EN.601.414/614 Computer Networks

Final Review

Xin Jin

Spring 2019 (MW 3:00-4:15pm in Shaffer 301)

JOHNS HOPKINS UNIVERSITY

https://github.com/xinjin/course-net

Final Exam

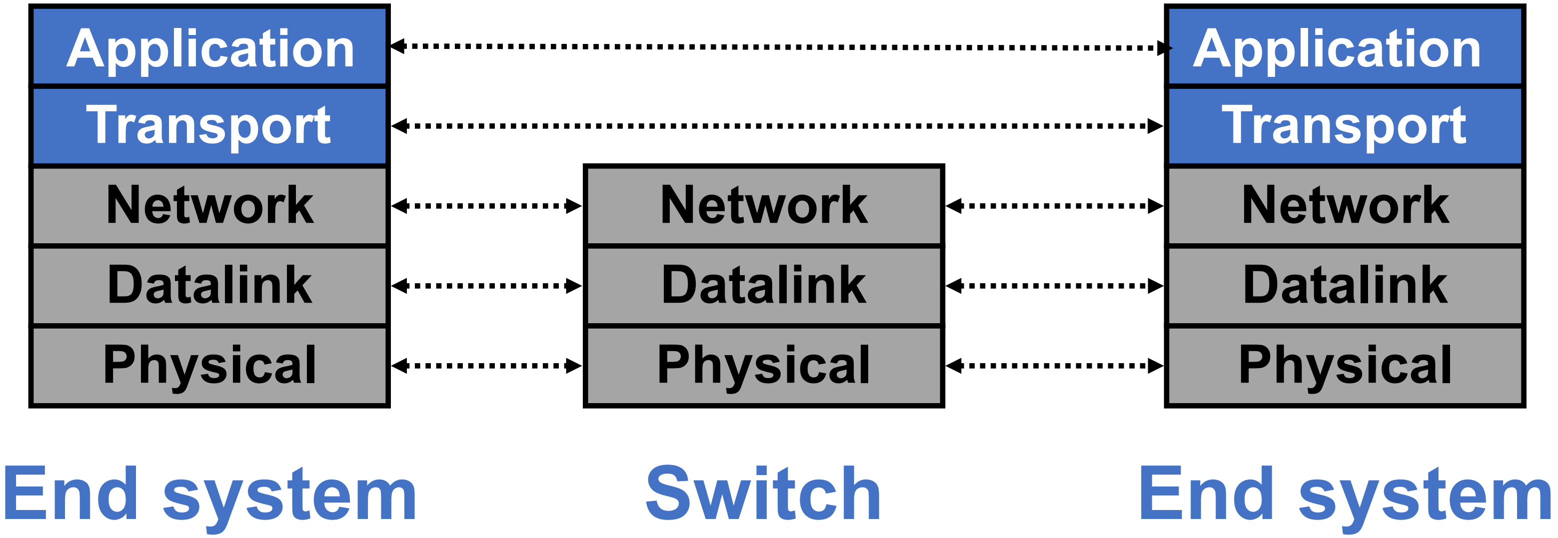
- **Time: 6pm-7:30pm, Wednesday, May 8**
- **Location: Shaffer 301**
- **Form: Closed-book**
  - Can bring TWO A4/letter papers with notes on both sides
  - Can bring a calculator
  - Anything else is prohibited
- **Focus on materials after midterm**
  - Materials before midterm will be tested, but not a focus

This review

- **Walk through what you're expected to know at this point: key topics, important aspects of each**
- **Not covered in review does NOT imply you don't need to know it**
  - ➢But if it's covered today, you should know it
- **Summarize, not explain**
  - ➢Stop me when you want to discuss something further!

The networking stack

- **Lower three layers implemented everywhere**
- **Top two layers implemented only at hosts**

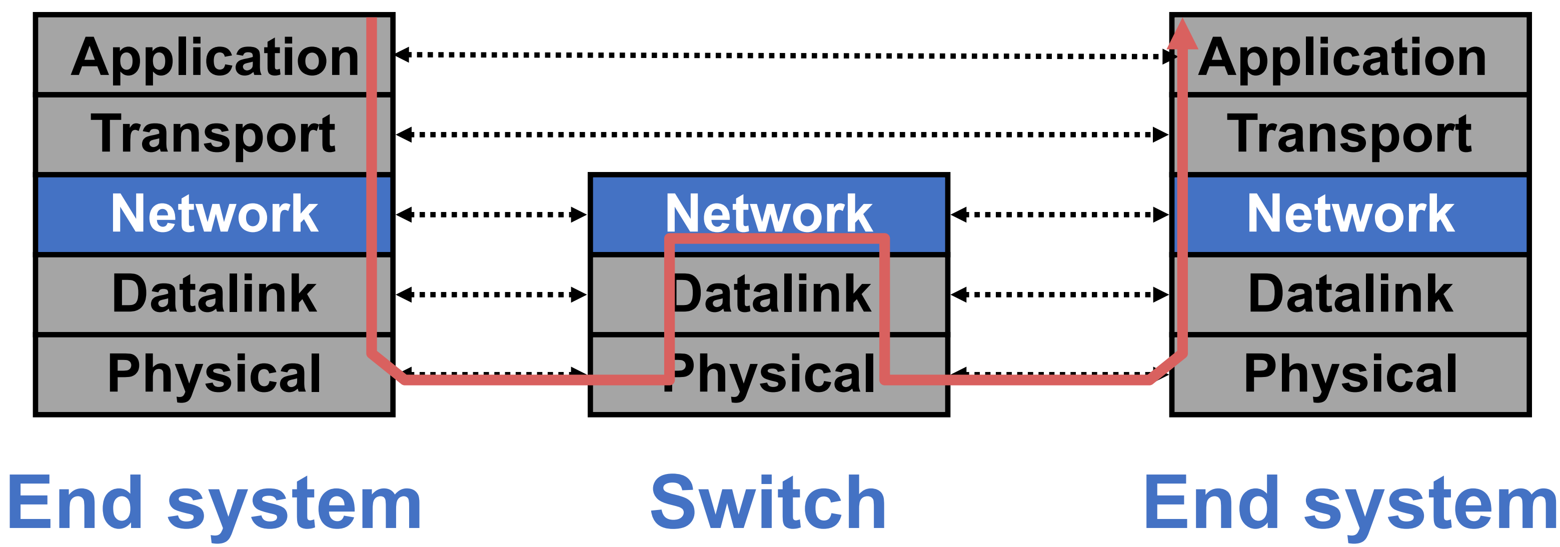


Topics

- **Network layer (lectures 11–15)**
  - ➢Data plane
  - ➢Intra- and inter-domain routing
- **Link layer (lectures 17–18)**
  - ➢Ethernet
- **Topics in networking (lectures 16, 19–21)**
  - ➢Programmable networks
  - ➢Wireless
  - ➢Security
  - ➢Misc

Network layer

- Present everywhere
- Performs addressing, forwarding, and routing, among other tasks

