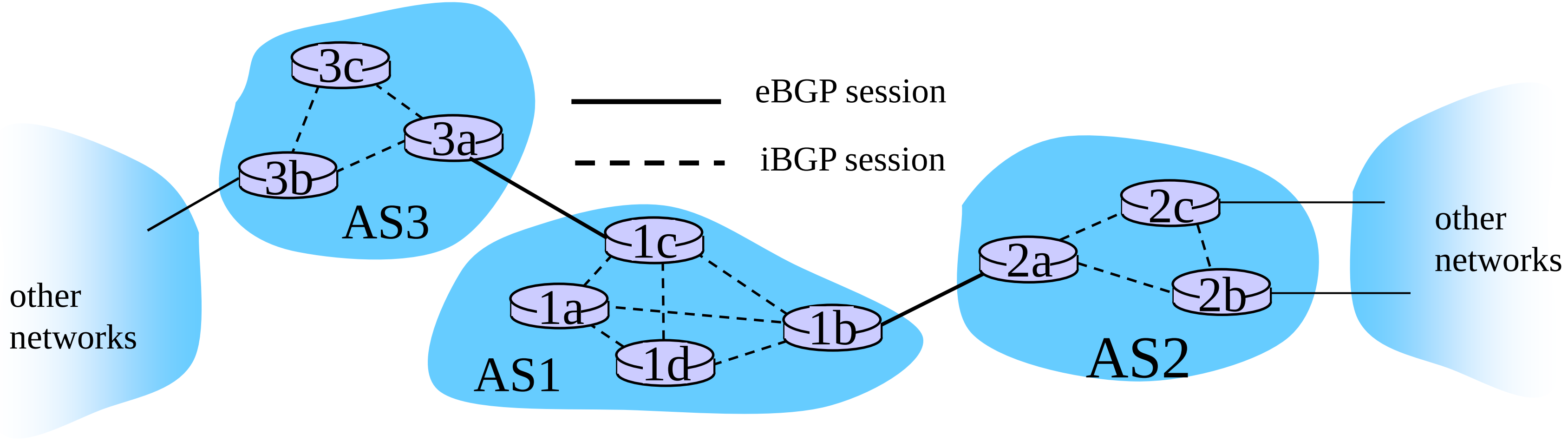
eBGP

- external BGP peering (eBGP)
  - between BGP speakers in different ASes
  - should be directly connected
  - never run an IGP between eBGP peers
- when AS3 advertises a prefix to AS1:
  - AS3 *promises* it will forward datagrams towards that prefix
  - AS3 can aggregate prefixes in its advertisement



iBGP

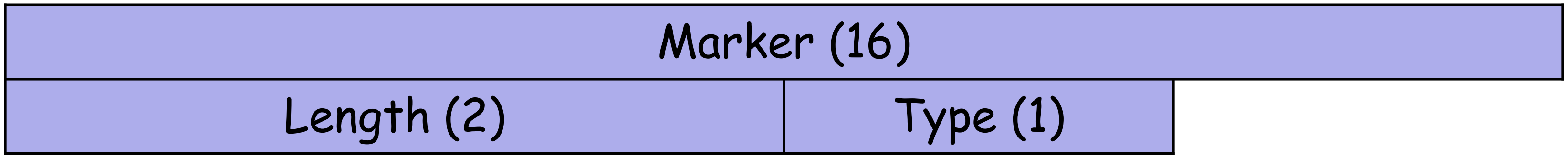
- internal BGP peering (iBGP)
  - peers within an AS; not required to be directly connected
    - IGP takes care of inter-BGP speaker connectivity
  - iBGP peers must be fully meshed (via loopback interface)
    - They originate connected networks
    - Pass on prefixes learned from outside the AS
    - Do not pass on prefixes learned form other iBGP speakers
- 1c can use iBGP do distribute prefix info to all routers in AS1; 1b can re-advertise info to AS2 over eBGP



BGP messages

- OPEN: opens TCP connection to peer and authenticates sender
- UPDATE: advertises new paths (or withdraws old paths)
- KEEPALIVE: keeps connection alive in absence of UPDATES; also ACKs OPEN request
- NOTIFICATION: reports errors in previous mssages; also used to close connection

