18-441/741: Computer Networks
Lectures 2: Protocol Stack

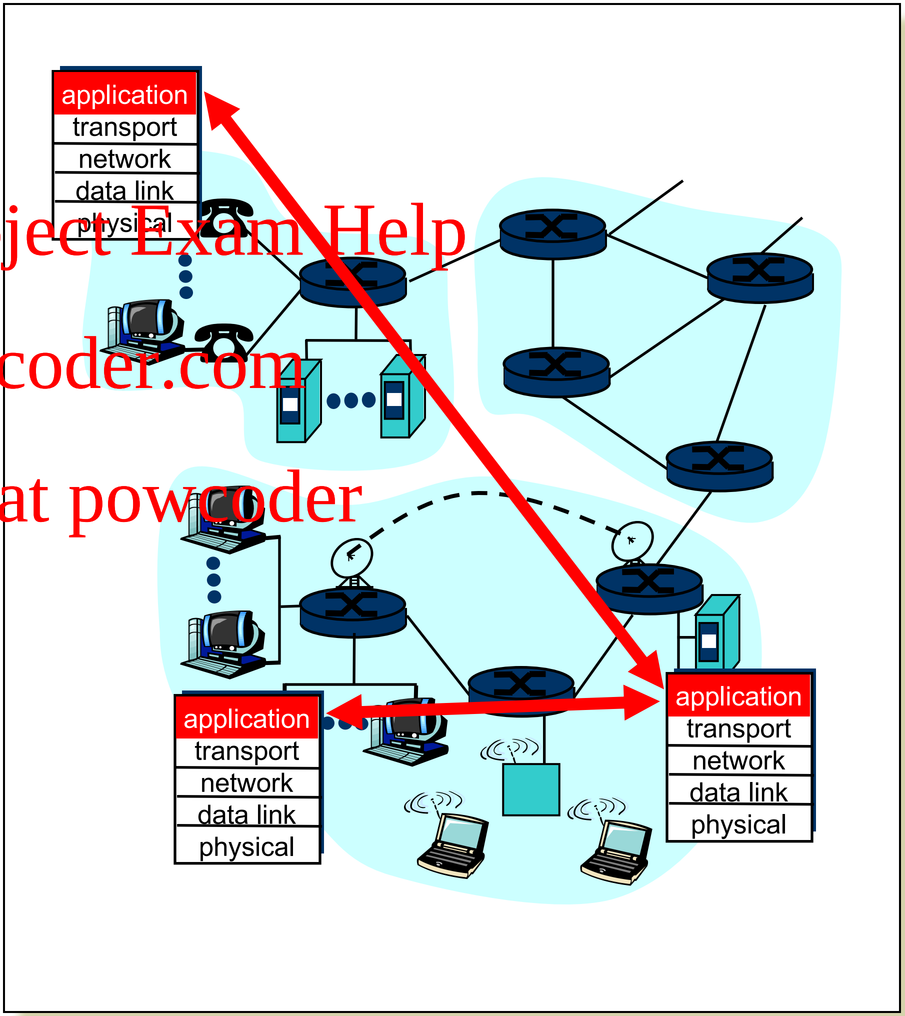
Swarun Kumar

Today's Lecture

- Network applications
  - Requirements
  - Latency and bandwidth
- Internet architecture
  - Protocols
  - A layered design
  - Life of a packet
- Network utilities

Applications and Application Protocols

- Application: communicating, distributed processes
  - Running in network hosts in "user space"
  - Exchange messages to implement app
  - e.g., email, file transfer, Web
- Application protocols
  - One "piece" of an app
  - Define messages exchanged by apps and actions taken
  - User services provided by lower layer protocols



Client-Server Paradigm

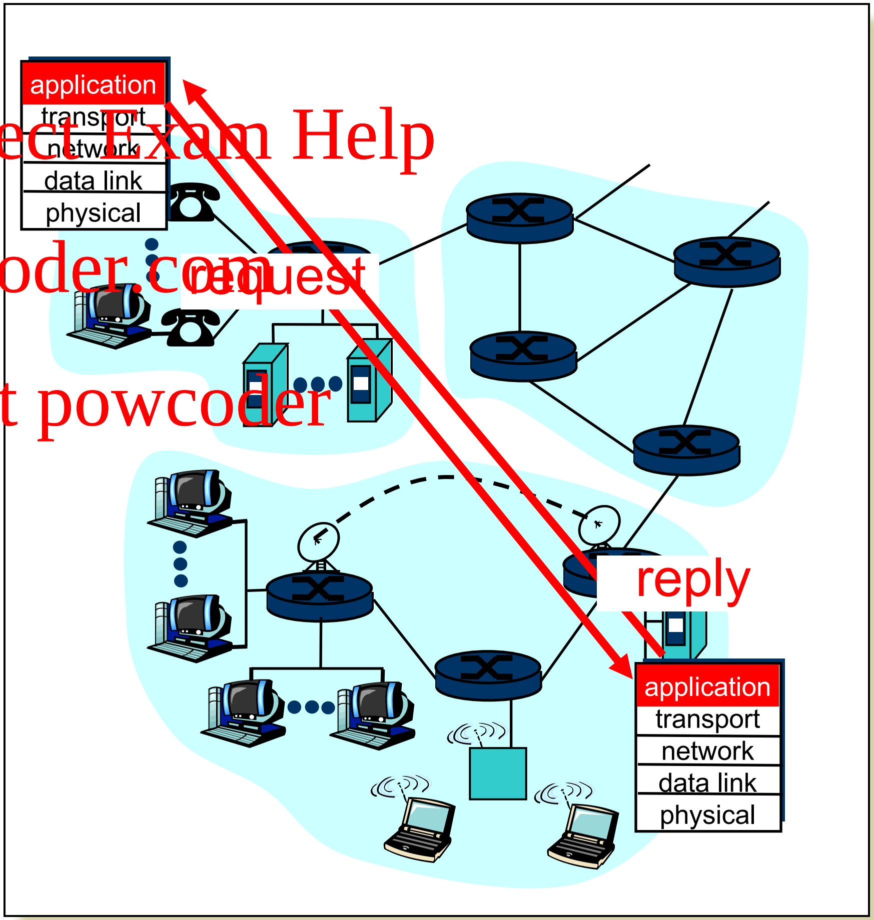
Typical network app has two pieces: *client* and *server*

Client:

- Initiates contact with server ("speaks first")
- Typically requests service
- Web: client is implemented in browser; e-mail: in mail reader

Server:

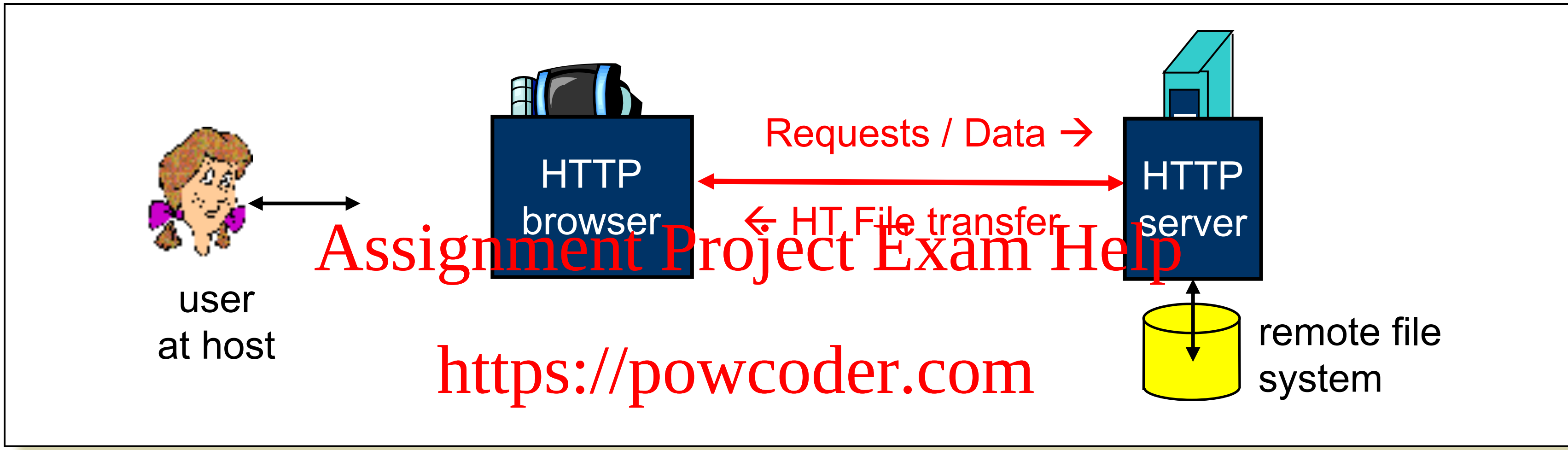
- Provides a service to client
- e.g., Web server sends requested Web page, mail server delivers e-mail



Yesterday's Applications

- FTP: transfer files to a host
  - No distributed file systems!
  - Mostly replaced by "the web" – http
- Telnet: use a computer remotely
  - Similar to ssh today (minus the security)
- Mail: exchange electronic e-mail
  - Similar today (kind of)
  - Initially host-to-host: name@my.computer.edu
- Already very useful!

HTTP: The Hypertext Transfer Protocol



- Transfer (hypertext) files as they are requested
- Client/server model that allows clients to access multiple servers as per their need
- The Hypertext markup language (HTML) allows for describing rich content (video, text, audio, images, etc)
- Project: will learn more about HTTP

Today's Applications

- Amazon, Facebook, etc.
  - 2009 quote: "*Amazon* found every 100ms of *latency* cost them 1% in sales"

- Video streaming (YouTube, Netflix, …)
  - Accounts for very high percentage of bandwidth
  - Interactive versus broadcast versus playback
  - What matters most?

- Audio and video conferencing (Skype, Facetime,…)
  - Traditional telephone app
  - What matters most?

Requirements

- Performance: latency and throughput
- Network reliability
  - Network service must always be available
- Security: for users and the network itself
  - Privacy, authentication, deal with various attacks, …
  - Attacks on the network, versus enabled by the network
- Scalability.
  - Scale to large numbers of users, traffic flows, …
- Manageability: monitoring, enforcing policies, billing, …

What Service Does an Application Need?

Data loss

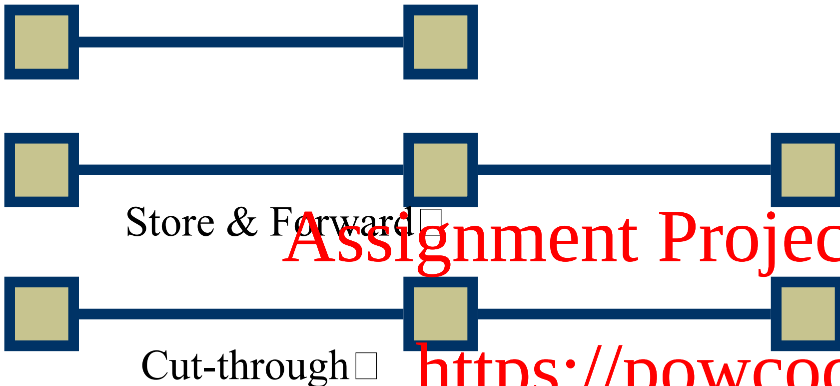
Timing

Bandwidth

Transport Service Requirements of Common Apps

Application	Data loss	Bandwidth	Time Sensitive
file transfer	no loss	elastic	no
e-mail	no loss	elastic	no
web documents	no loss	elastic	no
real-time audio/ video	loss-tolerant	audio: 5Kb-1Mb video:10Kb-5Mb	yes, 100's msec
stored audio/video	loss-tolerant	same as above	yes, few secs
interactive games	loss-tolerant	few Kbps	yes, 100's msec
financial apps	no loss	elastic	yes and no

A Closer Look at Packet Delay



Prop + xmit

2*(Prop + xmit)

2*prop + xmit

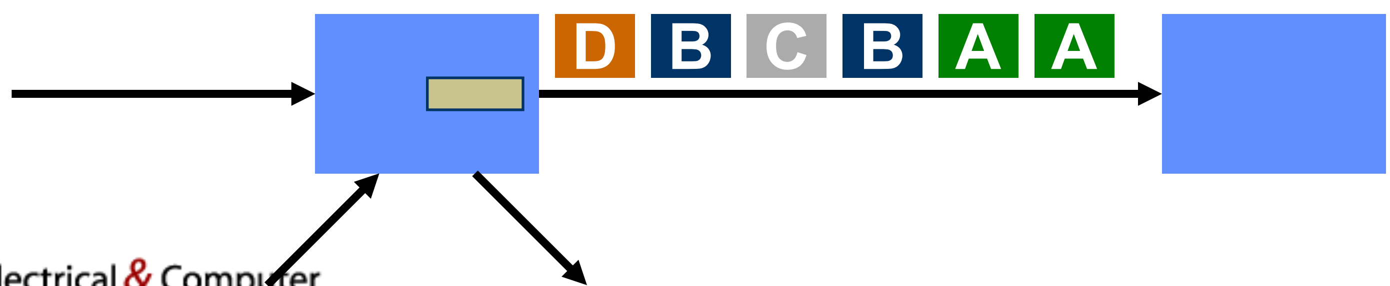
When does cut-through matter?

