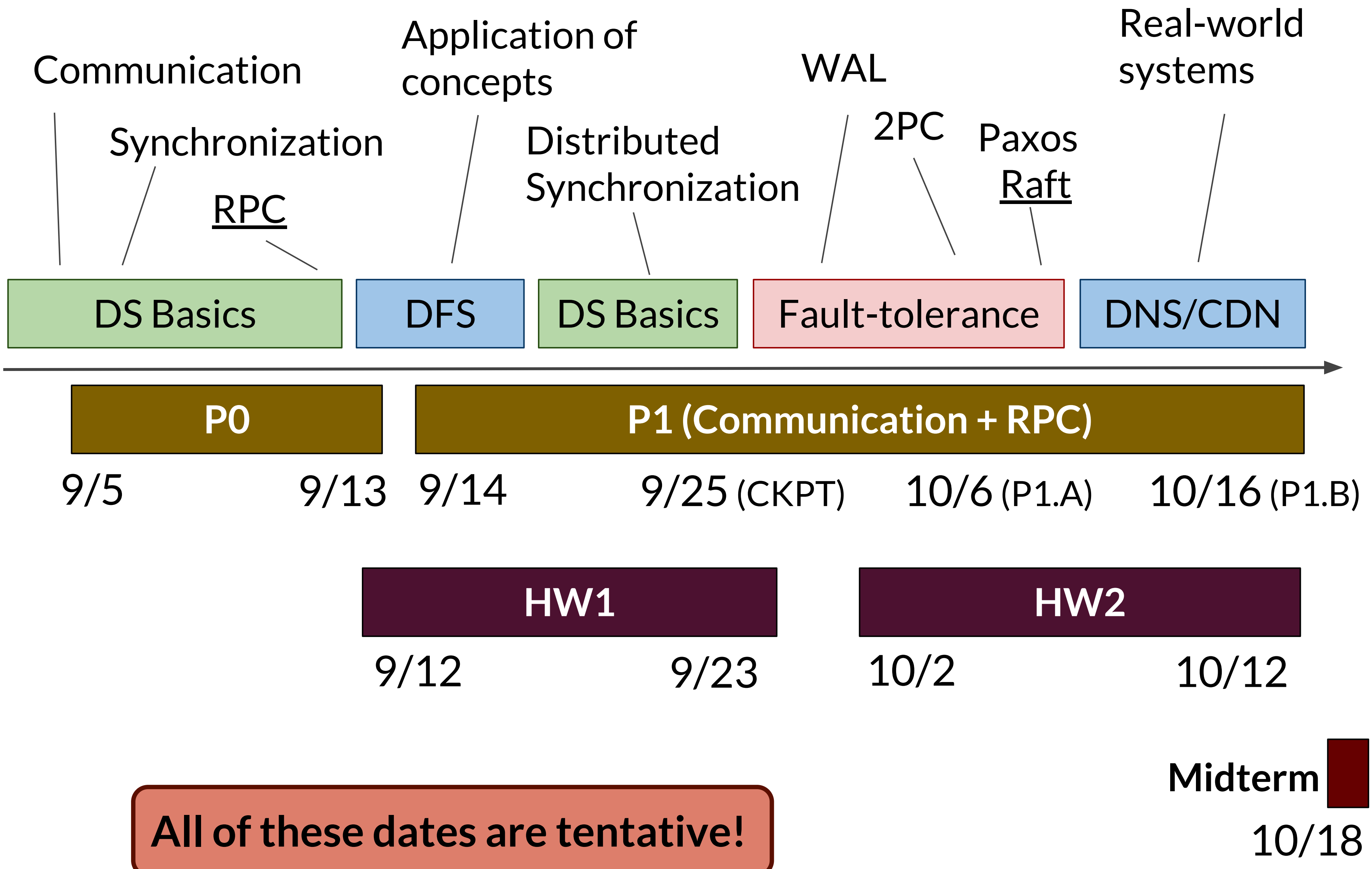
Distributed Systems

15-440/640

Fall 2018

2 – Communication:
The Internet in a Day

Overview: First Semester Half



Warning

These slides (and any other slides posted for class) are *NOT* meant to be a complete guide or class notes

- Please take notes in class
- There is often additional material presented
- Slides are often hard to understand after the fact

I will make sure to post at least a draft of the slides in advance of class

- Gives you a start on notes :-)

Distributed Systems

15-440/640

Fall 2018

2 – Communication:
The Internet in ~~a Day~~

really: in two days

Today’s & Tuesday’s Lecture

Network links and LANs

Inter-network Communication

Layering & Protocols

Internet design

Transport protocols

Application design

Distributed Systems vs. Networks

- Low level (c/go)
- Run forever
- Support others
- Adversarial environment
- Distributed & concurrent
- Resources matter

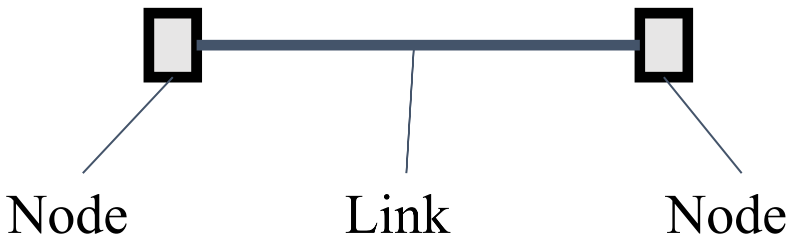
- Availability vs. correctness
- Common goals vs. policies
- Single vs. diverse implementations

Keep an eye out for…

- Modularity, Layering, and Decomposition:
  - Techniques for dividing the work of building systems
  - Hiding the complexity of components from each other
  - Hiding implementation details to deal with heterogeneity
- Naming/lookup/routing
- Resource sharing and isolation

- Models and assumptions about the environment and components
- Understanding and estimating performance

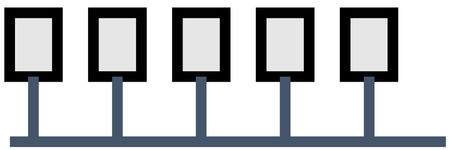
Basic Building Block: Links



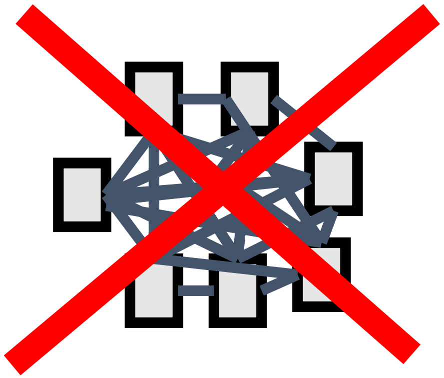
- Electrical questions
  - Voltage, frequency, …
  - Wired or wireless?

- Link-layer issues: How to send data?
  - When to talk – can either side talk at once?
  - What to say – low-level format?

What if we want more hosts?



One wire



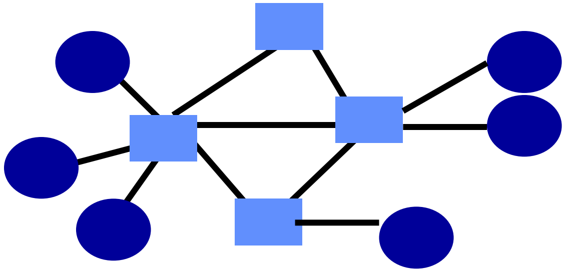
Wires for everybody!

- When to send?
- Who should receive?

- Scalability?

