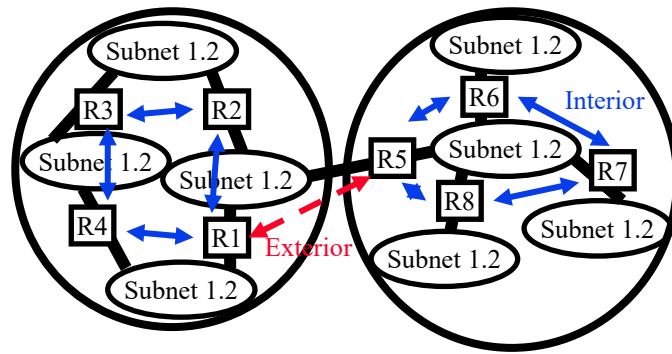


# The Network Layer: Control Plane



**Raj Jain**

Washington University in Saint Louis

Saint Louis, MO 63130

Jain@wustl.edu

Audio/Video recordings of this lecture are available on-line at:

<http://www.cse.wustl.edu/~jain/cse473-23/>

## Student Questions

- ❖ This symbol precedes questions asked during the Exam 2 Review session.



1. Routing Algorithms: Link-State, Distance Vector  
Dijkstra's algorithm, Bellman-Ford Algorithm
2. Routing Protocols: OSPF, BGP
3. SDN Control Plane
4. ICMP
5. SNMP

**Note:** This class lecture is based on Chapter 5 of the textbook (Kurose and Ross) and the figures provided by the authors.

## Student Questions

# Network Layer Functions

- ❑ Forwarding: Deciding what to do with a packet using a routing table  $\Rightarrow$  Data plane
- ❑ Routing: Making the routing table  $\Rightarrow$  Control Plane

## Student Questions



# Routing Algorithms

1. Graph abstraction
2. Distance Vector vs. Link State
3. Dijkstra's Algorithm
4. Bellman-Ford Algorithm

## Student Questions

# Rooting or Routing

- ❑ *Rooting* is what fans do at football games, what pigs do for truffles under oak trees in the Vaucluse, and what nursery workers intent on propagation do to cuttings from plants.
- ❑ *Routing* is how one creates a beveled edge on a tabletop or sends a corps of infantrymen into a full-scale, disorganized retreat.

## Student Questions

Ref: Piscitello and Chapin, "Open Systems Networking: TCP/IP and OSI," Adison-Wesley, 1993, p413

# Routeing or Routing

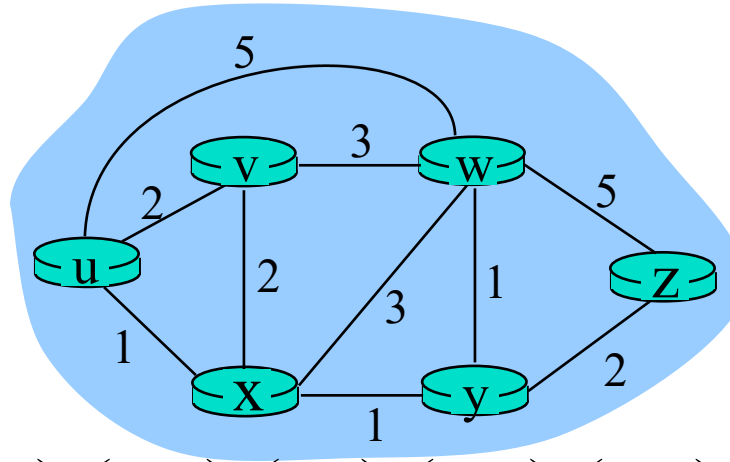
- ❑ Routeing: British
- ❑ Routing: American
- ❑ Since Oxford English Dictionary is much heavier than any other dictionary of American English, British English generally prevails in the documents produced by ISO and CCITT; wherefore, most of the international standards for routing standards use the routeing spelling.

## Student Questions

Ref: Piscitello and Chapin, "Open Systems Networking: TCP/IP and OSI," Addison-Wesley, 1993, p413

# Graph abstraction

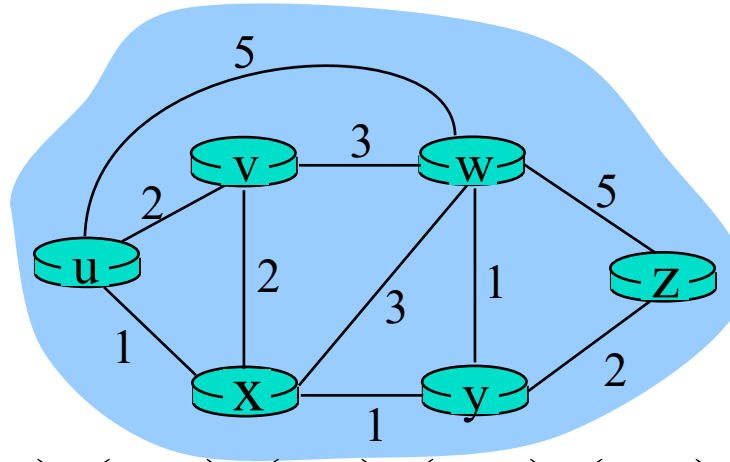
- ❑ Graph:  $G = (N, E)$
- ❑  $N =$  Set of routers  
 $= \{ u, v, w, x, y, z \}$
- ❑  $E =$  Set of links  
 $= \{ (u, v), (u, x), (u, w), (v, x), (v, w), (x, w), (x, y), (w, y), (w, z), (y, z) \}$
- ❑ Each link has a cost, e.g.,  $c(w, z) = 5$
- ❑ Cost of path  $(x_1, x_2, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$
- ❑ Routing Algorithms find the least cost path
- ❑ We limit to “Undirected” graphs, i.e., cost is same in both directions



## Student Questions

- ❑ What would the cost of the link represent? Throughput?  
*Opposite of nominal bit rate, delay, or distance*
- ❑ Do we have a cost for nodes?  
*Node cost is ignored. But it could be added to all links connected to that node.*
- ❑ Is the link cost function based on the RTT between the links or the distance?  
*See above.*

# Graph abstraction



- ❑ Graph:  $G = (N, E)$
- ❑  $N = \text{Set of routers}$   
 $= \{ u, v, w, x, y, z \}$
- ❑  $E = \text{Set of links}$   
 $= \{ (u, v), (u, x), (u, w), (v, x), (v, w), (x, w), (x, y), (w, y), (w, z), (y, z) \}$
- ❑ Each link has a cost, e.g.,  $c(w, z) = 5$
- ❑ Cost of path  $(x_1, x_2, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$
- ❑ Routing Algorithms find the least cost path
- ❑ We limit to “Undirected” graphs, i.e., cost is same in both directions

## Student Questions

- ❑ Do real-world routing algorithms use directed graphs as abstractions as well?

*Yes.*

- ❑ Can we apply Dijkstra's algorithm in this graph?

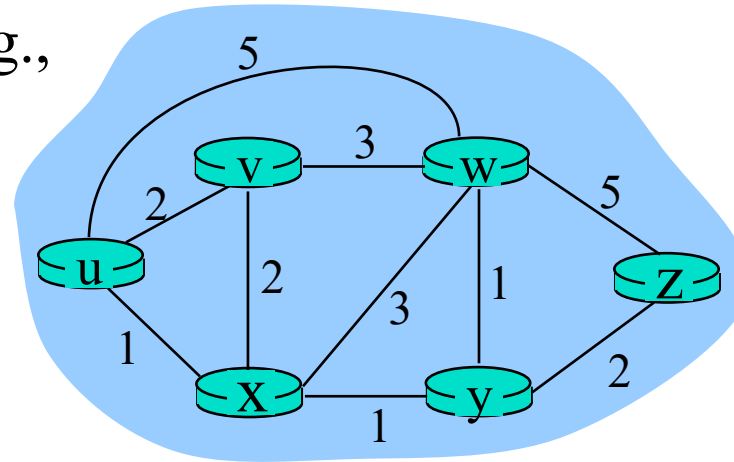
*Yes.*



# Distance Vector vs. Link State

## Distance Vector:

- ❑ Vector of distances to all nodes, e.g.,  
u: {u:0, v:2, w:5, x:1, y:2, z:4}
- ❑ Sent to neighbors, e.g.,  
u will send to v, w, x
- ❑ Large vectors to small # of nodes  
Tell about the world to neighbors
- ❑ Older method. Used in RIP.



## Link State:

- ❑ Vector of link cost to neighbors, e.g, u: {v:2, w:5, x:1}
- ❑ Sent to all nodes, e.g., u will send to v, w, x, y, z
- ❑ Small vectors to large # of nodes  
Tell about the neighbors to the world
- ❑ Newer method. Used in OSPF.

## Student Questions

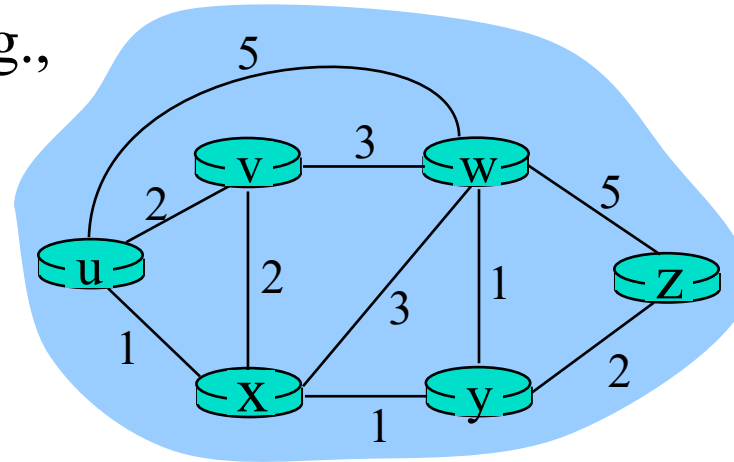
- ❑ Is there a reason, other than the fact that link-state algorithms do not encounter counting-to-infinity problems, that link-state is preferable to distance-vector?  
*No. But counting to infinity is a BIG problem.*
- ❑ Will distance vector and link state result in different routing tables?  
*No. The final answer is the same. But the number of iterations required to settle down after a change in the network is significantly different.*

- ❑ Why is it called a vector? It seems more like a set.  
*A vector is a set with one column.*
- ❑ What does RIP stand for? I couldn't find it in the list of acronyms.  
*Routing Information Protocol*
- ❑ What is meant by large vectors to a small # of nodes?  
*The number of elements in the vector is large.*

# Distance Vector vs. Link State

## Distance Vector:

- ❑ Vector of distances to all nodes, e.g.,  
u: {u:0, v:2, w:5, x:1, y:2, z:4}
- ❑ Sent to neighbors, e.g.,  
u will send to v, w, x
- ❑ Large vectors to small # of nodes  
Tell about the world to neighbors
- ❑ Older method. Used in RIP.



## Link State:

- ❑ Vector of link cost to neighbors, e.g, u: {v:2, w:5, x:1}
- ❑ Sent to all nodes, e.g., u will send to v, w, x, y, z
- ❑ Small vectors to large # of nodes  
Tell about the neighbors to the world
- ❑ Newer method. Used in OSPF.

## Student Questions

- ❑ How come distance vector is sent to fewer nodes than link state?  
*Link state is sent to the world.*
- ❑ When would it be better to use link state instead of distance vector, and vice versa?

*Discussed later in this module.*

- ❑ Why are we telling information about distance or cost to other nodes?

*To find the best route.*

- ❑ To which routing algorithm mentioned here does Dijkstra's algorithm belong? Link state?

*Yes*

# Dijkstra's Algorithm

- ❑ Goal: Find the least cost paths from a given node to all other nodes in the network
- ❑ Notation:
  - $c(i,j)$  = Link cost from  $i$  to  $j$  if  $i$  and  $j$  are connected
  - $D(k)$  = Total path cost from  $s$  to  $k$
  - $N'$  = Set of nodes so far for which the least cost path is known
- ❑ Method:
  - Initialize:  $N' = \{u\}$ ,  $D(v) = c(u,v)$  for all neighbors of  $u$
  - Repeat until  $N$  includes all nodes:
    - ❑ Find node  $w \notin N'$ , whose  $D(w)$  is minimum
    - ❑ Add  $w$  to  $N'$

## Student Questions

- ❑ Is Dijkstra's Algorithm ever implemented with a min-priority queue in Networking?

*Implementations are not standardized. So yes, someone may implement it using heaps.*

- ❑ Is there any tradeoff between making it faster vs. the space required?

*Yes. That's almost always true. Any computation can be made faster by caching.*

- ❑ Is Dijkstra's algorithm run on every node in the network or just a subset of the nodes? It seems that there might be some re-work to run on every node since the paths might have overlapped when computing different nodes.

*The algorithm is run on every **router**. Paths may **change** while the algorithm is still in progress. If that happens, the router noticing the change will send its new table, and the process will eventually end **iff** the configuration does not change again.*

---

# Dijkstra's Algorithm

- ❑ Goal: Find the least cost paths from a given node to all other nodes in the network
- ❑ Notation:  
 $c(i,j)$  = Link cost from  $i$  to  $j$  if  $i$  and  $j$  are connected  
 $D(k)$  = Total path cost from  $s$  to  $k$   
 $N'$  = Set of nodes so far for which the least cost path is known
- ❑ Method:
  - Initialize:  $N' = \{u\}$ ,  $D(v) = c(u,v)$  for all neighbors of  $u$
  - Repeat until  $N$  includes all nodes:
    - ❑ Find node  $w \notin N'$ , whose  $D(w)$  is minimum
    - ❑ Add  $w$  to  $N'$

## Student Questions

- ❑ How does Dijkstra's algorithm handle network changes or failures, and what strategies does it use to ensure network stability?

*See Slide 5.11*

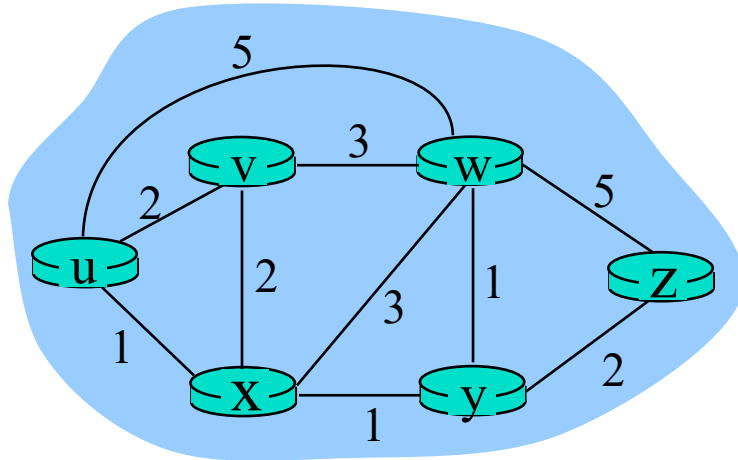
- ❑ Do we need to know how to write Dijkstra's algorithm in exam or need to know how it works?

*How to find the path using it in the exam.*

- ❑ Do we need to know the performance of Dijkstra's algorithm or other algorithms in this course?

