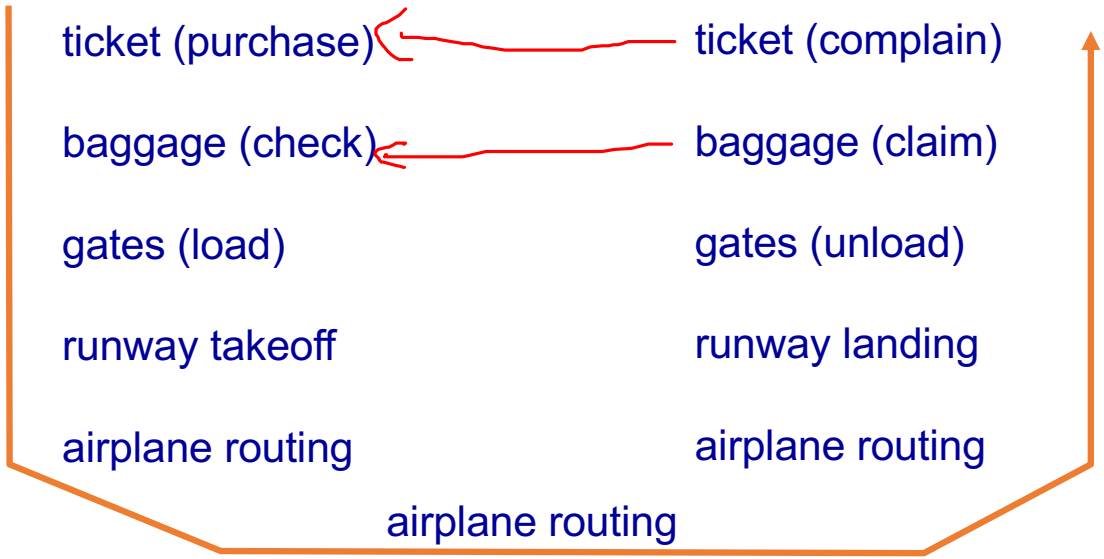
IT 304 Computer Networks
Introduction
Week 2-Lecture 3

Protocol "layers"

Networks are complex, with many "pieces":

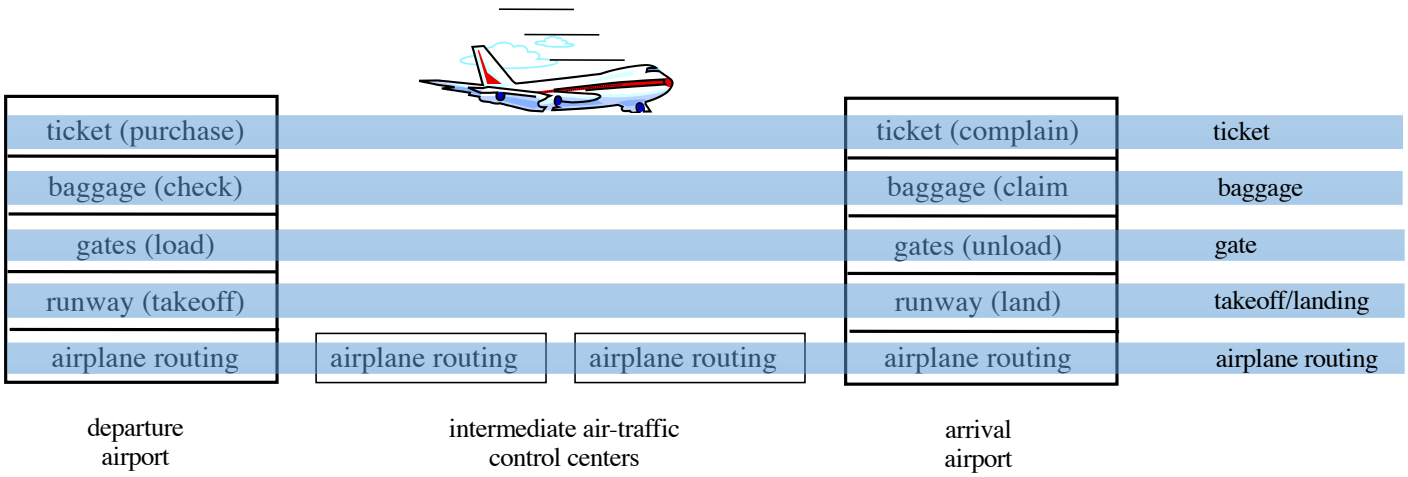
- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Organization of air travel



- a series of steps

Layering of airline functionality



layers: each layer implements a service

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

Question:

How is the Internet structured?