Multiplexing

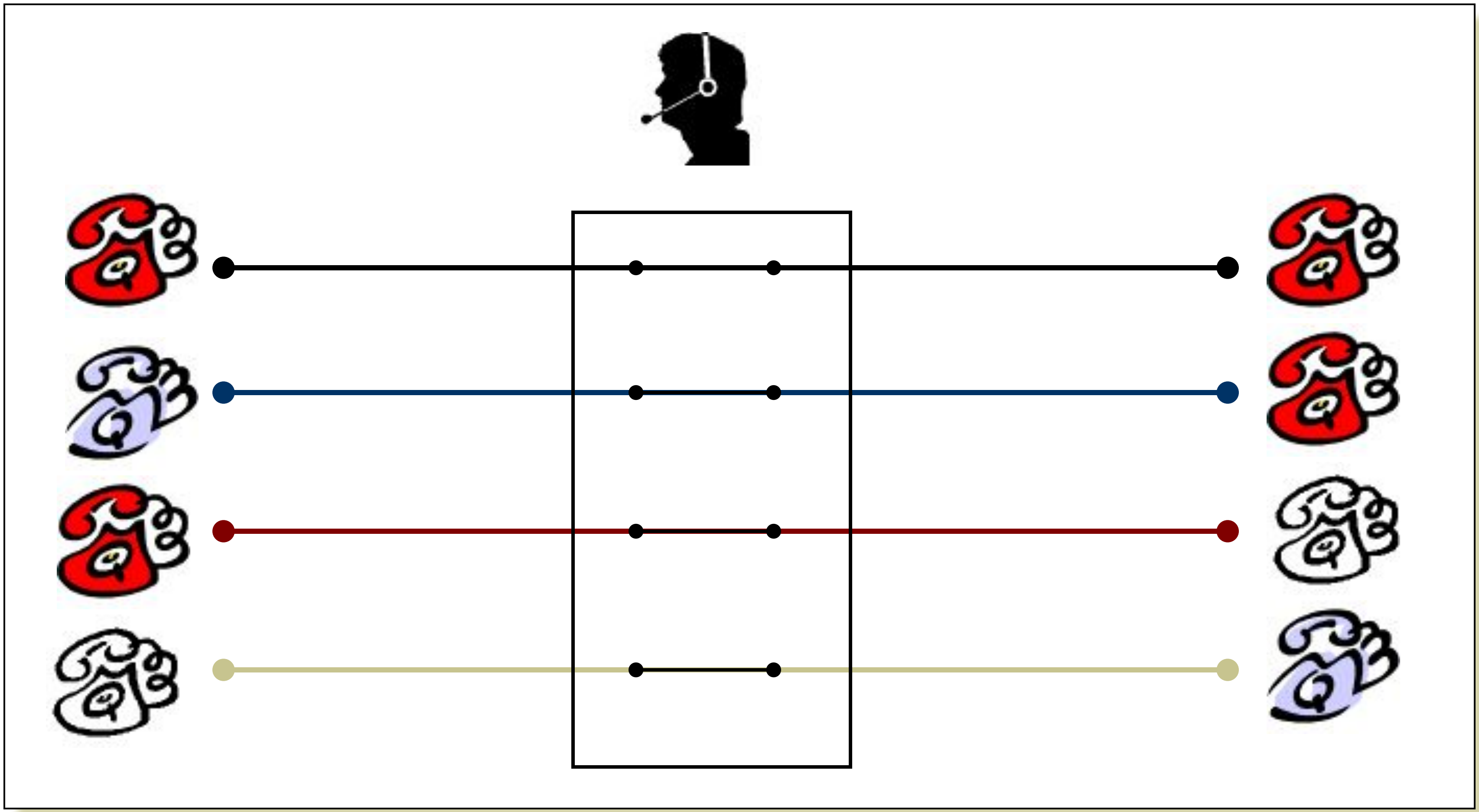
- Need to share network resources



- How? Switched network
  - Party “A” gets resources sometimes
  - Party “B” gets them sometimes
- Interior nodes act as “Switches”

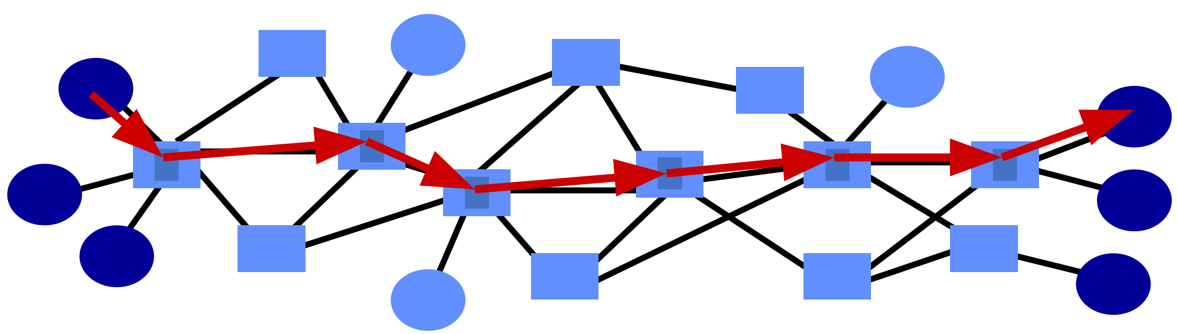
- What mechanisms to share resources?