*Yes.*

# Dijkstra's Algorithm: Example

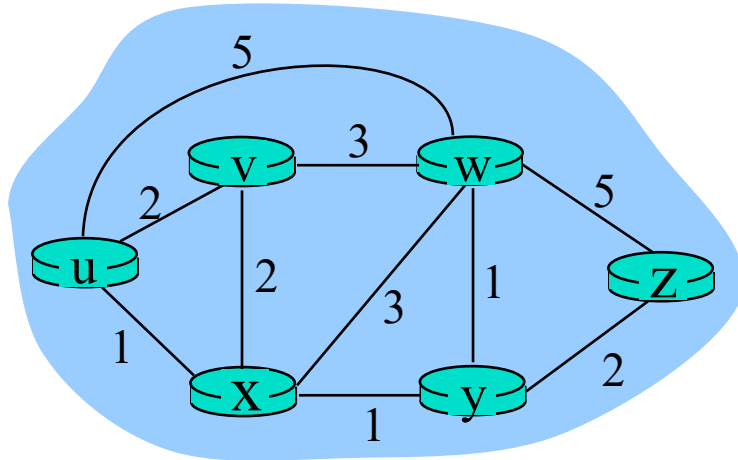


	$N'$	$D(v)$	Path	$D(w)$	Path	$D(x)$	Path	$D(y)$	Path	$D(z)$	Path
0	{u}	2	u-v	5	u-w	1	u-x	$\infty$	-	$\infty$	-
1	{u, x}	2	u-v	4	u-x-w			2	u-x-y	$\infty$	-
2	{u, x, y}	2	u-v	3	u-x-y-w					4	u-x-y-z
3	{u, x, y, v}			3	u-x-y-w					4	u-x-y-z
4	{u, x, y, v, w}									4	u-x-y-z
5	{u, x, y, v, w, z}										

## Student Questions

- Could you again go over the differences between Dijkstra's and Bellman-Ford's algorithms? It's easy to be confused about how exactly they are different. *Dijkstra broadcasts its link-state table to the entire network, and computation proceeds hop by hop.*
  - Bellman-Ford broadcasts its distance vector to its neighbors. Computation continues until the distance vectors do not change.
  - Link state tables are smaller than distance vectors.
  - Link state tables have to be sent to the entire network. Distance vectors have to be sent only to neighbors.
  - Can we go over another example of Dijkstra's algorithm?
- You can make many variations of this graph by changing the costs or source node.

# Dijkstra's Algorithm: Example



	$N'$	$D(v)$	Path	$D(w)$	Path	$D(x)$	Path	$D(y)$	Path	$D(z)$	Path
0	{u}	2	u-v	5	u-w	1	u-x	$\infty$	-	$\infty$	-
1	{u, x}	2	u-v	4	u-x-w			2	u-x-y	$\infty$	-
2	{u, x, y}	2	u-v	3	u-x-y-w					4	u-x-y-z
3	{u, x, y, v}			3	u-x-y-w					4	u-x-y-z
4	{u, x, y, v, w}									4	u-x-y-z
5	{u, x, y, v, w, z}										

## Student Questions

❑ What did the quiz mean by "not in the set of nodes"? I usually think of Dijkstra's as from one node in the graph to another.

*The 2<sup>nd</sup> column in the table is the set of nodes.*

❖ Can we go over P4 on page 439 of the book

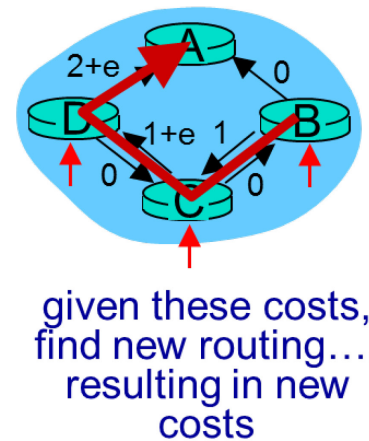
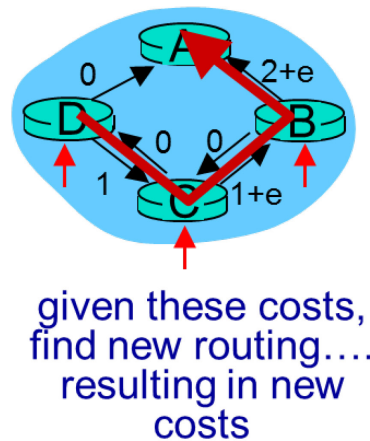
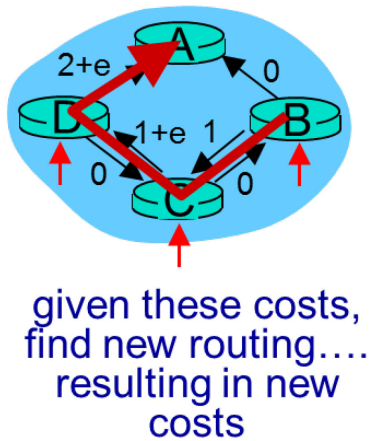
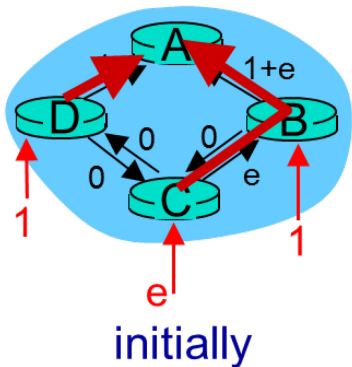
*This is Homework 5A, done 36 times. Good for practice.*

❖ Do we need to know the steps of Dijkstra's link-state routing?

*YES*

# Complexity and Oscillations

- ❑ *Algorithm complexity:  $n$  nodes*
  - Each iteration: need to check all nodes,  $w$ , not in  $N$
  - $n(n+1)/2$  comparisons:  $O(n^2)$
  - More efficient implementations possible:  $O(n \log n)$
- ❑ *Oscillations Possible: e.g., support link cost equals amount of carried traffic*



## Student Questions

- ❑ Why is it  $n+1$  in the complexity  $n(n+1)/2$ ?  
 $1+2+3+4+\dots+n = n(n+1)/2$
- ❑ Is the  $n$  in the runtime for Dijkstra's algorithm the number of routers?  
*Yes.*  
 $n = \text{number of nodes}$

❑ Will oscillation lead to a change in routing results?

*Yes.*

❑ Do we manage the cost dynamically according to the current state?

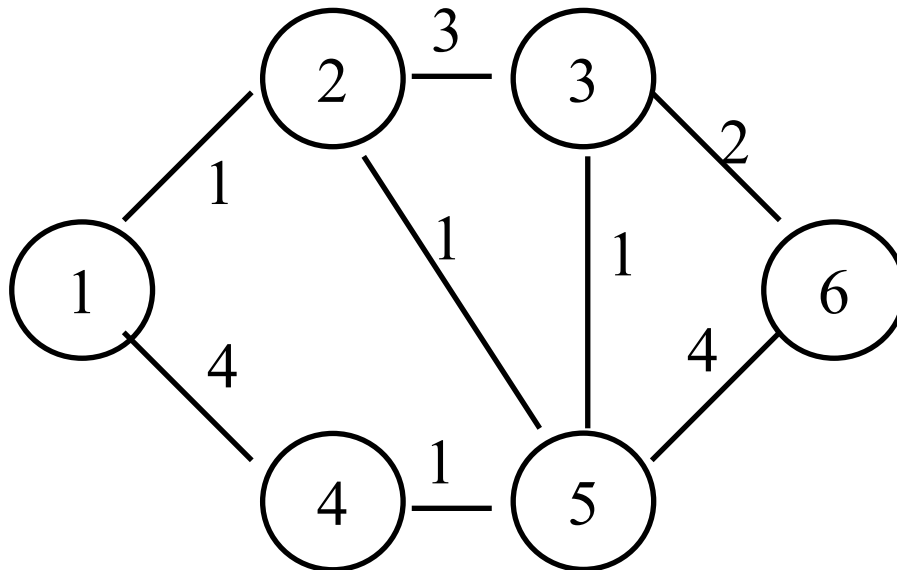
*If necessary, yes.*

❑ Can you explain again why is it  $n(n+1)/2$  comparisons?

$$n+(n-1)+(n-2)+\dots+1 = n(n+1)/2$$

# Homework 5A

[12 points] Prepare the routing calculation table for node 1 in the following network using Dijkstra's algorithm. Explain how you computed new entries in each row.



## Student Questions

❑ Should the routing table look like the one on slide 10? Which entries should each node have?

*No. Computation is shown in slide 5.10. Routing tables are shown in Slide 4.4.*

Prefix	Next Router	Interface
126.23.45.67/32	125.200.1.1	1
128.272.15/24	125.200.1.2	2
128.272/16	125.200.1.1	1



# Bellman-Ford Algorithm

## □ Notation:

$u$  = Source node

$c(i,j)$  = link cost from  $i$  to  $j$

$h$  = Number of hops being considered

$D_u(n)$  = Cost of  $h$ -hop path from  $u$  to  $n$

## □ Method:

1. Initialize:  $D_u(n) = \infty$  for all  $n \neq u$ ;  $D_u(u) = 0$
2. For each node:  $D_u(n) = \min_j [D_u(j) + c(j,n)]$
3. If any costs change, repeat step 2

## Student Questions

- When do we use Dijkstra's vs. Bellman-Ford's? Is one for distance vector and the other for link state?

*Bellman-Ford is a distance vector algorithm. Dijkstra is a link-state algorithm.*

- What would the difference between the Bellman-Ford and Dijkstra algorithms be?

*See Slide 5.17*

# Bellman Ford Example 1

**node x table**

		cost to		
		x	y	z
from	x	0	2	7
	y	$\infty$	$\infty$	$\infty$
	z	$\infty$	$\infty$	$\infty$

**node y table**

		cost to		
		x	y	z
from	x	$\infty$	$\infty$	$\infty$
	y	2	0	1
	z	$\infty$	$\infty$	$\infty$

**node z table**

		cost to		
		x	y	z
from	x	$\infty$	$\infty$	$\infty$
	y	$\infty$	$\infty$	$\infty$
	z	7	1	0

		cost to		
		x	y	z
from	x	0	2	3
	y	2	0	1
	z	7	1	0

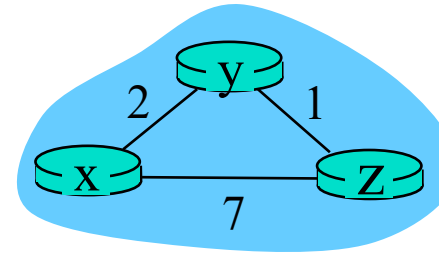
		cost to		
		x	y	z
from	x	0	2	7
	y	2	0	1
	z	7	1	0

		cost to		
		x	y	z
from	x	0	2	7
	y	2	0	1
	z	3	1	0

		cost to		
		x	y	z
from	x	0	2	3
	y	2	0	1
	z	3	1	0

		cost to		
		x	y	z
from	x	0	2	3
	y	2	0	1
	z	3	1	0

		cost to		
		x	y	z
from	x	0	2	3
	y	2	0	1
	z	3	1	0

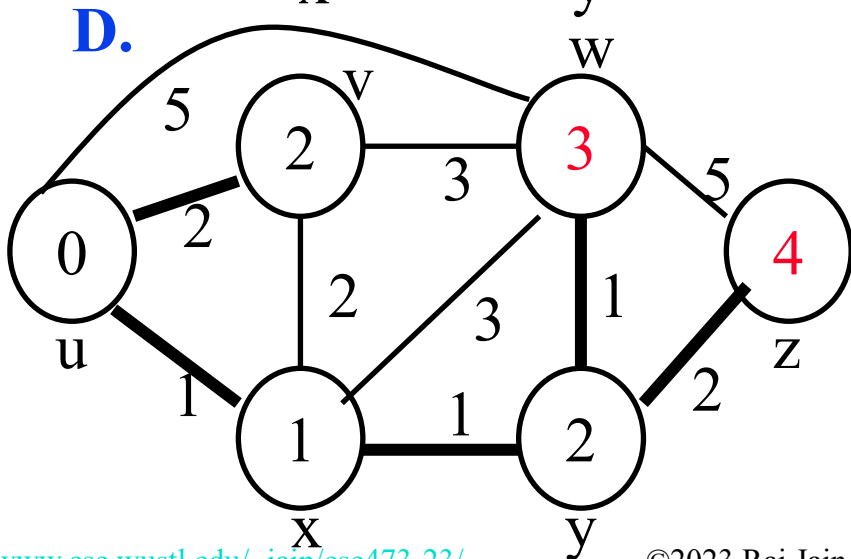
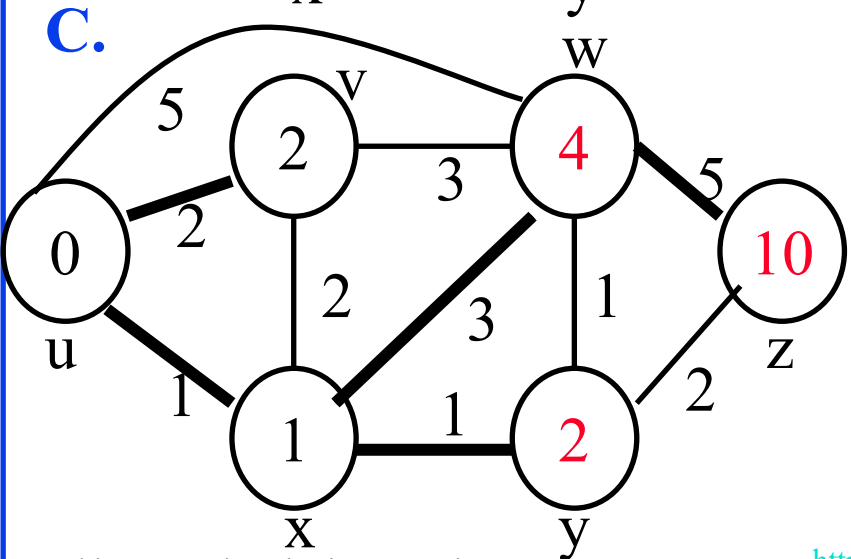
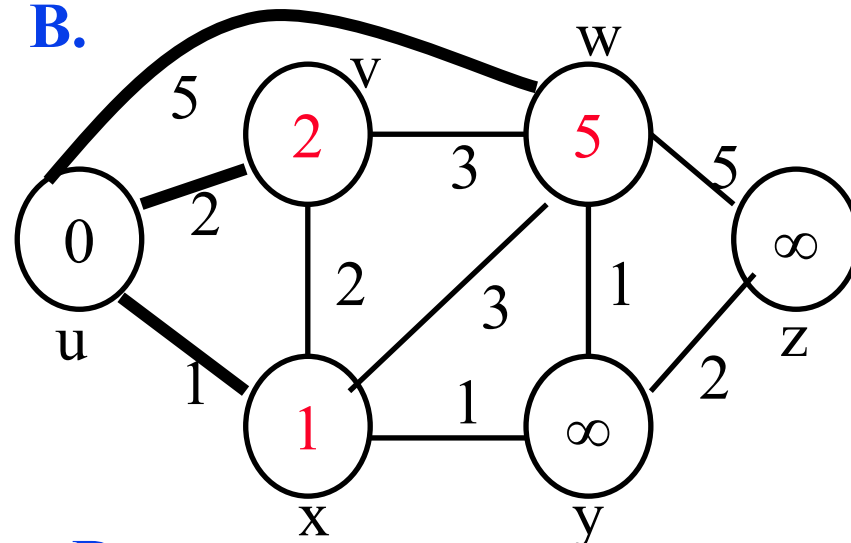
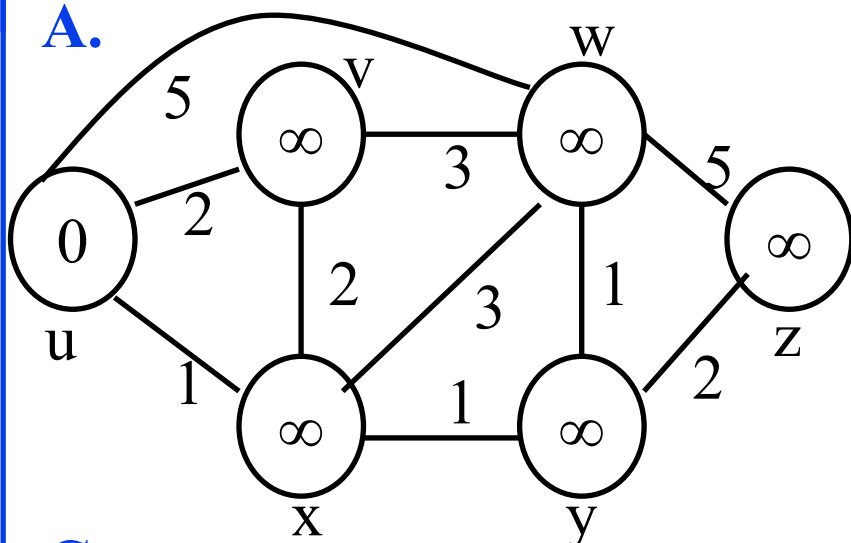


## Student Questions

Can we go through the example again, please?

