

---

# Internet Architecture I

## EECS 122: Lecture 2

---

Department of Electrical Engineering and Computer Sciences  
University of California  
Berkeley

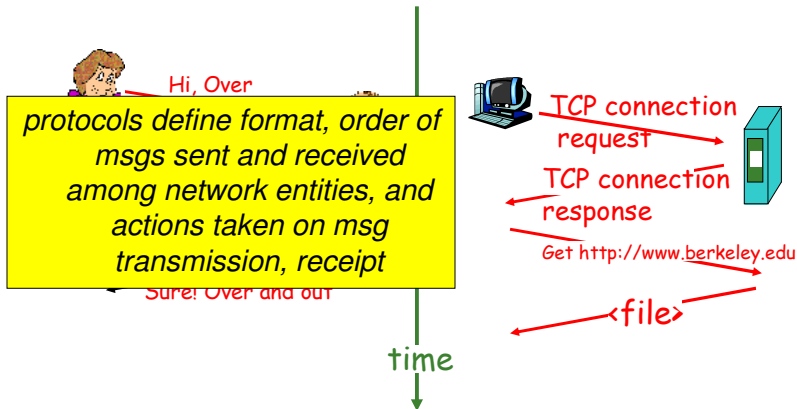
---

## Today's Outline

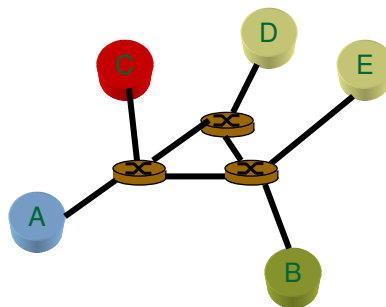
- Last lecture was a “lean back” overview of networks
- Today, let's begin “leaning forward”
  - How the functions of the internet are organized into layers
  - How these functions lead to a protocol stack
  - The End-to-End Principle
  - Packet Sniffing

# What's a protocol?

a human protocol and a computer network protocol:



# Back to a simple network...

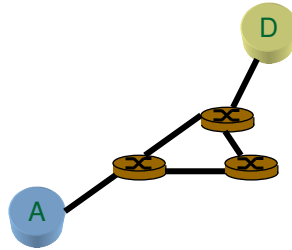


## Review:

Hosts connect to specialized devices called Routers. Routers are built (HW and Software) to discover the topology, select good paths and forward messages quickly. They also respond to changing network conditions.

“Hey, David!”

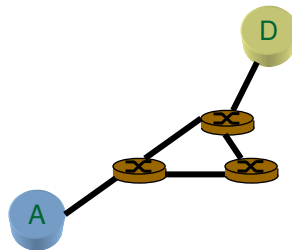
101110001111



Digitize the message

“Hey, David!”

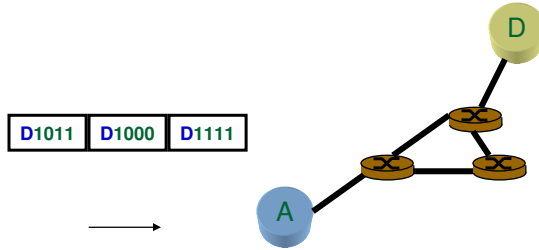
1011	1000	1111
------	------	------



Need a host address!

Packetize the message...  
Assume equal packet sizes for simplicity although this isn't true in IP

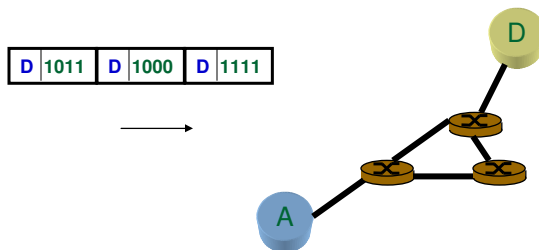
“Hey, David!”



Fix address width  
Need an address  
field!

Add a destination label

“Hey, David!”

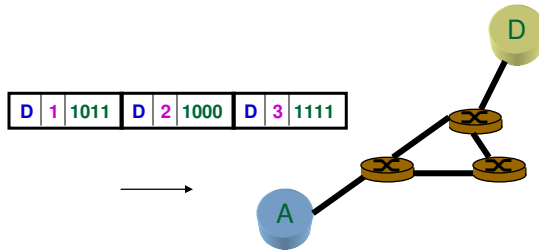


What if packets arrive  
out of order?  
What if packets are  
lost?