In the Old Days…Circuit Switching

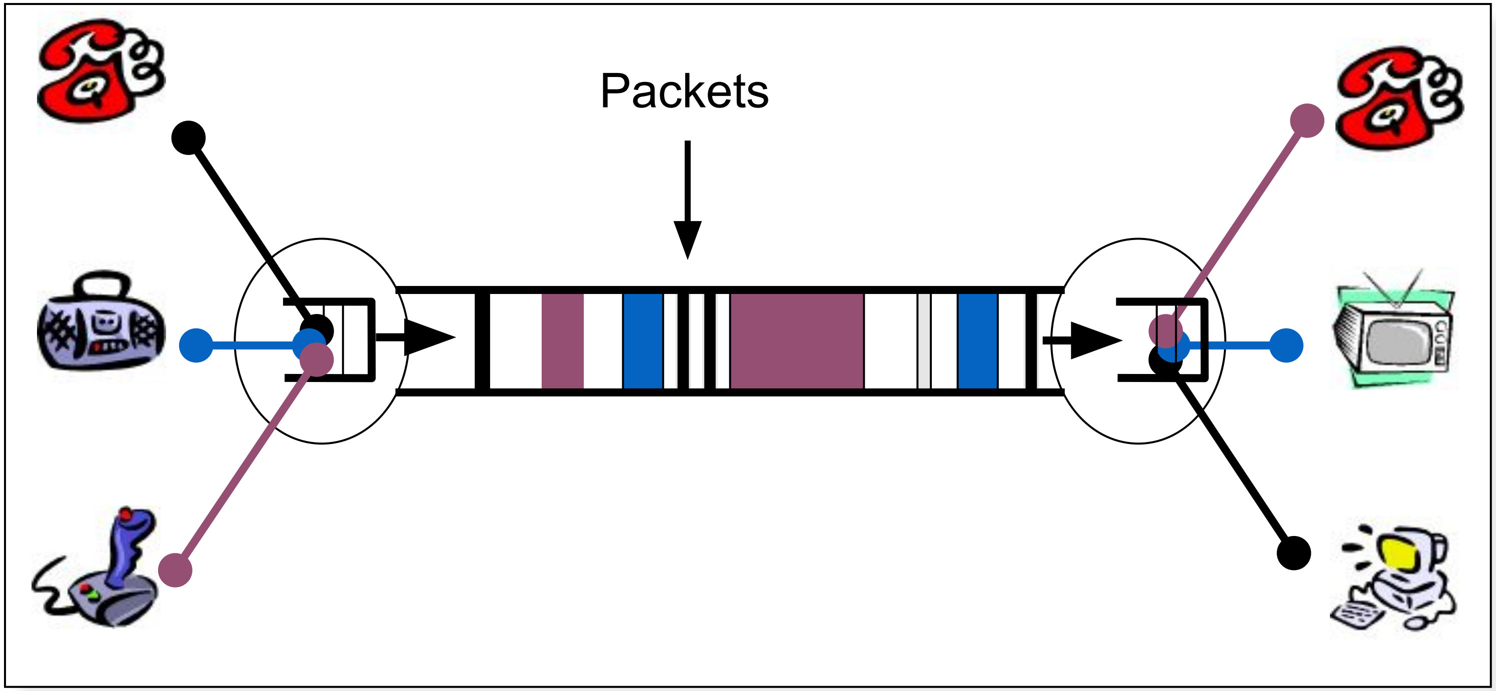


Packet Switching

- Source sends information as self-contained packets that have an address.
  - Source may have to break up single message in multiple
- Each packet travels independently to the destination host.
  - Switches use the address in the packet to determine how to forward the packets
  - Store and forward
- Analogy: a letter in surface mail.



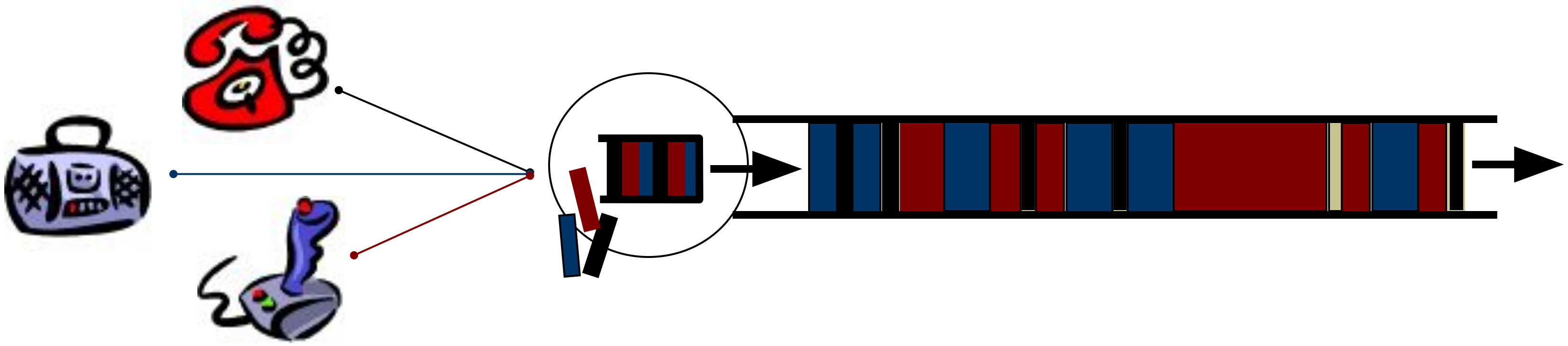
Packet Switching – Statistical Multiplexing



- Switches arbitrate between inputs
- Can send from *any* input that's ready
  - Links never idle when traffic to send
  - (Efficiency!)

What if Network is Overloaded?

Problem: Network Overload



Solution: Buffering and Congestion Control

- Short bursts: buffer
- What if buffer overflows?
  - Packets dropped
  - Sender adjusts rate until load = resources → “congestion control”

Model of a communication channel

- Latency - how long does it take for the first bit to reach destination

- Capacity - how many bits/sec can we push through? (often termed “bandwidth”)

- Jitter - how much variation in latency?

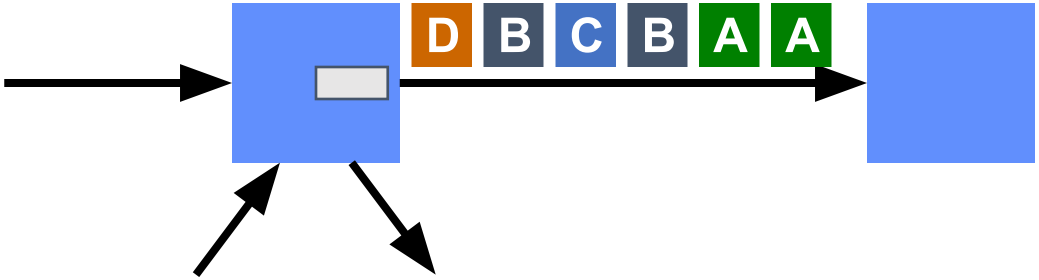
- Loss / Reliability - can the channel drop packets?

- Packet reordering

Packet Delay

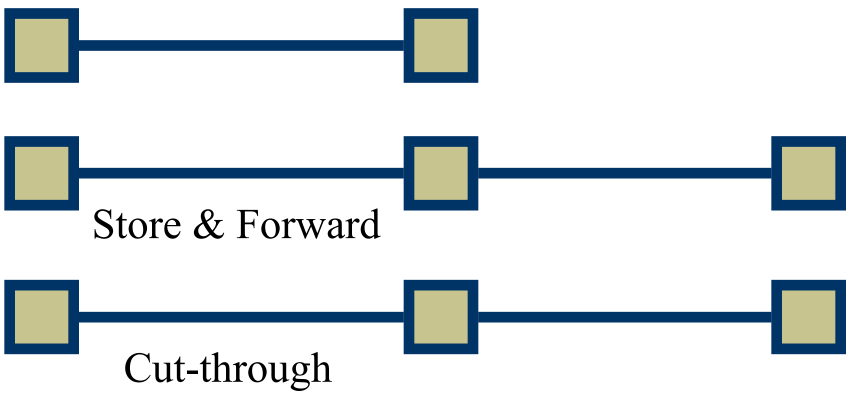
Sum of a number of different delay components:

- Propagation delay on each link.
  - Proportional to the length of the link
- Transmission delay on each link.
  - Proportional to the packet size and 1/link speed
- Processing delay on each router.
  - Depends on the speed of the router
- Queuing delay on each router.
  - Depends on the traffic load and queue size



Packet Delay

xmit = trans-
mission delay



Prop + xmit

2*(Prop + xmit)

2*prop + xmit

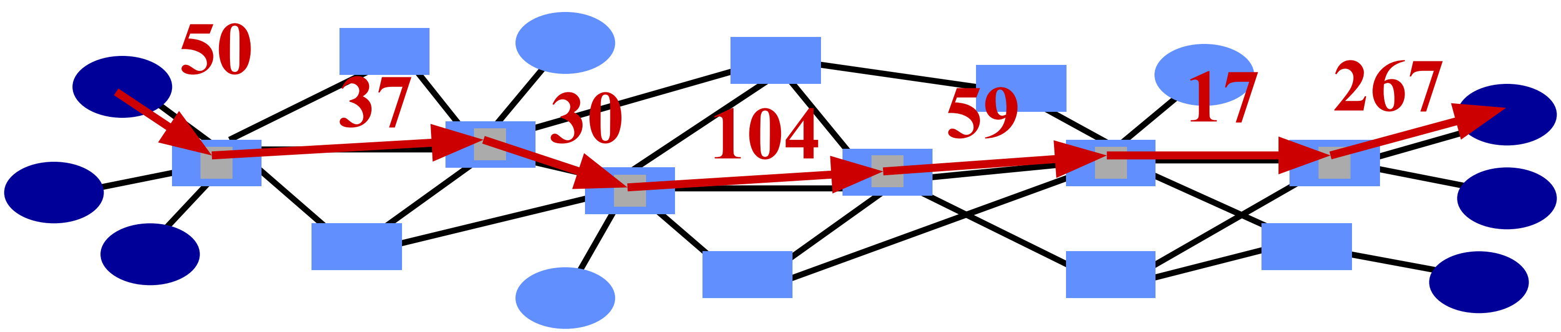
When does cut-through matter?