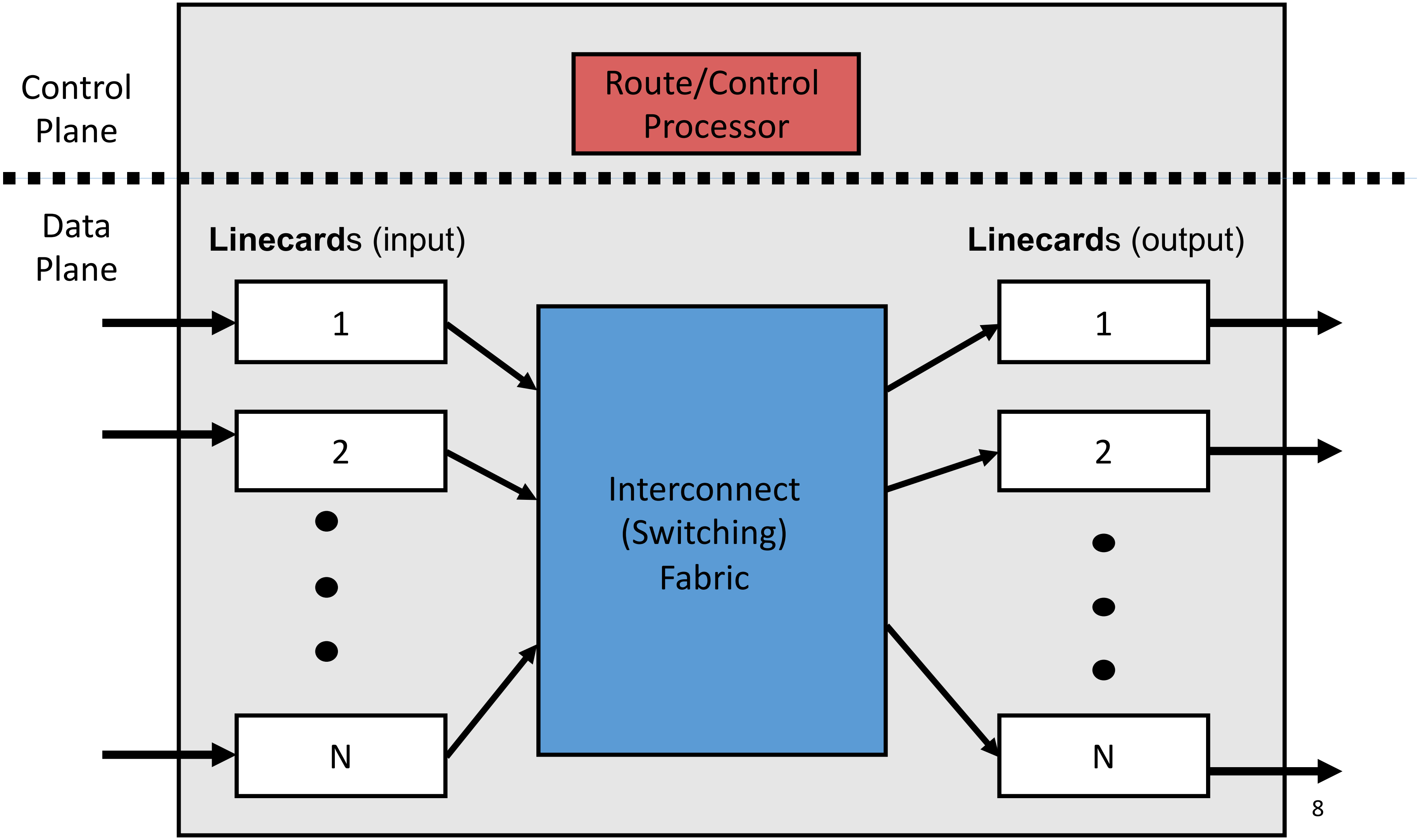


Forwarding vs. routing

- **Forwarding: "data plane"**
  - Directing one data packet
  - Each router using local routing state
- **Routing: "control plane"**
  - Computing the forwarding tables that guide packets
  - Jointly computed by routers using a distributed algorithm

- **Very different timescales!**

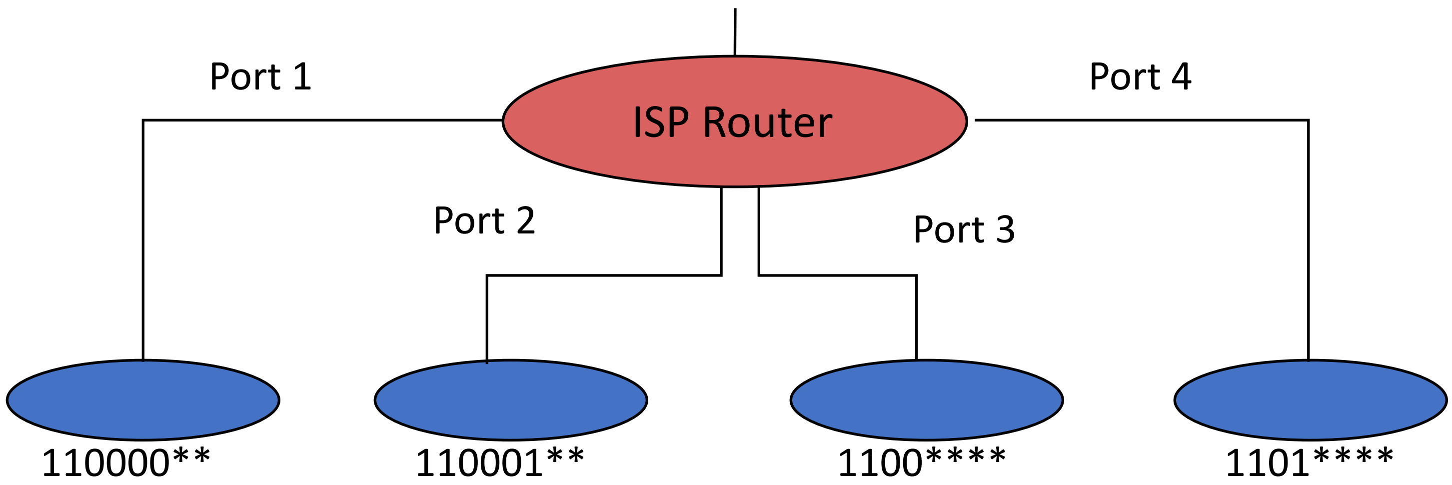
What's inside a router?



Looking up the output port

- **One entry for each address → 4 billion entries!**
- **For scalability, addresses are aggregated**

Longest prefix matching



Send to the port with the longest prefix match

Routing: Local vs. global view

- ***Local* routing state is the forwarding table in a single router**
  - By itself, the state in a single router cannot be evaluated
  - It must be evaluated in terms of the global context
- ***Global* state refers to the collection of forwarding tables in each of the routers**
  - Global state determines which paths packets take

“Valid” routing state

- **Global state is “valid” if it produces forwarding decisions that always deliver packets to their destinations**
- **Goal of routing protocols: compute valid state**
    - ➢ How can we tell if routing state if valid?