Why layering?

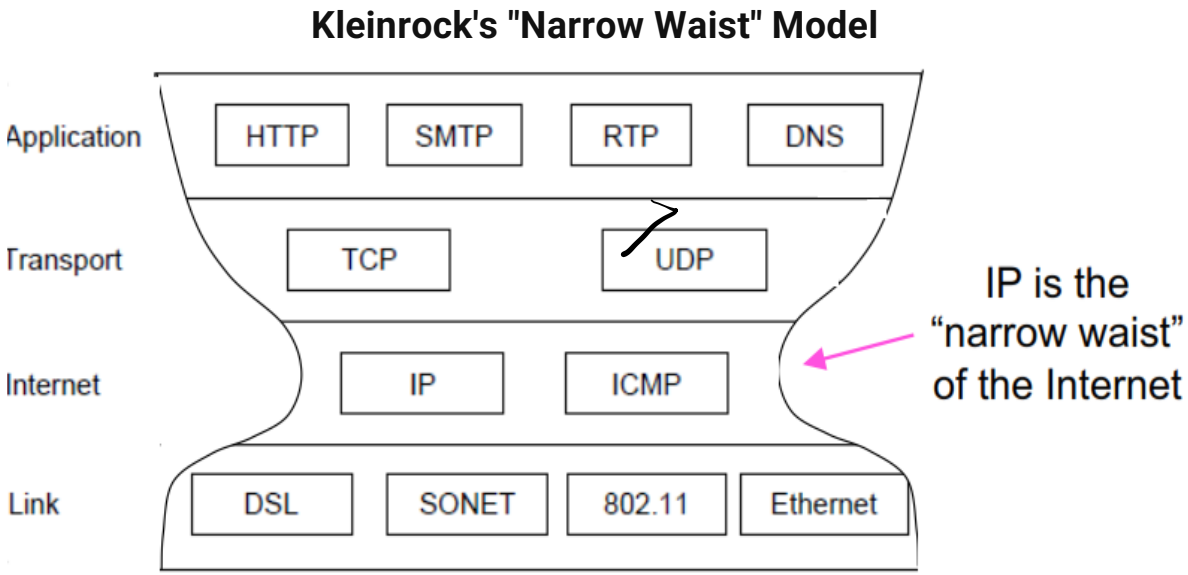
dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system