BGP Message Header Format



Marker (16)

Length (2)

Type (1)

- Marker: 16-byte field for compatibility
- Length: 2-byte unsigned integer indicates the total length of the message
- Type: 1-byte unsigned integer indicates the type code of the message
    - 1 – OPEN
    - 2 – UPDATE (most important and complicated)
    - 3 – NOTIFICATION
    - 4 - KEEPALIVE

OPEN Message Format

<table>
<tr><td colspan="4">Marker (16)</td></tr>
<tr><td colspan="2">Length (2)</td><td>Type (1)</td><td>Version (1)</td></tr>
<tr><td colspan="2">My Autonomous System (2)</td><td colspan="2">Hold Time (2)</td></tr>
<tr><td colspan="4">BGP Identifier (4)</td></tr>
<tr><td>Opt Par Len (1)</td><td colspan="3">Optional Parameters (variable)</td></tr>
</table>

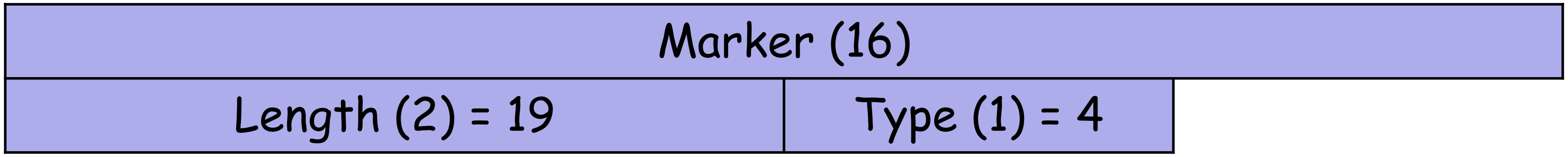
- Version: 1 byte indicates the protocol version
- My AS: 2 bytes indicate the ASN of the sender
- Hold Time: 2 bytes indicate the number of seconds the sender propose for the Hold Timer
- BGP ID: 4 bytes indicate the BGP identifier
- Use of option can be referred to RFC 3392

NOTIFICATION Message Format

<table>
<tr><td colspan="4">Marker (16)</td></tr>
<tr><td colspan="2">Length (2)</td><td>Type (1)</td><td>Error code (1)</td></tr>
<tr><td>Error subcode (1)</td><td colspan="3">Data (variable)</td></tr>
</table>

- Error codes:
  - 1: message header error
  - 2: OPEN message error
  - 3: UPDATE message error
  - 4: Hold time expired
  - …
- The length of data can be inferred by
  - Message length = 21 + data length
- How to respond to NOTIFICATION messages?
  - More BGP error handling details in the RFC

KEEPALIVE Message Format



Marker (16)

Length (2) = 19

Type (1) = 4

- KEEPALIVE is just the 19-byte message header
- Used to determine if peers are reachable
- Maximum inter-KEEPALIVE (typically 60s)
  - = 1/3 of Hold Time (typically 180s)
- Must not sent more frequently than 1 per second

UPDATE Message Format

<table>
<tr><td colspan="4">Marker (16)</td></tr>
<tr><td colspan="2">Length (2)</td><td>Type (1)</td><td></td></tr>
<tr><td colspan="2">Withdrawn Routes Length (2)</td><td colspan="2">Withdrawn Routes (variable)</td></tr>
<tr><td colspan="2">Path Attribute Length (2)</td><td colspan="2">Path Attributes (variable)</td></tr>
<tr><td colspan="4">Network Layer Reachability Information (variable)</td></tr>
</table>

- Withdrawn Routes: IP prefixes for the routes withdrawn
- Network Layer Reachability Information (NLRI): IP prefixes that could be reached from the advertised route
  - NLRI length can be inferred as:

UPDATE Message Len – 23 - Withdrawn Routes Len - Path Attribute Len

  - IP address prefixes are coded more compactly (refer to RFC)

UPDATE Message Format

Marker (16)	
Length (2)	Type (1)
Withdrawn Routes Length (2)	Withdrawn Routes (variable)
Path Attribute Length (2)	Path Attributes (variable)
Network Layer Reachability Information (variable)	