Necessary and sufficient condition

- **Global routing state is valid *if and only if*:**
  - ➢ There are no dead ends (other than destination)
  - ➢ There are no loops
- **A dead end is when there is no outgoing link (next-hop)**
  - ➢ A packet arrives, but the forwarding decision does not yield any outgoing link
- **A loop is when a packet cycles around the same set of nodes forever**

Least-cost routes

- **Least-cost routes provide an easy way to avoid loops**
  - No reasonable cost metric is minimized by traversing a loop
- **Least-cost paths form a spanning tree for each destination rooted at that destination**

Intra-domain routing

- **Link-state (LS) routing protocol**
  - Dijkstra’s algorithm
  - Broadcast neighbors’ info to everyone
- **Distance vector (DV) routing protocol**
  - Bellman-Ford algorithm
  - Gossip to neighbors about everyone

Similarities between LS and DV routing

- **Both are shortest-path based routing**
  - ➢ Minimizing cost metric (link weights) a common optimization goal
    - Routers share a common view as to what makes a path “good” and how to measure the “goodness” of a path
- **Due to shared goal, commonly used inside an organization**
  - ➢ RIP and OSPF are mostly used for intra-domain routing

Comparison of LS and DV routing

Messaging complexity

- **LS: with N nodes, E links, O(NE) messages sent**
- **DV: exchange between neighbors only**

Speed of convergence

- **LS: relatively fast**
- **DV: convergence time varies**
  - **Count-to-infinity** problem

Robustness: what happens if router malfunctions?

- **LS:**
  - Node can advertise incorrect link cost
  - Each node computes its *own* table
- **DV:**
  - Node can advertise incorrect path cost
  - Each node's table used by others (error propagates)

Addressing is key to scalable inter-domain routing

- **Ability to aggregate addresses is crucial for**
  - State: Small forwarding tables at routers
    - Much less than the number of hosts
  - Churn: Limited rate of change in routing tables

Classful addressing

- **Three classes**
  - ➢8-bit network prefix (Class A),
  - ➢16-bit network prefix (Class B), or
  - ➢24-bit network prefix (Class C)
- **Example: an organization needs 500 addresses.**
  - ➢ A single class C address is not enough (<500 hosts)
  - ➢ Instead, a class B address is allocated (~65K hosts)
    - Huge waste!

CIDR: Classless inter-domain routing

- **Flexible division between network and host addresses**
- **Offers a better tradeoff between size of the routing table and efficient use of the IP address space**

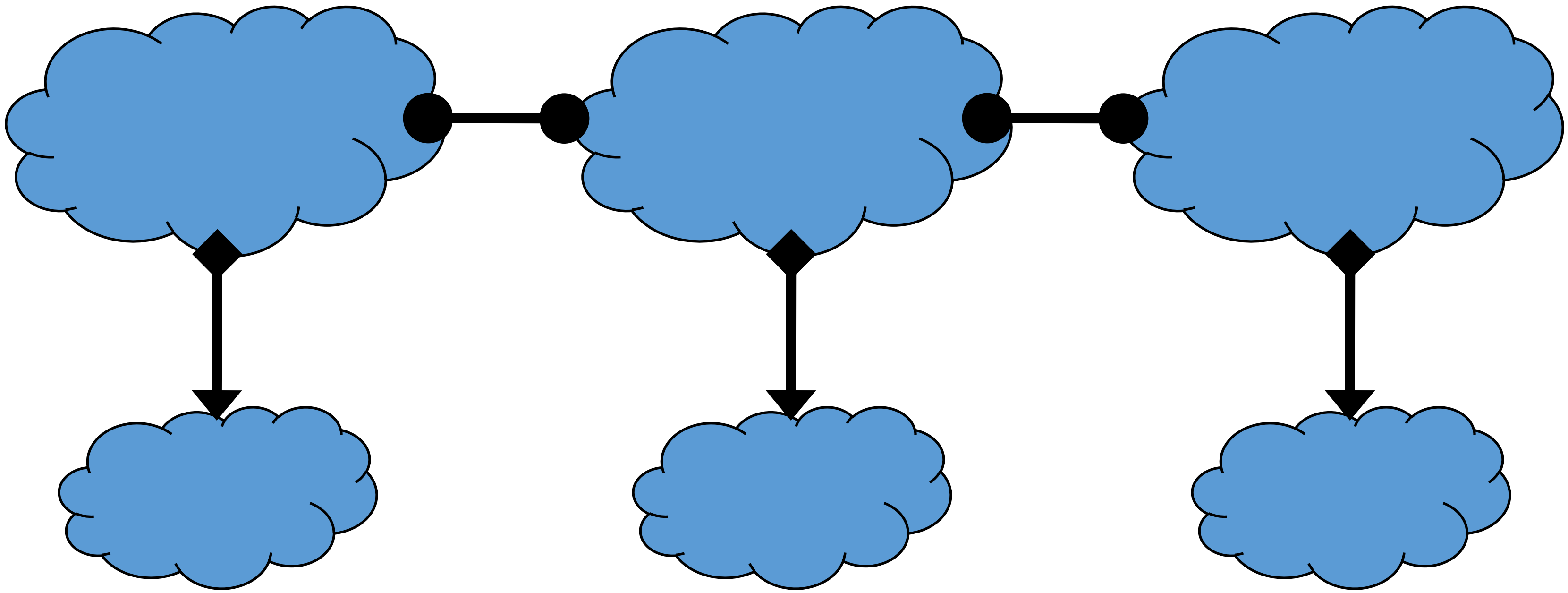
Administrative structure shapes Inter-domain routing

- **ASes want freedom to pick routes based on policy**
- **ASes want autonomy**
- **ASes want privacy**

Topology & policy shaped by inter-AS business relationship

- **Three basic kinds of relationships between ASes**
  - ➢AS A can be AS B's customer
  - ➢AS A can be AS B's provider
  - ➢AS A can be AS B's peer
- **Business implications**
  - ➢Customer pays provider
  - ➢Peers don't pay each other
    - Exchange roughly equal traffic