Next: Routers have finite speed (processing delay)

Routers may buffer packets (queueing delay)

Packet Delay Components

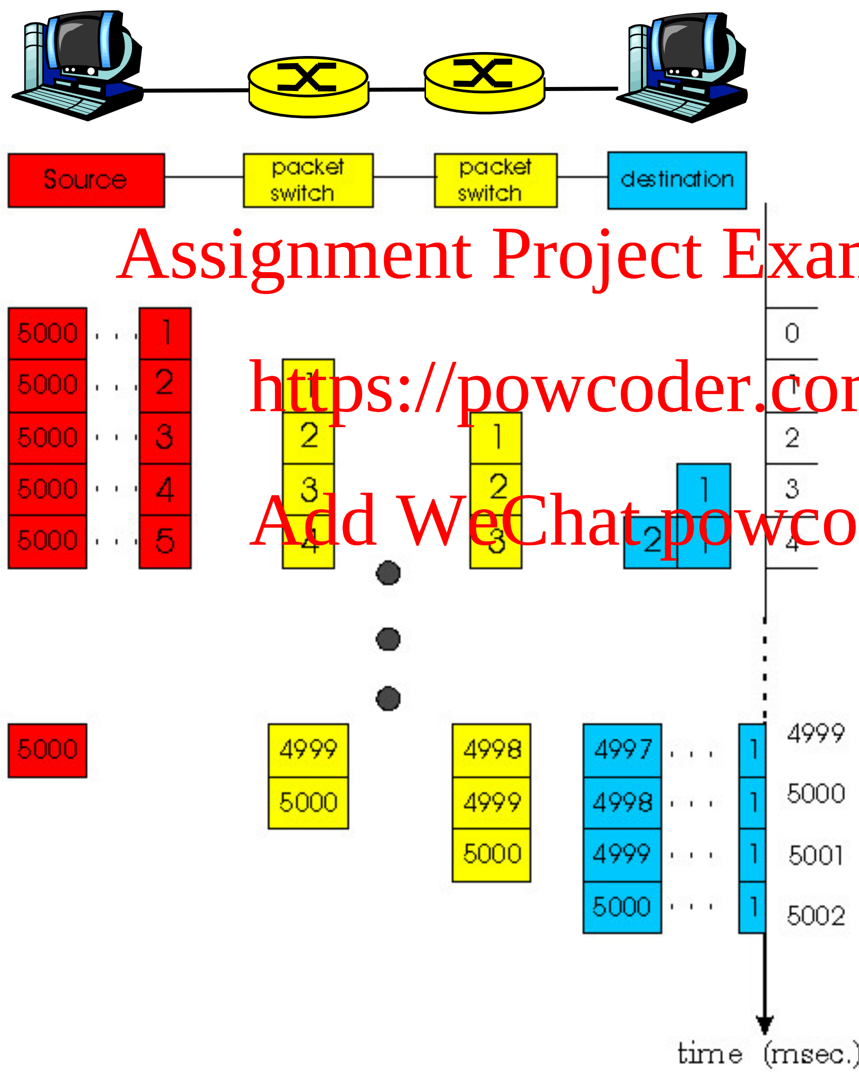
- Sum of a number of different delay components.
- Propagation delay on each link.
  - Proportional to the length of the link
- Transmission delay on each link.
  - Proportional to the packet size and 1/link speed
- Processing delay on each router.
  - Depends on the speed of the router
- Queuing delay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



A Word about Units

- What do “Kilo” and “Mega” mean?
  - Depends on context
- Storage works in powers of two.
  - 1 Byte = 8 bits
  - 1 KByte = 1024 Bytes
  - 1 MByte = 1024 Kbytes
- Networks work in decimal units.
  - Network hardware sends bits, not Bytes
  - 1 Kbps = 1000 bits per second
  - To avoid confusion, use 1 Kbit/second
- Why? Historical: CS versus ECE.

Application-level Delay



Delay of one packet

$$\text{Delay}^{*} + \frac{\text{Size}}{\text{Throughput}}$$

Average sustained throughput

Units: seconds + bits/(bits/seconds)

* For minimum sized packet

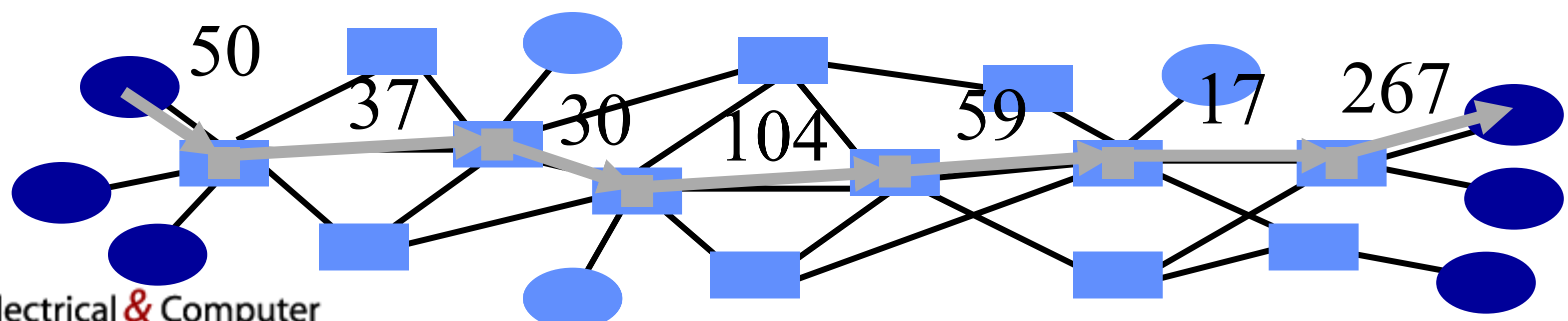
Sample Quiz Question

- How long does it take to send a 100 Kbit file?
  - Assume a perfect world.. Fill in the blanks..
- Is the transfer latency or throughput limited?
- What about a 10 Kbit file?

Latency \ Throughput	100 Kbit/s	1 Mbit/s	100 Mbit/s
500 µsec			
10 msec			
100 msec			

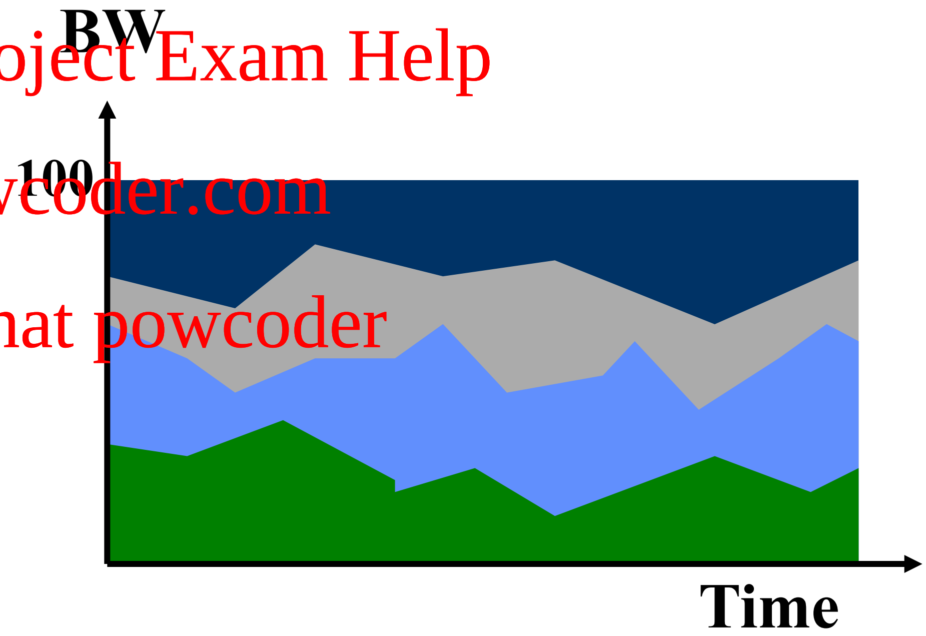
A Closer Look at Throughput

- 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 matter why the link is slow!
  - Low link bandwidth
  - Many users sharing the link bandwidth



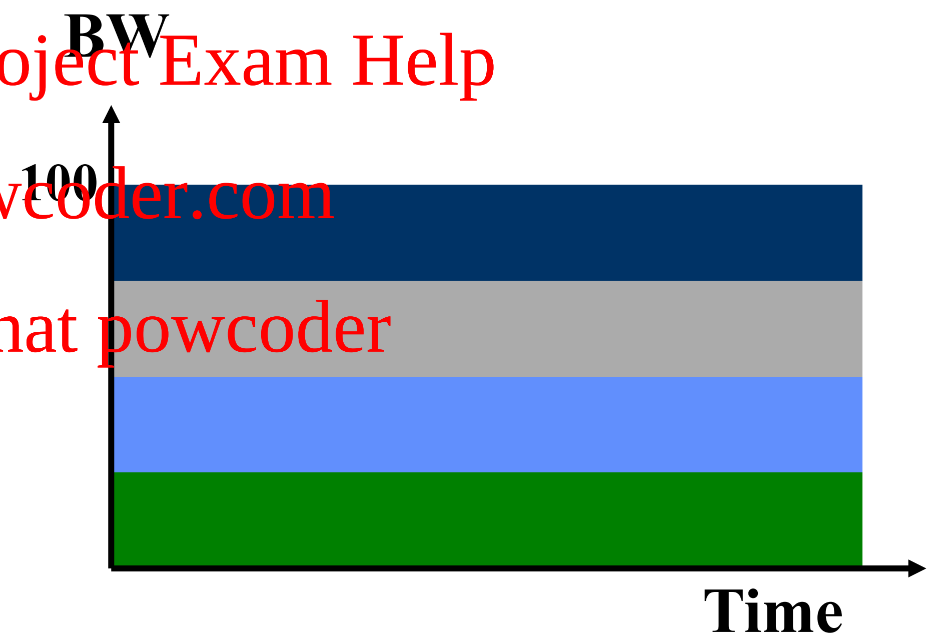
Bandwidth Sharing

- Bandwidth received on the bottleneck link determines end-to-end throughput.
- Router before the bottleneck link decides how much bandwidth each user gets.
  - Users that try to send at a higher rate will see packet loss
- User bandwidth can fluctuate quickly as flows are added or end, or as flows change their transmit rate.

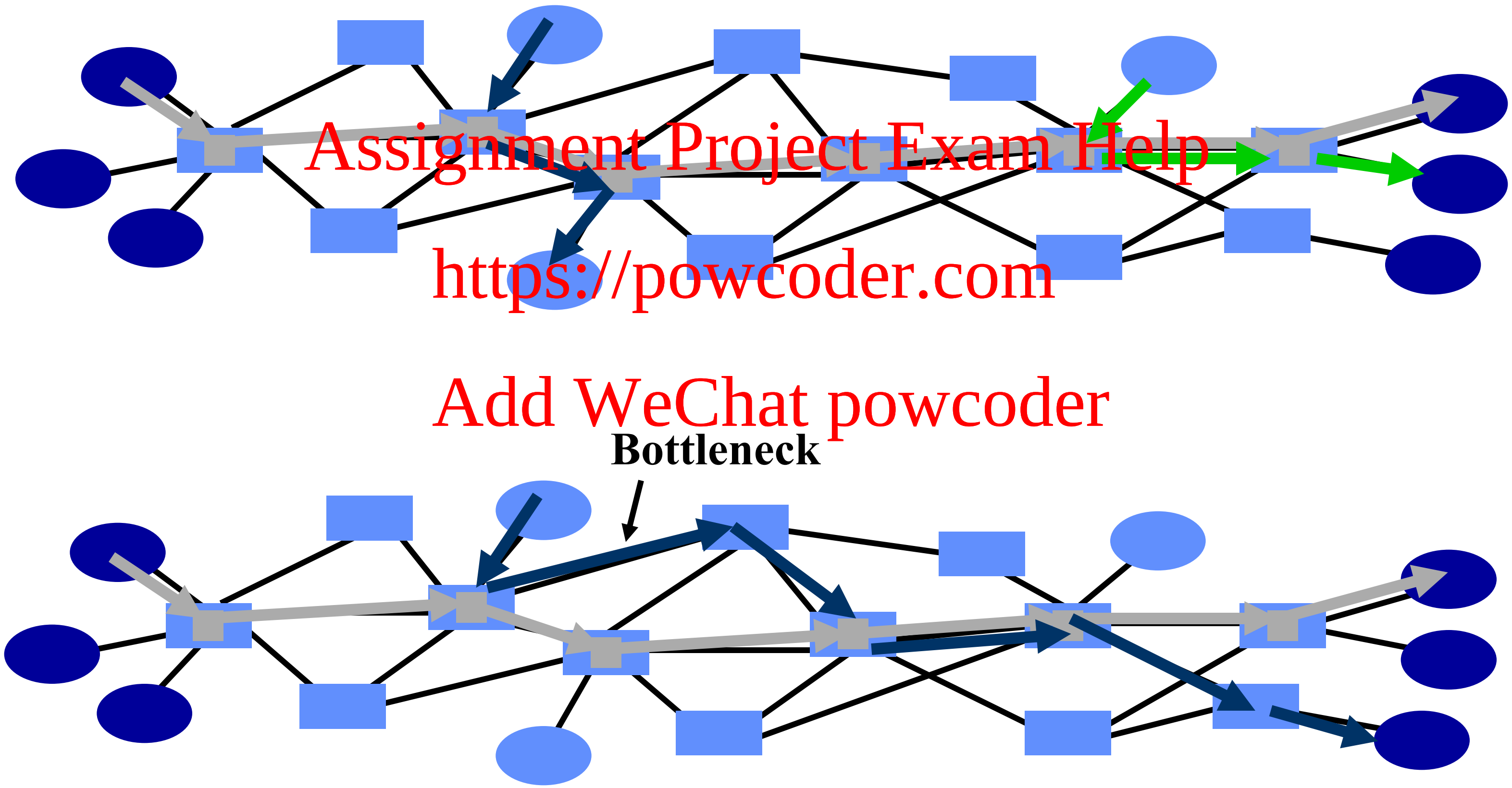


Fair Sharing of Bandwidth

- All else being equal, fair means that users get equal treatment.
  - Sounds fair
- When things are not equal, we need a policy that determines who gets how much bandwidth.
  - Users who pay more get more bandwidth
  - Users with a higher “rank” get more bandwidth
  - Certain classes of applications get priority



But It is Not that Simple



Electrical & Computer ENGINEERING

Today's Lecture

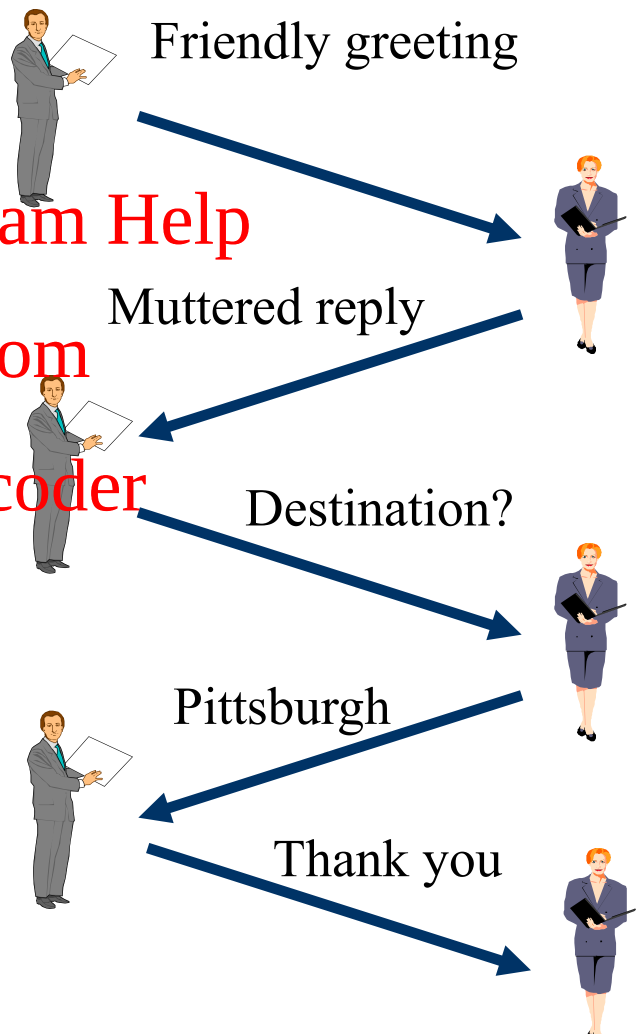
- Network applications
  - Requirements
  - Latency and bandwidth
- Internet architecture
  - Protocols
  - A layered design
  - Life of a packet
- Network utilities

Lots of Protocols (and Acronyms!)