- ❑ Can only advertise one feasible route for the NLRI
- ❑ Can withdraw multiple routes in an UPDATE message
- ❑ Should not have the NLRI prefix in Withdrawn Routes
  - ❖ otherwise, should treat as if Withdrawn Routes do not contain the address prefix

Withdrawn Routes

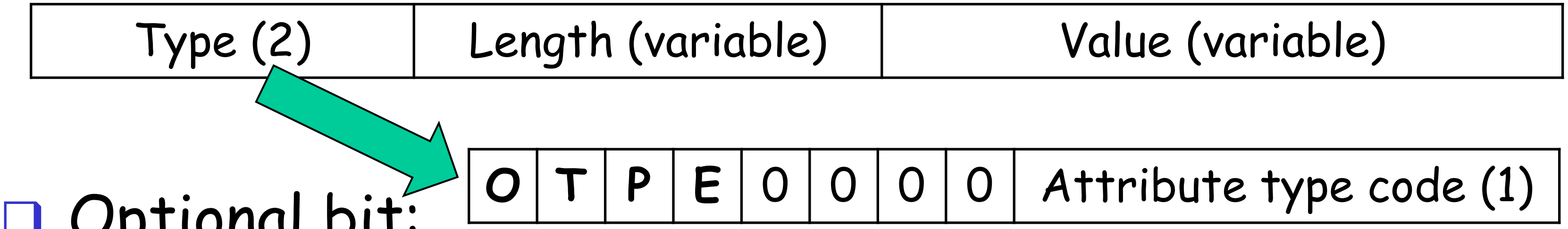
- No expiration timer for the routes like RIP
- Invalidate routes are actively withdrawn by the original advertiser
- Or use UPDATE message to replace the existing routes
- All routes from a peer become invalid when the peer goes down

BGP Path Attributes

- Fall into four separate categories:
  1. Well-known mandatory
  2. Well-known discretionary
  3. Optional transitive
  4. Optional non-transitive
- Some implementation rules:
  - Must recognize all well-known attributes
  - Mandatory attributes must be included in UPDATE messages that contain NLRI
  - Once a BGP peer updates well-known attributes, it must pass them to its peers

Path Attribute Format

- Each path attribute is a triple



Type (2)	Length (variable)	Value (variable)

O	T	P	E	0	0	0	0	Attribute type code (1)

- Optional bit:
  - whether the attribute is optional (1) or well-known (0)
- Transitive bit:
  - whether the attribute should be forwarded along the AS path; well-known attribute must have 1 for transitive bit
- Partial bit:
  - optional transitive attribute is unrecognized (1)
  - set value 0 for attributes in other categories
- Extended Length bit:
  - 1-byte (0) or 2-byte (1) for the length field