Business relationships



Relations between ASes

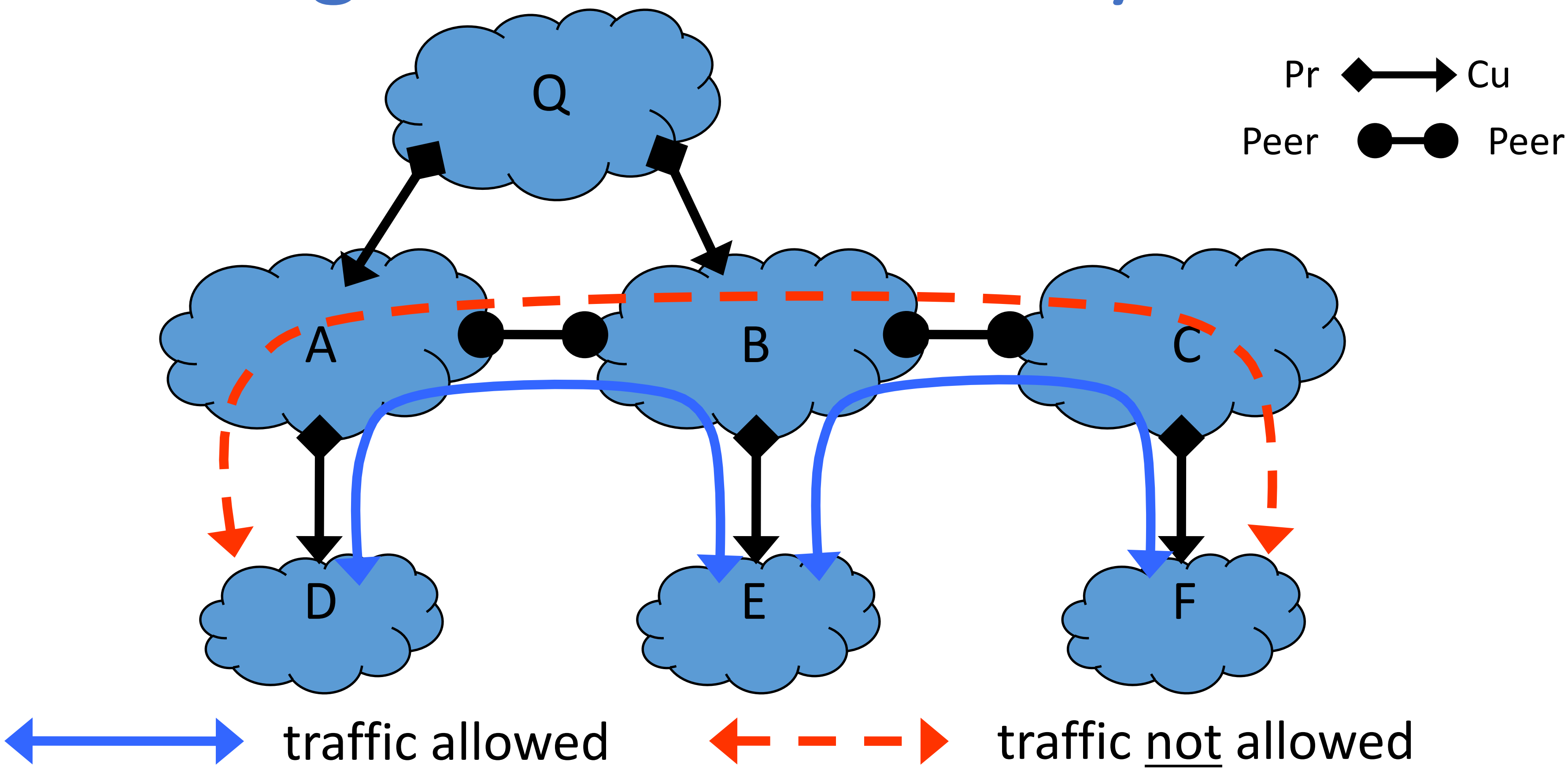
provider → customer

peer — peer

Business implications

- Customers pay provider
- Peers don't pay each other

Routing follows the money!



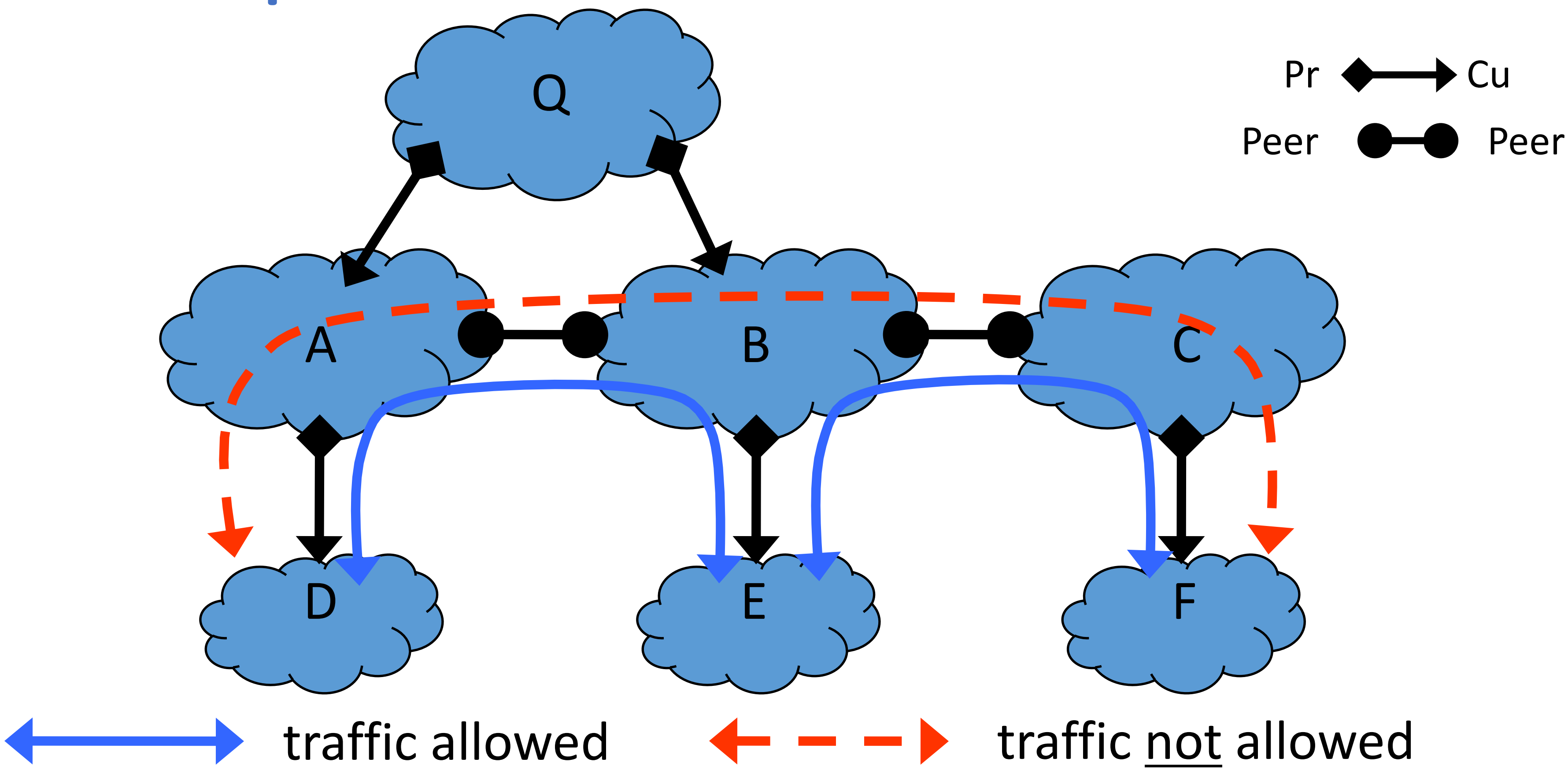
- ASes provide “transit” between their customers
- Peers do not provide transit between other peers

Valley-Free Routing

- **Number links as (+1, 0, -1) for customer-to-provider, peer and provider-to-customer**
- **In any path should only see sequence of +1, followed by at most one 0, followed by sequence of -1**