- IP: Internet protocol
- UDP: User datagram protocol
- TCP: Transmission control protocol
- FTP: File transfer protocol
- SMTP: Simple mail transfer protocol
- HTTP: Hypertext transfer protocol
- ARP: Address resolution protocol
- BGP: Border gateway protocol
- ICMP: Internet control message protocol
- DHCP: Dynamic host configuration protocol
- And many more …

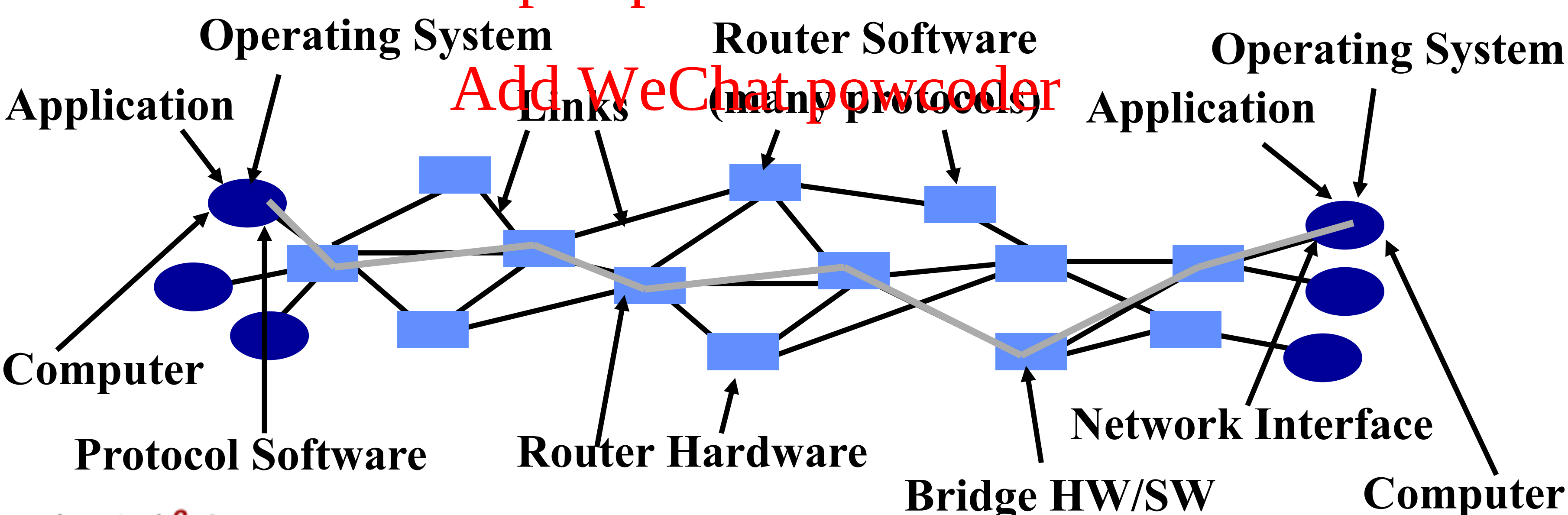
What are Protocols?

- An agreement between parties on how communication should take place
- Module in layered structure
- Protocols define: Interface to peer (syntax & semantics)
  - Actions taken on receipt of a messages
  - Format and order of messages
  - Error handling, termination, ordering of requests, etc.
- Example: Buying airline ticket

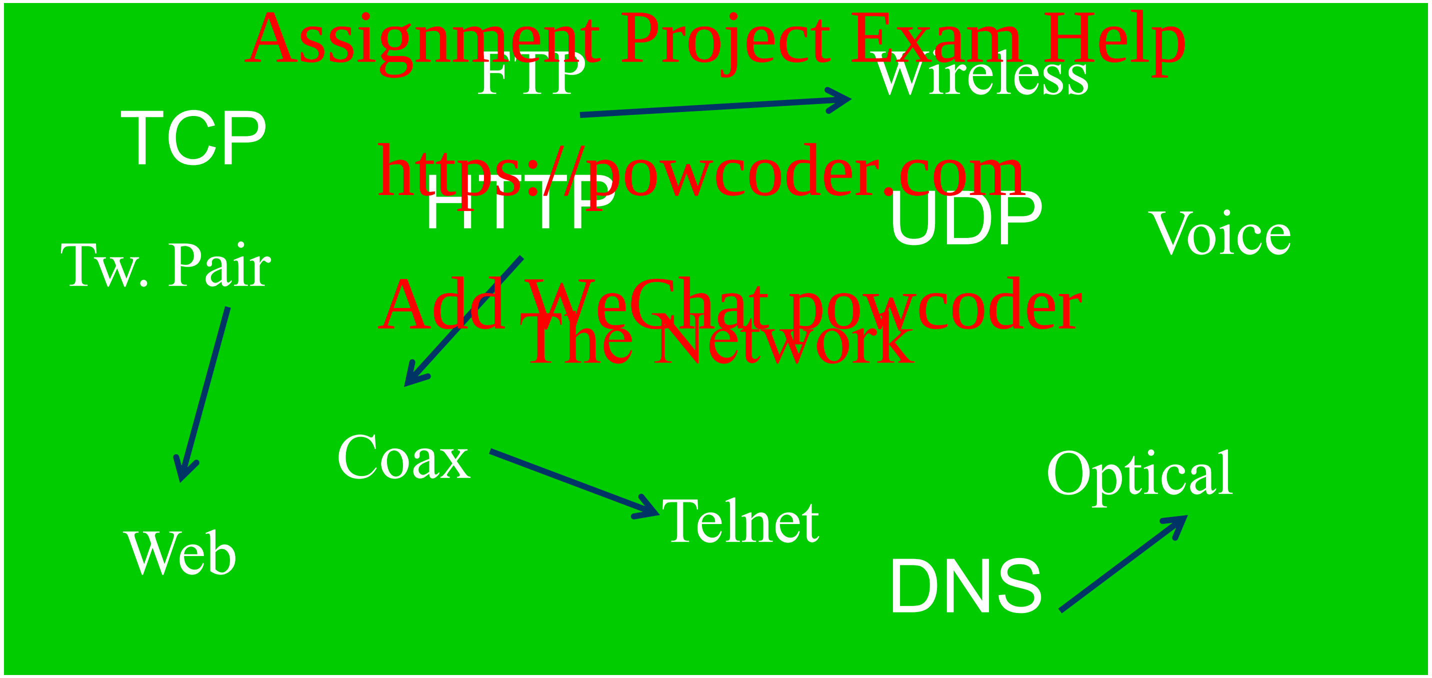


How to Design a Network?

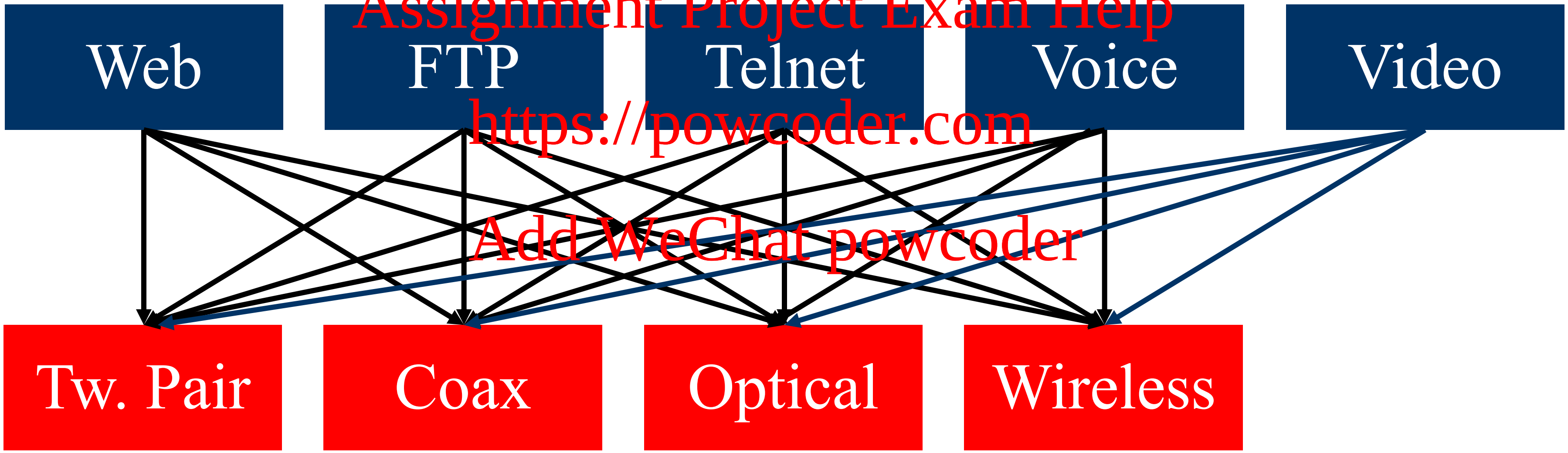
- Has many users
- Offers diverse services
- Mixes very diverse technologies
- Components built by many companies
- Diverse ownership
- Can evolve over time



Solution #1

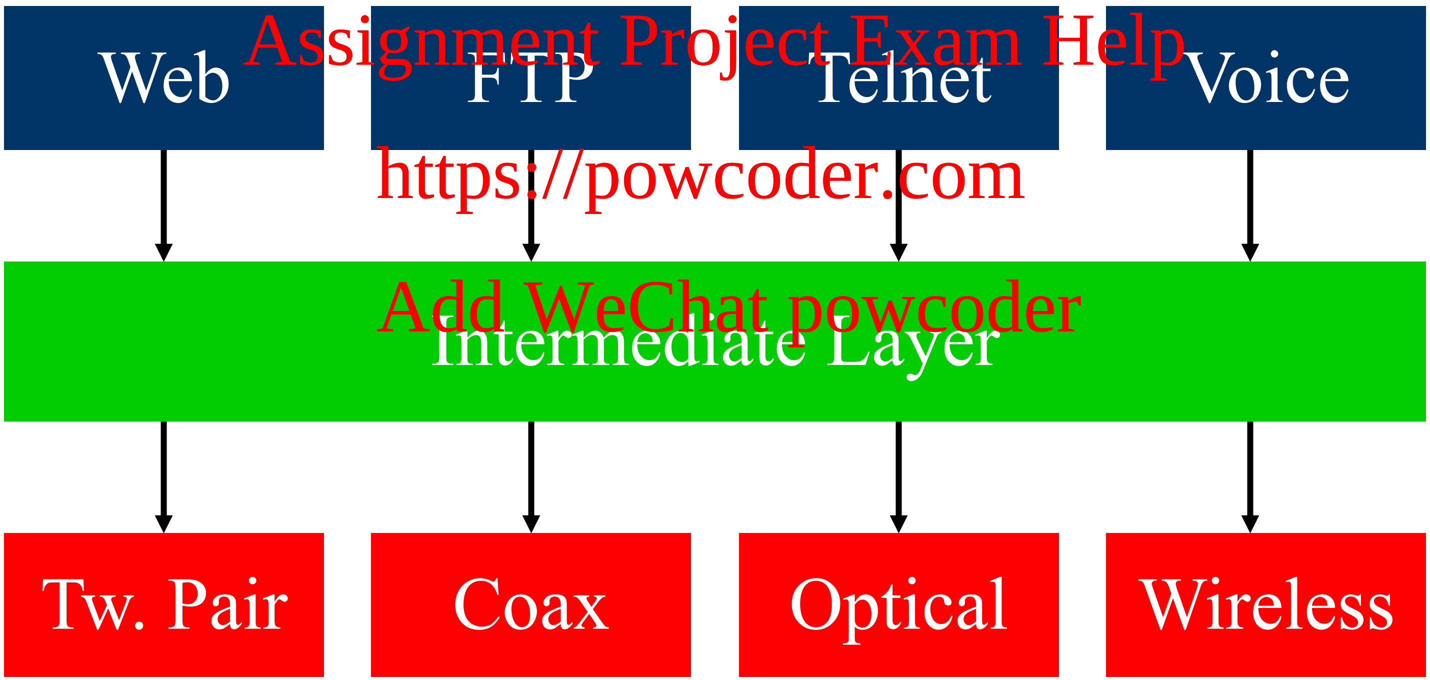

FTP
Wireless
TCP
HTTP
UDP
Voice
Tw. Pair
The Network
Coax
Telnet
Optical
Web
DNS

Solution #2?