Next: back-of-the-envelope calculations

(What's the xmit? Which protocol are we using?)

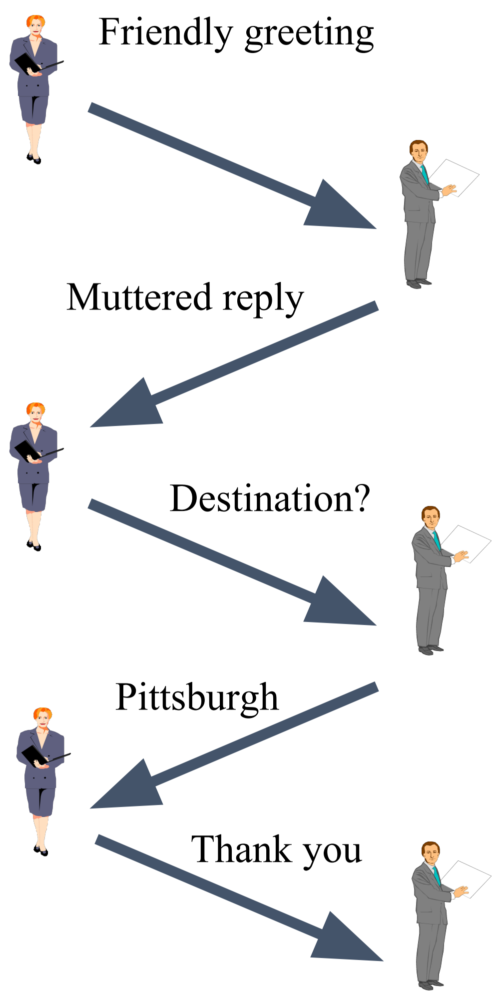
Sustained Throughput

- When streaming packets, the network works like a pipeline.
  - All links forward different packets in parallel
- Throughput is determined by the slowest stage.
  - Called the bottleneck link
- Does not really matter why the link is slow.
  - Low link bandwidth
  - Many users sharing the link bandwidth



Polite Networks: Stop & Wait Protocol

- Protocol: agreement between parties on how communication should take place
- Stop & wait: sender sends a single packet to receiver & waits for an acknowledgment
- Example: polite conversation, buying airline ticket over phone



Back-of-the-Envelope Bandwidth Calculation

- Cross country latency
  - Distance/speed = 5 * 10^6m / 2x10^8m/s = 25 * 10^-3 s = 25ms
  - 50ms round-trip-time (RTT) for one bit
- Link speed (capacity) 100Mbps
- Packet size = 1250 bytes = 10 kbits
  - Packet size on networks usually = 1500bytes across wide area or 9000bytes in local area

- 1 packet takes
  - 10k/100M = .1 ms to transmit
  - 25ms to reach there
  - ACKs are small → so 0ms to transmit
  - 25ms to get back
- Effective bandwidth = 10kbits/50.1ms = 200kbits/sec ☹

Think about this…

- What if we sent two packets before waiting for an ACK
  - What if we sent N packets?
  - How many packets do we need to send before we use up the capacity of the link?

Performance is critical challenge in DS

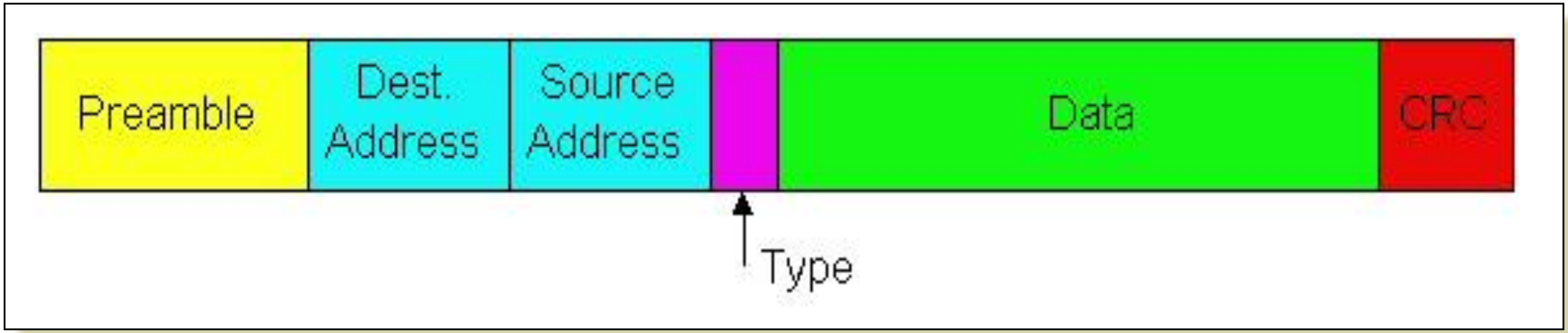
- Examples on HW1

Back to real-world networks: what does a network packet look like?

What does a packet look like?

Example: Ethernet Packet

- Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

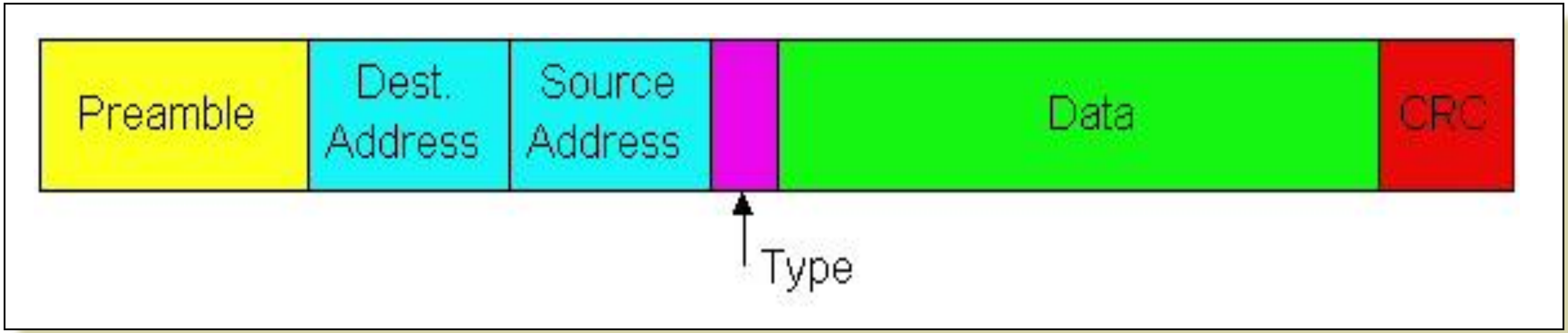


- Addresses:
  - 6 bytes
  - Each adapter is given a globally unique address at manufacturing time
    - Address space allocated to manufacturers, 24 bits identify manufacturer: e.g., 0:0:15:* → 3com
    - Frame is received by all adapters on a LAN and dropped if address does not match
  - Special addresses
    - Broadcast – FF:FF:FF:FF:FF:FF is “everybody”
    - Range of addresses allocated to multicast

What does a packet look like?