Add Sequence Numbers!

Now the routers can look for a fixed width address  
How many bits wide should it be if our network can be at  
most 1000 nodes?  
In IP the address is 32 bits wide

## “Hey, David!”

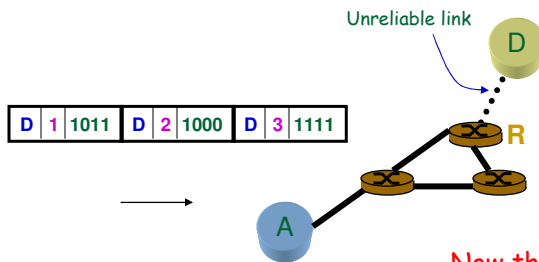


What if packets arrive out of order?  
What if packets are lost?

Add Sequence Numbers!

- Node D can reorder packets by sequence number
- Can run a protocol between the hosts that will cause A to retransmit lost packets

## “Hey, David!”

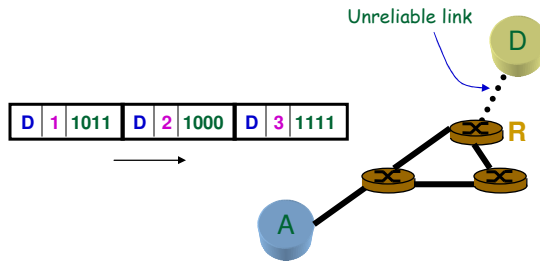


Packets are mostly lost over the unreliable link.

Now there are two data recovery protocols running simultaneously

Retransmitting all the way from A adds load to the network  
A protocol between R and D which recovers packets lost over the link could get the message to D much faster than having all retransmissions sent from A

## “Hey, David!”

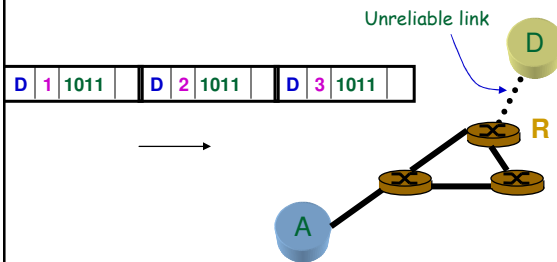


Packets are mostly lost over the unreliable link. No need to retransmit all the way from A

Add field for link sequence numbers...

Why not just use the host sequence numbers?

## “Hey, David!”



What if packet 2 is lost before reaching router, R...?

If the link device at D is using the purple seq numbers it will keep asking for 2

The link sequence number field is filled out/rewritten at each hop.

## What's an Application Protocol?

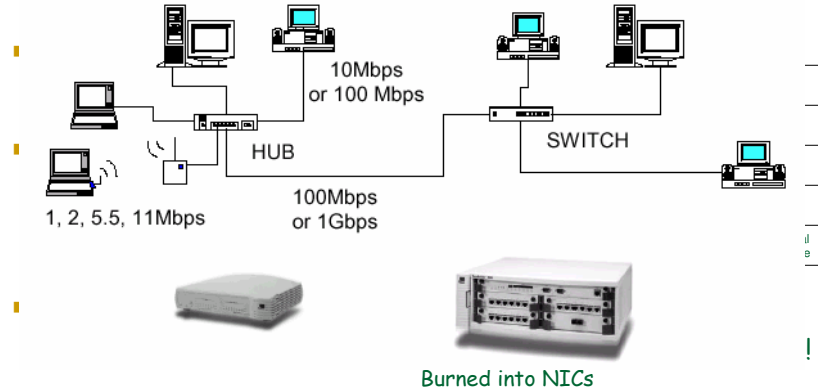
- Examples
  - The IM protocol
  - HTTP
  - Mail
  - Bittorrent
- What about a protocol that routers use to learn the network topology?