Sure

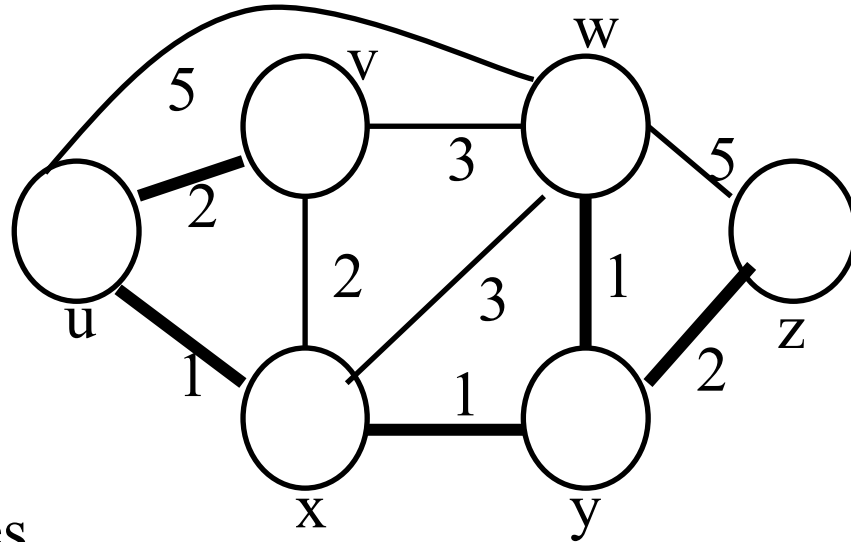
# Bellman-Ford Example 2



## Student Questions

❖ Can you reclarify this example?  
*Sure.*

# Bellman-Ford: Tabular Method



If cost changes  
 $\Rightarrow$  Recompute the costs to all neighbors

h	D(v)	Path	D(w)	Path	D(x)	Path	D(y)	Path	D(z)	Path
0	$\infty$	-	$\infty$	-	$\infty$	-	$\infty$	-	$\infty$	-
1	2	u-v	5	u-w	1	u-x	$\infty$	-	$\infty$	-
2	2	u-v	4	u-x-w	1	u-x	2	u-x-y	10	u-w-z
3	2	u-v	3	u-x-y-w	1	u-x	2	u-x-y	4	u-x-y-z
4	2	u-v	3	u-x-y-w	1	u-x	2	u-x-y	4	u-x-y-z

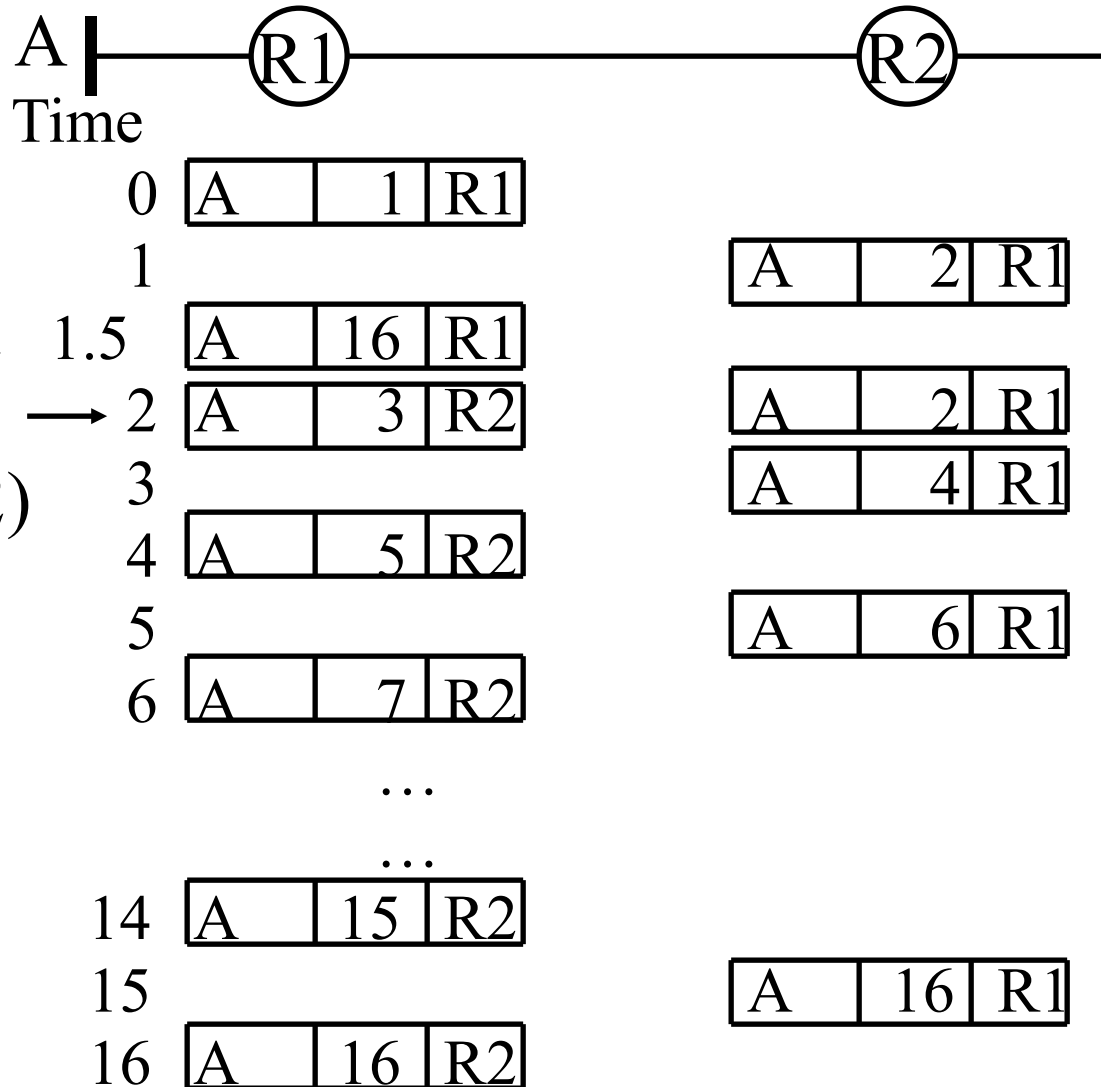
## Student Questions

- In the last iteration of the Bellman-Ford Algorithm, there is no update on the distance; what is this last iteration do?  
*Verifies that there is no update.*

---

- For what cases is Bellman-Ford used, and for what cases is Dijkstra's better?  
*Next slide.*

# Counting to Infinity Problem



R1 loses A  
R1 hears from R2  
(Before it tells R2)

## Student Questions

- How would routers combat a counting-to-infinity problem?

*Using Link-State routing algorithms.*

- Is this referring to the Bellman-Ford Algorithm?

*Yes*

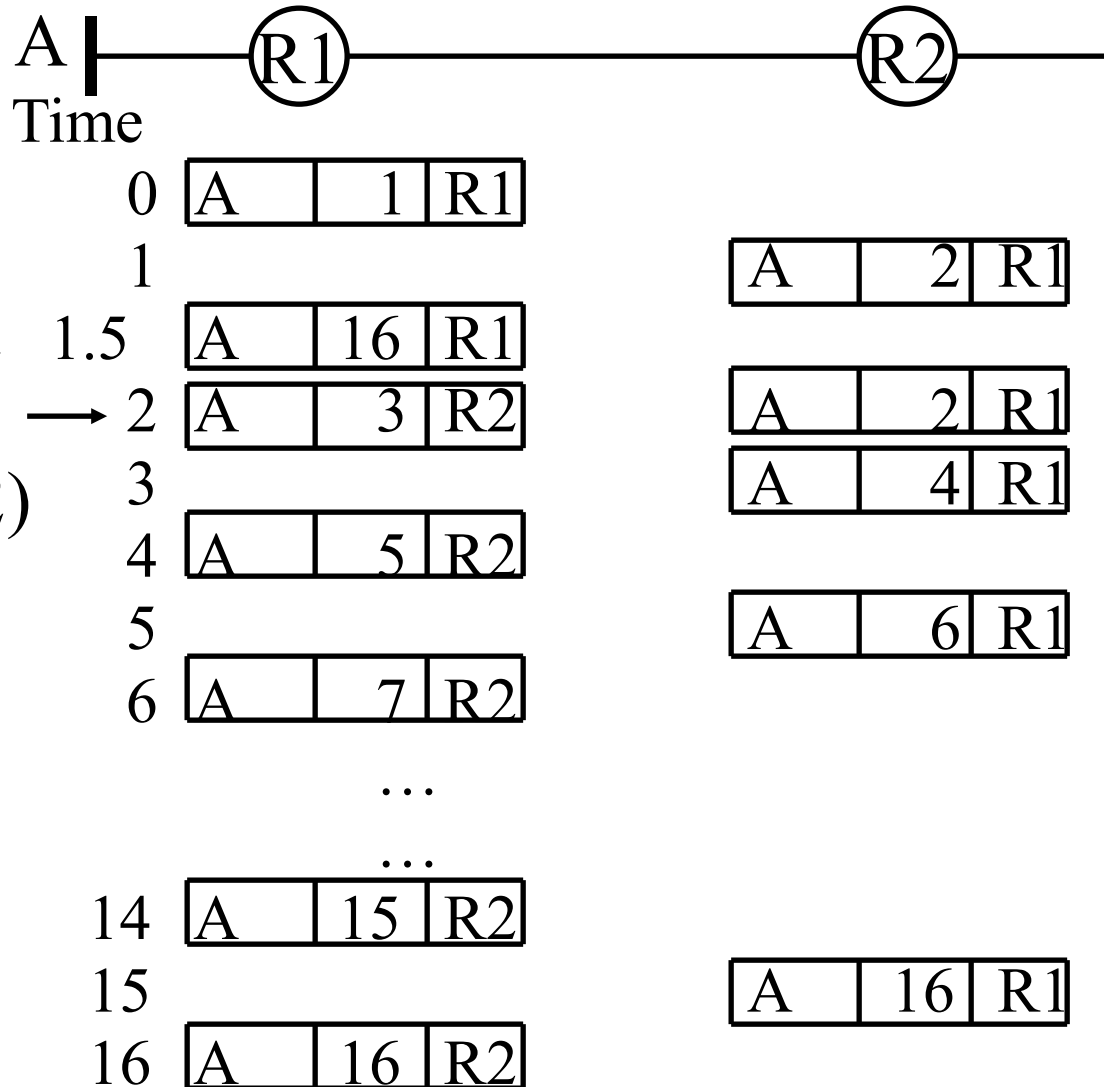
- Didn't R2 know the cost to A when it sent its cost to A to R1? If so, why couldn't R2 figure out the final state of the network?

*R2's cost to A is 2 at that time. It sends that to R1. This example is highly simplified. Actual cycles may be pretty big.*

- How could we prevent counting to infinity?

*Use link-state algorithms.*

# Counting to Infinity Problem



R1 loses A  
R1 hears from R2  
(Before it tells R2)

## Student Questions

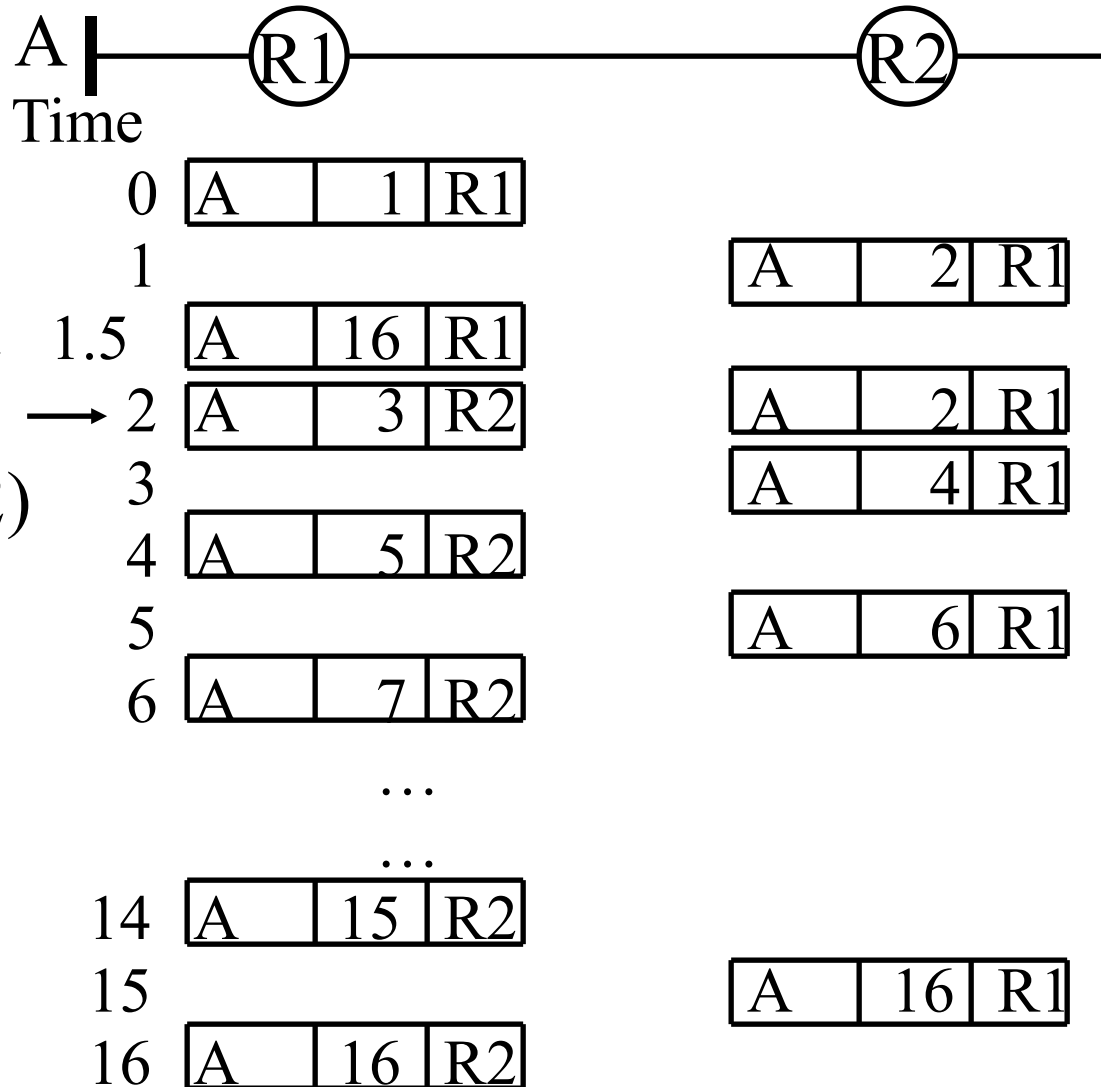
❖ Can we go over what is counting to infinity again?

*Sure.*

❖ Why is the counting to infinity case a problem? Isn't it technically true that the cost to the "lost" router is infinity, since there is no longer a valid path to it.

*It takes a long time for all routers to know that the cost is infinity.*

# Counting to Infinity Problem

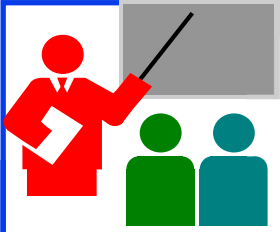


R1 loses A  
 R1 hears from R2  
 (Before it tells R2)

## Student Questions

❖ The textbook said the link state routing algorithm has better robustness because the DV algorithm propagates the error of one node to every other node in the network, whereas LS only influences its neighbor. Could you give a concrete example?