Example: Ethernet Packet

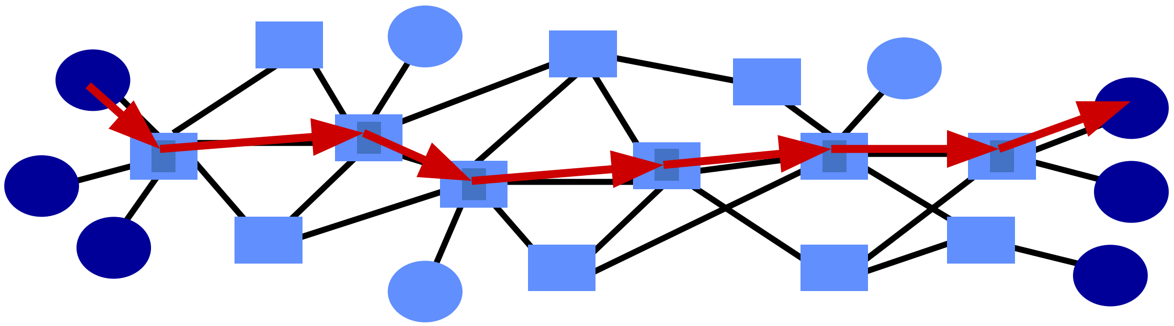
- Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame



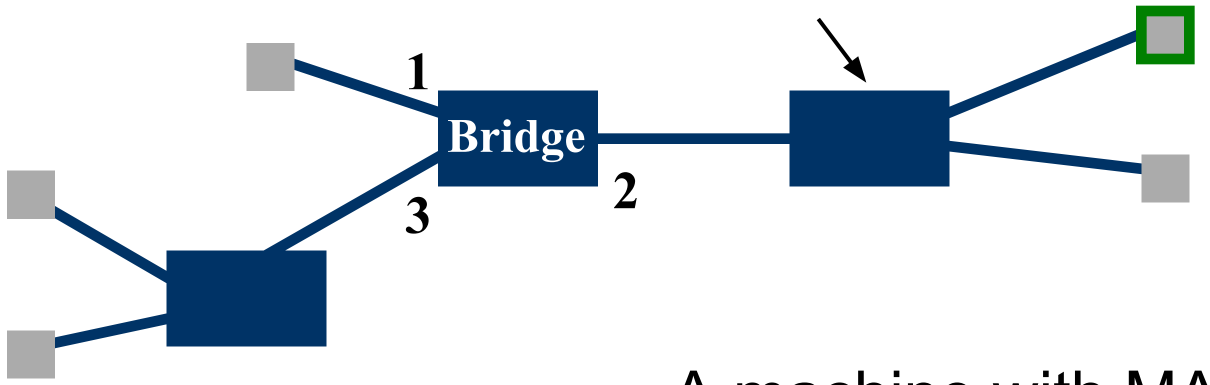
- Each protocol layer needs to provide hooks to upper layer protocols
  - Demultiplexing: identify which upper layer protocol packet belongs to
  - E.g., port numbers allow TCP/UDP to identify target application
  - Ethernet uses Type field
- Type: 2 bytes
  - Indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk

Packet Switching

- Source sends information as self-contained packets that have an address.
  - Source may have to break up single message in multiple packets
- Each packet travels independently to the destination host.
  - Switches use the address in the packet to determine how to forward the packets
  - Store and forward
- Analogy: a letter in surface mail.



Frame Forwarding



MAC Address	Port	Age
A21032C9A591	1	36
99A323C90842	2	01
8711C98900AA	2	15
301B2369011C	2	16
695519001190	3	11

- A machine with <u>MAC Address</u> lies in the direction of number <u>port</u> of the bridge
- For every packet, the bridge “looks up” the entry for the packets destination MAC address and forwards the packet on that port.
  - Other packets are broadcast – why?
- Timer is used to flush old entries

Today's & Tuesday's Lecture

Network links and LANs

Inter-network Communication

Layering & Protocols

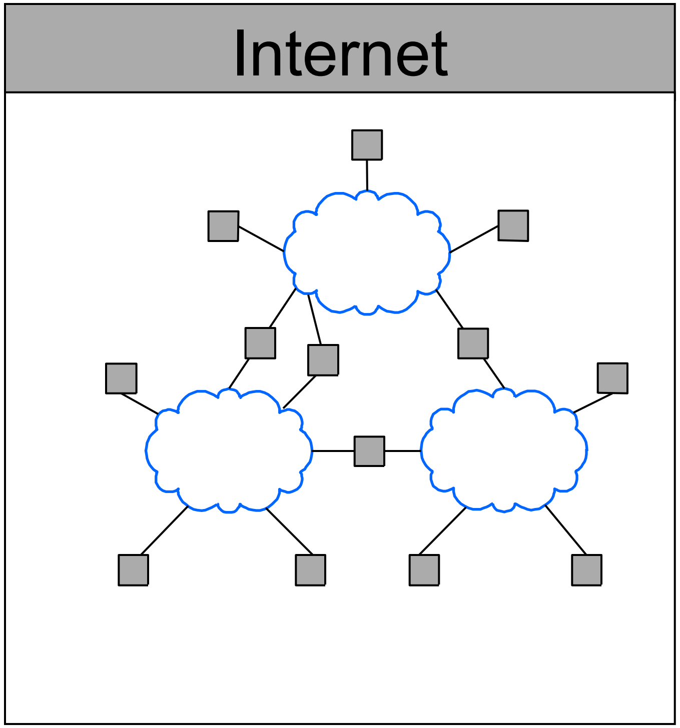
Internet design

Transport protocols

Application design

Internet

- An inter-net: a network of networks.
  - Networks are connected using routers that support communication in a hierarchical fashion
  - Often need other special devices at the boundaries for security, accounting, ..
- The Internet: the interconnected set of networks of the Internet Service Providers (ISPs)
  - About 17,000 different networks make up the Internet

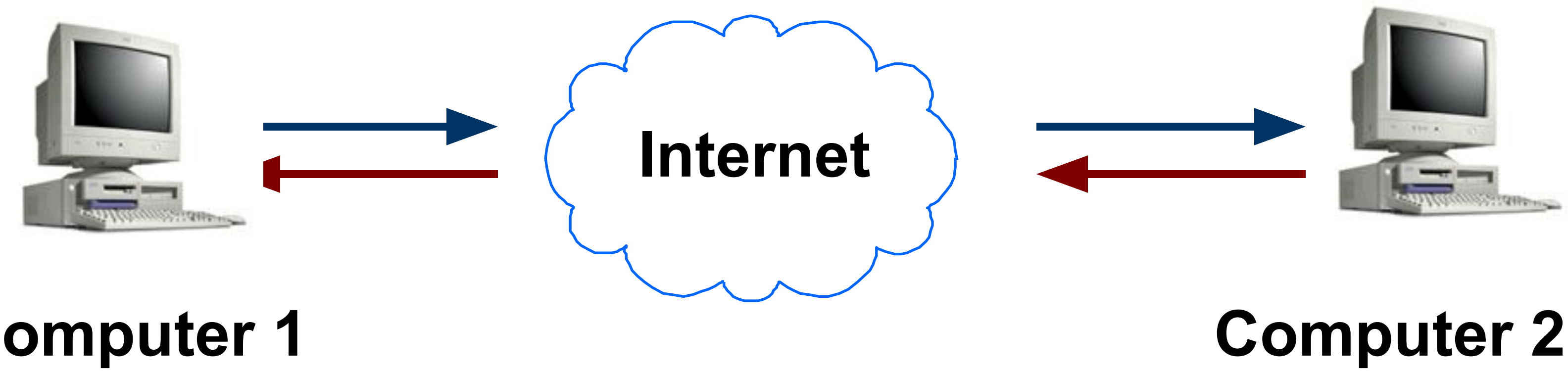


Challenges of an internet

- Heterogeneity
  - Address formats
  - Performance – bandwidth/latency
  - Packet size
  - Loss rate/pattern/handling
  - Routing
  - Diverse network technologies → satellite links, cellular links, carrier pigeons
  - In-order delivery