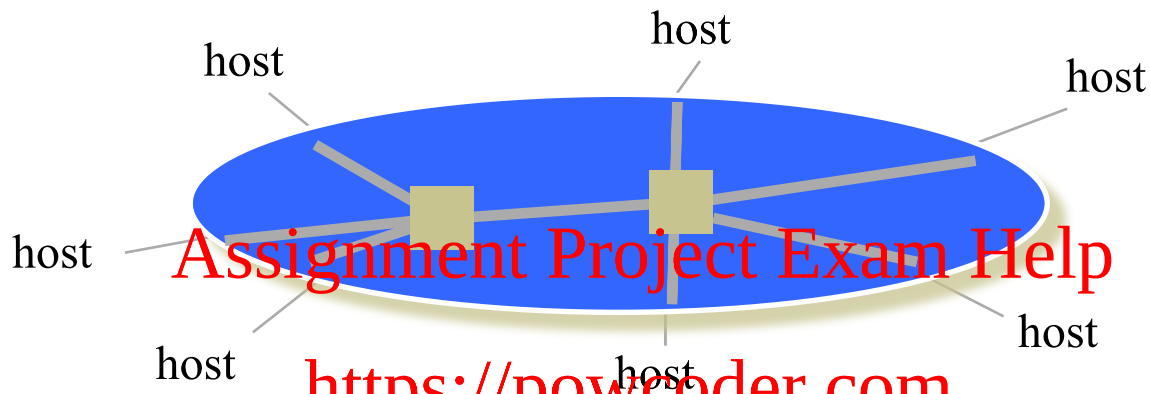


Electrical & Computer ENGINEERING

Solution #3

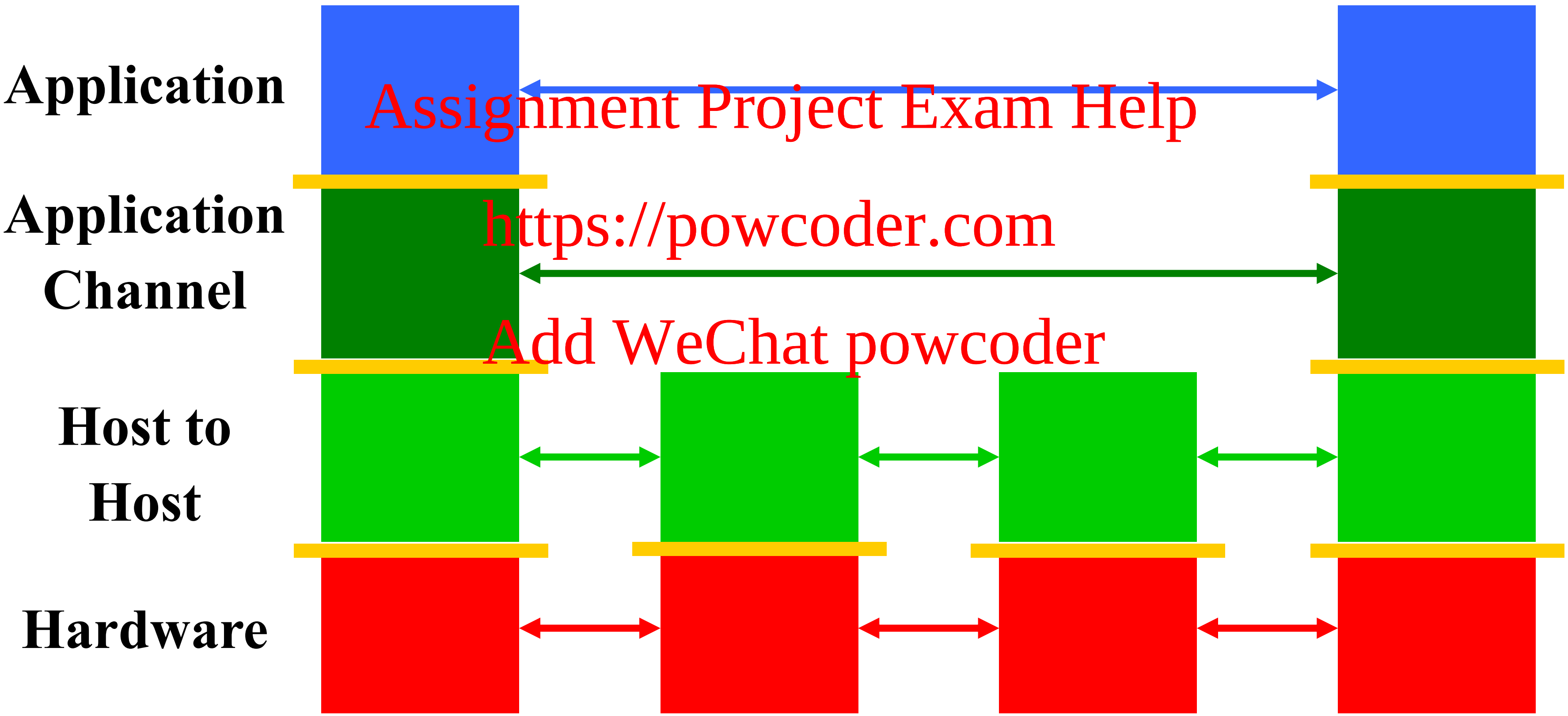

Web
FTP
Telnet
Voice
Intermediate Layer
Tw. Pair
Coax
Optical
Wireless

Types of Protocols



- Core network: responsible for transferring data between a sending and receiving host.
- End-to-end protocols: present a network service to applications and users.
  - May add value to the core network protocols
- Driven by differences in constraints: scalability, power, management, speed, etc.

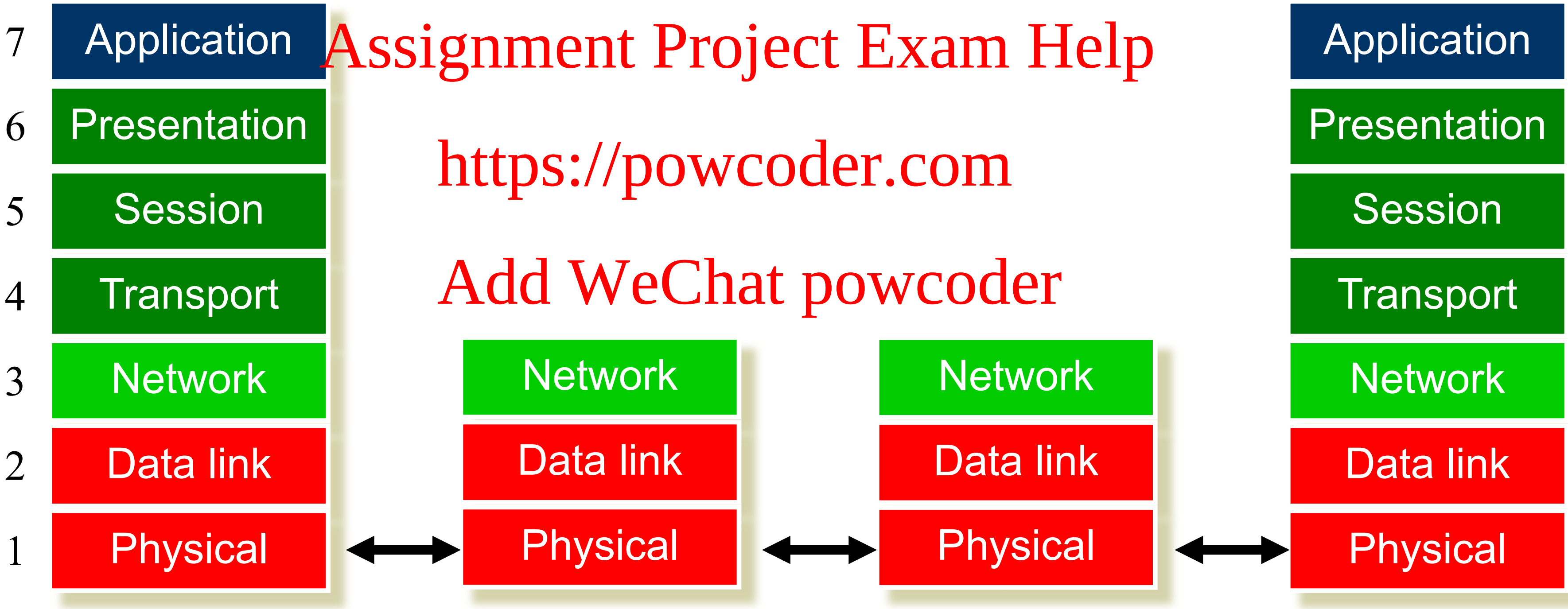
Protocol and Service Levels



Layering: modular approach to network functionality

A Layer Network Model

The Open Systems Interconnection (OSI) Model



Layering Characteristics

- Each layer relies on services from layer below and exports services to layer above
- Interface defines interaction with peer on other hosts – called protocols
- Modules hide implementation - layers can change without disturbing other layers (black box)

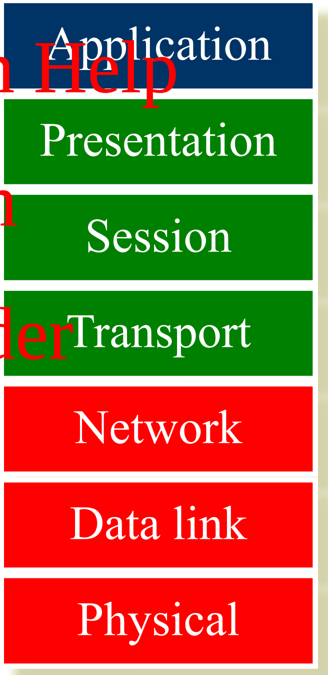
OSI Model: 7 Protocol Layers

- Physical: how to transmit bits
- Data link: how to transmit frames
- Network: how to route packets
- Transport: how to send packets end2end
- Session: how to tie flows together
- Presentation: byte ordering, security
- Application: everything else

- TCP/IP has been amazingly successful, and it is not based on rigid OSI model. The OSI model has been successful at shaping thought

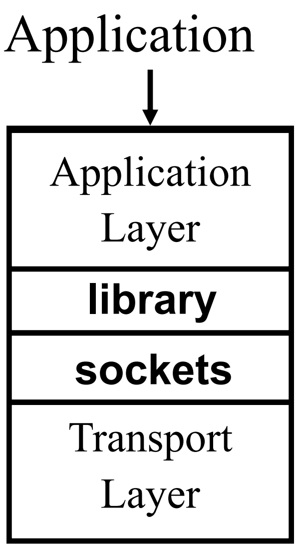
Different Sources of Components

- Application: web server/browser, mail, distributed game,..
- Presentation/session
  - Often part of application
- Transport/network
  - Typically part of the operating system
- Datalink
  - Often written by vendor of the network interface hardware
- Physical
  - Hardware: card and link



Application & Upper Layers

- Application Layer: Provides services that are frequently required by applications: DNS, web acess, file transfer, email…
- Presentation Layer: machine-independent representation of data…
- Session Layer: dialog management, recovery from errors, …



Mostly incorporated into Application Layer

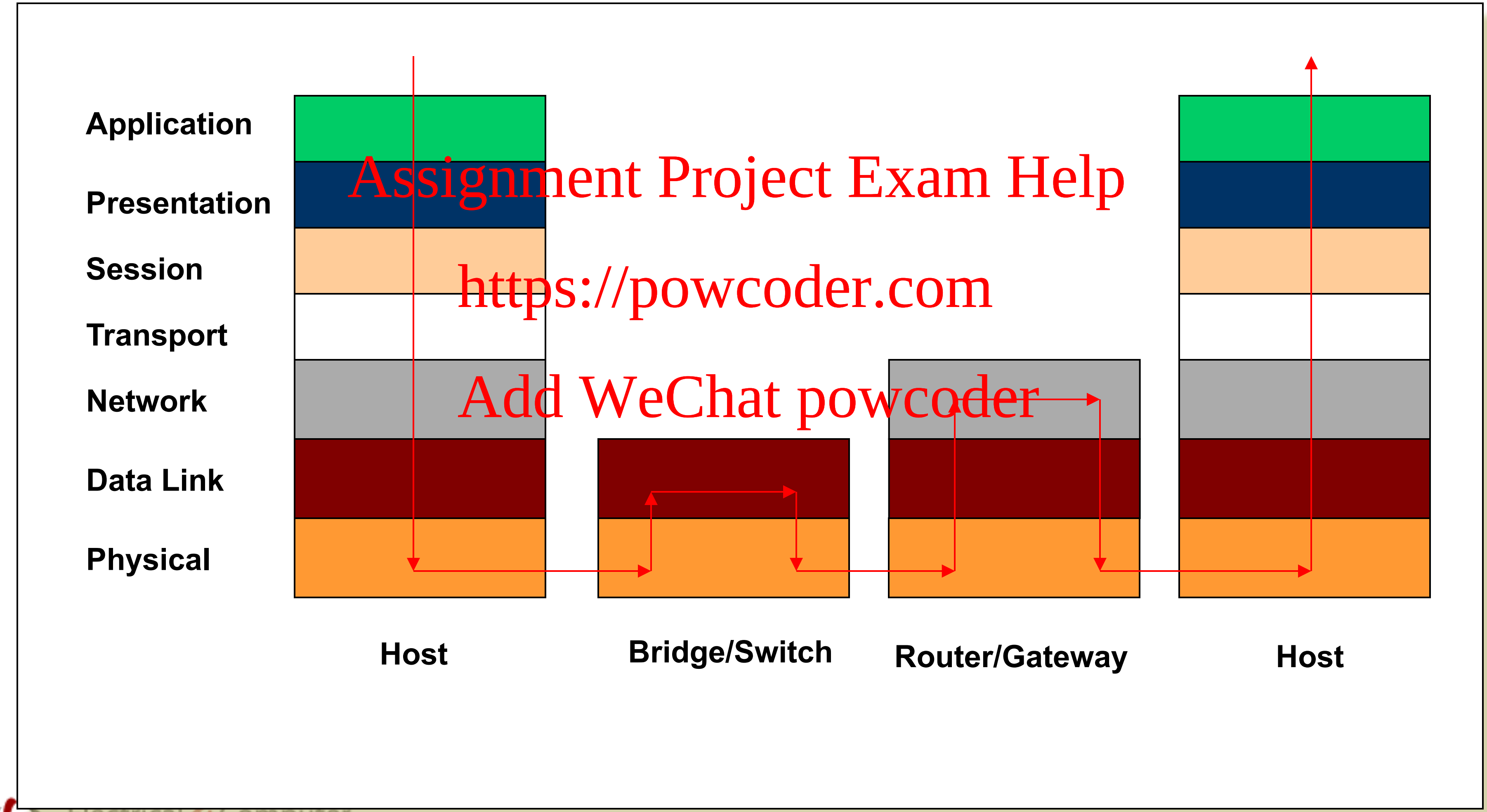
The Internet Engineering Task Force

- Standardization is key to network interoperability
  - The hardware/software of communicating parties are often not built by the same vendor → yet they can communicate because they use the same protocol