The TCP/IP Model

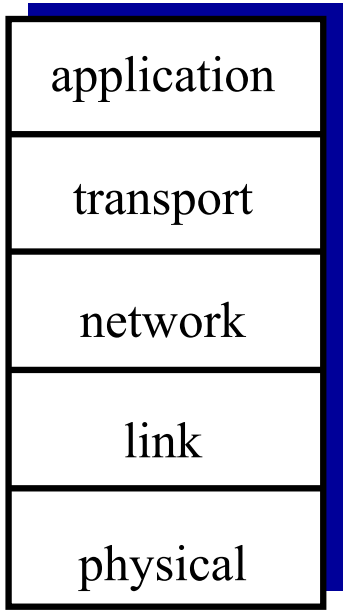
Vint Cerf and **Bob Kahn**

- Link layer
- Internet layer
- Transport layer
- Application layer



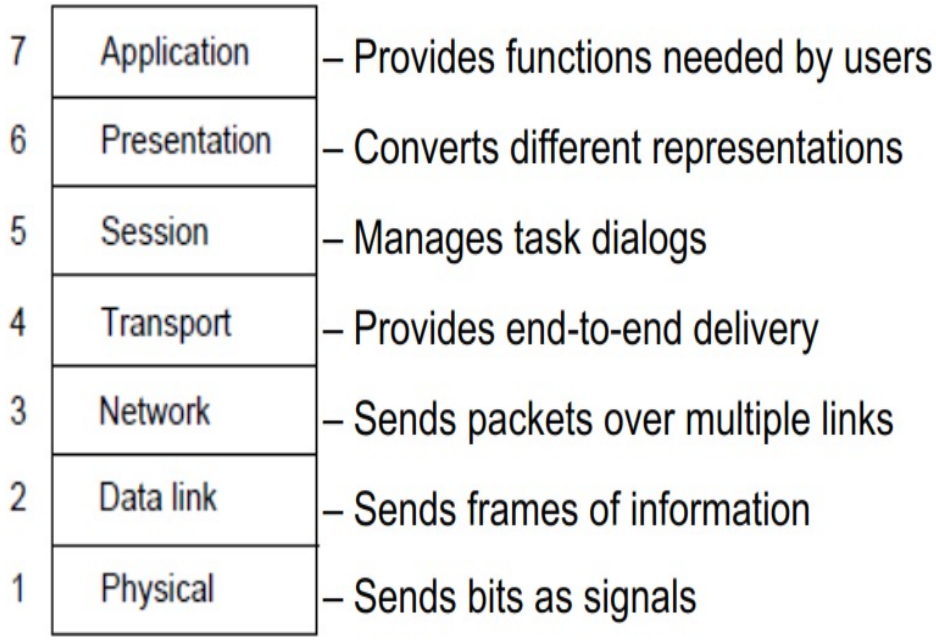
Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet (802.3), WiFi (802.11), PPP
- *physical:* bits "on the wire"



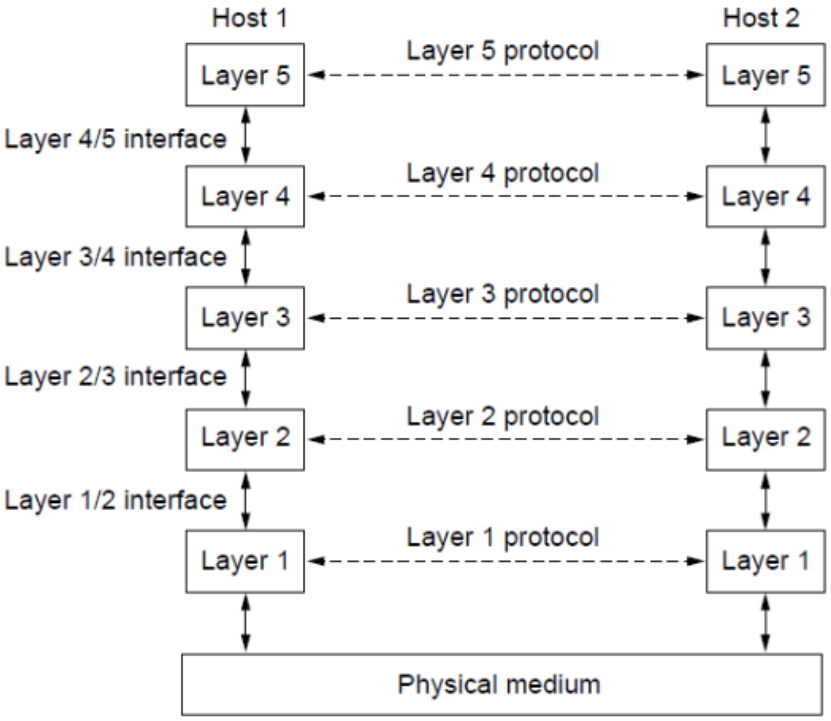
The Open Systems Interconnection (OSI) Reference Model by International Standards Organization (ISO)

- Layers created for different abstractions
- Each layer performs well-defined function
- Function of layer chosen with standards in mind
- Minimize information flow across layer interfaces
- Find the optimum number of layers



Protocol Layers (1)

Protocol layering is the main structuring method used to divide up network functionality.