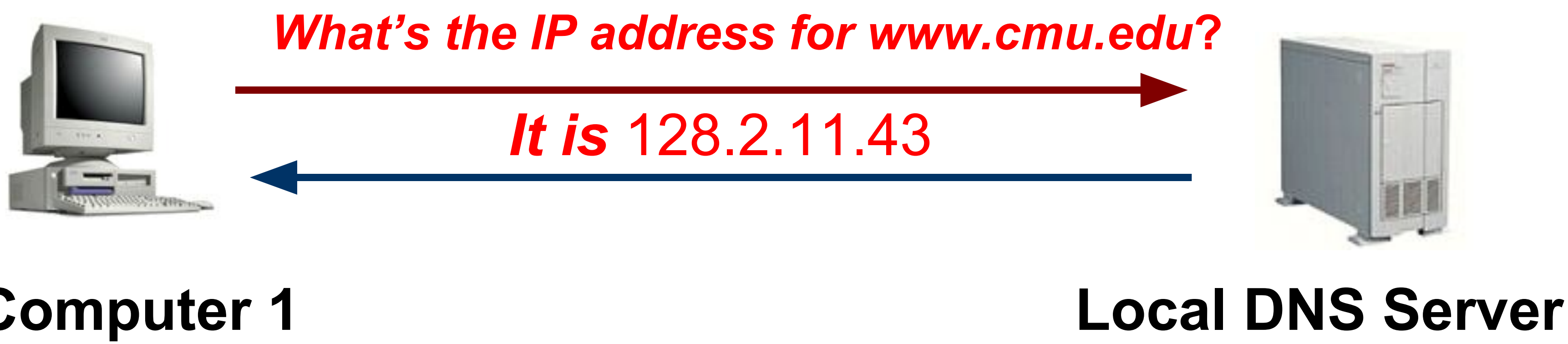
- Need a “standard” that everyone can use → IP

How To Find Nodes?