- **Example question: show why the traffic is allowed or not allowed in the previous slide with valley-free routing**

Example Question

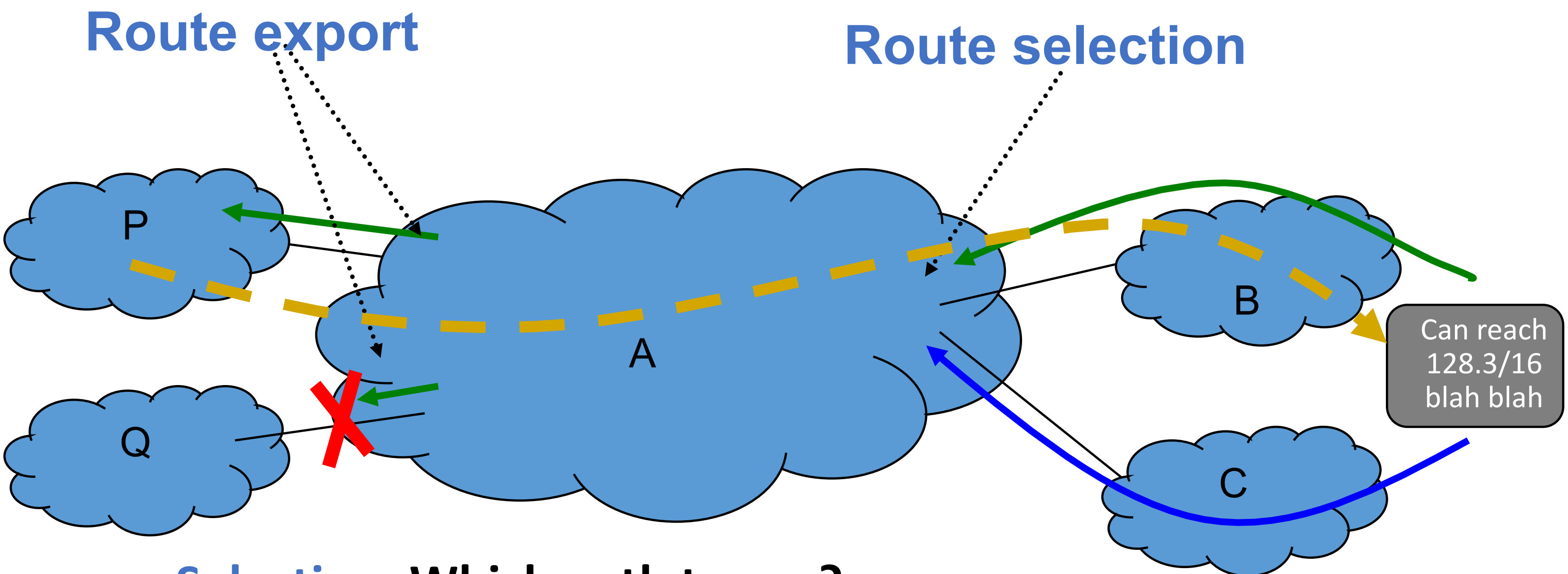


- **Show why the traffic is allowed or not allowed with valley-free routing**

BGP inspired by Distance-Vector with four differences

- **Shortest-path routes may not be picked to enforce policy**
- **Path-Vector routing to avoid loops**
- **Selective route advertisement may affect reachability**
- **Routes may be aggregated for scalability**

Policy dictates how routes are "selected" and "exported"



- **Selection: Which path to use?**
  - Controls whether/how traffic leaves the network
- **Export: Which path to advertise?**
  - Controls whether/how traffic enters the network

eBGP, iBGP, and IGP

- **eBGP: BGP sessions between border routers in different ASes**
  - Learn routes to external destinations
- **iBGP: BGP sessions between border routers and other routers within the same AS**
  - Distribute externally learned routes internally
- **IGP: "Interior Gateway Protocol" = Intra-domain routing protocol**
  - Provide internal reachability
  - E.g., OSPF, RIP

eBGP, iBGP, and IGP together

- **Learn routes to external destination using eBGP**
- **Distribute externally learned routes internally using iBGP**
- **Travel shortest path to egress using IGP**

Topics

- **Network layer (lectures 11–15)**
  - Data plane
  - Intra- and inter-domain routing
- **Link layer (lectures 17–18)**
  - Ethernet
- **Topics in networking (lectures 16, 19–21)**
  - Programmable networks
  - Wireless
  - Security
  - Misc

Data link layer

- **Provides four primary services**
  - Framing
    - Encapsulates network layer data
  - Link access
    - Medium access control (MAC) protocol defines when to transmit frames
  - Reliable delivery
    - Primarily for mediums with high error rates (e.g., wireless)
  - Error detection and correction

Point-to-point vs. broadcast medium

- **Point-to-point: dedicated pairwise communication**
  - E.g., long-distance fiber link
  - E.g., Point-to-point link b/n Ethernet switch and host
- **Broadcast: shared wire or medium**
  - Traditional Ethernet (pre ~2000)
  - 802.11 wireless LAN

Random access MAC protocols

- **When node has packet to send**
  - Transmit at full channel data rate **w/o** coordination
- **Two or more transmitting nodes $\Rightarrow$ collision**
  - Data lost
- **Random access MAC protocol specifies**
  - How to detect and recover from collisions
- **Examples**
  - ALOHA and Slotted ALOHA
  - CSMA, CSMA/CD, CSMA/CA (wireless)

CSMA (Carrier Sense Multiple Access)

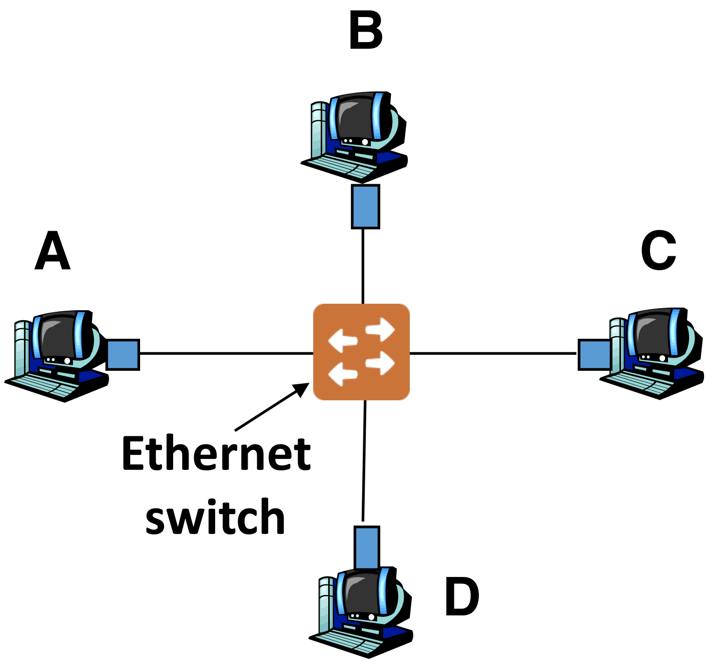
- **CSMA: listen before transmit**
  - If channel sensed idle: transmit entire frame
  - If channel sensed busy, defer transmission
- **Human analogy: don’t interrupt others!**
- **Does not eliminate all collisions**
  - Why?