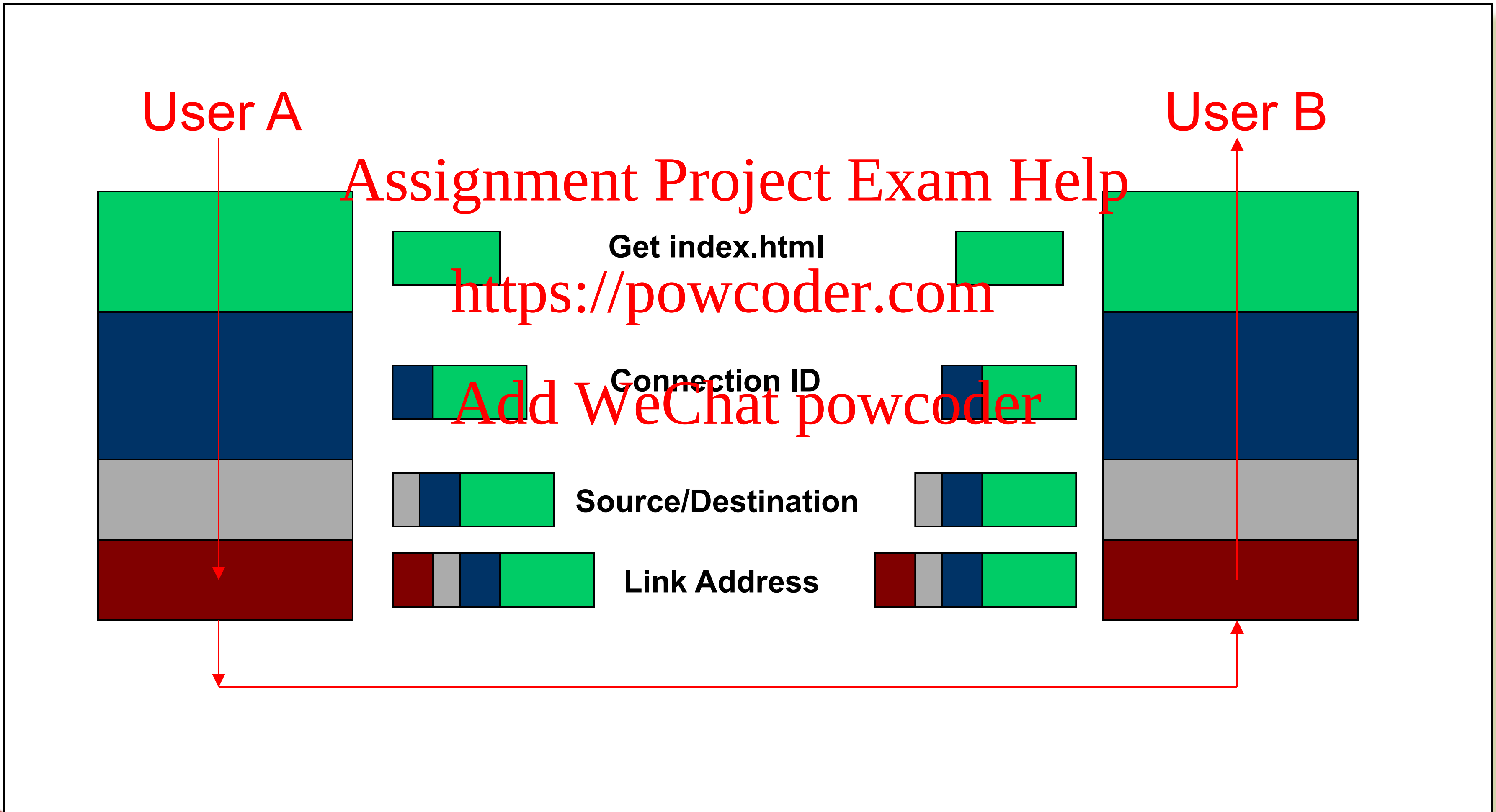
- Internet Engineering Task Force
  - Based on working groups that focus on specific issues

- Request for Comments
  - Document that provides information or defines standard
  - Requests feedback from the community
  - Can be "promoted" to standard under certain conditions
    - consensus in the committee
    - interoperating implementations
  - Project 1 will look at the Internet Relay Chat (IRC) RFC

Life of Packet

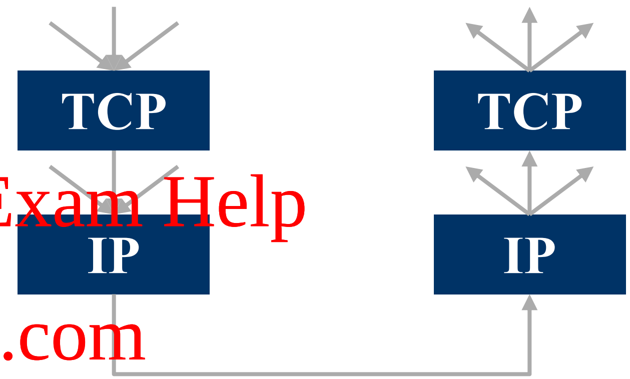


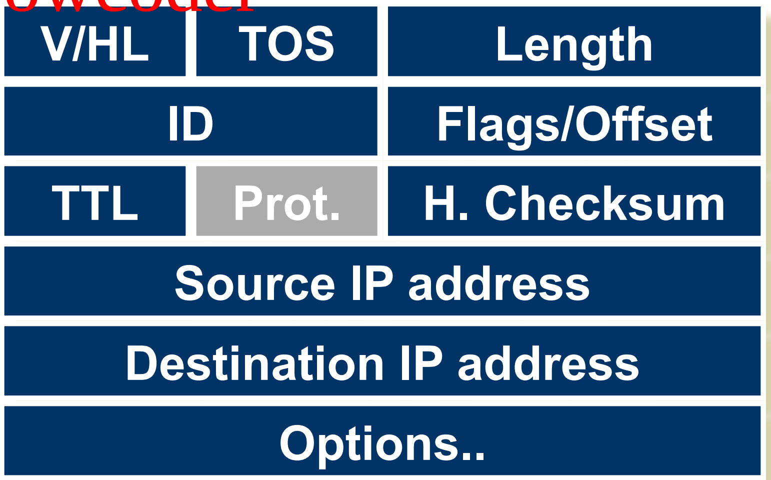
Layer Encapsulation



Multiplexing and Demultiplexing

- There may be multiple implementations of each layer.
  - How does the receiver know what version of a layer to use?
- Each header includes a demultiplexing field that is used to identify the next layer.
  - Filled in by the sender
  - Used by the receiver
- Multiplexing occurs at multiple layers. E.g., IP, TCP, …