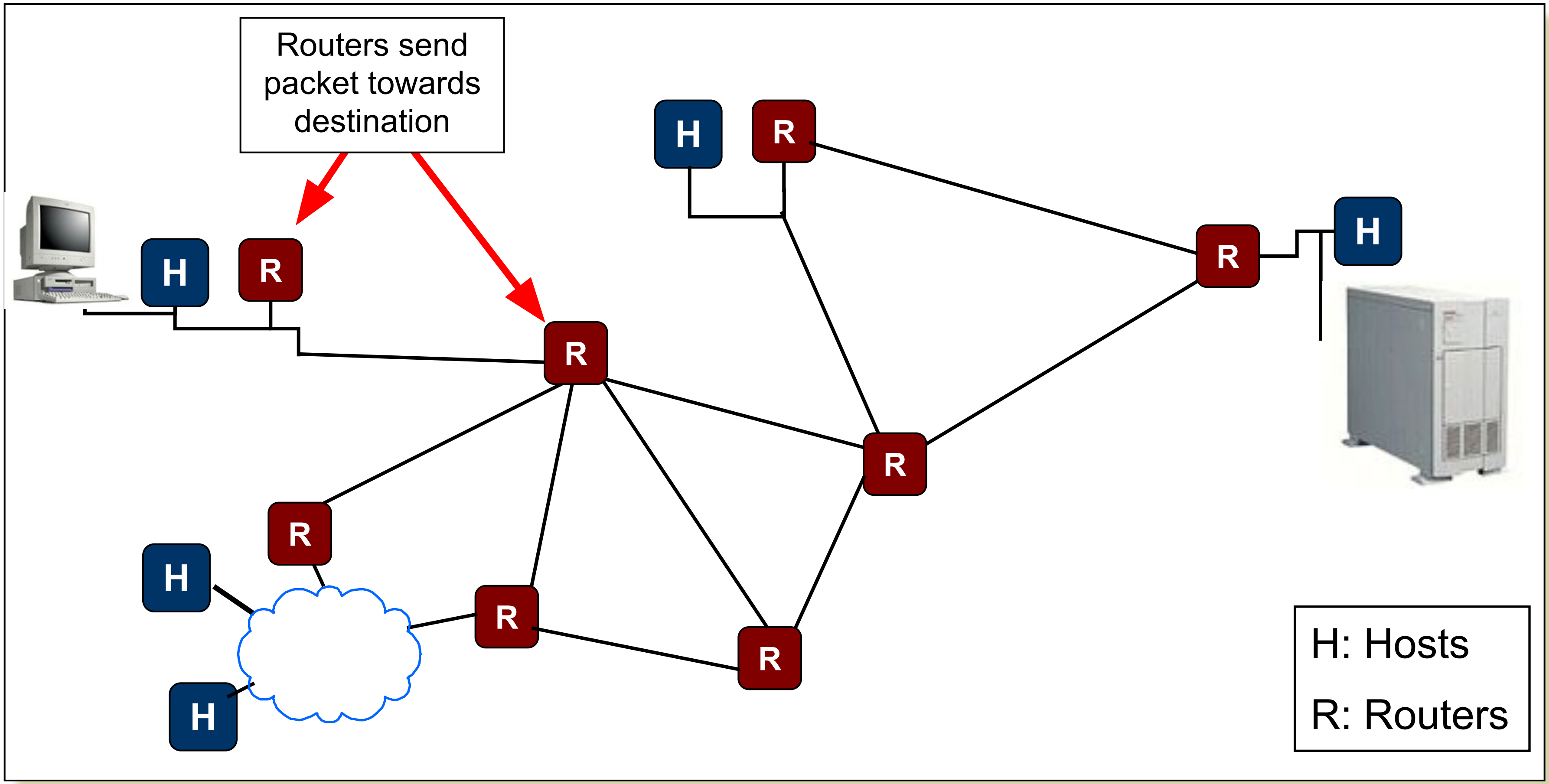
Need naming and routing

Naming



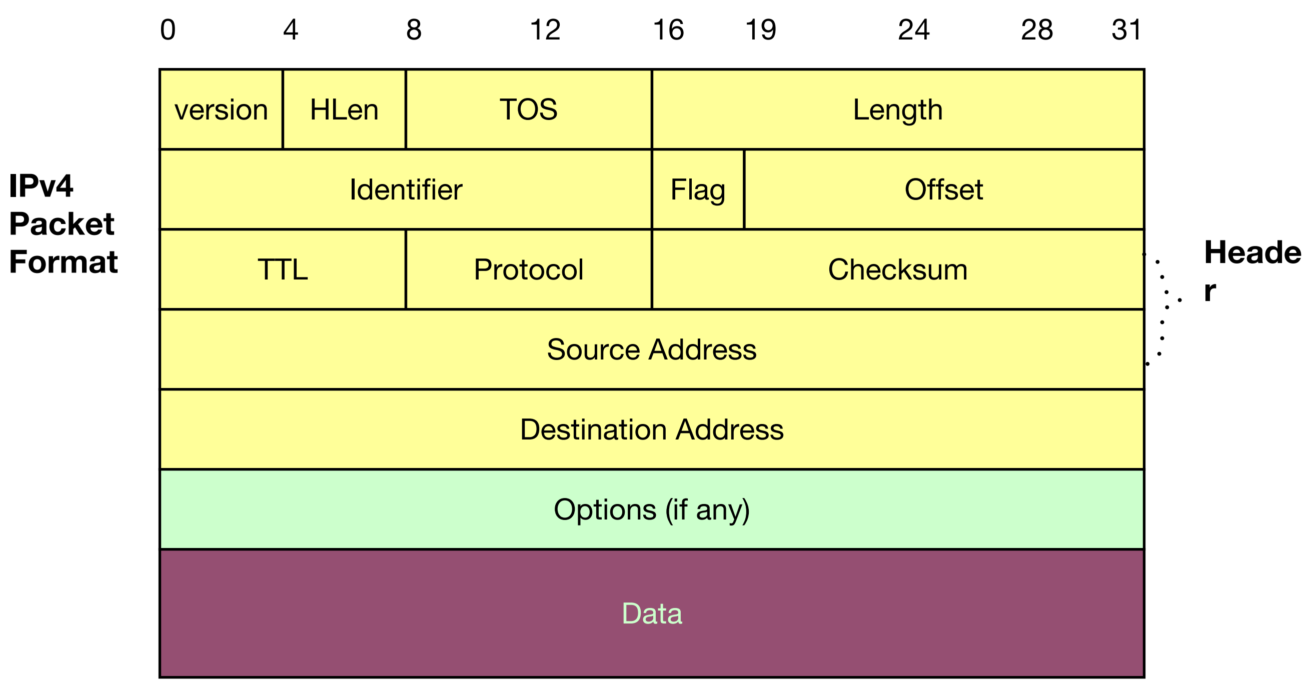
Translates human readable names to logical endpoints

Routing


Routers send packet towards destination
H
R
H
R
R
R
R
R
R
R
R
H
H
H
H: Hosts
R: Routers

IP Packets/Service Model

- Low-level communication model provided by Internet
- Datagram
  - Each packet self-contained
    - All information needed to get to destination
    - No advance setup or connection maintenance
  - Analogous to letter or telegram



Aside: Interaction with Link Layer

- How does one find the Ethernet address of a IP host?
- ARP
  - Broadcast search for IP address
    - E.g., “who-has 128.2.184.45 tell 128.2.206.138” sent to Ethernet broadcast (all FF address)
  - Destination responds (only to requester using unicast) with appropriate 48-bit Ethernet address
    - E.g, “reply 128.2.184.45 is-at 0:d0:bc:f2:18:58” sent to 0:c0:4f:d:ed:c6

IP Addresses: How to Get One?

Network (network portion):

- Get allocated portion of ISP's address space:

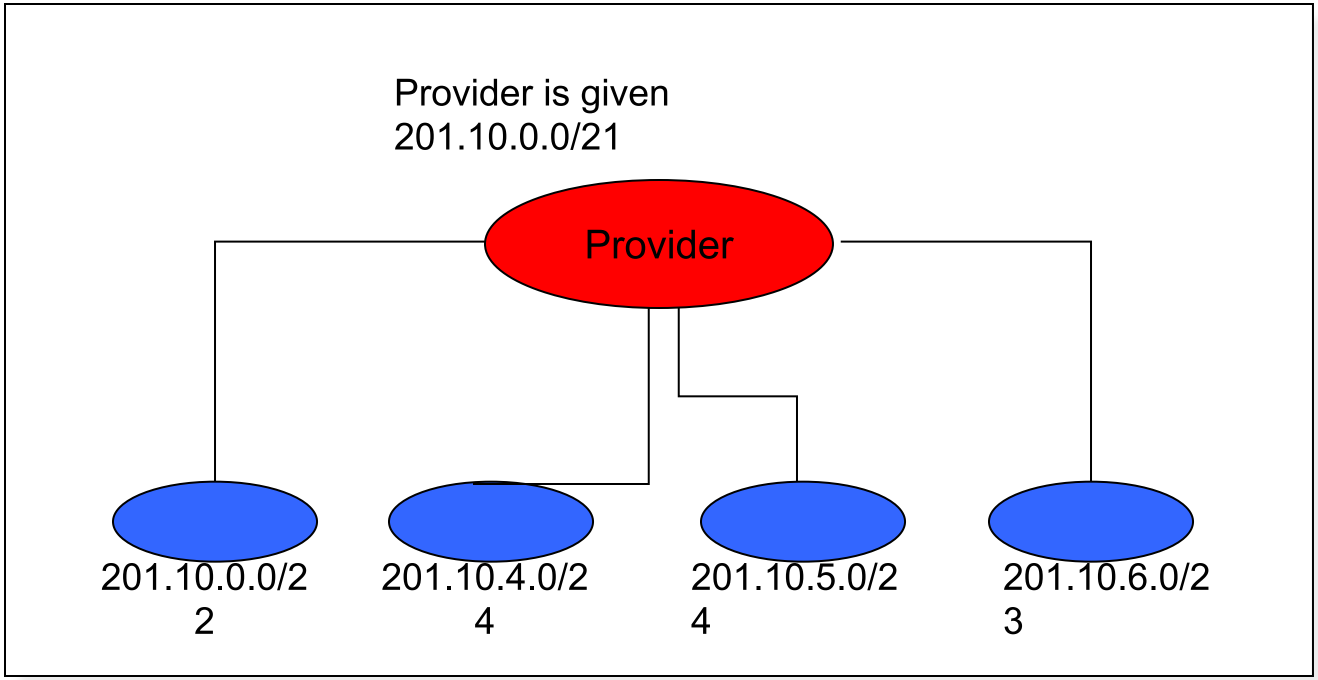
ISP's block	11001000 00010111 00010000 00000000	200.23.16.0/20
Organization 0	11001000 00010111 00010000 00000000	200.23.16.0/23
Organization 1	11001000 00010111 00010010 00000000	200.23.18.0/23
Organization 2	11001000 00010111 00010100 00000000	200.23.20.0/23
...	….. ….	….
Organization 7	11001000 00010111 00011110 00000000	200.23.30.0/23

IP Addresses: How to Get One?