CSMA/CD (Collision Detection)

- **CSMA/CD: carrier sensing, deferral as in CSMA**
  - ➢Collisions detected within short time
  - ➢Colliding transmissions aborted, reducing wastage
- **Collision detection easy in wired (broadcast) LANs**
  - ➢Compare transmitted, received signals
- **Collision detection difficult in wireless LANs**

Why switched Ethernet?

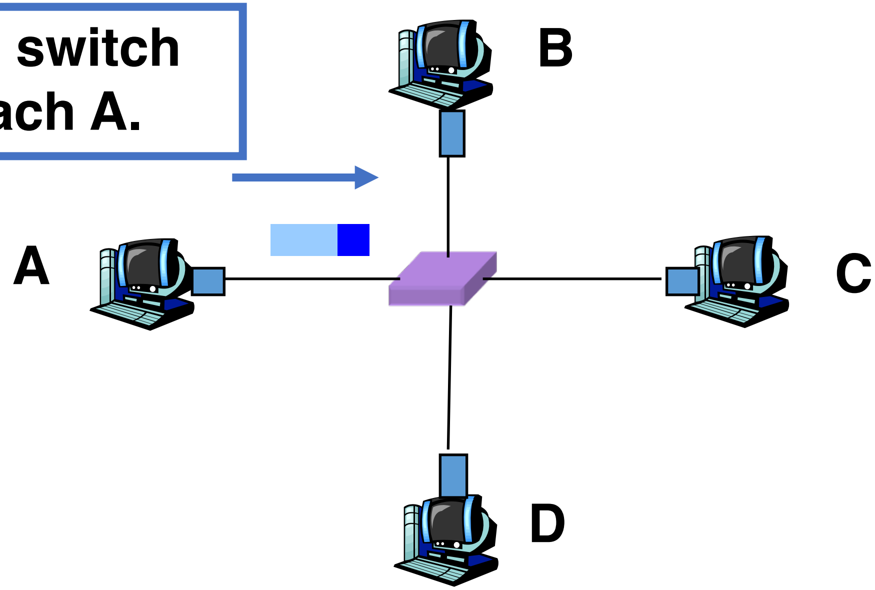
- **Enables concurrent communication**
  - ➢ Host A can talk to C, while B talks to D
  - ➢ No collisions and no need for CSMA/CD
  - ➢ No constraints on link lengths, etc.



Ethernet switches are "self learning"

- **When a packet arrives:**
  - ➢Inspect source MAC address, associate with incoming port
  - ➢Store mapping in the switch table
  - ➢Use time-to-live field to eventually forget mapping

Packet tells switch how to reach A.



ARP and DHCP

- **Link layer discovery protocols**
  - ➢ARP → Address Resolution Protocol
  - ➢DHCP → Dynamic Host Configuration Protocol
  - ➢Confined to a single local-area network (LAN)
  - ➢Rely on broadcast capability

Key ideas in both ARP and DHCP

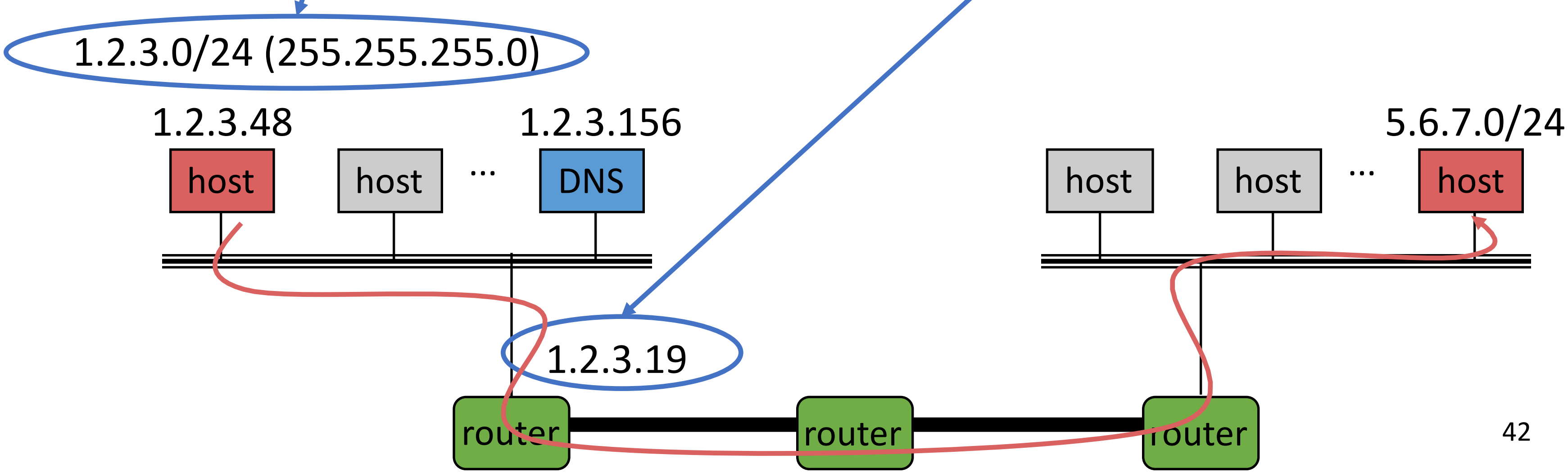
- Broadcasting: **Can use broadcast to make contact**
  - Scalable because of limited size
- Caching: **remember the past for a while**
  - Store the information you learn to reduce overhead
- Soft state: **eventually forget the past**
  - Associate a time-to-live field with the information
  - … and either refresh or discard the information
  - Key for robustness in the face of unpredictable change

ARP: Address Resolution Protocol

- **Every host maintains an ARP table**
  - List of (IP address → MAC address) pairs
- **Consult the table when sending a packet**
  - Map dest. IP address to dest. MAC address
  - Encapsulate (IP) data packet with MAC header; xmit
- **What if IP address not in the table?**
  - Sender broadcasts: Who has IP address 1.2.3.156?
  - Receiver replies: MAC address 58-23-D7-FA-20-B0
  - Sender caches result in its ARP table