*This refers to the “Counting to Infinity” problem.*



# Routing Algorithms: Summary

1. Distance Vectors: Distance to all nodes in the network sent to neighbors. Small # of large messages.
2. Link State: Cost of link to neighbors sent to the entire network. Large # of small messages.
3. Dijkstra's algorithm is used to compute the shortest path using the link state
4. Bellman Ford's algorithm is used to compute shortest paths using distance vectors
5. Distance Vector algorithms suffer from the count-to-infinity problem

## Student Questions

- ❑ Could you explain again the meaning of link states algorithms sending "Large # of small messages" and distance vector algorithms sending "Small # of large messages?"  
*-Link state tables consist of the cost of each link connected to that router. The size is small. But it has to be broadcast to the entire network.*  
*-Distance vectors consist of distances to all nodes in the network. The size can be huge. But it has to be sent only to neighbors.*

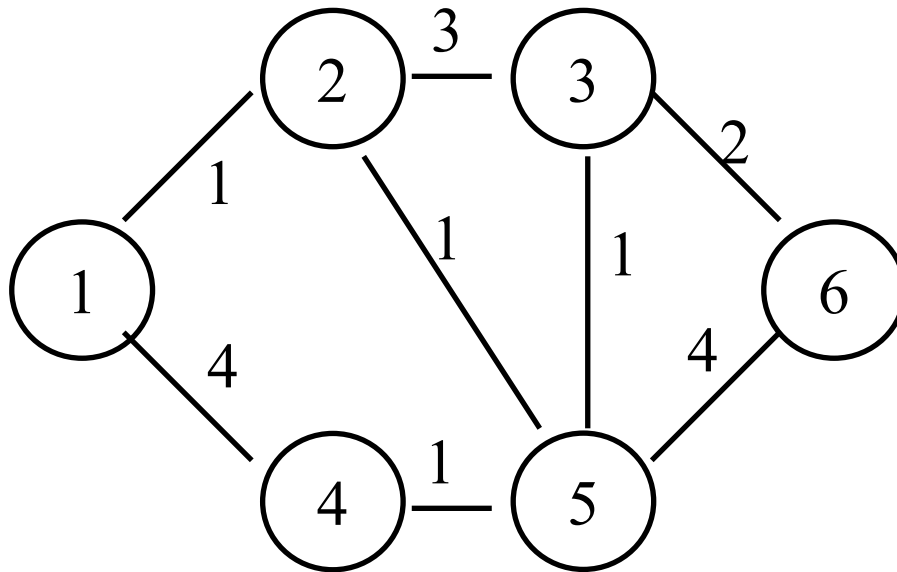
- ❑ Why does the distance vector suffer from counting to infinity while the link state does not?  
*In the link state, broken links are immediately announced to the world.*
- ❑ What is the difference between the routing algorithm and Dijkstra's algorithm?  
*Dijkstra's algorithm is an example of a routing algorithm.*

Ref: Read Section 5.2 of the textbook and try review questions R3-R6.



# Homework 5B

[10 points] Prepare the routing calculation *table* for node 1 in the following network using the Bellman-Ford Algorithm. Explain how you computed new entries in each row.



## Student Questions

- Do we also need explanations in words or just our equations?

Yes

- Should I have five rows in total?

Yes



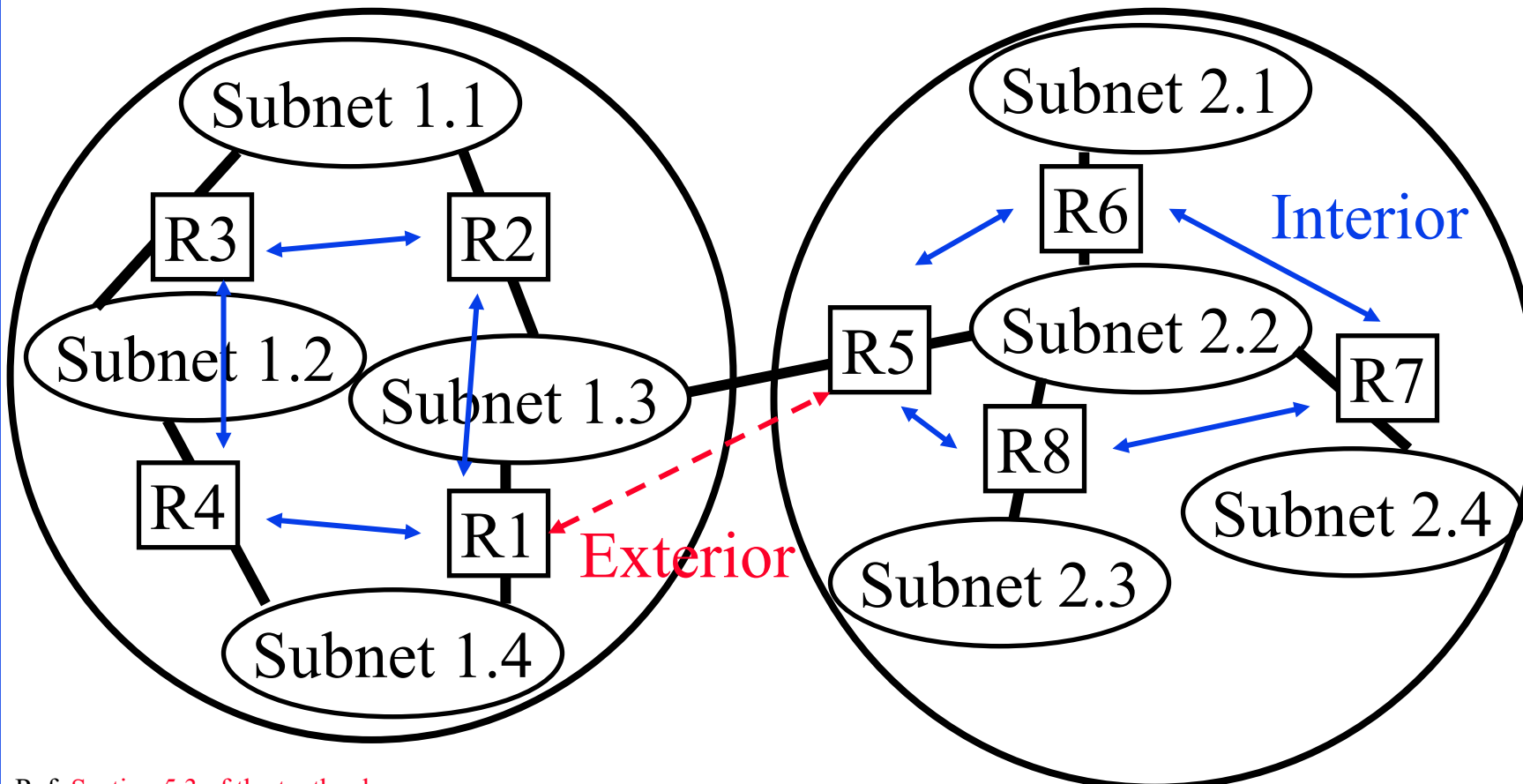
# Routing Protocols

1. Autonomous Systems (AS)
2. Open Shortest Path First (OSPF)
  - OSPF Areas
3. Border Gateway Protocol (BGP)

## Student Questions

# Autonomous Systems

- ❑ An internet connected by homogeneous routers under the administrative control of a single entity



## Student Questions

- ❑ Is an Autonomous System just an area owned by an ISP?

*An enterprise or an ISP can own an autonomous system. For example, WUSTL.edu could be one autonomous system. WUSTL is not an ISP. It is an enterprise customer. Actually, WUSTL.edu consists of at least two autonomous systems: Med school and Danforth.*

- ❑ Why are all the subnets in diagram 1.2?

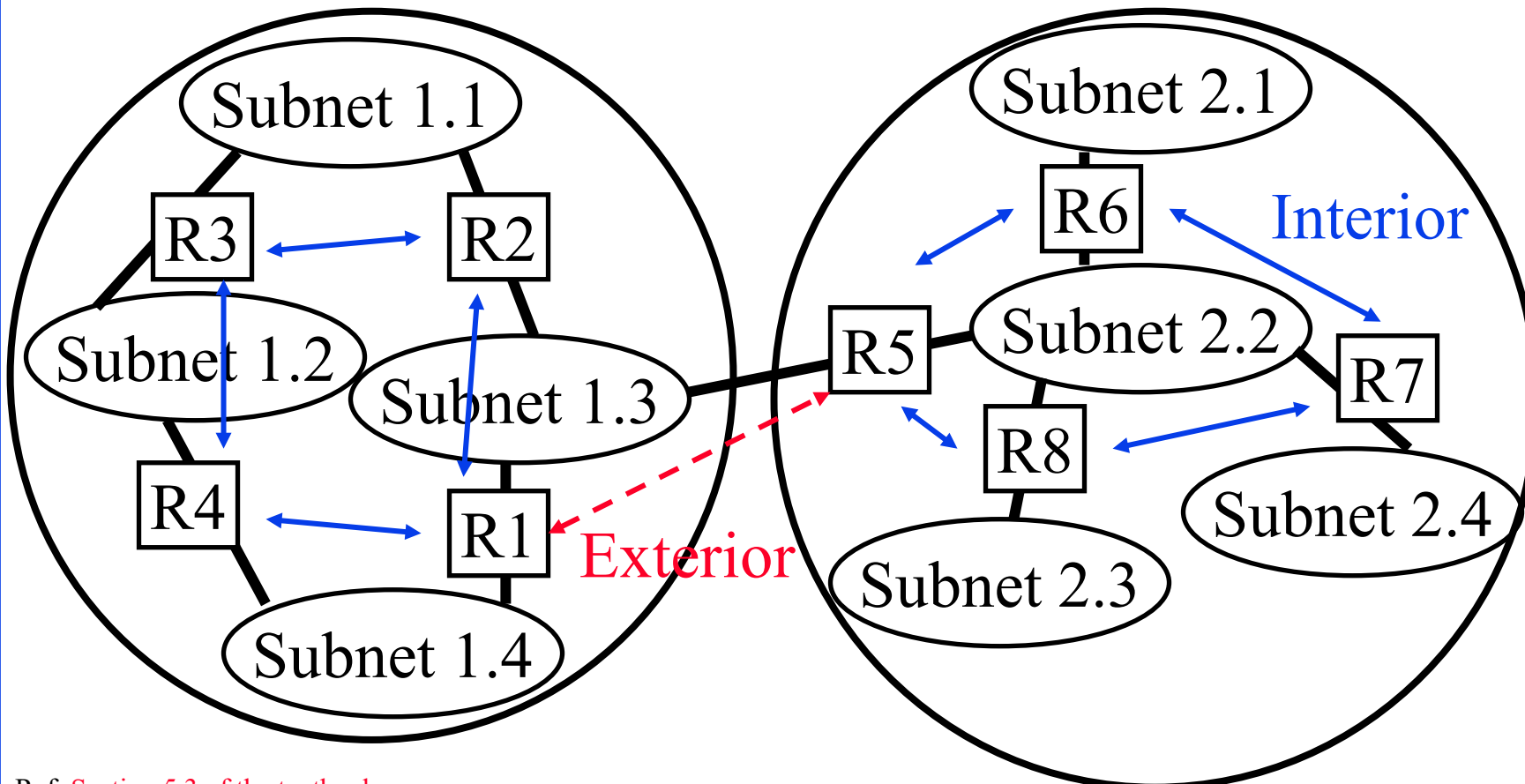
*Error corrected. Thank you.*

- ❑ How do we know which one is interior and which one is exterior?
- Network administrators know.*

- ❑ Do Autonomous Systems include access points/Wi-Fi extenders?
- Access points and extenders are layer-2 devices.*

# Autonomous Systems

- ❑ An internet connected by homogeneous routers under the administrative control of a single entity



## Student Questions

- ❑ Are the routers between networks, such as WashU's or my home network, also part of an AS? Are there any routers that are not part of an AS?  
*Some routers belong to carrier AS and others to enterprise AS.*
- ❑ What are some of the current and emerging applications of autonomous systems in network management and security, and how do these applications improve network performance?  
*ASs are used for routing.*
- ❑ What do you mean by a homogeneous router? Are the routers sharing the same IP address?

Ref: Section 5.3 of the textbook.

Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse473-23/>

©2023 Raj Jain

# Routing Protocols

- ❑ Interior Router Protocol (IRP): Used for passing routing information among routers internal to an autonomous system. Also known as IGP.
  - Examples: RIP, OSPF, IGRP
- ❑ Exterior Router Protocol (ERP): Used for passing routing information among routers between autonomous systems. Also known as EGP.
  - Examples: EGP, BGP, IDRP
  - Note: EGP is a class as well as an instance in that class.

## Student Questions

- ❑ Do we combine IRP and ERP for a real-world transmission? How to switch between two protocols?  
*OSPF for interior. BGP for exterior.*
- ❑ Do the class EGP and instance EGP stand for the same thing?  
*No. One member of EGP class is EGP.*
- ❑ The book didn't really mention anything about EGP in the instance. What is the difference between EGP and BGP?  
*BGP is an EGP.*

# Open Shortest Path First (OSPF)

- ❑ Uses true metrics (not just hop count)
- ❑ Uses subnet masks
- ❑ Allows load balancing across equal-cost paths
- ❑ Supports type of service (ToS)
- ❑ Allows external routes (routes learned from other autonomous systems)
- ❑ Authenticates route exchanges
- ❑ Quick convergence
- ❑ Direct support for multicast
- ❑ Link state routing  $\Rightarrow$  Each router broadcasts its connectivity with neighbors to the entire network

## Student Questions

- ❑ What do you mean by saying using true metrics? The *Hop count does not reflect the cost. Some hops are more expensive than others hops. True metrics would reflect actual costs.*

- ❑ Doesn't IP-anycast violate the rule that computers need to have different IP addresses?

*-Anycast means "to anyone" in the set. For example, anycasting a question "What time is it?" to students in this class will result in a response from any one of the students. One response is sufficient in this case.*

*-Multicast means to "everyone" in the set. For example, "please submit your questions by midnight" needs to be multicast. Anycast will not work. Individual IPs will be too much work.*

- ❑ What is meant by external routes?

*Routes learned from other autonomous systems*

# Open Shortest Path First (OSPF)

- ❑ Uses true metrics (not just hop count)
- ❑ Uses subnet masks
- ❑ Allows load balancing across equal-cost paths
- ❑ Supports type of service (ToS)
- ❑ Allows external routes (routes learned from other autonomous systems)
- ❑ Authenticates route exchanges
- ❑ Quick convergence
- ❑ Direct support for multicast
- ❑ Link state routing  $\Rightarrow$  Each router broadcasts its connectivity with neighbors to the entire network

## Student Questions

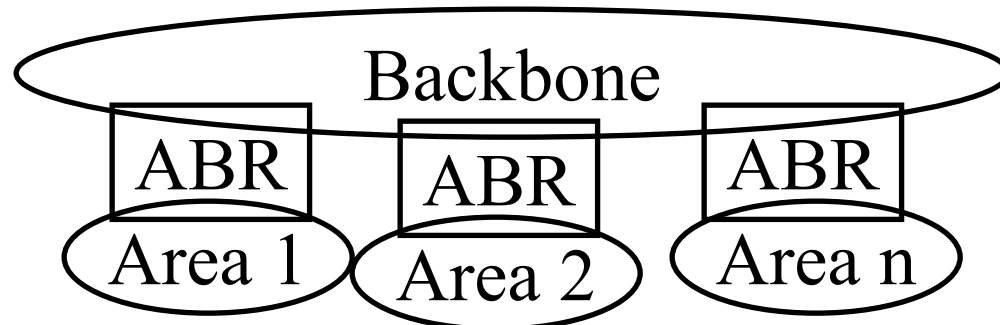
- ❖ Why does OSPF include external routes? Doesn't it work for routing within an AS?

*Tells how to get out of AS.*

- ❖ The textbook says that OSPF uses the link weights, so if the administrator wanted, they could set all link costs to 1 to have minimum-hop routing, while the lecture says that it uses true metrics such as load and speed. Which is correct?

*Admin can set weights.*

# OSPF Areas



- ❑ Large networks are divided into areas to reduce routing traffic.
- ❑ Link-State Advertisements (LSAs) are flooded throughout the area.
- ❑ Area border routers (ABRs) summarize the topology and transmit it to the backbone area.
- ❑ Backbone routers forward it to other areas
- ❑ ABRs connect an area with the backbone area.  
ABRs contain OSPF data for all backbone areas.
- ❑ If there is only one area in the AS, there is no backbone area, and there are no ABRs.

## Student Questions

- ❑ What is LSA?