## What did we learn?

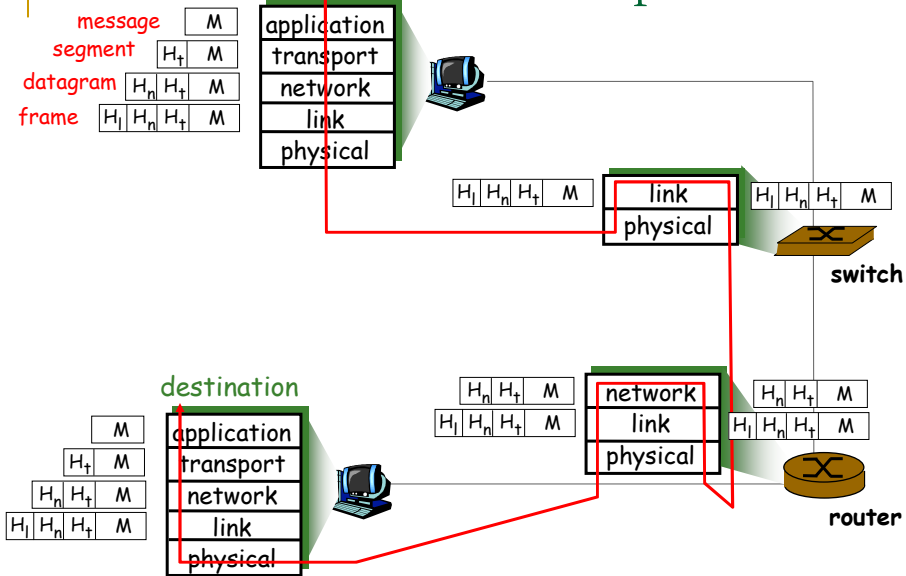
- Need separate headers for:
  - Link level reliable transmission
  - Network routing
  - Host-Host Reliable transmission: Transport
  - Application protocols
- Different devices look at different headers
  - Link Level: Data Link Control (DLC)
  - Network Level: Router
  - Transport Level: Hosts
- Two devices that look at the same header are **“peers”**
- Each set of peers runs its own protocols
- Protocols at one “level” can be improved/changed without affecting the correctness of protocols at another level

# What about a broadcast system?

- Example: Satellite.

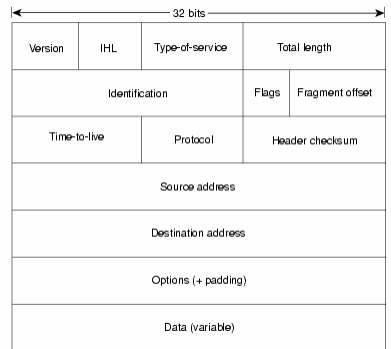


# Encapsulation

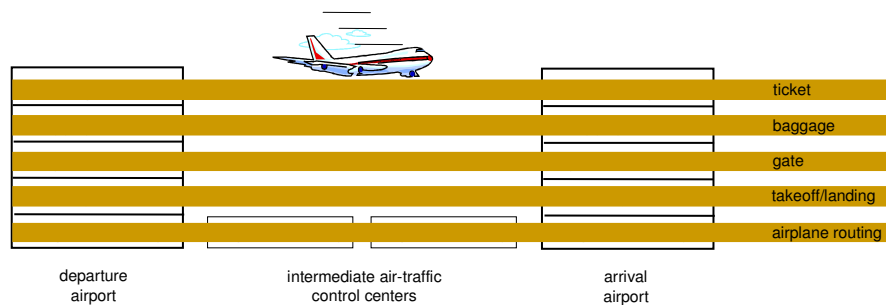




# The IP Packet Format



# Layering of airline functionality

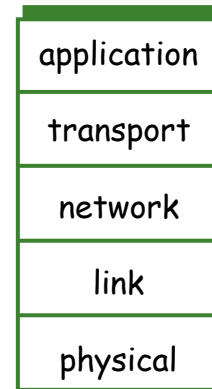


**Layers:** each layer implements a service

- via its own internal-layer actions
- relying on services provided by layer below

## Internet protocol stack

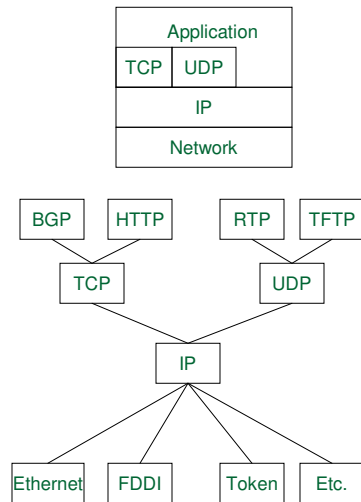
- **application**: supporting network applications
  - FTP, SMTP, HTTP
- **transport**: host-host data transfer
  - TCP, UDP
- **network**: routing of datagrams from source to destination
  - IP, routing protocols
- **link**: data transfer between neighboring network elements
  - PPP, Ethernet
- **physical**: bits “on the wire”



## Question

- Can multiple \_\_\_\_\_ protocols co-exist on the internet?
  1. Transport
  2. Routing
  3. Link
- Yes to all three!