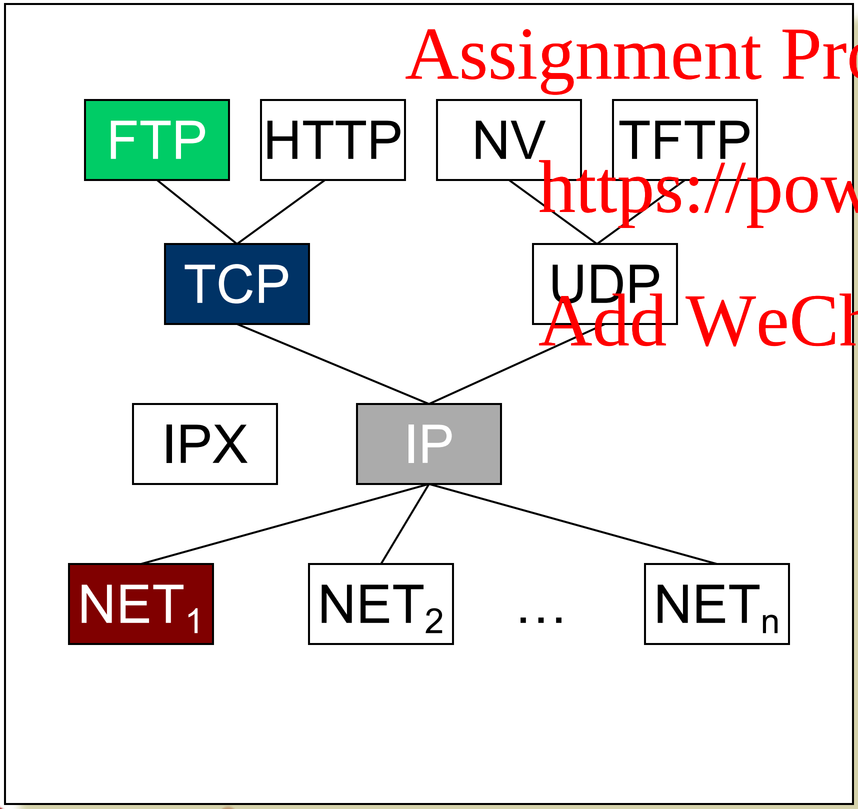


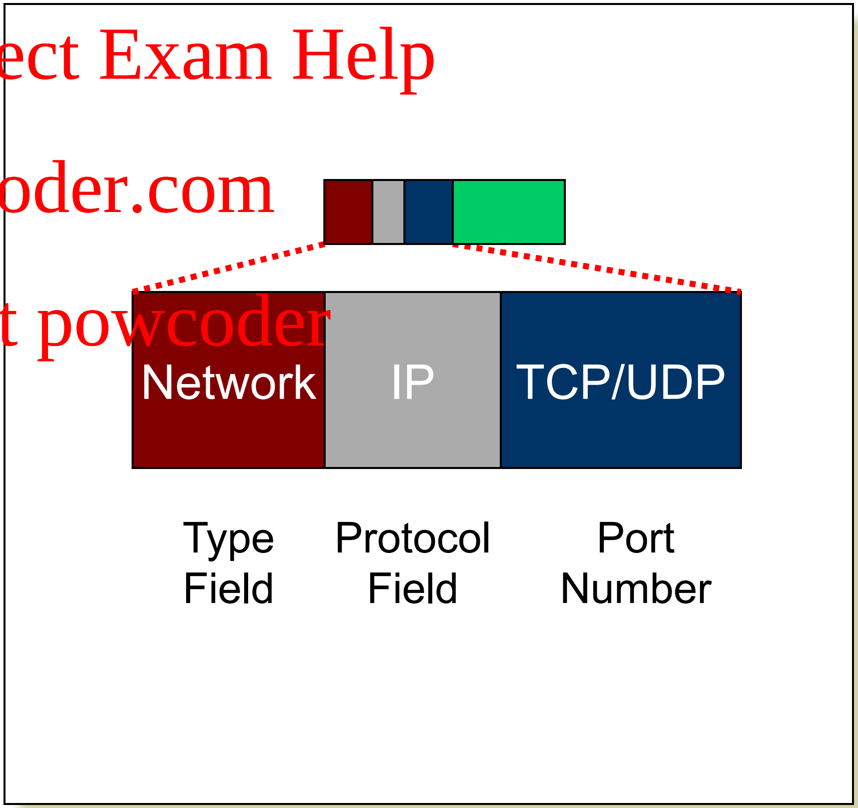


Electrical & Computer ENGINEERING

Protocol Demultiplexing

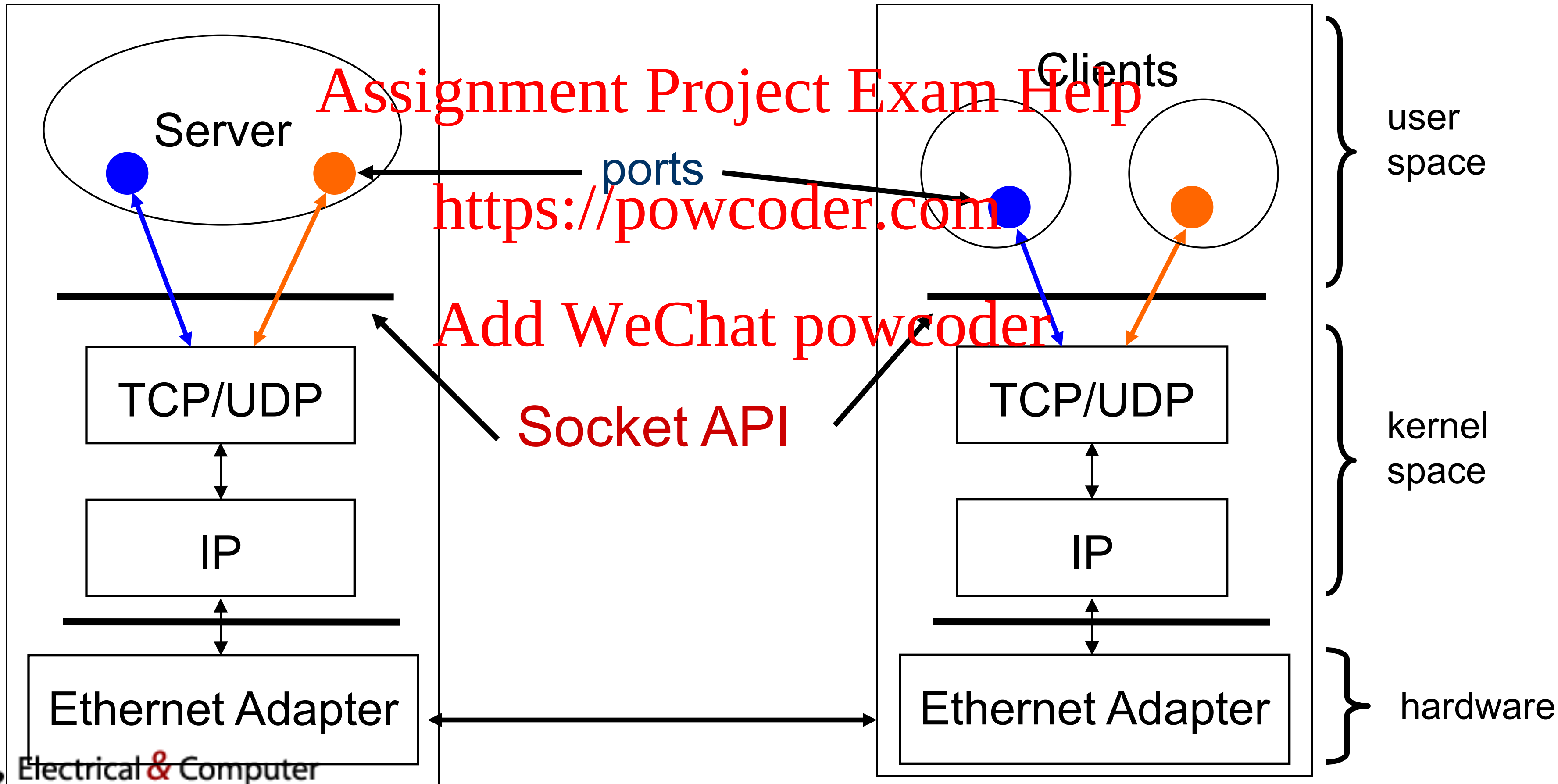
- Multiple choices at each layer



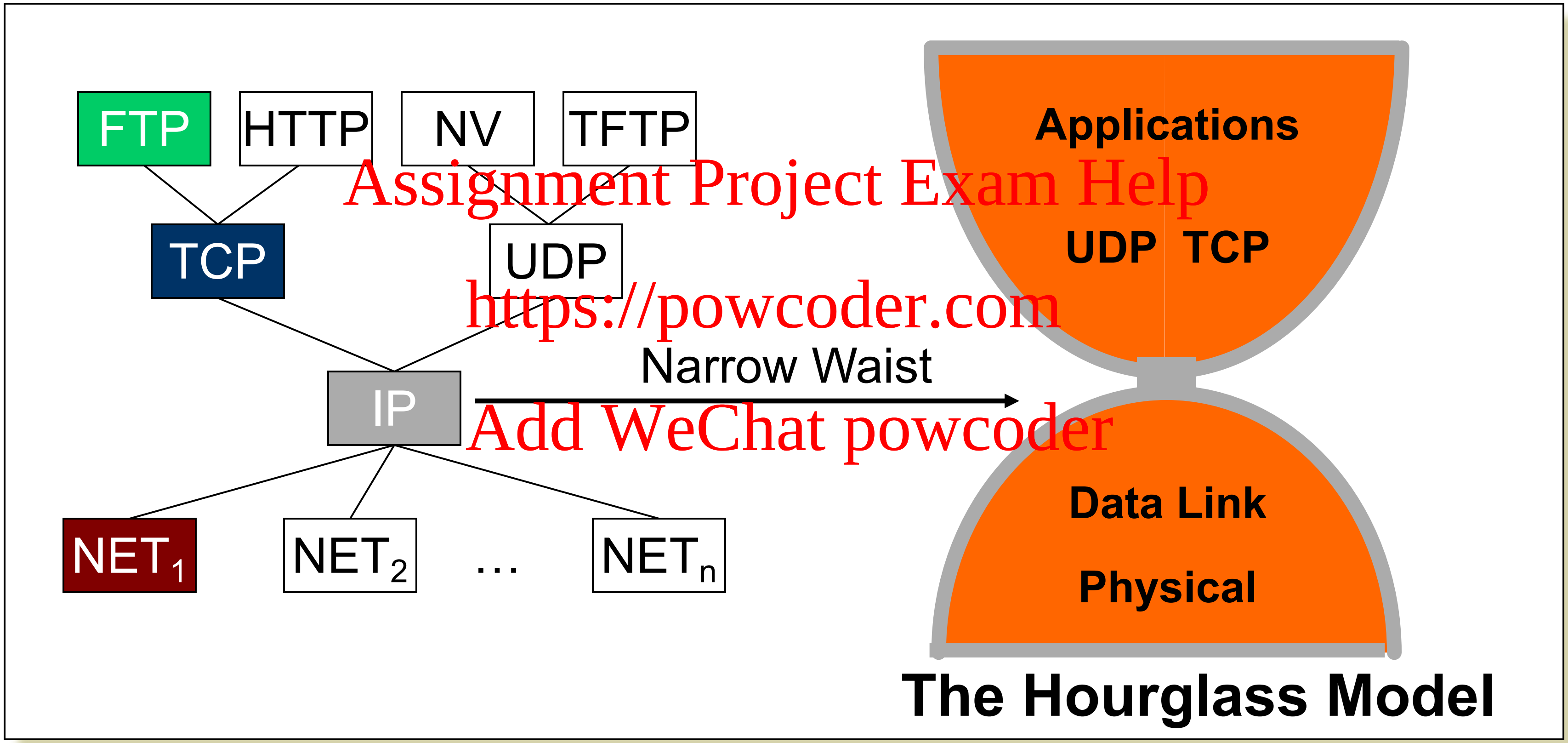


Server and Client

Server and Client exchange messages over the network through a common Socket API



The Internet Protocol Suite



The waist facilitates interoperability
… but evolution is hard

IP based on a Minimalist Approach

- Dumb network
  - IP provide minimal functionalities to support connectivity
    - Addressing, forwarding, routing
- Smart end system
  - Transport layer or application performs more sophisticated functionalities
    - Flow control, error control, congestion control
- Advantages
  - Accommodate heterogeneous technologies (Ethernet, modem, satellite, wireless)
  - Support diverse applications (telnet, ftp, Web, X windows)
  - Decentralized network administration

Sample Quiz Question

- Question: Which of these will be hardest launch at Internet-scale: a new version of TCP, a new version of IP, or a new version of WiFi?

- Answer: New IP (why?)

Today's Lecture

- Network applications
  - Requirements
  - Latency and bandwidth
- Internet architecture
  - A layered design
  - Protocols
  - Life of a packet
- Network utilities

Network tools

- ping
- traceroute
- ipconfig
- tcpdump
- …

ping

- Application to determine if host is reachable
- Based on Internet Control Message Protocol
  - ICMP informs source host about errors encountered in IP packet processing by routers or by destination host
  - ICMP Echo message requests reply from destination host
- PING sends echo message & sequence #
- Determines reachability & round-trip delay
- Sometimes disabled for security reasons

```
PING google.com (68.65.124.59): 56 data bytes
64 bytes from 68.65.124.59: icmp_seq=0 ttl=60 time=13.022 ms
64 bytes from 68.65.124.59: icmp_seq=1 ttl=60 time=16.723 ms
64 bytes from 68.65.124.59: icmp_seq=2 ttl=60 time=16.057 ms
64 bytes from 68.65.124.59: icmp_seq=3 ttl=60 time=13.777 ms
64 bytes from 68.65.124.59: icmp_seq=4 ttl=60 time=17.644 ms
^C
--- google.com ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 13.022/15.445/17.644/1.760 ms
```

```
Swaruns-MacBook-Pro-4:~ swarun$ ping weibo.com
PING weibo.com (123.125.104.197): 56 data bytes
64 bytes from 123.125.104.197: icmp_seq=0 ttl=41 time=232.848 ms
64 bytes from 123.125.104.197: icmp_seq=1 ttl=41 time=297.449 ms
64 bytes from 123.125.104.197: icmp_seq=2 ttl=41 time=316.311 ms
64 bytes from 123.125.104.197: icmp_seq=3 ttl=41 time=235.104 ms
^C
--- weibo.com ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 232.848/270.428/316.311/37.066 ms
```

traceroute

- Find route from local host to a remote host
- Time-to-Live (TTL)
  - IP packets have TTL field that specifies maximum # hops traversed before packet discarded
  - Each router decrements TTL by 1
  - When TTL reaches 0 packet is discarded
- Traceroute
  - Send UDP to remote host with TTL=1
  - First router will reply ICMP Time Exceeded Message
  - Send UDP to remote host with TTL=2, …
  - Each step reveals next router in path to remote host
- **`tracert`** (windows), **`tracepath`** (linux)

```
Swaruns-MacBook-Pro-4:~ swarun$ traceroute facebook.com
traceroute to facebook.com (31.13.69.228), 64 hops max, 52 byte packets
 1  pod-w-vl75.gw.cmu.net (128.237.128.1)  62.820 ms  5.246 ms  1.859 ms
 2  core255-pod-w-cyh.gw.cmu.net (128.2.255.241)  1.966 ms  1.548 ms  1.755 ms
 3  pod-i-dcns-core255.gw.cmu.net (128.2.255.194)  86.446 ms  2.744 ms  1.807 ms
 4  100.121.0.37 (100.121.0.37)  2.053 ms  2.089 ms  2.169 ms
 5  tr-cps.pennren.3rox.net (147.73.18.143)  2.336 ms  2.851 ms  2.080 ms
 6  et-4-0-0.512.sdn-sw.pitt.net.internet2.edu (198.71.47.181)  9.345 ms  4.421 ms  3.444 ms
 7  et-8-3-0.4079.sdn-sw.ashb.net.internet2.edu (162.252.70.52)  12.930 ms  12.143 ms  12.822 ms
 8  lo-0.8.rtr.ashb.net.internet2.edu (64.57.29.131)  12.823 ms  12.352 ms  12.783 ms
 9  ae14.pr06.iad3.tfbnw.net (103.4.97.230)  14.077 ms
    dc5.pr01.iad2.tfbnw.net (206.126.236.191)  13.729 ms
    ae21.pr05.iad3.tfbnw.net (103.4.97.144)  13.153 ms
10  po104.psw04.iad3.tfbnw.net (31.13.28.99)  12.135 ms
    po106.psw04.iad3.tfbnw.net (157.240.43.131)  12.683 ms
    po105.psw01.iad3.tfbnw.net (31.13.31.169)  11.950 ms
11  173.252.67.27 (173.252.67.27)  12.508 ms
    173.252.67.111 (173.252.67.111)  13.686 ms
    173.252.67.139 (173.252.67.139)  12.432 ms
12  edge-star-mini-shv-01-iad3.facebook.com (31.13.69.228)  12.942 ms  12.665 ms  12.358 ms
```

ipconfig

- Utility in Microsoft Windows to display TCP/IP information about a host
- Many options
  - Simplest: IP address, subnet mask, default gateway for the host
  - Information about each IP interface of a host
    - DNS hostname, IP addresses of DNS servers, physical address of network card, IP address, …
  - Renew IP address from DHCP server

```
Swaruns-MacBook-Pro-4:~ swarun$ ifconfig en0
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether b8:f6:b1:1b:9a:d5
        inet6 fe80::32:6dc1:b470:7030%en0 prefixlen 64 secured scopeid 0x7
        inet 128.237.142.42 netmask 0xfffff000 broadcast 128.237.143.255
        nd6 options=201<PERFORMNUD,DAD>
        media: autoselect
        status: active
```

netstat

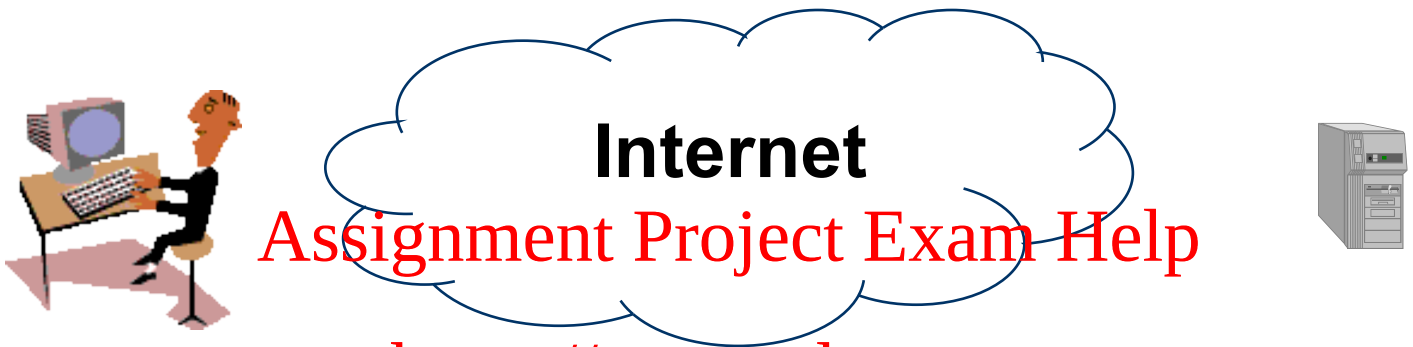
- Queries a host about TCP/IP network status
- Status of network drivers & their interface cards
  - #packets in, #packets out, errored packets, …
- State of routing table in host
- TCP/IP active server processes
- TCP active connections

```
Swaruns-MacBook-Pro-4:~ swarun$ netstat
Active Internet connections
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp4       0      0  swaruns-mbp-4.wv.61551 198.98.22.58.https     ESTABLISHED
tcp4       0      0  swaruns-mbp-4.wv.61540 ec2-52-4-191-33..https ESTABLISHED
tcp4       0      0  swaruns-mbp-4.wv.61536 upload-lb.eqiad..https ESTABLISHED
tcp4       0      0  swaruns-mbp-4.wv.61535 text-lb.eqiad.wi.https ESTABLISHED
tcp4       0      0  swaruns-mbp-4.wv.61530 www.ece.cmu.edu.https  ESTABLISHED
tcp4       0      0  swaruns-mbp-4.wv.61523 172.217.3.98.http      ESTABLISHED
tcp4      31      0  swaruns-mbp-4.wv.61491 162.125.33.7.https     CLOSE_WAIT
tcp4       0      0  swaruns-mbp-4.wv.61441 172.217.10.10.https    CLOSE_WAIT
tcp4       0      0  swaruns-mbp-4.wv.61427 172.217.10.234.https   CLOSE_WAIT
tcp4       0      0  swaruns-mbp-4.wv.61420 172.217.3.106.https    CLOSE_WAIT
tcp4       0      0  swaruns-mbp-4.wv.61419 172.217.6.202.https    CLOSE_WAIT
tcp4       0      0  swaruns-mbp-4.wv.61404 151.101.202.49.https   ESTABLISHED
tcp4       0      0  swaruns-mbp-4.wv.61397 151.101.202.49.https   ESTABLISHED
```