### *Link State Advertisements*

- ❑ How does the flooding of link states in a network affect network congestion?

### *Not much*

- ❑ When saying, ABR contains data for two areas, which two areas' data does it contain? Are two areas chosen randomly?

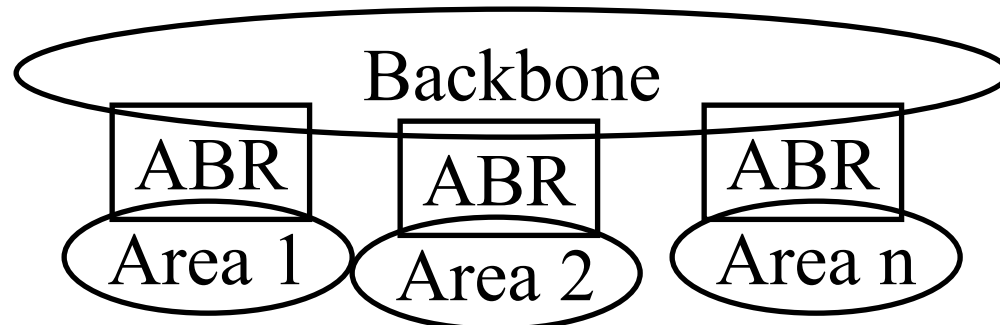
### *The ABR connects the two areas.*

- ❖ Why does ABR has OSPF areas?  
Which two area's information does it have?

### *See above.*



# OSPF Areas



- ❑ Large networks are divided into areas to reduce routing traffic.
- ❑ LSAs are flooded throughout the area.
- ❑ Area border routers (ABRs) summarize the topology and transmit it to the backbone area.
- ❑ Backbone routers forward it to other areas
- ❑ ABRs connect an area with the backbone area.  
ABRs contain OSPF data for all backbone areas.
- ❑ If there is only one area in the AS, there is no backbone area, and there are no ABRs.

## Student Questions

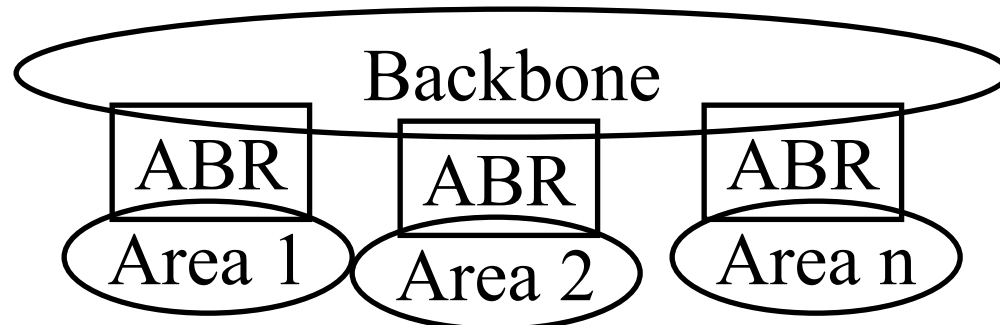
- ❑ Does this picture show one autonomous system? Does the AS include the ABR and backbone?

*Yes.*

- ❑ So, can one ABR have information about two other ABRs?

*ABR contains OSPF data for all backbone areas.*

# OSPF Areas



- ❑ Large networks are divided into areas to reduce routing traffic.
- ❑ LSAs are flooded throughout the area.
- ❑ Area border routers (ABRs) summarize the topology and transmit it to the backbone area.
- ❑ Backbone routers forward it to other areas
- ❑ ABRs connect an area with the backbone area.  
ABRs contain OSPF data for **all areas**.
- ❑ If there is only one area in the AS, there is no backbone area, and there are no ABRs.

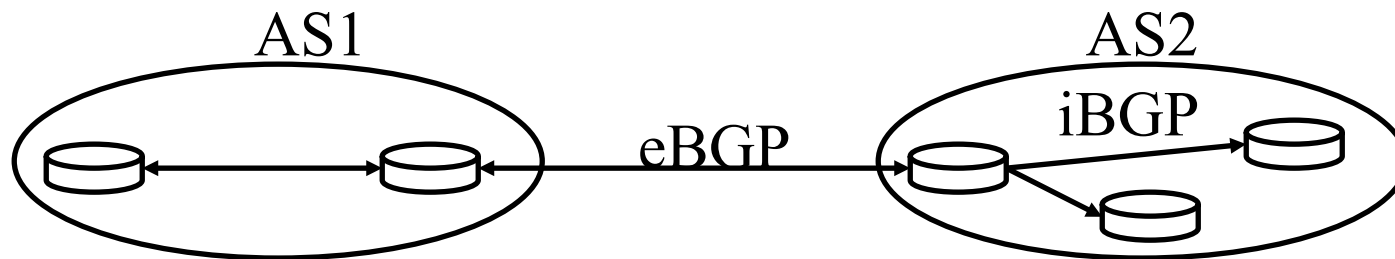
## Student Questions

- ❖ The lecture slide says "All backbone areas" instead of "the backbone area," implying there can be more than one. Is this correct?

*Changed to "all areas"*

# Border Gateway Protocol

- ❑ Inter-autonomous system protocol [RFC 1267]
- ❑ Used since 1989 but not extensively until recently
- ❑ Runs on TCP (segmentation, reliable transmission)
- ❑ Advertises all transit ASs on the path to a destination address
- ❑ A router may receive multiple paths to a destination  $\Rightarrow$  Can choose the best path
- ❑ iBGP is used to forward paths between two peers in the same AS. eBGP is used to exchange paths between ASs.



Ref: Section 5.4 of the textbook.

Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse473-23/>

©2023 Raj Jain

5.25a

## Student Questions

- ❑ What's the difference between iBGP and OSPF since they both work in AS?

*If BGP is between peers of an AS, it is called iBGP. The two peers may be separated by other ASs.*

- ❑ If the path is handled by OSPF, what is the purpose of iBGP?

*See above.*

- ❑ Why does interior BGP exist if there is OSPF?

*See above.*

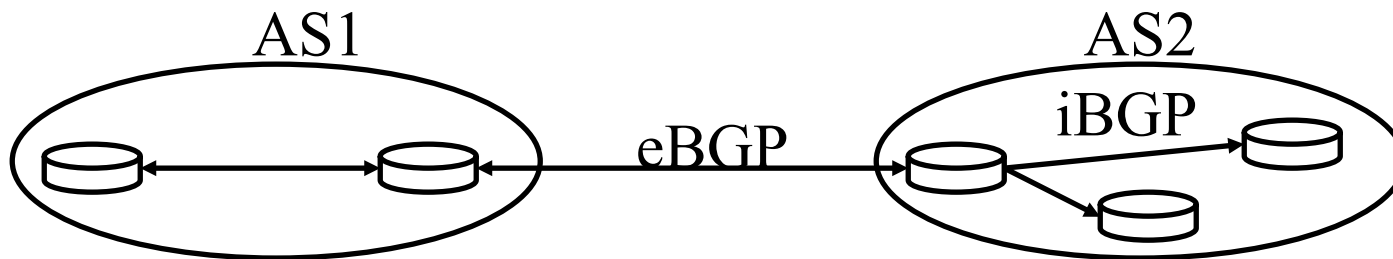
- ❖ *Why is iBGP needed? Why isn't OSPF used until we hit a gateway router for the AS. (4 other variations of this question)*

*See the first question above.*

Next Q 5.25b

# Border Gateway Protocol

- ❑ Inter-autonomous system protocol [RFC 1267]
- ❑ Used since 1989 but not extensively until recently
- ❑ Runs on TCP (segmentation, reliable transmission)
- ❑ Advertises all transit ASs on the path to a destination address
- ❑ A router may receive multiple paths to a destination  $\Rightarrow$  Can choose the best path
- ❑ iBGP is used to forward paths between two peers in the same AS. eBGP is used to exchange paths between ASs.



Ref: Section 5.4 of the textbook.

Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse473-23/>

©2023 Raj Jain

## Student Questions

- ❖ The textbook mentioned that BGP uses a TCP connection to communicate with edge routers belonging to other AS. I previously thought routers did not have layer four.

*They do not change headers of higher layers for datagrams being forwarded. However, they use all layers for internal operation and management.*

# BGP Operations

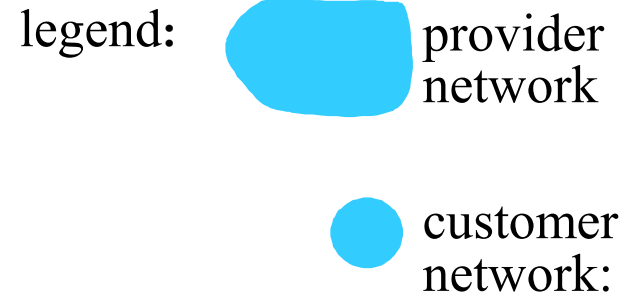
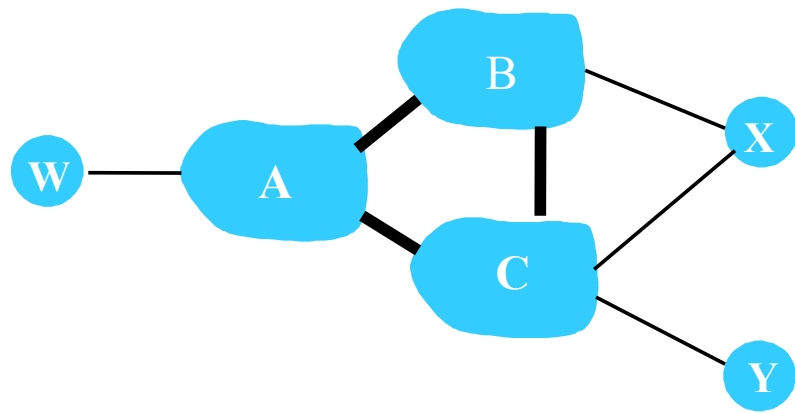
- ❑ BGP systems initially exchange entire routing tables. Afterward, only updates are exchanged.
- ❑ BGP messages have the following information:
  - Origin of path information: RIP, OSPF, ...
  - AS\_Path: List of ASs on the path to reach the dest
  - Next\_Hop: IP address of the border router to be used as the next hop to reach the dest
  - Unreachable: If a previously advertised route has become unreachable
- ❑ BGP speakers generate update messages to all peers when it selects a new route, or some route becomes unreachable.

## Student Questions

- ❑ When BGP hops between ABRs, do the datagrams enter the AS or remain on the exterior? If BGP only sends updates to neighbors, is it also susceptible to a count-to-infinity problem?

*BGP is not a distance vector algorithm. It belongs to another class called “path vector” algorithms. It does not have a count-to-infinity problem.*

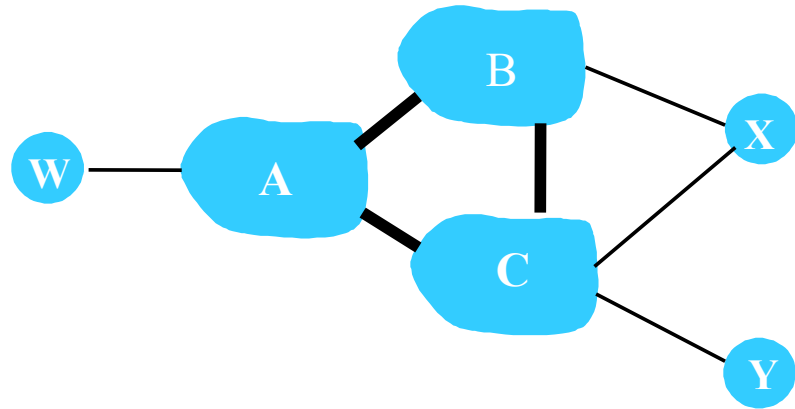
# BGP Routing Policy Example





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

## Student Questions

# BGP Routing Policy Example (Cont)



legend:  provider network  
 customer network:

- ❑ A advertises path A-W to B
- ❑ B advertises path B-A-W to X
- ❑ Should B advertise path B-A-W to C?
  - No way! B gets no “revenue” for routing C-B-A-W since neither W nor C are B’s customers
  - B wants to force C to route to w via A
  - B wants to route *only* to/from its customers!

## Student Questions

- ❑ What is the relationship between the routing protocols and the path algorithms like Dijkstra's and Bellman Ford's? Is one used by the other?

*Protocols use algorithms.*

# Intra- vs. Inter-AS Routing

## ❑ Policy:

- Inter-AS: admin wants control over how its traffic is routed and who routes through its net.
- Intra-AS: single admin, so no policy decisions are needed

## ❑ Scale:

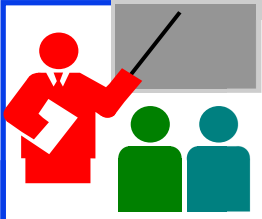
- Hierarchical routing saves table size, reduced update traffic

## ❑ Performance:

- Intra-AS: can focus on performance
- Inter-AS: policy may dominate over performance

## Student Questions





# Routing Protocols: Summary

1. OSPF uses link-state routing and divides the autonomous systems into multiple areas.  
Area border router, AS boundary router
2. BGP is an inter-AS protocol  $\Rightarrow$  Policy driven

## Student Questions

- Can you point in the graph of 5.21 about the three kinds of routers again?

*Designated routers were not shown.  
Used when there are multiple  
routers on a single Local Area  
Network.*



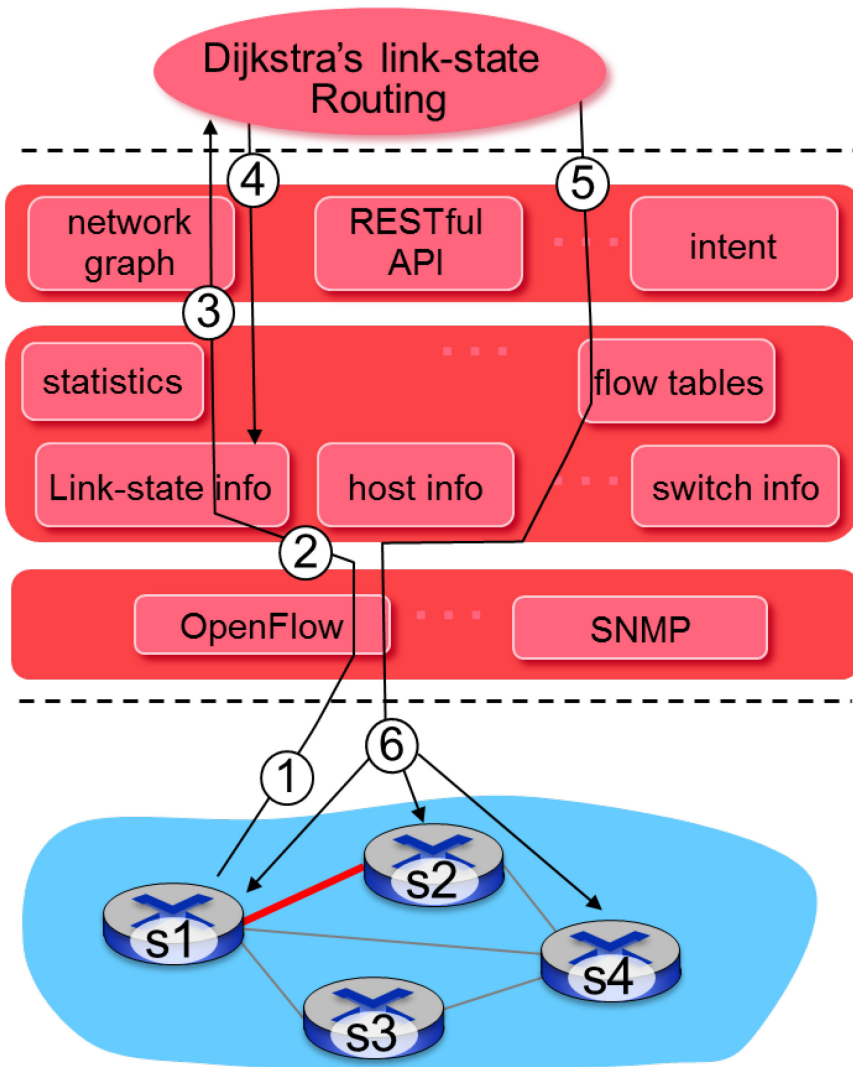
Ref: Read Section 5.3 and 5.4 of the textbook and try review questions R7-R13.

Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse473-23/>

©2023 Raj Jain

# SDN Control Plane

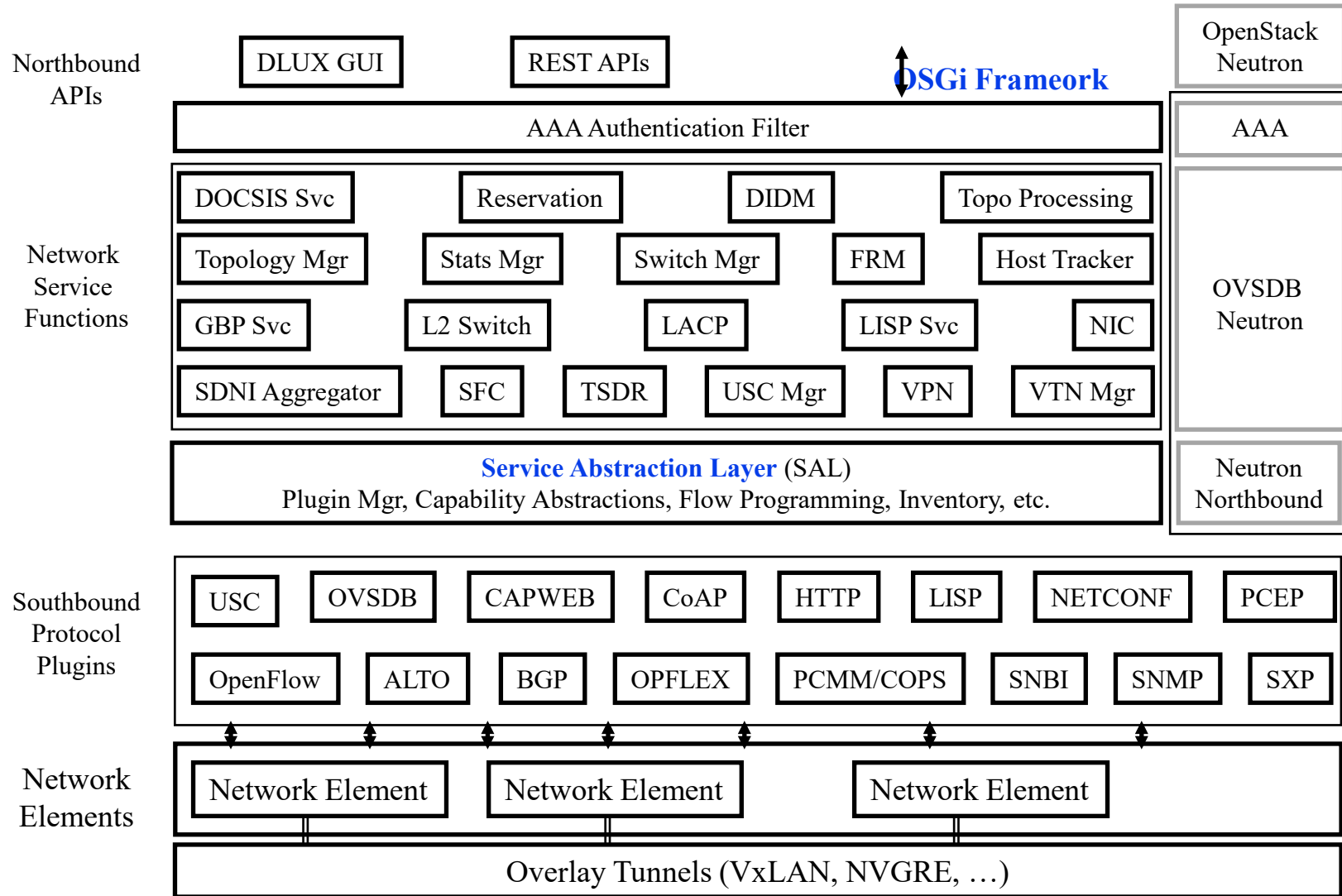


- ① S1, experiencing link failure using OpenFlow port status message to notify controller
- ② SDN controller receives OpenFlow message, updates link status info
- ③ Dijkstra's routing algorithm application has previously registered to be called when ever link status changes. It is called.
- ④ Dijkstra's routing algorithm access network graph info, link state info in controller, computes new routes

## Student Questions

- Are any special techniques used to optimize Dijkstra's performance in practice?  
*Yes. But beyond the scope of this course.*

# Controller Example: OpenDaylight



## Student Questions

- Is the SAL responsible for translating Northbound APIs into Southbound Protocol Plugins?

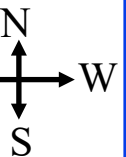
*It is responsible for translating and submitting network service function requests to southbound protocol plugins. Also, it is responsible for translating and submitting responses from southbound protocol plugins to network service functions.*

- Why are so many protocols needed? Do they all do similar things?

*No, they solve different problems.*

- What exactly are northbound and southbound?

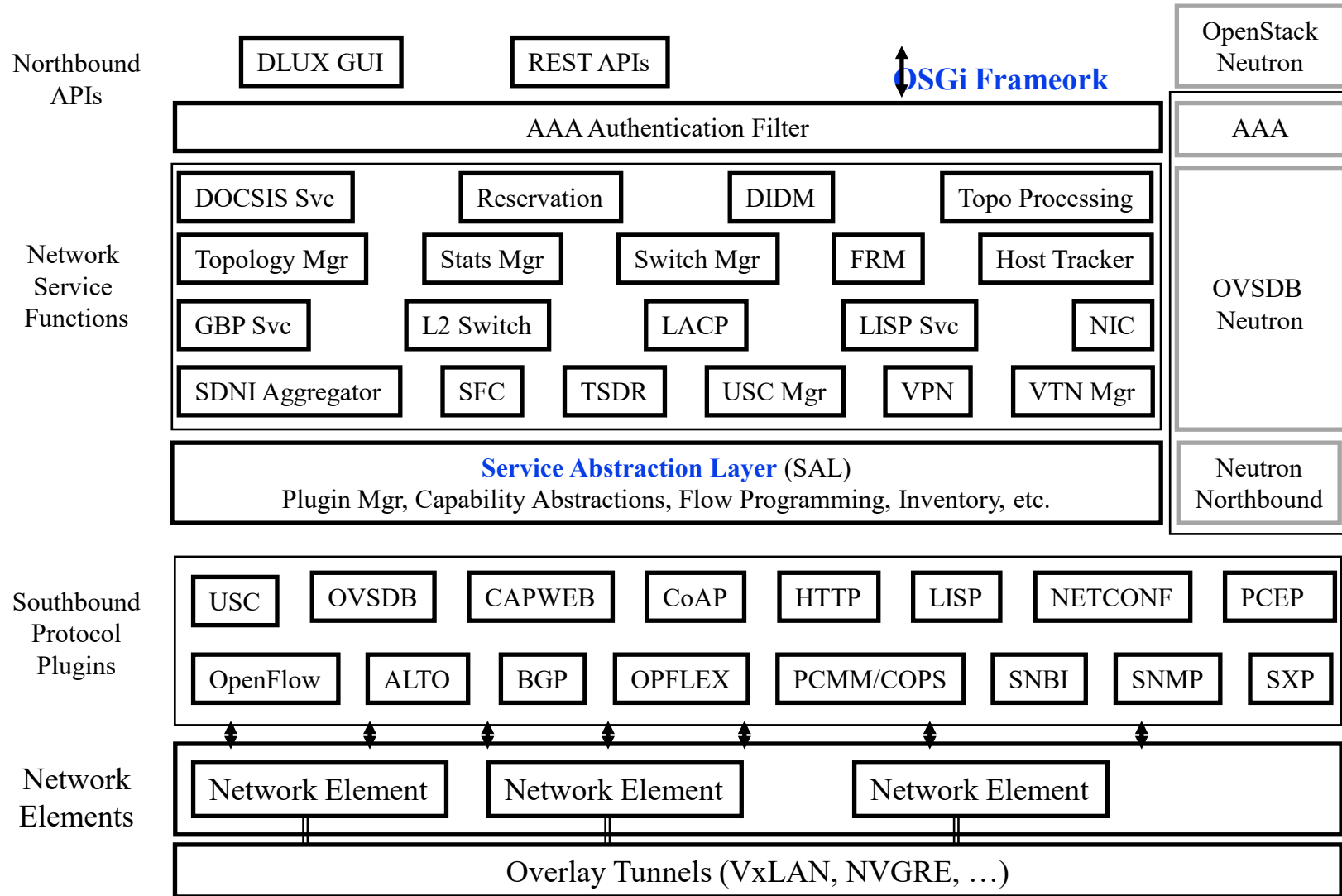
*North is upward on a map. E ← N → W*



- What are network elements?

*Network Hardware*

# Controller Example: OpenDaylight



## Student Questions

- ❖ Is "northbound" equivalent to higher abstraction layers (like applications) and "southbound" equivalent to lower layers (like hardware)?

*Higher = upper = North*

# OpenDaylight SDN Controller

- ❑ Multi-company collaboration under the Linux Foundation
- ❑ Many projects, including OpenDaylight Controller
- ❑ Dynamically linked into a Service Abstraction Layer (SAL)  
⇒ SAL figures out how to fulfill the service requested by higher layers irrespective of the southbound protocol.
- ❑ Modular design
- ❑ A rich set of North-bound APIs via **RESTful** (Web page-like) services

## Student Questions

❖ Can you explain SAL a little more?

*Sure*

❖ What exactly does “RESTful” mean?

*Representational State Transfer = State-less like Web*

Ref: **Read Section 5.5** and try review questions R14-R18.

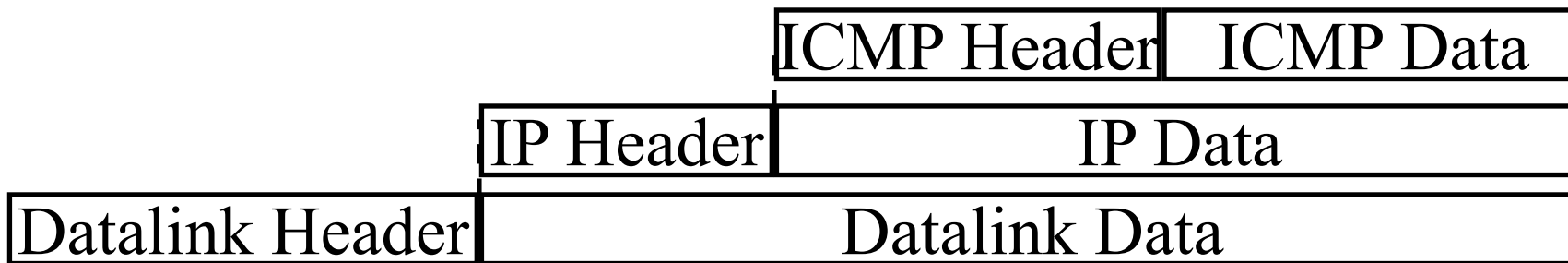
Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse473-23/>

©2023 Raj Jain

# ICMP

- ❑ Internet Control Message Protocol
- ❑ Required companion to IP. Provides feedback from the network.
- ❑ ICMP: Used by IP to send error and control messages
- ❑ ICMP uses IP to send its messages (Not UDP)
- ❑ ICMP does not report errors on ICMP messages.
- ❑ ICMP reports error only on the first fragment



## Student Questions

❑ Can we say ICMP is a layer 3.5 protocol?  
*Not really. It is a component of the Layer 3 protocol. IP cannot run without ICMP. Generally, each layer needs its management and security protocols. Sometimes these are built-in. In other cases, separate and many different standards (protocols) exist.*

❑ What if an error starts to occur on the second fragment? Is this possible? If so, how will ICMP handle this, then?

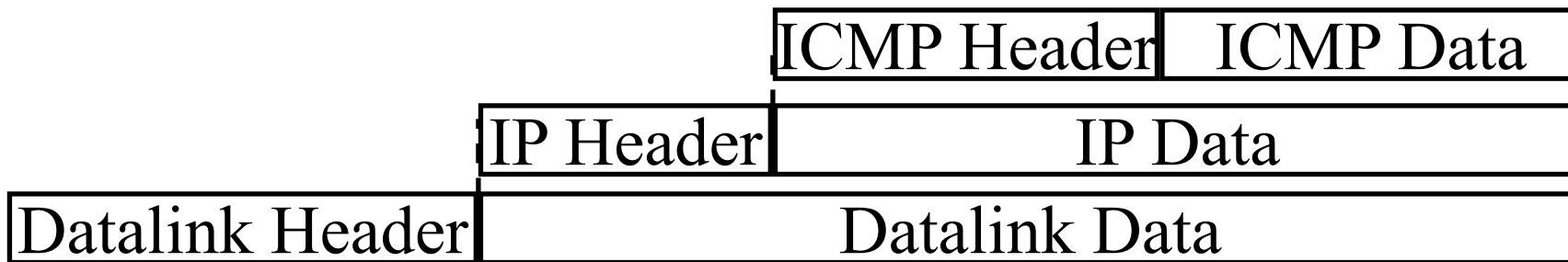
*Those errors are handled by TCP/UDP checksum.*

❑ Can you explain why an error is only reported on the first fragment?

*The first fragment contains the original IP header.*

# ICMP

- ❑ Internet Control Message Protocol
- ❑ Required companion to IP. Provides feedback from the network.
- ❑ ICMP: Used by IP to send error and control messages
- ❑ ICMP uses IP to send its messages (Not UDP)
- ❑ ICMP does not report errors on ICMP messages.
- ❑ ICMP reports error only on the first fragment



## Student Questions

- ❖ The book, page 423, says that ICMP is commonly used for error reporting. What motivates using ICMP messages that describe router advertisement and discovery? Could OpenFlow implement the same functionality that ICMP does with sending messages?

*OpenFlow works in one management domain. ICMP is worldwide.*

- ❖ Is everything after the "IP header" an IP datagram payload containing the ICMP header and message?

*Yes.*

# ICMP: Message Types

IP Header	
Type of Message	8b
Error Code	8b
Checksum	16b
Parameters, if any	Var
Information	Var

Type	Message
0	Echo reply
3	Destination unreachable
4	Source quench
5	Redirect
8	Echo request
11	Time exceeded
12	Parameter unintelligible
13	Time-stamp request
14	Time-stamp reply
15	Information request
16	Information reply
17	Address mask request
18	Address mask reply

## Student Questions

- Can you explain more about the error packet?

*ICMP Messages have "Type," which indicates what that message is for. Not all ICMP messages are "Error messages." The error code field indicates the type of error encountered while processing a datagram.*

- If all necessary information is included in the ICMP header (type of message, error code), what is included in ICMP data?

*ICMP data consists of the IP header and some parts of the IP data.*

- ❖ For ICMP Information request/reply, do we specify the information in the ICMP data field?

*See above.*



# ICMP: Message Types

IP Header	
Type of Message	8b
Error Code	8b
Checksum	16b
Parameters, if any	Var
Information	Var

Type	Message
0	Echo reply
3	Destination unreachable
4	Source quench
5	Redirect
8	Echo request
11	Time exceeded
12	Parameter unintelligible
13	Time-stamp request
14	Time-stamp reply
15	Information request
16	Information reply
17	Address mask request
18	Address mask reply

## Student Questions

- ❖ Should I check both ICMP checksum and IP checksum when I receive an ICMP packet?  
*Yes.*

# ICMP Messages

- ❑ Source Quench: Please slow down!  
I just dropped one of your datagrams.
- ❑ Time Exceeded: Time to live field in one of your packets became zero.” or “Reassembly timer expired at the destination.
- ❑ Fragmentation Required: Datagram was longer than MTU, and “No Fragment bit” was set.
- ❑ Address Mask Request/Reply: What is the subnet mask on this net? Replied by “Address mask agent”.
- ❑ PING uses ICMP echo
- ❑ Tracert uses TTL expired

## Student Questions

- ❑ What is the type code for fragmentation required? Why does it not appear in the chart in the previous slide?