## An Advanced View of Internet Layering



- Almost Any kind of application can write directly on IP
  - Including new transport protocols
- IP cannot be avoided
- As long as the routers speak IP, any application that can make do with datagram service can be written and implemented on the end devices.
  - No co-ordination, standards activity etc. is required!!

## Internet End-to-End Argument (Saltzer, Reed and Clark 1984)

- Implement a network function at the end hosts unless it cannot be implemented correctly in this manner.

OR

- “Don’t implement a function at the lower levels of the system unless it can be completely implemented at this level” (Peterson and Davie)
- This principle was a fundamental guiding principle for the first phase of the internet

## E2E Argument Applied

- Routing
  - Just doing it on the hosts would eliminate routers...and the internet would not scale
- Error Recovery
  - Can't just implement it at the lower layers since packets can be lost within the host as well
    - Must implement at the host so don't implement it at the link level
- Security
  - Similar argument as for error recovery
- E2E argument is a great principle, but it is violated a lot

## The downside of layering

- Efficiency
  - Suboptimal network behavior
    - Transport layer can't get information from the routers
  - Added Overhead
    - Fragmentation and reassembly
- Confusion in actual networks
  - What layer does the function "security" belong to?
  - Is routing just a network layer function?
  - Layer 2 , Layer 4 and Layer 4-7 switches!

## Example: Layer 4-7 switch

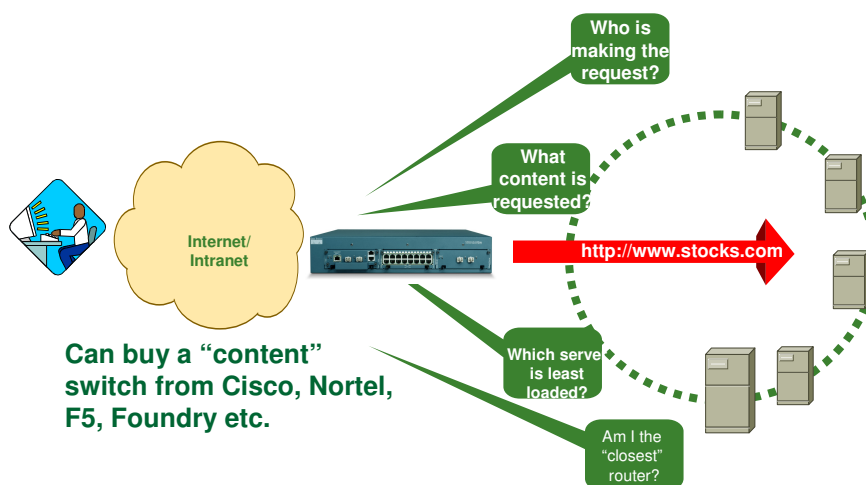
- For <http://www.stocks.com>
- What if we want many servers to appear as one?
  - Load balancing for server performance
    - Number of TCP connections
    - Server capacity
  - Load balance over different geographies
- All requests for that url go to a device called a Layer 4-7 switch

January 19, 2006

EECS122 Lecture 2 (AKP)

25

## A Layer 4-7 Switch



January 19, 2006

EECS122 Lecture 2 (AKP)

26

## Where are internet protocols standardized?

- The Internet Engineering Task Force:
  - <http://www.ietf.org>
- IAB
- Areas
- Working Groups
- Drafts, RFC, Proposed Standards
- Being forced to evolve to deal with the total commercialization of the internet
  - Good: Output has immediate global impact
  - Bad:
    - Many more demands of the network
    - Weaknesses in the architecture have “immediate global impact”

## Be a protocol detective...

- Download a free packet sniffer, Ethereal from [www.ethereal.com](http://www.ethereal.com)
- A packet sniffer allows you to examine the packets sent from and received at your host.
- Classified by protocol.
- This allows you to see how your host is communicating with other devices of the internet

## Summary/Questions

- How the functions of the internet are organized into layers
- How these functions lead to a protocol stack
- The End-to-End Principle
- Packet Sniffing

Next time we will study Delay Models and how to think about network performance