Common Path Attributes

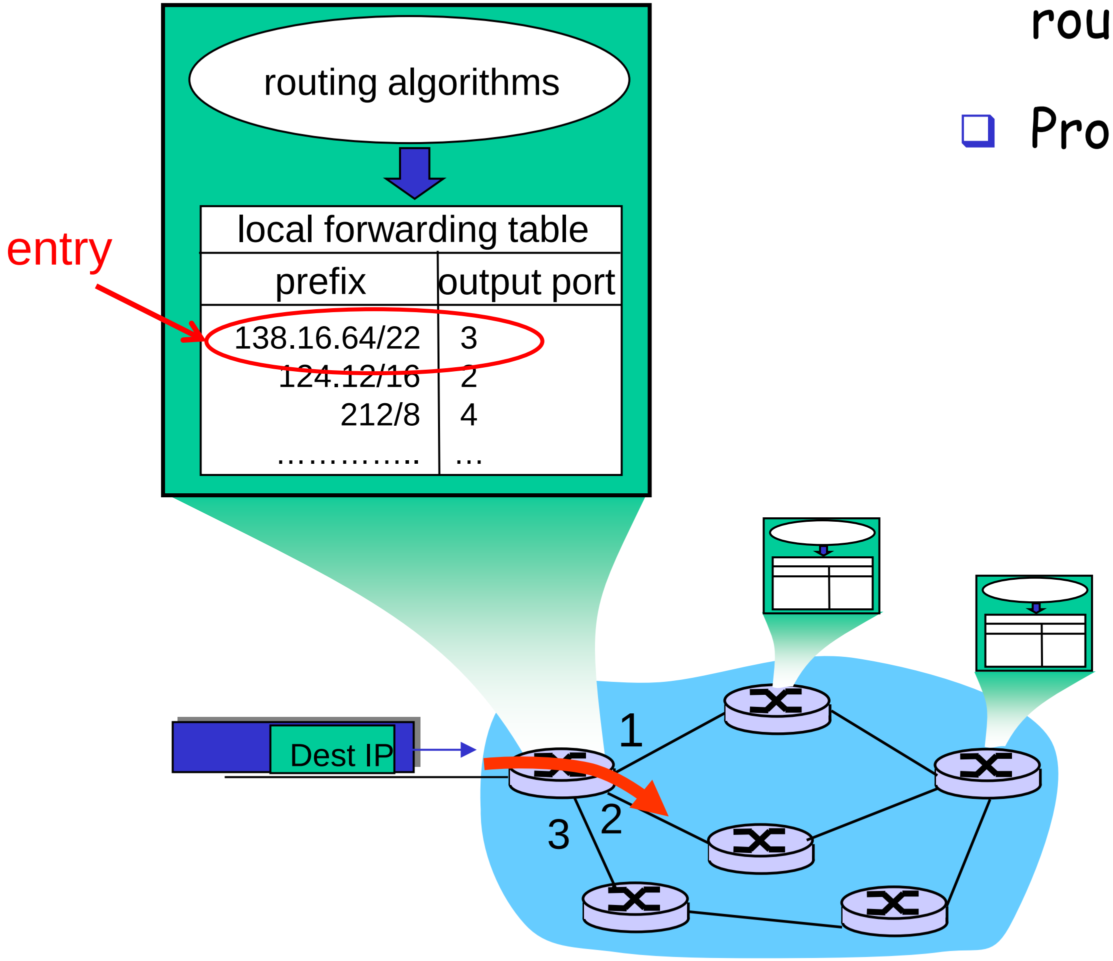
Attribute Name	Type code	Category
ORIGIN	1	Well-Known Mandatory
AS_PATH	2	Well-Known Mandatory
NEXT_HOP	3	Well-Known Mandatory
LOCAL_PREF	5	Well-Known Discretionary
ATOMIC_AGGREGATE	6	Well-Known Discretionary
AGGREGATOR	7	Optional Transitive
COMMUNITY	8	Optional Transitive
MULTI_EXIT_DISC (MED)	4	Optional Non-Transitive

Well-Known mandatory attributes

- ORIGIN:
  - conveys the origin of the prefix
  - historical attribute used in transition from EGP to BGP
- AS-PATH:
  - contains ASes through which NLRI has passed
  - expressed as a sequence, e.g., AS 79, AS 11 … , or a set
- NEXT-HOP:
  - indicates IP address of the router in the next-hop AS. (may be multiple links from current AS to next-hop-AS)

How does entry get in forwarding table?

Assume prefix is in another AS.