`tcpdump` and Network Protocol Analyzers

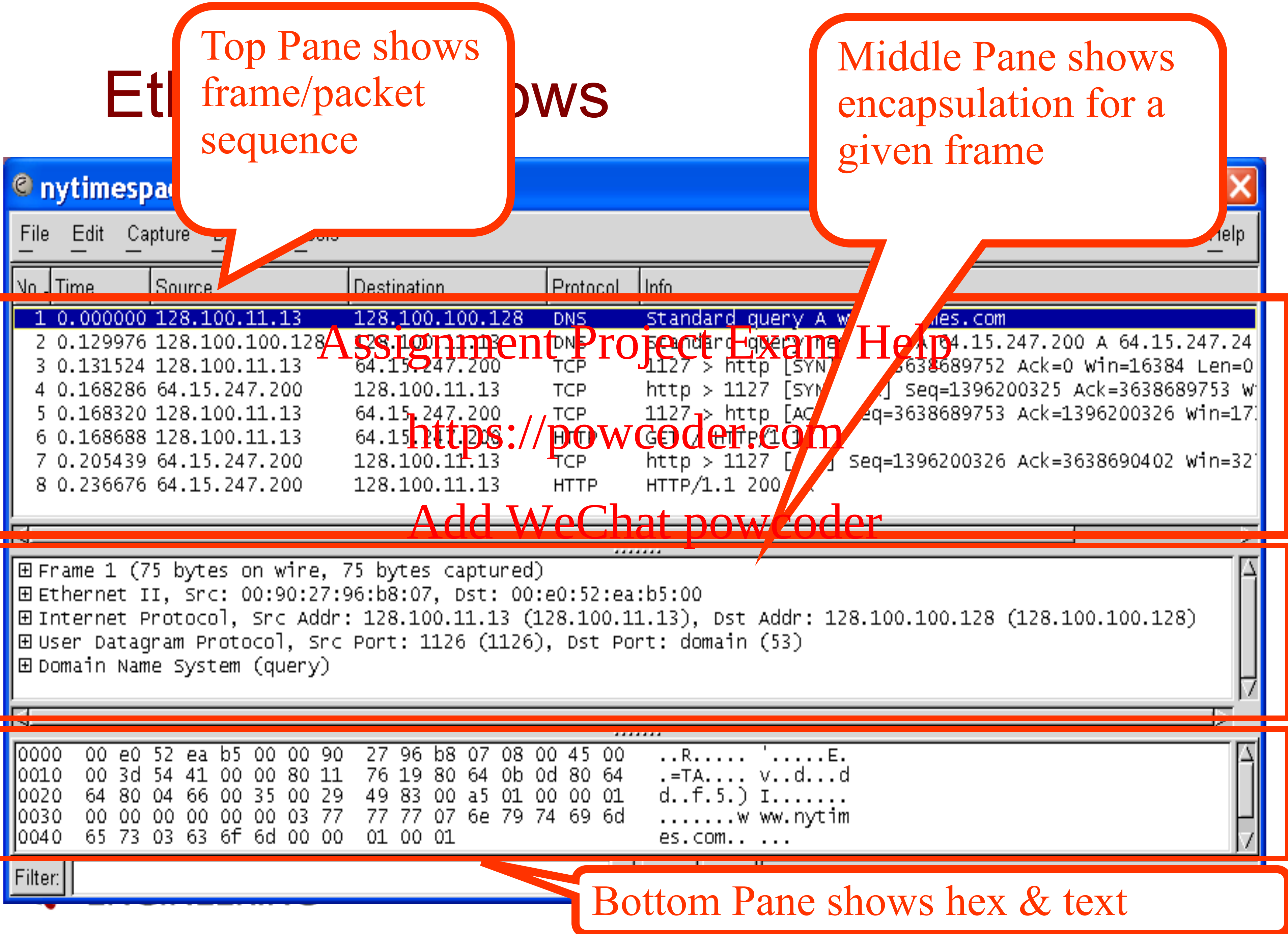
- `tcpdump` program captures IP packets on a network interface (usually Ethernet NIC)
- Filtering used to select packets of interest
- Packets & higher-layer messages can be displayed and analyzed
- `tcpdump` basis for many network protocol analyzers for troubleshooting networks
- We use the open source Ethereal analyzer to generate examples (or wireshark, etc.)
  - www.ethereal.com

How the layers work together: Network Analyzer Example



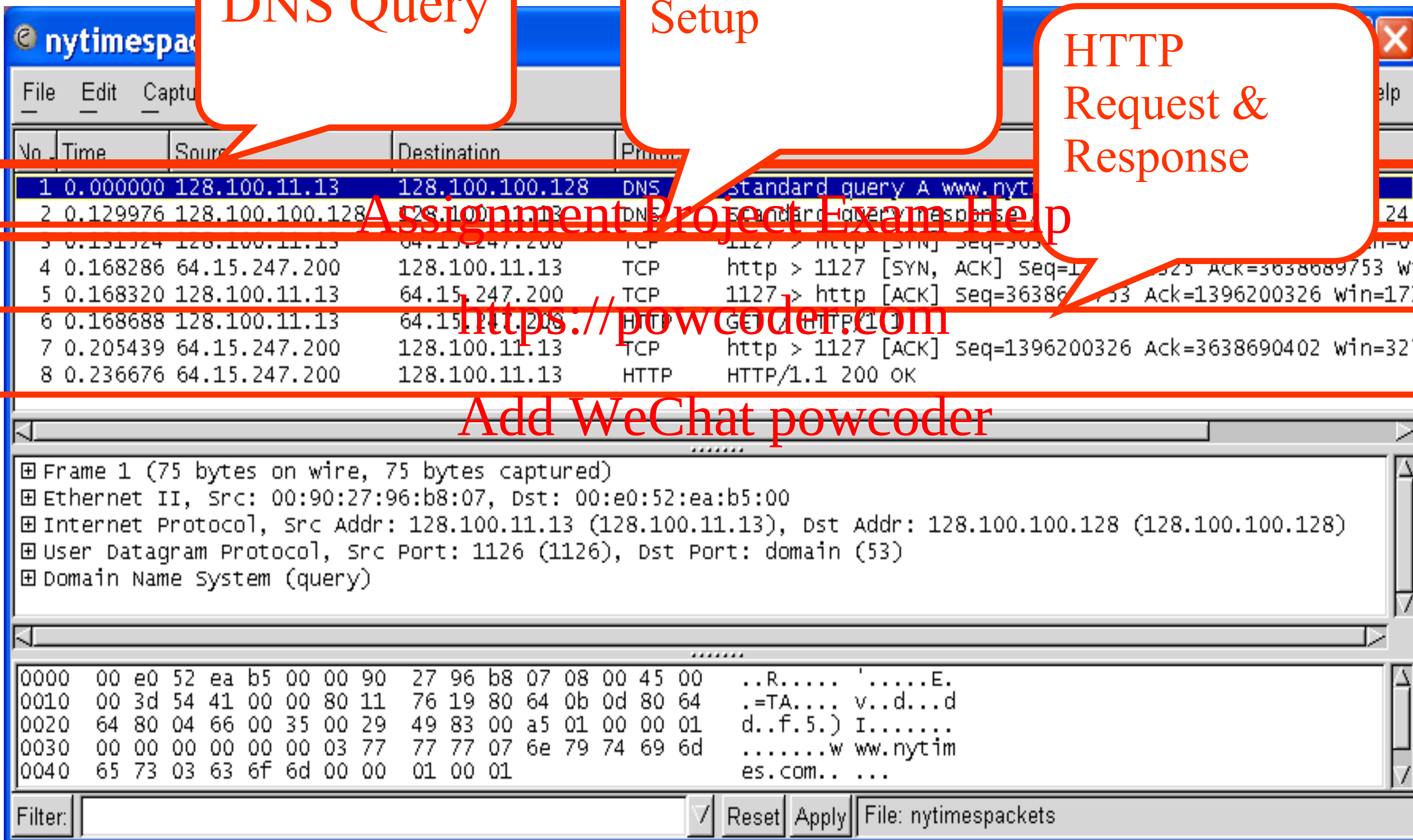
- User clicks on http://www.nytimes.com/
- *Ethereal* network analyzer captures all frames observed by its Ethernet NIC (or Wireshark)
- Sequence of frames and contents of frame can be examined in detail down to individual bytes

Et ows

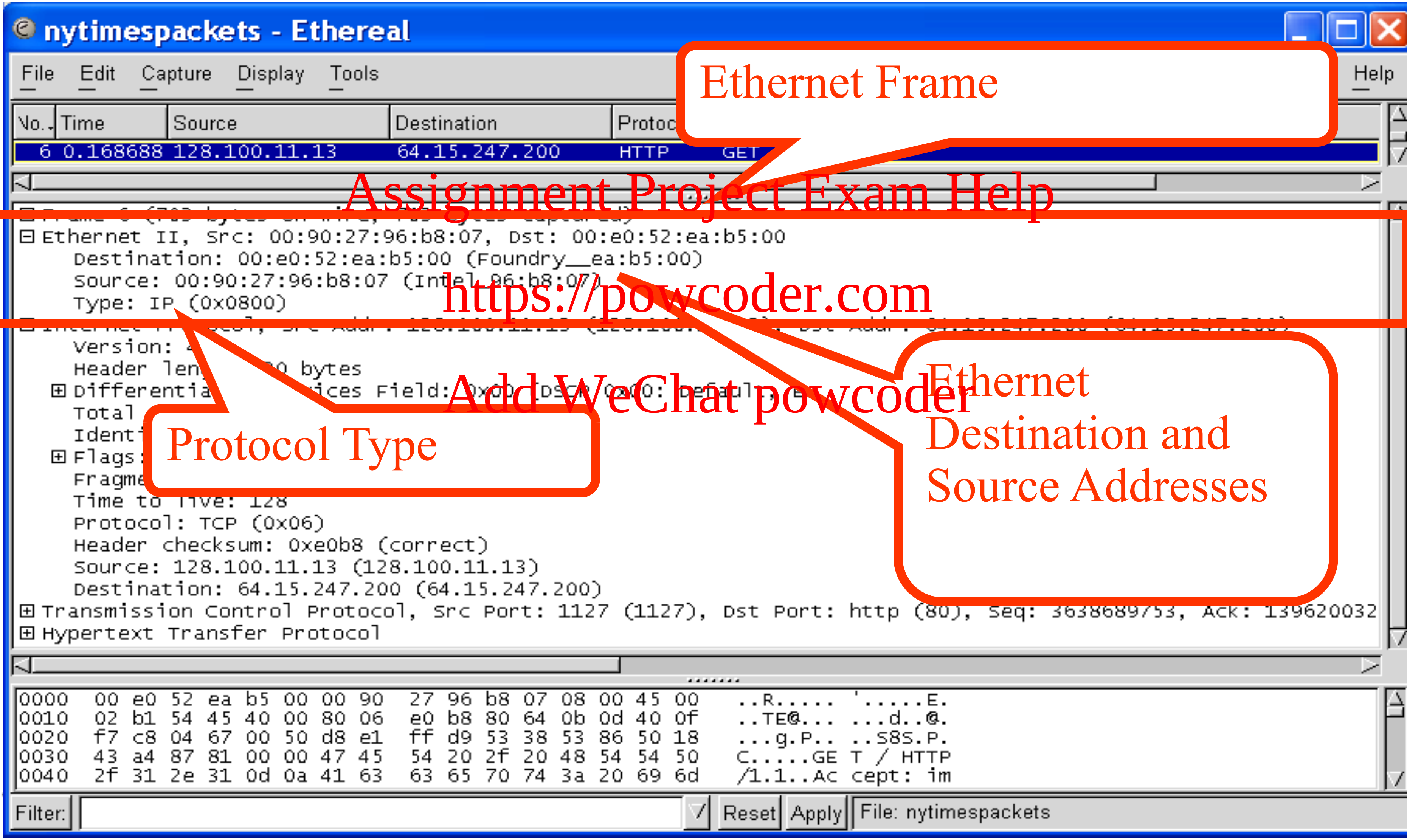


Top Pane shows frame/packet sequence
Middle Pane shows encapsulation for a given frame
nytimespa
File Edit Capture
Help
No. Time Source Destination Protocol Info
1 0.000000 128.100.11.13 128.100.100.128 DNS Standard query A w
2 0.129976 128.100.100.128 DNS 64.15.247.200 A 64.15.247.24
3 0.131524 128.100.11.13 64.15.247.200 TCP 1127 > http [SYN 638689752 Ack=0 Win=16384 Len=0
4 0.168286 64.15.247.200 128.100.11.13 TCP http > 1127 [SYN Seq=1396200325 Ack=3638689753
5 0.168320 128.100.11.13 64.15.247.200 TCP 1127 > http [AC eq=3638689753 Ack=1396200326 Win=17
6 0.168688 128.100.11.13 HTTP GET
7 0.205439 64.15.247.200 128.100.11.13 TCP http > 1127 Seq=1396200326 Ack=3638690402 Win=32
8 0.236676 64.15.247.200 128.100.11.13 HTTP HTTP/1.1 200
Frame 1 (75 bytes on wire, 75 bytes captured)
Ethernet II, Src: 00:90:27:96:b8:07, Dst: 00:e0:52:ea:b5:00
Internet Protocol, Src Addr: 128.100.11.13 (128.100.11.13), Dst Addr: 128.100.100.128 (128.100.100.128)
User Datagram Protocol, Src Port: 1126 (1126), Dst Port: domain (53)
Domain Name System (query)
0000 00 e0 52 ea b5 00 00 90 27 96 b8 07 08 00 45 00 ..R..... '.....E.
0010 00 3d 54 41 00 00 80 11 76 19 80 64 0b 0d 80 64 .=TA.... v..d...d
0020 64 80 04 66 00 35 00 29 49 83 00 a5 01 00 00 01 d..f.5.) I.......
0030 00 00 00 00 00 00 03 77 77 77 07 6e 79 74 69 6d ....w ww.nytim
0040 65 73 03 63 6f 6d 00 00 01 00 01 es.com.. ...
Filter:
Bottom Pane shows hex & text