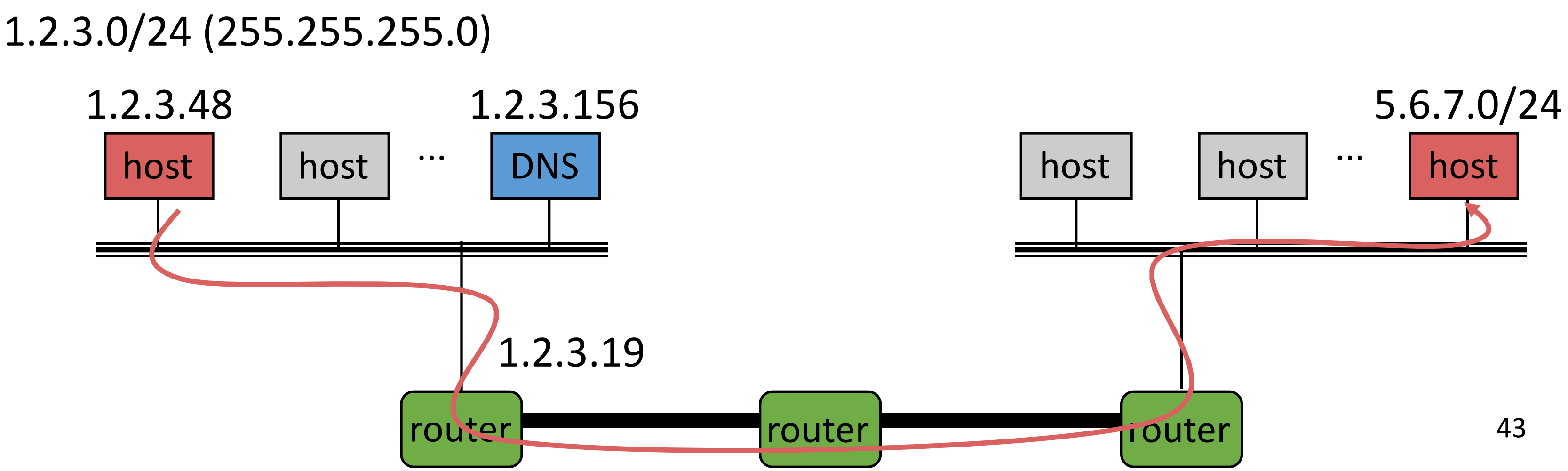
What if the destination is remote?

- **Look up the MAC address of the first hop router**
  - 1.2.3.48 uses ARP to find MAC address for first-hop router **1.2.3.19** rather than ultimate destination IP address
- **How does the red host know the destination is not local?**
  - Uses netmask (discovered via DHCP)
- **How does the red host know about 1.2.3.19?**
  - Also DHCP



Example Question

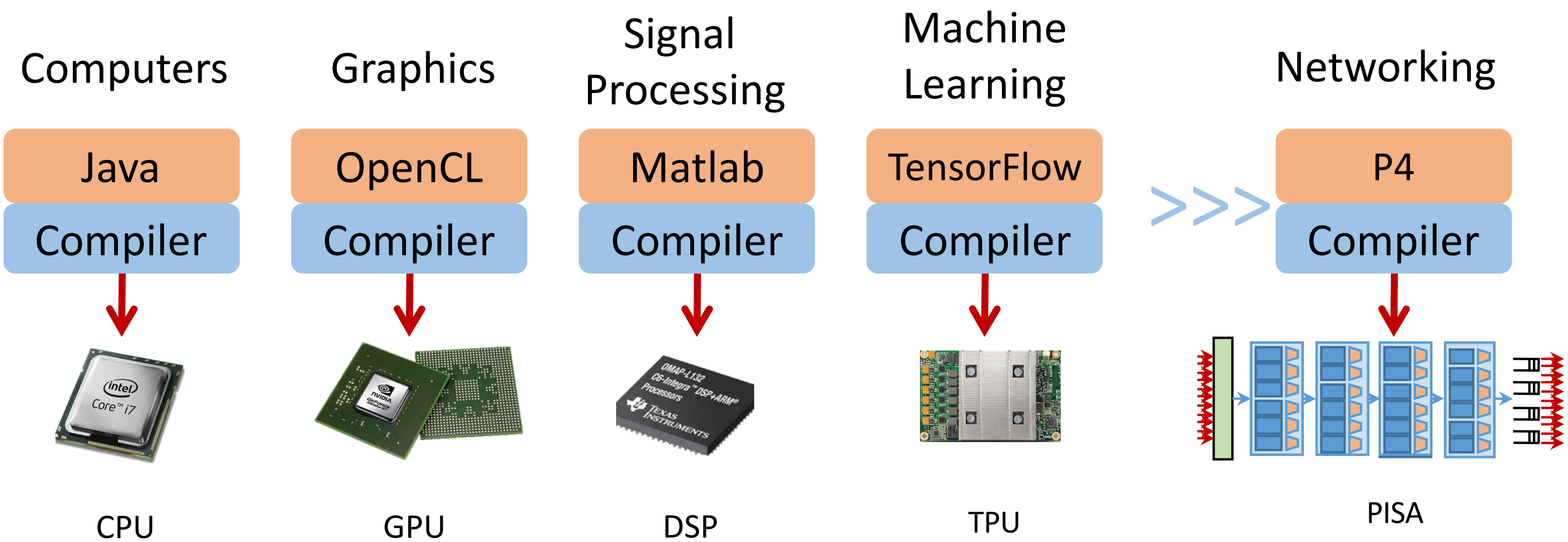
- **Describe the process for 1.2.3.48 to access www.foo.com hosted on 5.6.7.10**



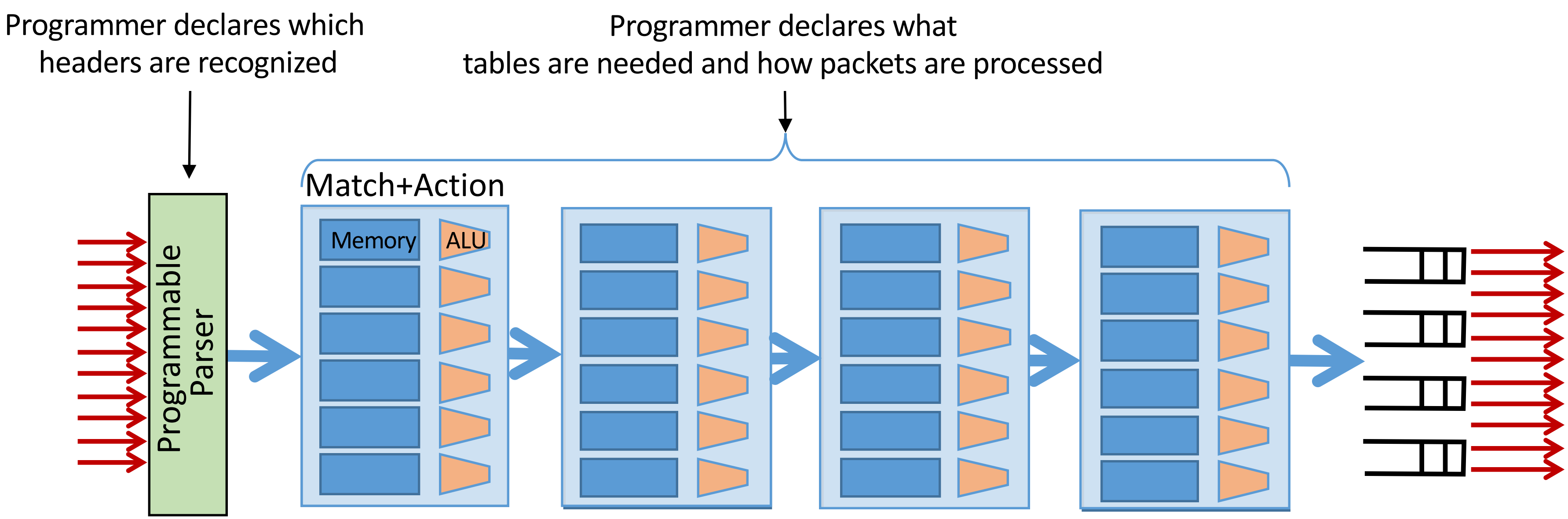
Topics

- **Network layer (lectures 11–15)**
  - Data plane
  - Intra- and inter-domain routing
- **Link layer (lectures 17–18)**
  - Ethernet
- **Topics in networking (lectures 16, 19–21)**
  - Programmable networks
  - Wireless
  - Security
  - Misc