*The list on the previous slide is partial. For a complete list of possibilities, please see the RFC.*

# Trace Route Example

```
C:\>tracert www.google.com
```

```
Tracing route to www.l.google.com [74.125.93.147]  
over a maximum of 30 hops:
```

```
 1  3 ms  1 ms  1 ms 192.168.0.1  
 2 12 ms 10 ms  9 ms bras4-10.stlsmo.sbcglobal.net [151.164.182.113]  
 3 10 ms  8 ms  8 ms dist2-vlan60.stlsmo.sbcglobal.net [151.164.14.163]  
 4  9 ms  7 ms  7 ms 151.164.93.224  
 5 25 ms 22 ms 22 ms 151.164.93.49  
 6 25 ms 22 ms 22 ms 151.164.251.226  
 7 30 ms 28 ms 28 ms 209.85.254.128  
 8 61 ms 57 ms 58 ms 72.14.236.26  
 9 54 ms 52 ms 51 ms 209.85.254.226  
10 79 ms 160 ms 67 ms 209.85.254.237  
11 66 ms 57 ms 68 ms 64.233.175.14  
12 60 ms 58 ms 58 ms qw-in-f147.google.com [74.125.93.147]
```

```
Trace complete.
```

## Student Questions

- If the route changes during a traceroute, is there any way to know? Is a change prevented? Or is it not necessary?

*Traceroute gives the actual route used for the message. It can change between two traceroutes.*

# Lab 5A: ICMP

- ❑ [14 points] Download the Wireshark traces from <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip>
- ❑ Open *icmp-ethereal-trace-1* in Wireshark. Select **View → Expand All**. Answer the following questions:
  1. Examine Frame 3.
    - A. What is the IP address of your host? What is the IP address of the destination host?
    - B. Why is it that an ICMP packet does not have source and destination port numbers?
    - C. What are the ICMP type and code numbers? What other fields does this ICMP packet have? How many bytes are the checksum, sequence number, and identifier fields?

## Student Questions

## Lab 5A (Cont)

2. Examine Frame 4. What are the ICMP type and code numbers?
  - ❑ Open *icmp-ethereal-trace-2* in Wireshark. Answer the following questions:
3. Examine Frame 2. What fields are included in this ICMP error packet?
4. Examine Frames 100, 101, and 102. How are these packets different from the ICMP error packet 2? Why are they not error packets?

### Student Questions



# Network Management

- ❑ What is Network Management?
- ❑ Components of Network Management
- ❑ How is Network Managed?
- ❑ SNMP protocol

## Student Questions

- ❖ On page 426 of the textbook, the framework for network management is described very in depth. What extent of the definitions for each part of the framework are we expected to know for the exam?

*Everything in the book and slides.*

# What is Network Management?

- ❑ Traffic on Network = Data + Control + Management
- ❑ **Data** = Bytes/Messages sent by users
- ❑ **Control** = Bytes/messages added by the system to properly transfer the data (e.g., routing messages)
- ❑ **Management** = Optional messages to ensure that the network functions properly and to handle the issues arising from the malfunction of any component
- ❑ If all components function properly, control is still required, but management is optional.
- ❑ Examples:
  - Detecting failures of an interface card at a host or a router
  - Monitoring traffic to aid in resource deployment
  - Intrusion Detection

## Student Questions

# Components of Network Management

## 1. Fault Management:

Detect, log, and respond to fault conditions

## 2. Configuration Management:

Track and control which devices are on or off

## 3. Accounting Management:

Monitor resource usage for records and billing

## 4. Performance Management:

Measure, report, analyze, and control traffic, messages

## 5. Security Management:

Enforce a policy for access control, authentication, and authorization

## FCAPS

### Student Questions

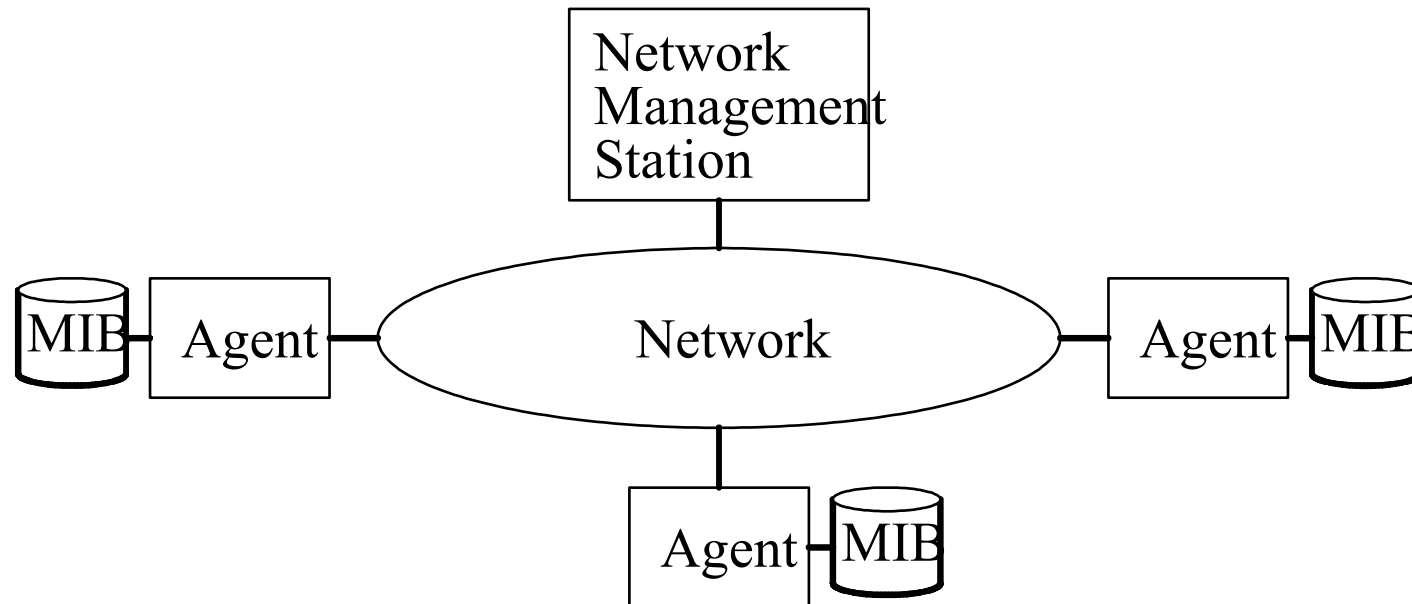
- Can performance measurements also be used for accounting management if performance includes resource usage/reports?

*Yes, any data can be used for multiple purposes.*



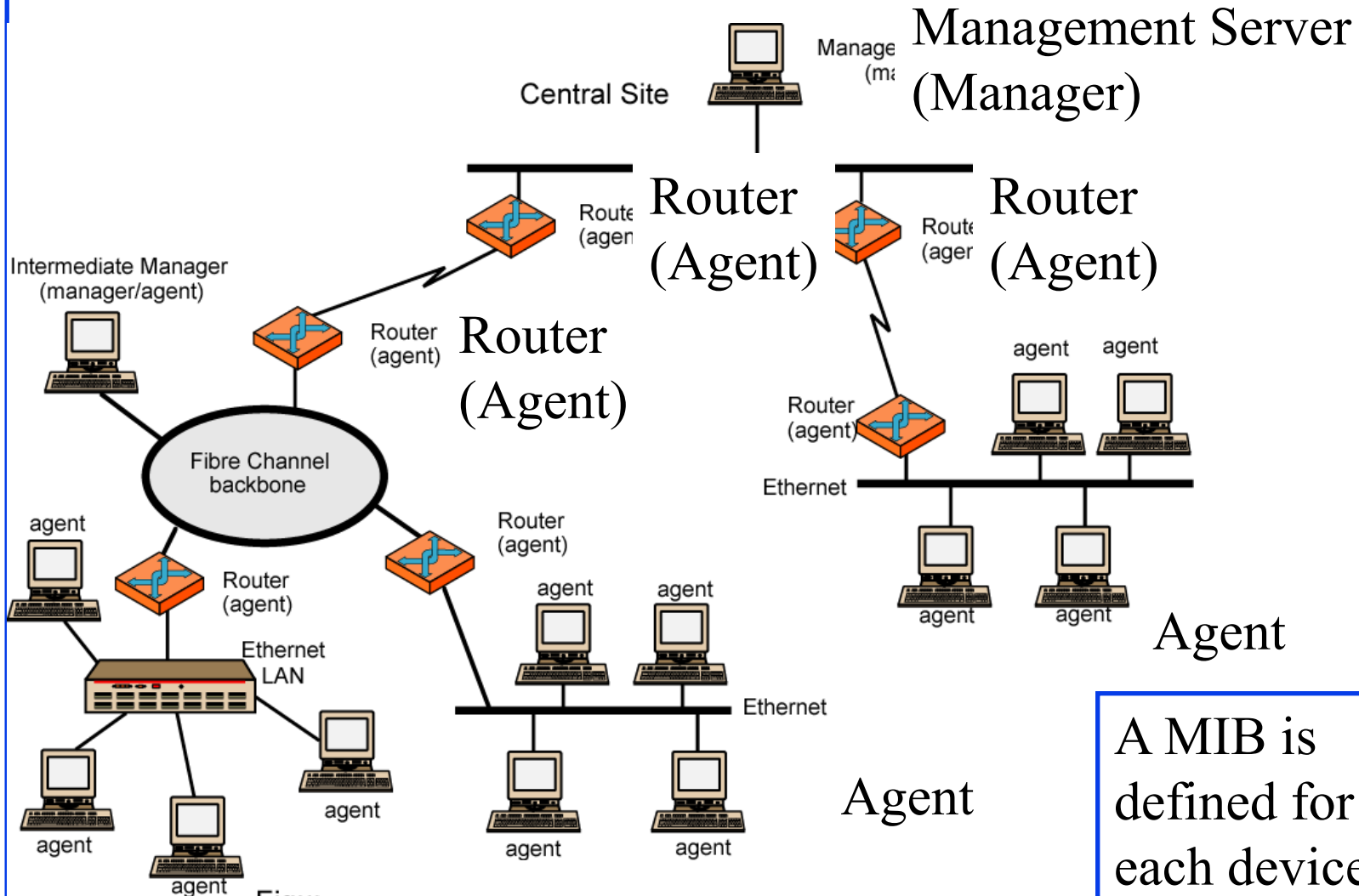
# How is Network Managed?

- ❑ Management = Initialization, Monitoring, Control
- ❑ Manager, Agents, and Management Information Base (MIB)



## Student Questions

# Example of Network Management



Figure

A MIB is defined for each device

## Student Questions

# SNMP

- ❑ Based on Simple Gateway Management Protocol (SGMP) – RFC 1028 – Nov 1987
- ❑ SNMP = **S**imply **N**ot **M**y **P**roblem [Marshall Rose]  
*Simple* Network Management Protocol
- ❑ RFC 1058, April 1988
- ❑ Only Five commands

## Command

## Meaning

get-request

Fetch a value

get-next-request

Fetch the next value (in a tree)

get-response

Reply to a fetch operation

set-request

Store a value

trap

An event

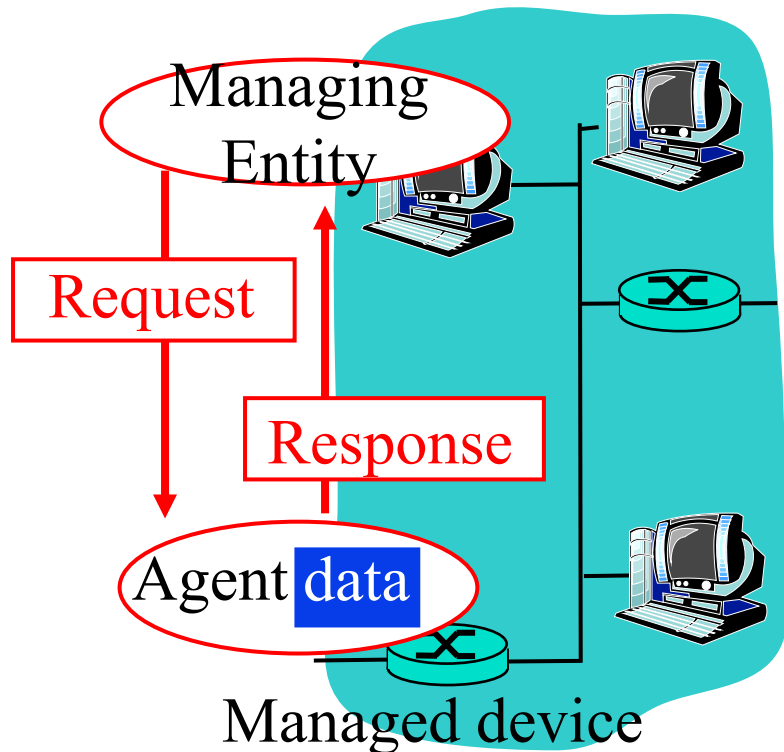
## Student Questions

- ❑ What is the meaning of value here? What do they represent?

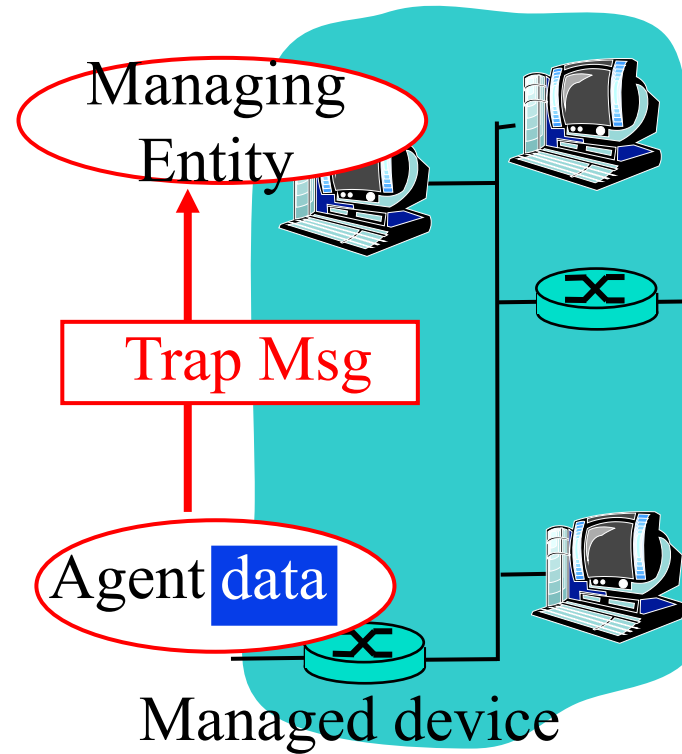
*Generally, counters and parameters*

# SNMP protocol

Two ways to convey MIB info, commands:



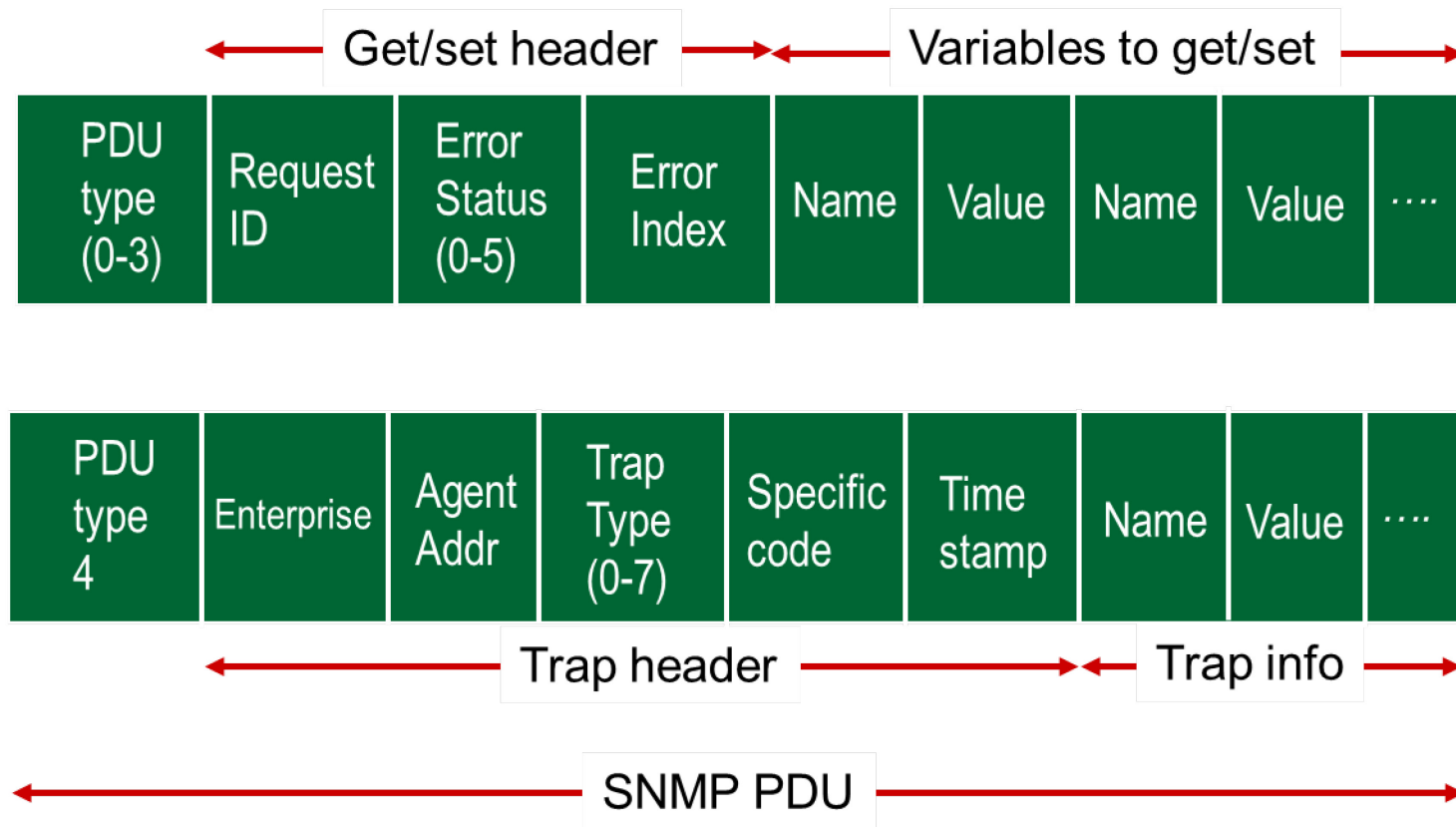
Request/response mode



Trap mode

## Student Questions

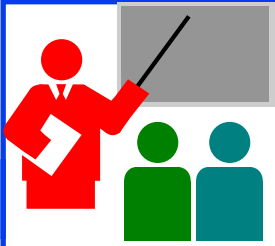
# SNMP Message Formats



## Student Questions

- ❑ How many variables are there to get/set? Is the length standard and padded if unused or dynamically allocated?

Dynamic.



# Network Management: Summary

1. Management = Initialization, Monitoring, and Control
2. Standard MIBs defined for each object
3. SNMP = Only 5 commands in the first version

## Student Questions

*What kinds of fields does a MIB contain?*

Parameters and counters

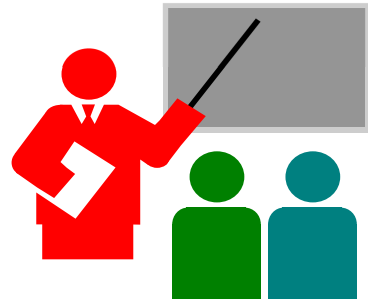
Ref: Read Section 5.7 of the textbook and try review questions R21-R23.

Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse473-23/>

©2023 Raj Jain

# Network Layer Control Plane: Summary



1. Dijkstra's algorithm allows path computation using link state
2. Bellman Ford's algorithm allows path computation using distance vectors.
3. OSPF is a link state IGP.
4. BGP is an EGP and uses path vectors
5. SDN controllers use various algorithms for the centralized computation of paths and other policies
6. ICMP is an IP control protocol used to convey errors
7. SNMP is the simple network management protocol to manage all devices and protocols in a network

## Student Questions

❖ What are the hardware and software advantages and disadvantages between decentralized and centralized routing algorithms?

*Centralization makes management easier.*

# Lab 5B: ICMP Ping Programming

[25 points] In this lab, you will gain a better understanding of Internet Control Message Protocol (ICMP). You will learn to implement a Ping application using ICMP request and reply messages.

Ping is a computer network application used to test whether a particular host is reachable across an IP network. It is also used to self-test the network interface card of the computer or as a latency test. It works by sending ICMP “echo reply” packets to the target host and listening for ICMP “echo reply” replies. The “echo reply” is sometimes called a pong. Ping measures the round-trip time, records packet loss, and prints a statistical summary of the echo reply packets received (the minimum, maximum, and the mean of the round-trip times and in some versions the standard deviation of the mean).

Your task is to develop your own Ping application in Python. Your application will use ICMP but, in order to keep it simple, will not exactly follow the official specification in RFC 1739. Note that you will only need to write the client side of the program, as the functionality needed on the server side is built into almost all operating systems.

You should complete the Ping application so that it sends ping requests to a specified host separated by approximately one second. Each message contains a payload of data that includes a timestamp. After sending each packet, the application waits up to one second to receive a reply. If one second goes by without a reply from the server, then the client assumes that either the ping packet or the pong packet was lost in the network (or that the server is down).

## Student Questions

- ❖ Should I check the ICMP type and code in Lab 5B? Is it enough to only check the ICMP type?

*Code gives a reason for type. It is required for some types, e.g., destination unreachable. You may get the destination unreachable to some echo requests. So yes, you should check both.*



# Lab 5B (Cont)

## Code

Below you will find the skeleton code for the client. You are to complete the skeleton code. The places where you need to fill in code are marked with #Fill in start and #Fill in end. Each place may require one or more lines of code. This code was written for **Python V2.7** and may not run on higher versions.

## Additional Notes

In “receiveOnePing” method, you need to receive the structure ICMP\_ECHO\_REPLY and fetch the information you need, such as checksum, sequence number, time to live (TTL), etc. Study the “sendOnePing” method before trying to complete the “receiveOnePing” method.

You do not need to be concerned about the checksum, as it is already given in the code.

This lab requires the use of raw sockets. In some operating systems, you may need **administrator/root privileges** to be able to run your Pinger program.

## Testing the Pinger

First, test your client by sending packets to localhost, that is, 127.0.0.1.

Then, you should see how your Pinger application communicates across the network by pinging servers in different continents. **See additional hints on slide 5.62.**

## What to Hand in

- ❑ You will hand in the complete client code and screenshots of your Pinger output for four target hosts: north-america.pool.ntp.org, europe.pool.ntp.org, asia.pool.ntp.org, south-america.pool.ntp.org

## Student Questions

# Lab 5B (Cont)

## Skeleton Python Code for the ICMP Pinger

```
from socket import *
import os
import sys
import struct
import time
import select
import binascii
ICMP_ECHO_REQUEST = 8

def checksum(string):
    csum = 0
    countTo = (len(string) // 2) * 2
    count = 0
    while count < countTo:
        thisVal = ord(string[count+1]) * 256 + ord(string[count])
        csum = csum + thisVal
        csum = csum & 0xffffffff
        count = count + 2

    if countTo < len(string):
        csum = csum + ord(string[len(string) - 1])
        csum = csum & 0xffffffff

    csum = (csum >> 16) + (csum & 0xffff)
    csum = csum + (csum >> 16)
    answer = ~csum
    answer = answer & 0xffff
    answer = answer >> 8 | (answer << 8 & 0xff00)
    return answer
```

## Student Questions

## Lab 5B (Cont)

```
def receiveOnePing(mySocket, ID, timeout, destAddr):
    timeLeft = timeout
    while 1:
        startedSelect = time.time()
        whatReady = select.select([mySocket], [], [], timeLeft)
        howLongInSelect = (time.time() - startedSelect)
        if whatReady[0] == []: # Timeout
            return "Request timed out."
        timeReceived = time.time()
        recPacket, addr = mySocket.recvfrom(1024)
        #Fill in start
        #Fetch the ICMP header from the IP packet
        #Fill in end
        timeLeft = timeLeft - howLongInSelect
        if timeLeft <= 0:
            return "Request timed out."
```

### Student Questions

# Lab 5B (Cont)

```
def sendOnePing(mySocket, destAddr, ID):
    # Header is type (8), code (8), checksum (16), id (16), sequence (16)
    myChecksum = 0
    # Make a dummy header with a 0 checksum
    # struct -- Interpret strings as packed binary data
    header = struct.pack("bbHHh", ICMP_ECHO_REQUEST, 0, myChecksum, ID, 1)
    data = struct.pack("d", time.time())
    # Calculate the checksum on the data and the dummy header.
    myChecksum = checksum(str(header + data))

    # Get the right checksum, and put in the header
    if sys.platform == 'darwin':
        # Convert 16-bit integers from host to network byte order
        myChecksum = htons(myChecksum) & 0xffff
    else:
        myChecksum = htons(myChecksum)
    header = struct.pack("bbHHh", ICMP_ECHO_REQUEST, 0, myChecksum, ID, 1)
    packet = header + data

    mySocket.sendto(packet, (destAddr, 1)) # AF_INET address must be tuple, not str
    # Both LISTS and TUPLES consist of a number of objects
    # which can be referenced by their position number within the object.
```

## Student Questions

# Lab 5B (Cont)

```
def doOnePing(destAddr, timeout):
    icmp = getprotobyname("icmp")
    # SOCK_RAW is a powerful socket type. For more details: http://sock-raw.org/papers/sock\_raw
    mySocket = socket(AF_INET, SOCK_RAW, icmp)
    myID = os.getpid() & 0xFFFF # Return the current process i
    sendOnePing(mySocket, destAddr, myID)
    delay = receiveOnePing(mySocket, myID, timeout, destAddr)
    mySocket.close()
    return delay
```

```
def ping(host, timeout=1):
    # timeout=1 means: If one second goes by without a reply from the server,
    # the client assumes that either the client's ping or the server's pong is lost
    dest = gethostbyname(host)
    print("Pinging " + dest + " using Python:")
    print("")
    # Send ping requests to a server separated by approximately one second
    while 1 :
        delay = doOnePing(dest, timeout)
        print(delay)
        time.sleep(1)# one second
    return delay
```

## Student Questions

# Acronyms

- ❑ ABR Area border router
- ❑ API Application Programming Interface
- ❑ AS Autonomous System
- ❑ ASBR Autonomous System Boundary Router
- ❑ BDR Backup Designated Router
- ❑ BGP Border Gateway Protocol
- ❑ BR Backbone Router
- ❑ CAPWAP Control and Provisioning of Wireless Access Points
- ❑ CCITT Consultative Committee for International Telegraph and Telephone (now ITU-T)
- ❑ CoAP Constrained Application Protocol
- ❑ COPS Common Open Policy Service
- ❑ DIDM Device Identifier and Driver Management
- ❑ DLUX OpenDaylight User Interface
- ❑ DOCSIS Data over Cable Service Interface Specification
- ❑ DR Designated Router
- ❑ eBGP exterior BGP

## Student Questions

# Acronyms (Cont)

- ❑ EGP            External Gateway Protocol
- ❑ ERP            Exterior Router Protocol
- ❑ FCAPS        Fault Configuration Accounting Performance and Security
- ❑ FRM            Forwarding Rules Manager
- ❑ GBP            Group Based Policy
- ❑ GUI            Graphical User Interface
- ❑ HTTP         Hyper-Text Transfer Protocol
- ❑ iBGP          interior BGP
- ❑ ICMP          IP Control Message Protocol
- ❑ ID             Identifier
- ❑ IDR            ICMP Router Discovery Protocol
- ❑ IGP            Interior Gateway Protocol
- ❑ IGRP          Interior Gateway Routing Protocol
- ❑ IP             Internet Protocol
- ❑ IRP            Interior Router Protocol
- ❑ ISO            International Standards Organization

## Student Questions

## Acronyms (Cont)

- ❑ LACP Link Aggregation Control Protocol
- ❑ LSA Link State Advertisements
- ❑ MIB Management Information Base
- ❑ MTU Maximum Transmission Unit
- ❑ NETCONF Network Configuration Protocol
- ❑ NIC Network Interface Card
- ❑ OSGi Open Service Gateway Initiative
- ❑ OSI Open Service Interconnection
- ❑ OSPF Open Shortest Path First
- ❑ OVSDB Open V-Switch Database
- ❑ PCEP Path Computation Element Protocol
- ❑ PCMM Packet Cable Multimedia
- ❑ REST Representational State Transfer
- ❑ RESTful Representational State Transfer
- ❑ RFC Request for Comments
- ❑ RIP Routing Information Protocol
- ❑ SAL Service Abstraction Layer

## Student Questions



# Acronyms (Cont)

- ❑ SDN Software Defined Networking
- ❑ SDNI SDN domains interface
- ❑ SFC Service Function Chaining
- ❑ SGMP Simple Gateway Management Protocol
- ❑ SNBI Secure Network Bootstrapping Interface
- ❑ SNMP Simple Network Management Protocol
- ❑ SXP SGT (Security Group Tags) Exchange Protocol
- ❑ TCP Transmission Control Protocol
- ❑ ToS Type of Service
- ❑ TSDR Time Series Data Repository
- ❑ TTL Time to Live
- ❑ UDP User Datagram Protocol
- ❑ USC Unified Secure Channel
- ❑ VPN Virtual Private Network
- ❑ VTN Virtual Tenant Network

## Student Questions

# Scan This to Download These Slides



Raj Jain

<http://rajjain.com>

[http://www.cse.wustl.edu/~jain/cse473-23/i\\_5n1c.htm](http://www.cse.wustl.edu/~jain/cse473-23/i_5n1c.htm)

## Student Questions

- ❑ Can you explain how collisions are avoided by randomizing the execution of LS algorithms at each node and only running it at one node at a single time?

*Collision avoidance usually requires someone to wait a random amount of time. However, clocks at different nodes are not synchronized, so they are already random.*

# Related Modules



CSE 567: The Art of Computer Systems Performance Analysis  
[https://www.youtube.com/playlist?list=PLjGG94etKypJEKjNAa1n\\_1X0bWWNyZcof](https://www.youtube.com/playlist?list=PLjGG94etKypJEKjNAa1n_1X0bWWNyZcof)

CSE473S: Introduction to Computer Networks (Fall 2011),  
[https://www.youtube.com/playlist?list=PLjGG94etKypJWOSPMh8Azcg5e\\_10TiDw](https://www.youtube.com/playlist?list=PLjGG94etKypJWOSPMh8Azcg5e_10TiDw)



CSE 570: Recent Advances in Networking (Spring 2013)  
<https://www.youtube.com/playlist?list=PLjGG94etKypLHyBN8mOgwJLHD2FFIMGq5>

CSE571S: Network Security (Spring 2011),  
<https://www.youtube.com/playlist?list=PLjGG94etKypKvzfVtutHcPFJXumyyg93u>



Video Podcasts of Prof. Raj Jain's Lectures,  
<https://www.youtube.com/channel/UCN4-5wzNP9-ruOzQMs-8NUw>

## Student Questions

# Lab5B Hints

- ❑ You only need to fill out the unpacking of the ICMP reply, check ICMP header fields, and return RTT time.

The program is supposed to hear the ICMP\_ECHO\_REPLY. And in the fill area, you should check the ICMP type, code, and ID. And measure the RTT using the sent time inside the data field of the ICMP\_ECHO\_REPLY. If you don't return the RTT time inside the receiveOnePing function, the code will print timeout.

- ❑ You should have a socket object or something inside the whatReady list
- ❑ You do not need to verify the checksum of the ICMP packet.
- ❑ Do not run the program on a virtual machine. Otherwise, you may get Received ICMP packet type 8 always.
- ❑ If you copy the code from the slide, the compiler may miss some indents resulting in all pings giving timeouts. So make sure that all indents are correct.

## Student Questions