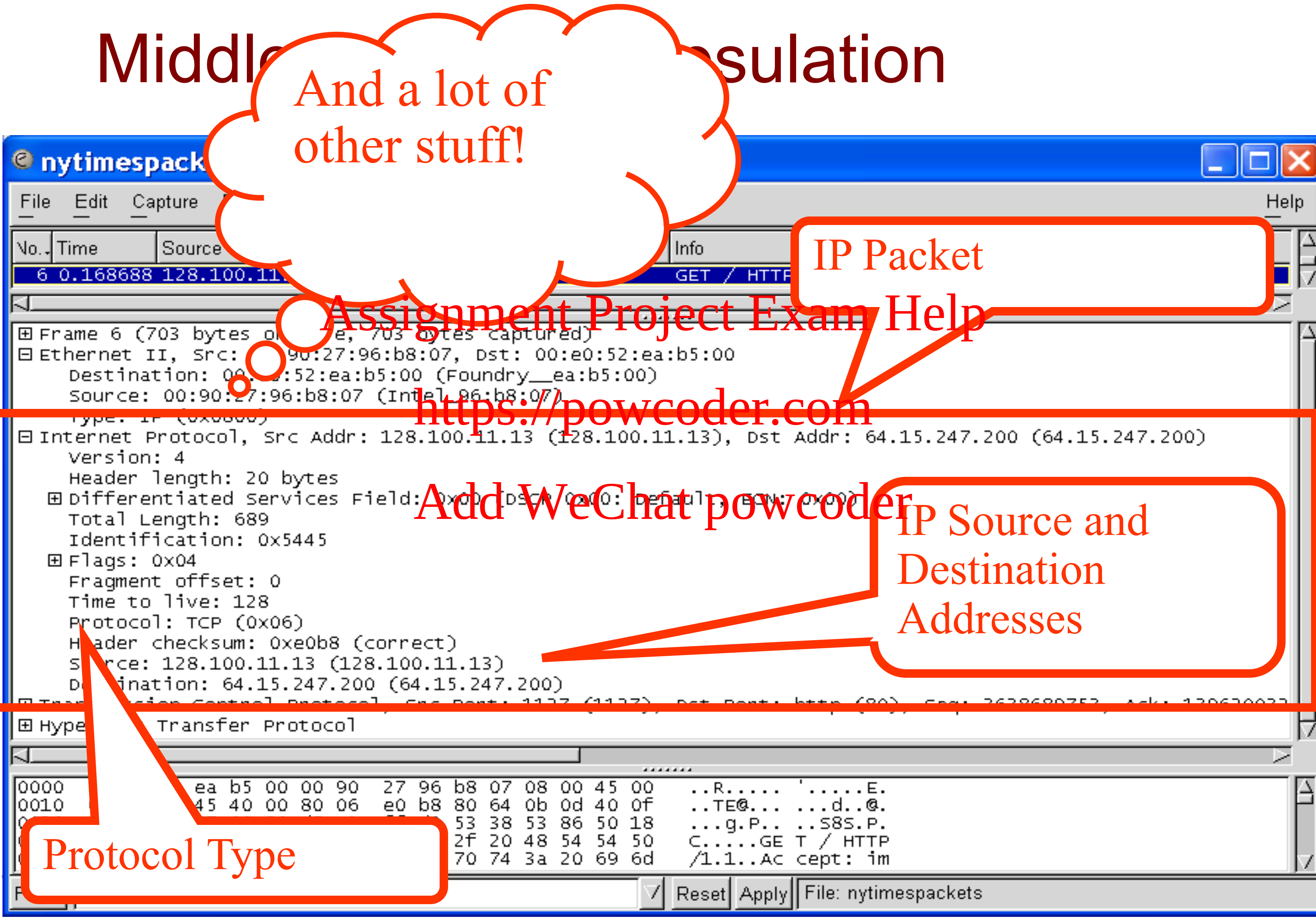
Top pane: fram… …e



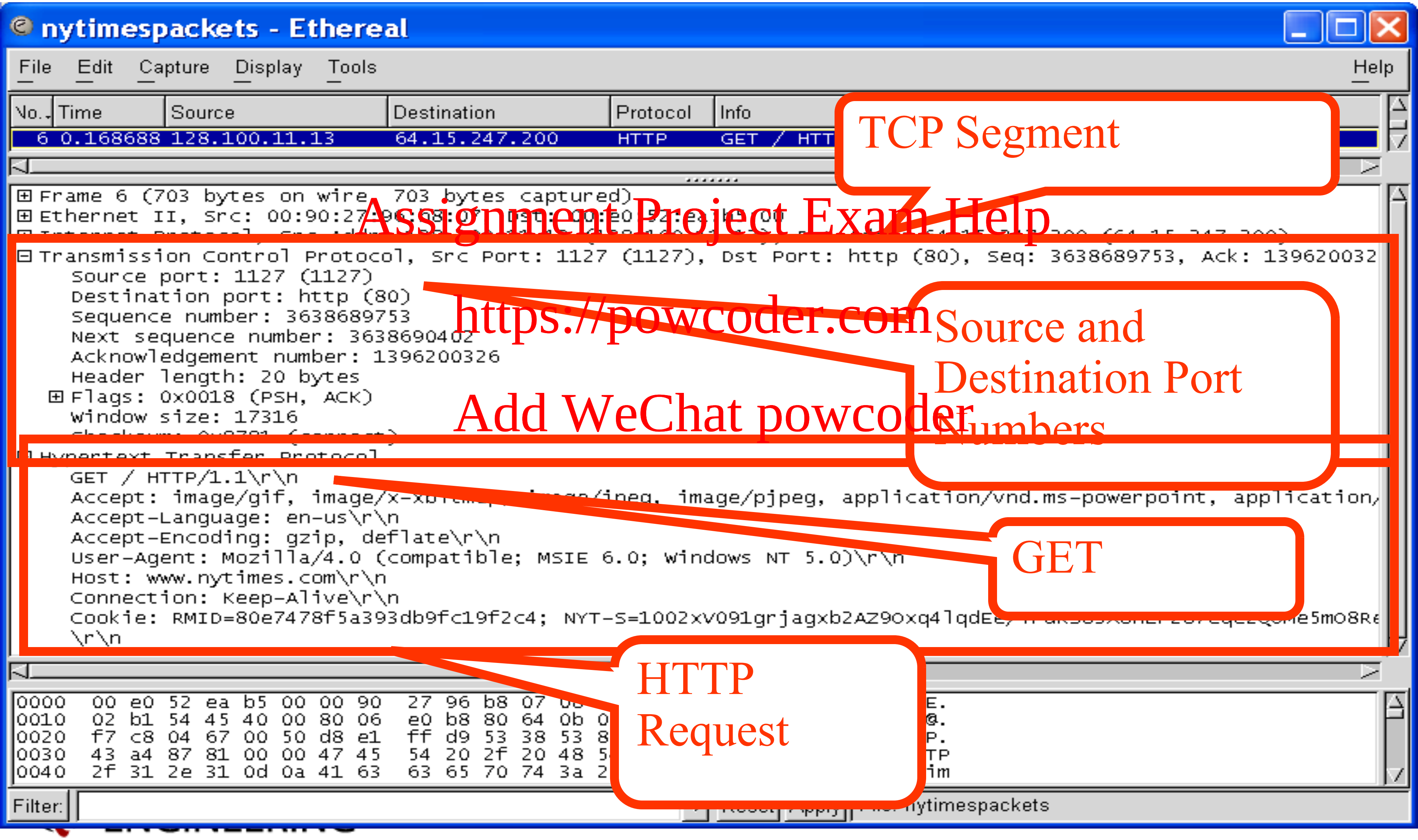
Middle pane: Encapsulation



Middl sulation



Middle pane: Encapsulation



Goals [Clark88]

0 **Connect existing networks**

initially ARPANET and ARPA packet radio network

1. Survivability

ensure communication service even in the presence of network and router failures

2. Support multiple types of services

3. Must accommodate a variety of networks
4. Allow distributed management
5. Allow host attachment with a low level of effort
6. Be cost effective
7. Allow resource accountability

Principle: End-to-End Argument (Saltzer'81)

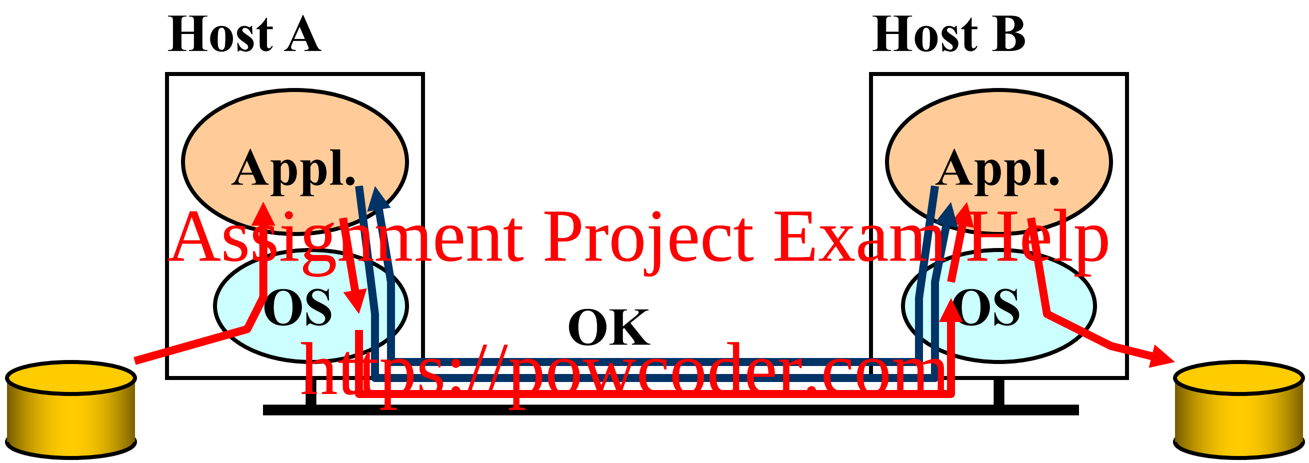
- Focus of the paper is "system"
  - Not a pure networking paper

- Deals with where to place functionality
  - Inside the network (in switching elements)

  - At the edges

- Argument: Some functions can only be correctly implemented by the endpoints – do not try to implement these elsewhere
  - Not a law – more of a "best practices"

Example: Reliable File Transfer



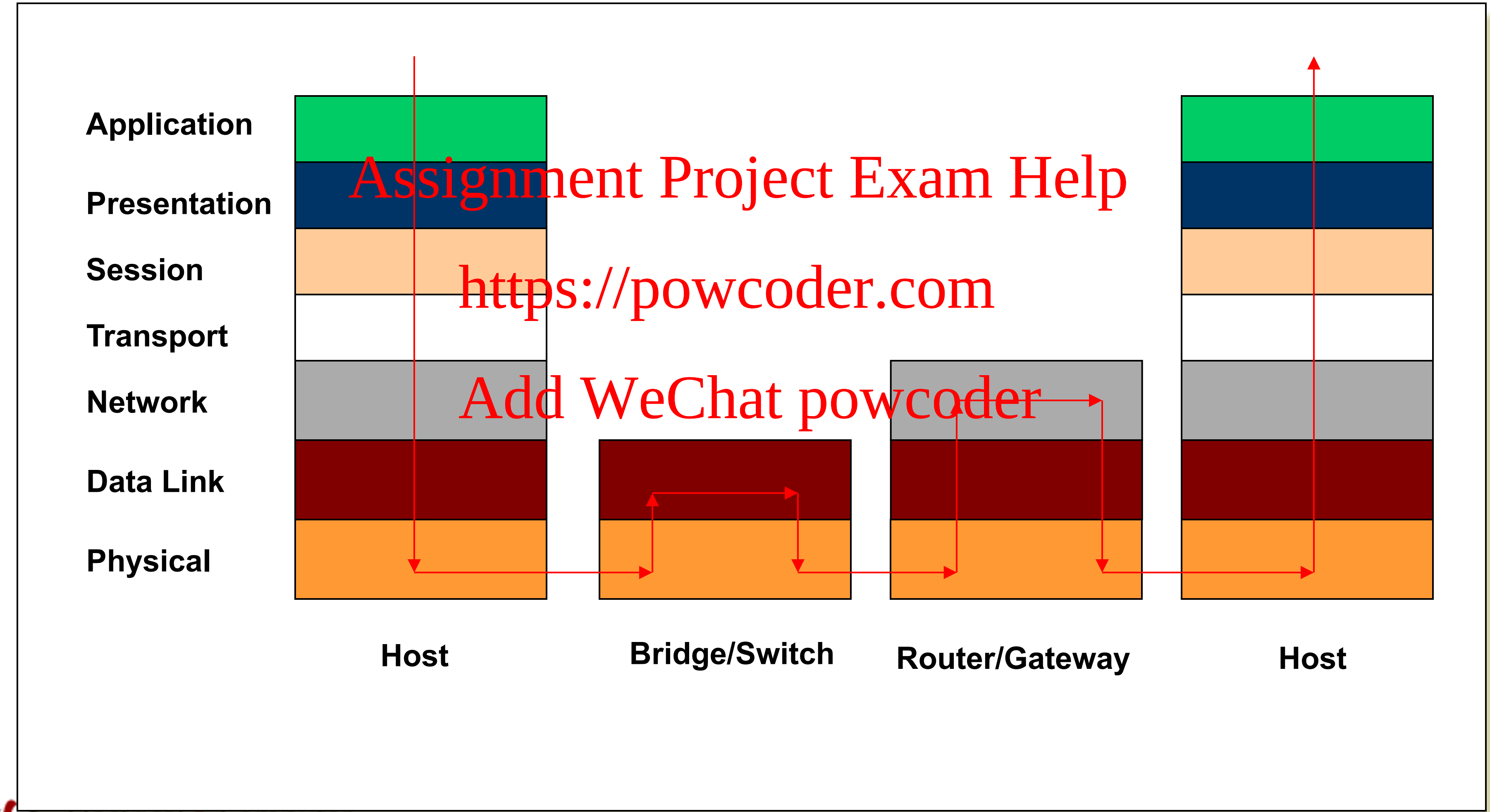
- Solution 1: make each step reliable, and then concatenate them
- Solution 2: end-to-end check and retry

Sample Quiz Question

- Question: A switch and a router both cost $100 and have similar specs and achieve similar performance in packet switching/routing. As a rational buyer, which should you buy?

- Answer: The router (why?)

Life of Packet



Next Lecture

- The “PHY”