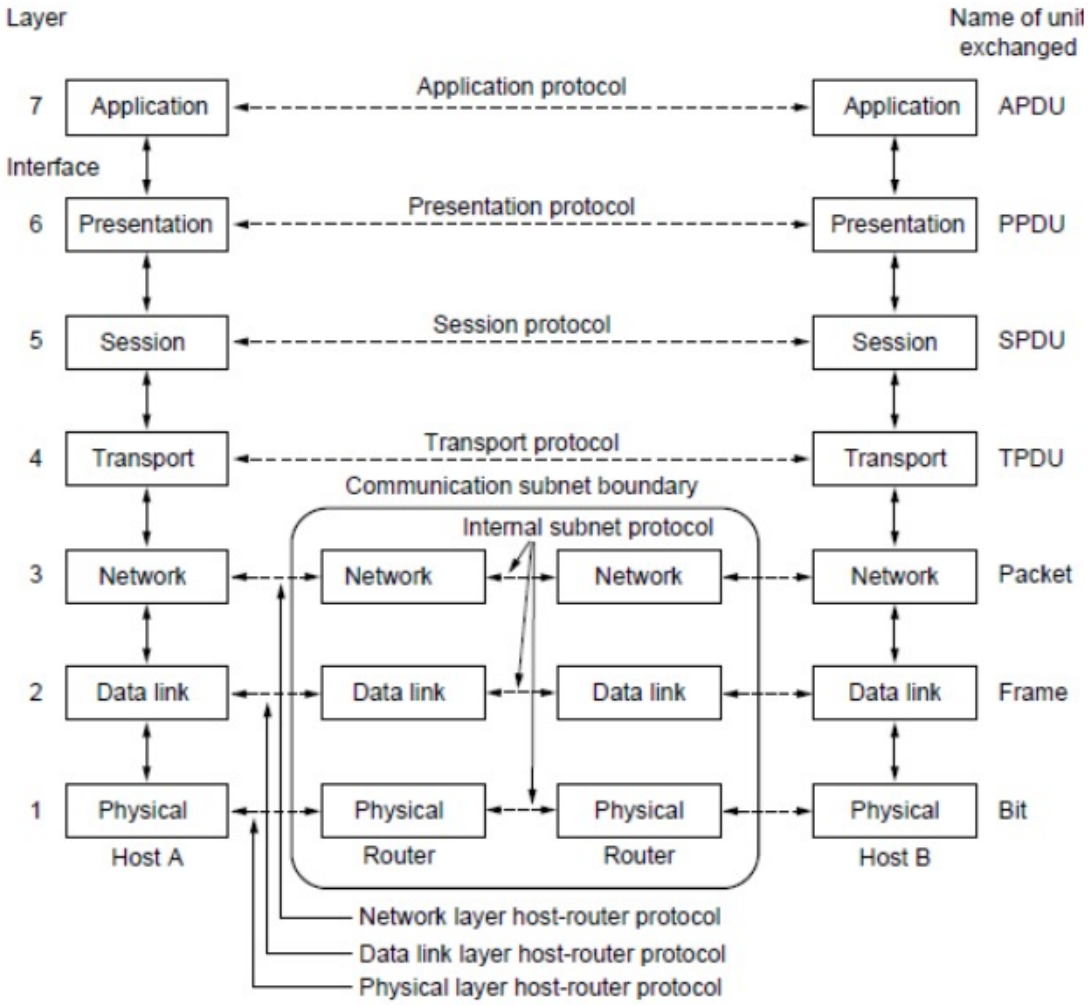


- Each protocol instance talks virtually to its peer
- Each layer communicates only by using the one below
- Lower layer services are accessed by an interface
- At bottom, messages are carried by the medium

Design Issues for the Layers

Each layer solves a particular problem but must include mechanisms to address a set of recurring design issues

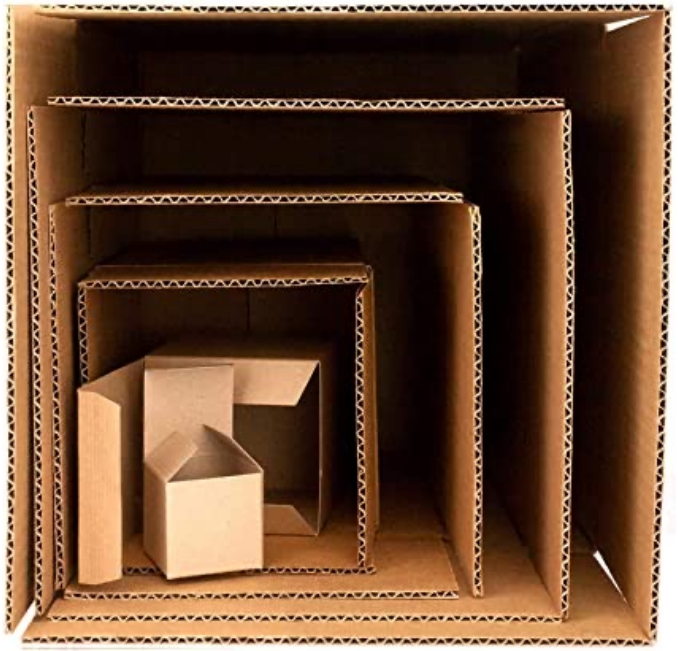
Issue	**Example mechanisms at different layers**
Reliability despite failures	Codes for error detection/correction Routing around failures
Network growth and evolution	Addressing and naming Protocol layering
Allocation of resources like bandwidth	Multiple access Congestion control
Security against various threats	Confidentiality of messages Authentication of communicating parties