local forwarding table

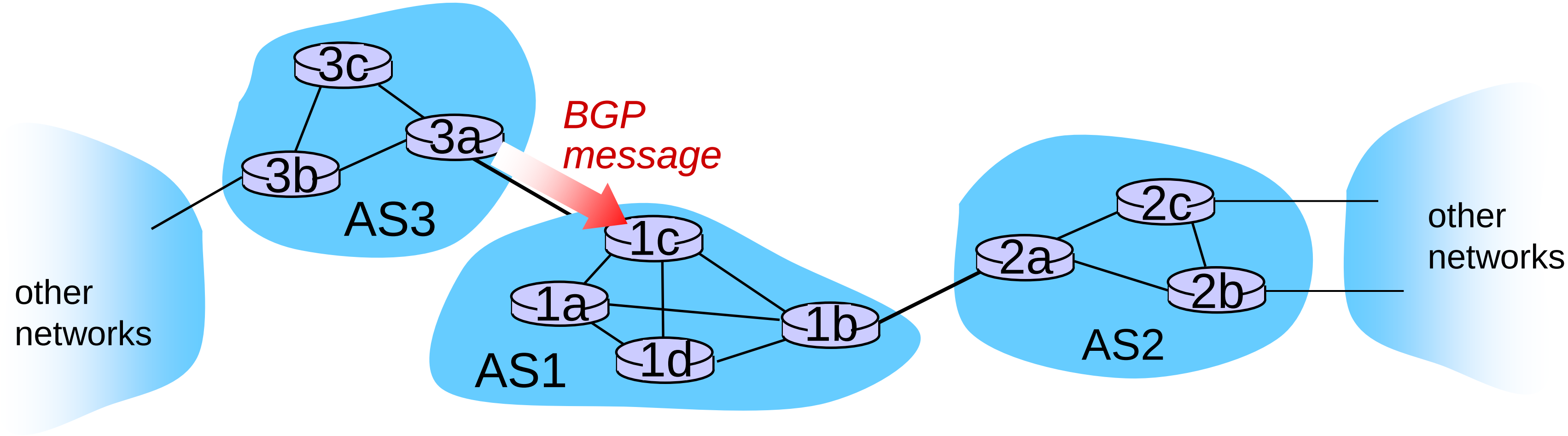
prefix	output port
138.16.64/22	3
124.12/16	2
212/8	4
…………..	…

- ❑ Ties together hierarchical routing with BGP and OSPF.
- ❑ Provides nice overview of BGP!

High-level overview

1. Router becomes aware of IP prefix
2. Router determines the output port for the IP prefix
3. Router enters the prefix-port pair in forwarding table

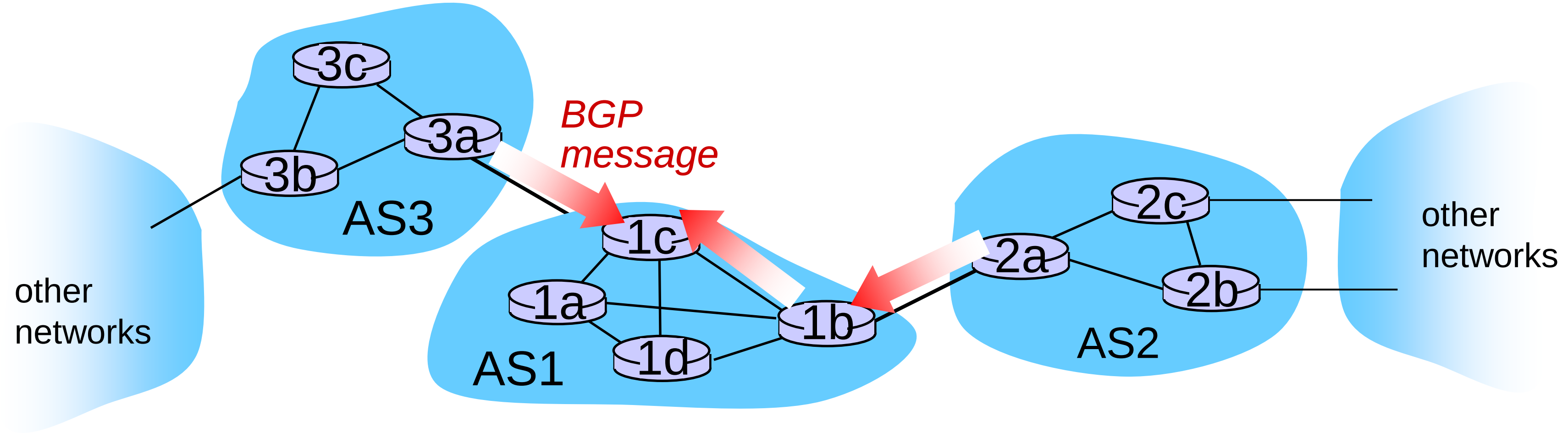
Becomes aware of destination prefix



- BGP message contains "routes"
- route = prefix + attributes: AS-PATH, NEXT-HOP,…
- Example: route:

  Prefix: 138.16.64/22; AS-PATH: AS3 AS131;
  NEXT-HOP: 201.44.13.125

Router may receive multiple routes



- Router may receive multiple routes for <u>same</u> destination prefix
- The router has to select one route