Domain Specific Processors



PISA: Protocol Independent Switch Architecture



All stages are identical – makes PISA a good "compiler target"

Summary of Programamable Networks

1. **Chip speed: We can now make programmable switch chips as fast as fixed ones.**
2. **Chip technology: The difference in chip area and power between “programmable” and “fixed function” is going away.**
3. **Chip complexity: There are now too many protocols to correctly hard-code in silicon.**
4. **New ideas: Beautiful new ideas are owned by the programmer, not the chip designer.**
5. **Level playing field: Lets us create a solid platform, an abstraction layer, upon which more will be built**

To learn more, visit P4.org

Wireless link characteristics

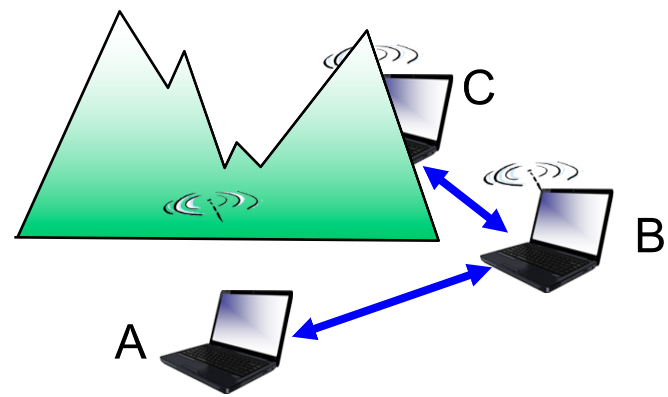
- **Three important differences from wired link …**
  - Decreased signal strength: Radio signal attenuates as it propagates through matter (path loss)
  - Multipath propagation: Radio signal reflects off objects ground, arriving at destination at slightly different times
  - Interference from other sources: Standardized wireless network frequencies (e.g., 2.4 GHz) shared by other devices (e.g., phone); devices (motors) interfere as well
- **… make communication across (even a point-to-point) wireless link much more "difficult"**

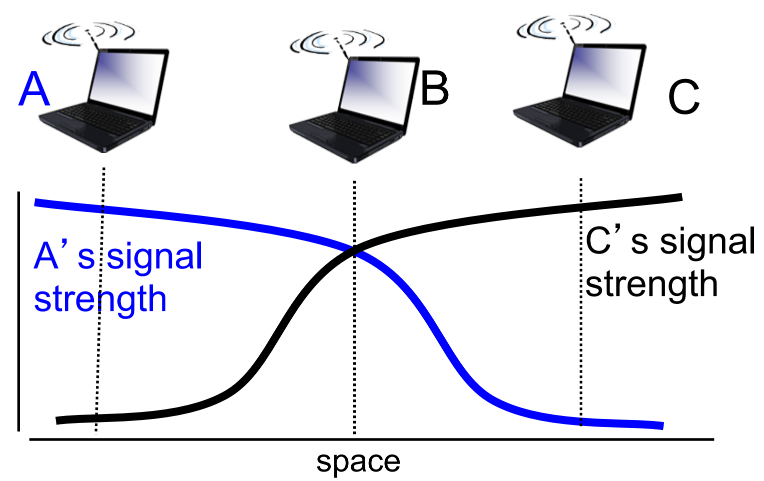
Wireless network characteristics

- **Multiple wireless senders and receivers create many problems**
  - Multiple access issues
  - Hidden terminal problem

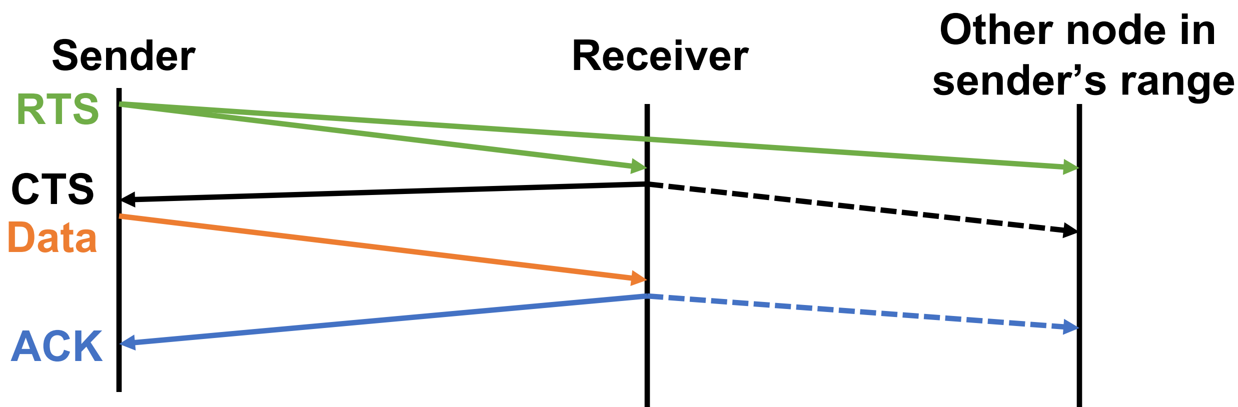
Hidden terminal problem

- **B, A hear each other**
- **B, C hear each other**
- **A, C can not hear each other**
- **Hence, A, C are unaware of their interference at B**





CSMA/CA



- **Before every data transmission**
  - ➢ Sender sends a Request to Send (RTS) frame with the length of transmission and the destination
  - ➢ Receiver respond with a Clear to Send (CTS) frame
  - ➢ Sender sends data
  - ➢ Receiver sends an ACK
- **If sender doesn't get a CTS back, it assumes collision**

General goals for communication security: CIA

- **Confidentiality**
  - No one read our communication
  - Cryptography
- **Message Integrity**
  - No one can modify our communication w/o detection
  - Verification
- **Availability and Authentication**
  - Redundancy, DoS/DDoS prevention
  - Only we can access our data and communicate on our behalf

Computer networks is a fast growing area

- **Software-Defined Networking**
- **Cloud Computing**
- **Network Virtualization**
- **Network Testing and Verification**
- **Big Network Data Processing**
- **AI and Networking**
- **Internet of Things**
- **Bitcoin and Blockchain**

Summary

- **Final exam:**
  - ➢Time: 6pm-7:30pm, Wednesday, May 8
  - ➢Location: Shaffer 301
- **2019 Fall: EN.601.714 Advanced Computer Networks**
- **THANK YOU SO MUCH!!!**

Thanks!
Q&A