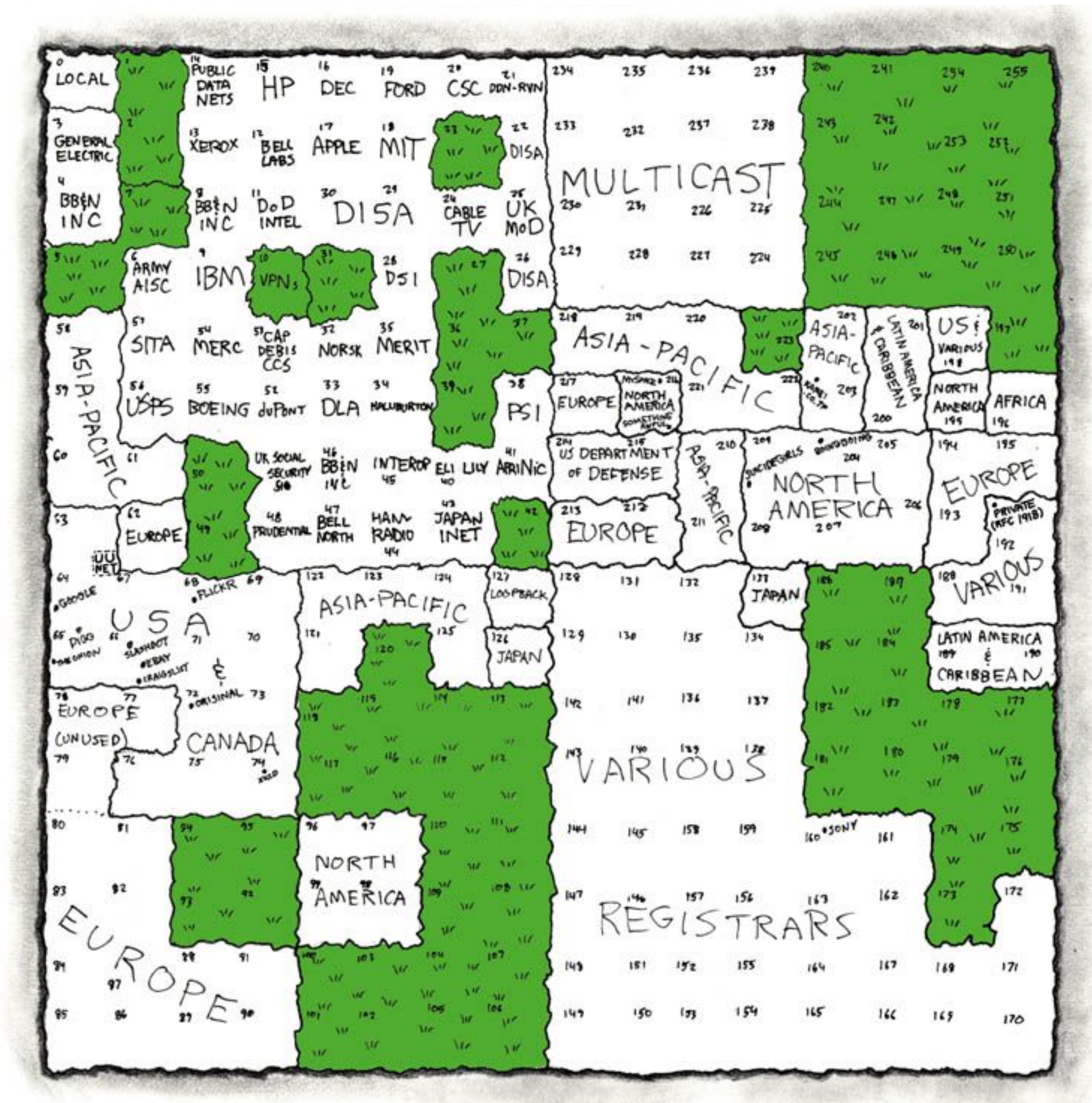
- How does an ISP get block of addresses?
  - From **Regional Internet Registries** (RIRs)
    - ARIN (North America, Southern Africa), APNIC (Asia-Pacific), RIPE (Europe, Northern Africa), LACNIC (South America)

- How about a single host?
  - Hard-coded by system admin in a file
  - DHCP: Dynamic Host Configuration Protocol: dynamically get address: "plug-and-play"
    - Host broadcasts "DHCP discover" msg
    - DHCP server responds with "DHCP offer" msg
    - Host requests IP address: "DHCP request" msg
    - DHCP server sends address: "DHCP ack" msg

CIDR IP Address Allocation

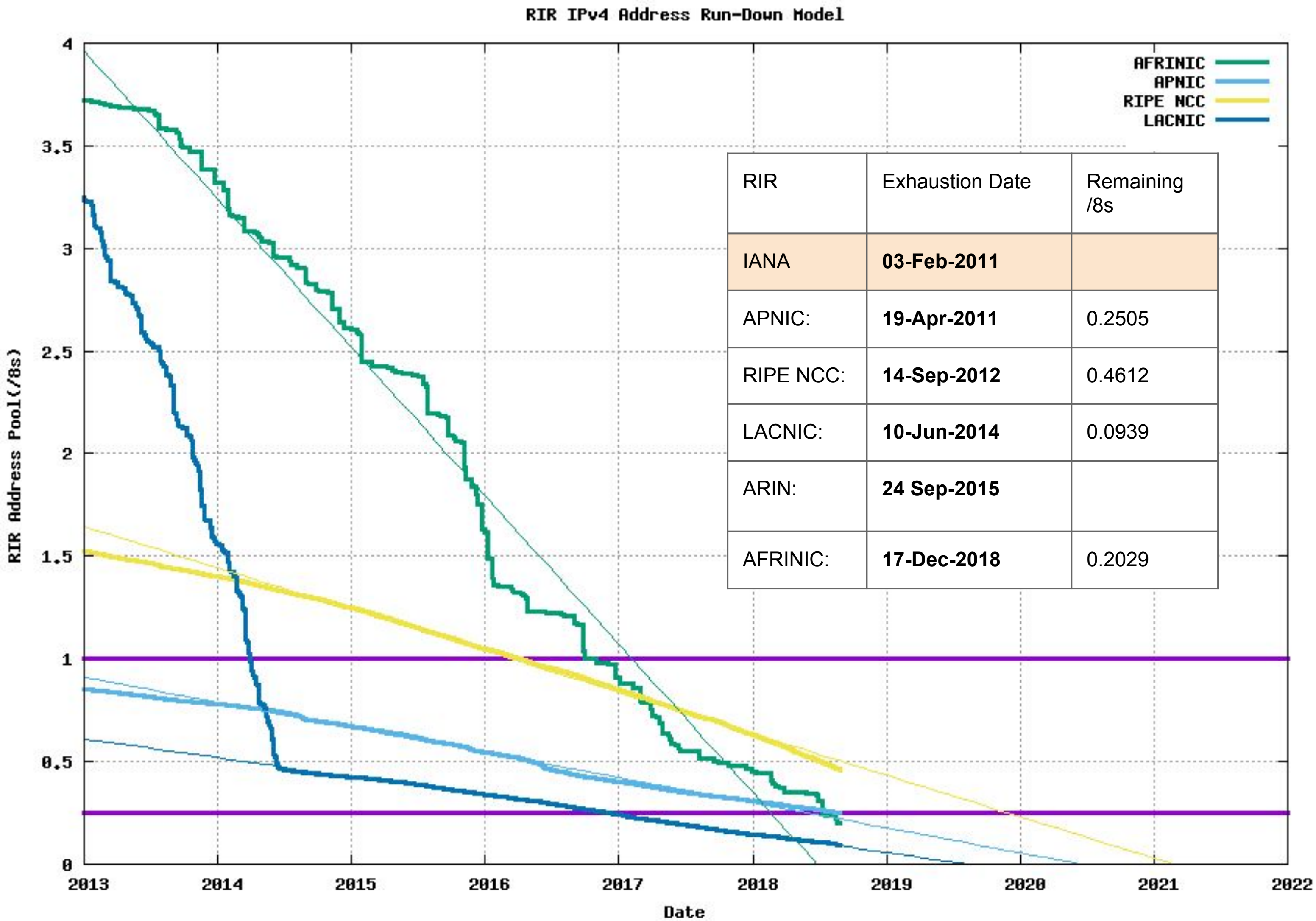


Map of the Internet (IPv4, 2006)



http://xkcd.com/195/

Map of the Internet (IPv4, 2006)



RIR	Exhaustion Date	Remaining /8s
IANA	**03-Feb-2011**	
APNIC:	**19-Apr-2011**	0.2505
RIPE NCC:	**14-Sep-2012**	0.4612
LACNIC:	**10-Jun-2014**	0.0939
ARIN:	**24 Sep-2015**	
AFRINIC:	**17-Dec-2018**	0.2029

https://www.potaroo.net/tools/ipv4/

What now?

- Last /8 given to RIR in 1/2011
- Mitigation
  - Reclaim addresses (give back class A=/8 networks)
  - More NAT?
  - Resale markets
  - Slow down allocation from RIRs to LIRs (i.e. ISPs)
- IPv6?

Next Tuesday’s Lecture

Network links and LANs

Inter-network Communication

Layering & Protocols

Internet design

Transport protocols

Application design