Select best BGP route to prefix

- ❑ Router selects route based on shortest AS-PATH

- ❖ Example:

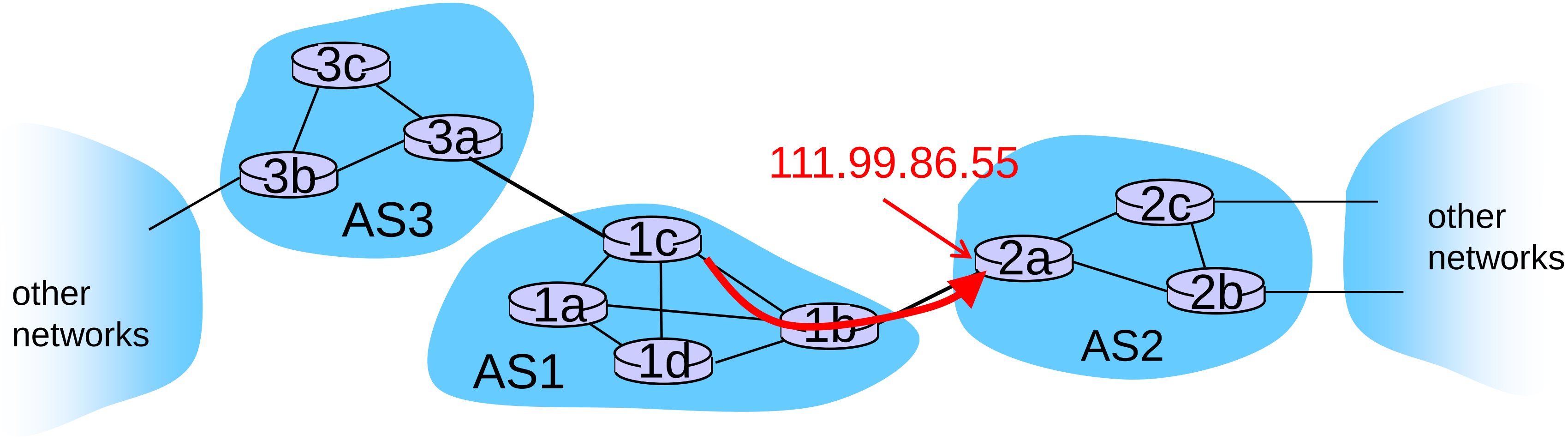
select

  - ❖ AS2 AS17 to 138.16.64/22
  - ❖ AS3 AS131 AS201 to 138.16.64/22

- ❖ What if there is a tie? will come back to that!

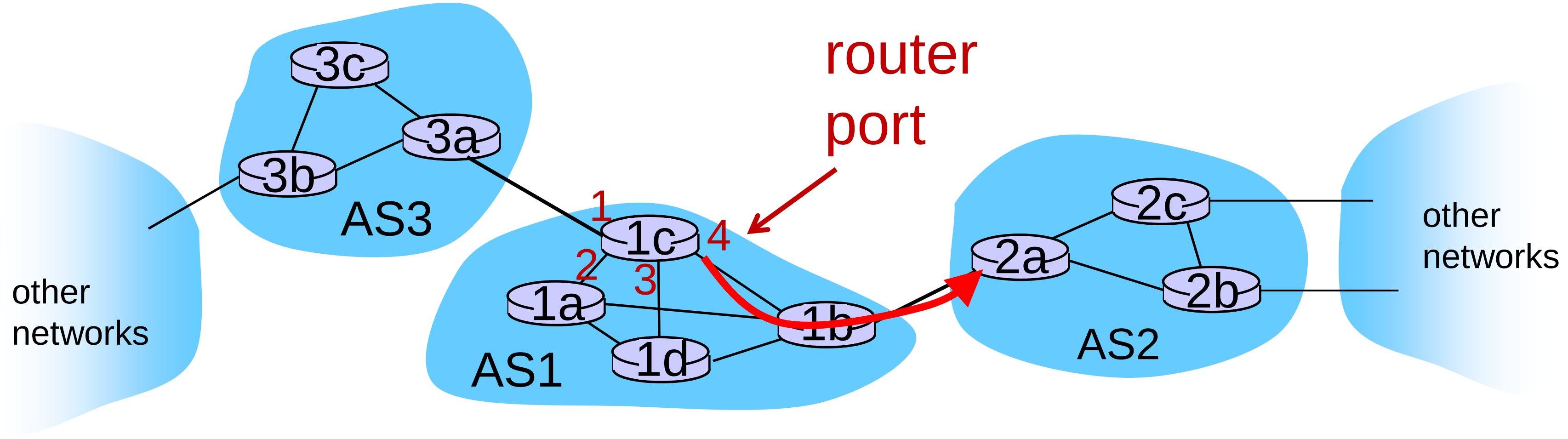
Find best intra-route to BGP route

- Use selected route's NEXT-HOP attribute
  - Route's NEXT-HOP attribute is the IP address of the router interface that begins the AS PATH.
- Example:
  - AS-PATH: AS2 AS17; NEXT-HOP: 111.99.86.55
- Router uses OSPF to find shortest path from 1c to 111.99.86.55



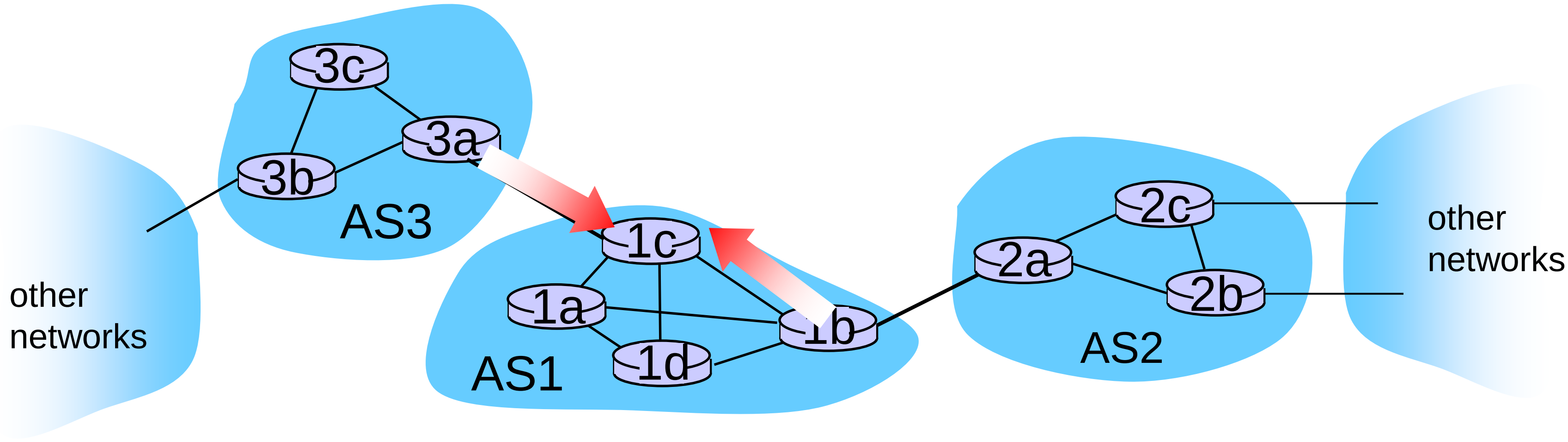
Router identifies port for route

- Identifies port along the OSPF shortest path
- Adds prefix-port entry to its forwarding table:
  - (138.16.64/22 , port 4)



Hot Potato Routing

- ❑ if there exists two or more best inter-routes
- ❑ then choose route with closest NEXT-HOP
  - ❖ Use OSPF to determine which gateway is closest
  - ❖ Q: From 1c, chose AS3 AS131 or AS2 AS17?
  - ❖ A: route AS3 AS131 since it is closer



How does entry get in forwarding table?

Summary

1. Router becomes aware of prefix
   - via BGP route advertisements from other routers
2. Determine router output port for prefix
   - Use BGP route selection to find best inter-AS route
   - Use OSPF to find best intra-AS route leading to best inter-AS route
   - Router identifies router port for that best route
3. Enter prefix-port entry in forwarding table