PDU is Protocol Data Unit

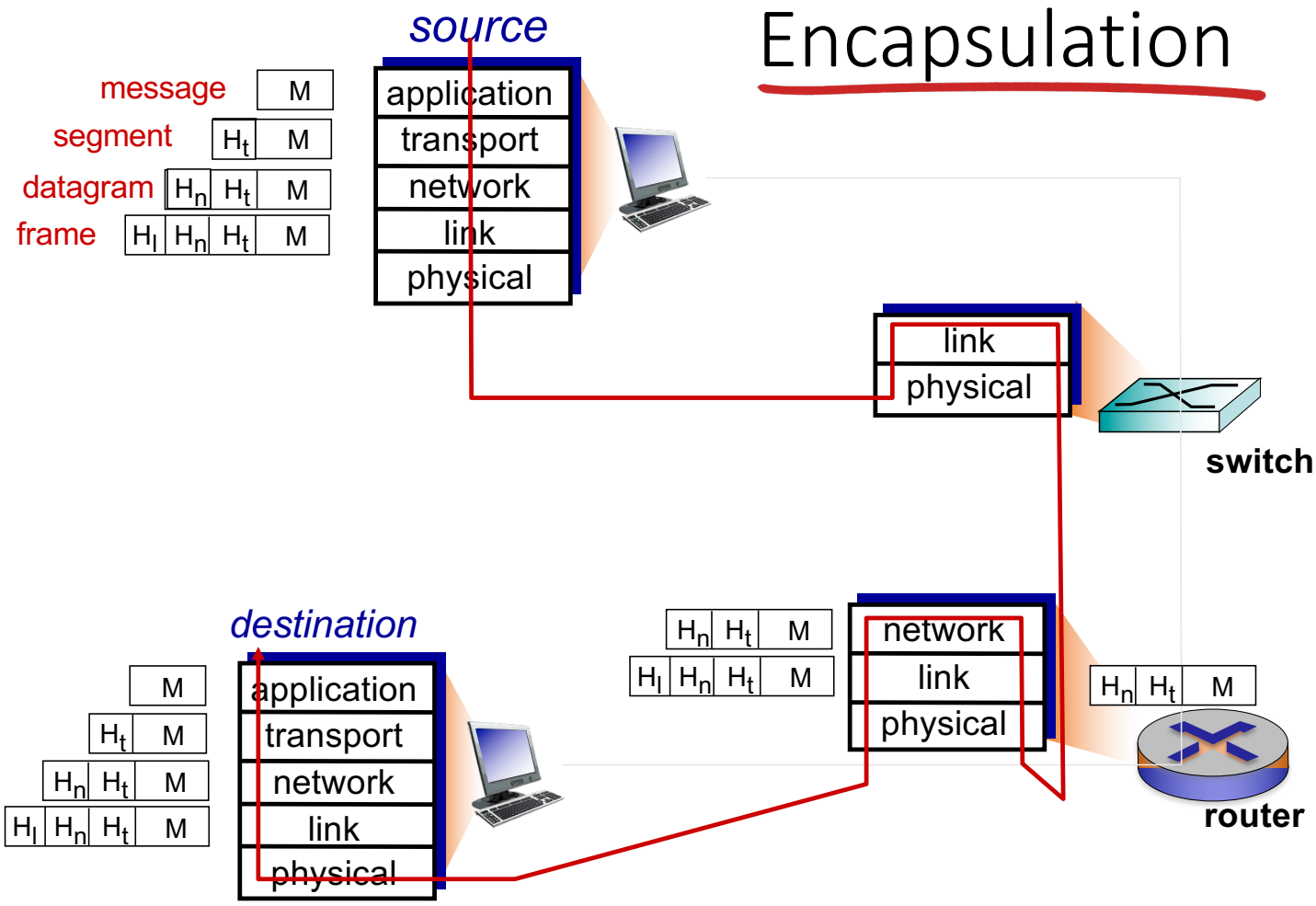
Encapsulation

- Who is the source (S) computer?
- Who is the destination (D) computer?
- Who is passing this message to whom?



A gift box example for encapsulating / packing message from S to D

Encapsulation

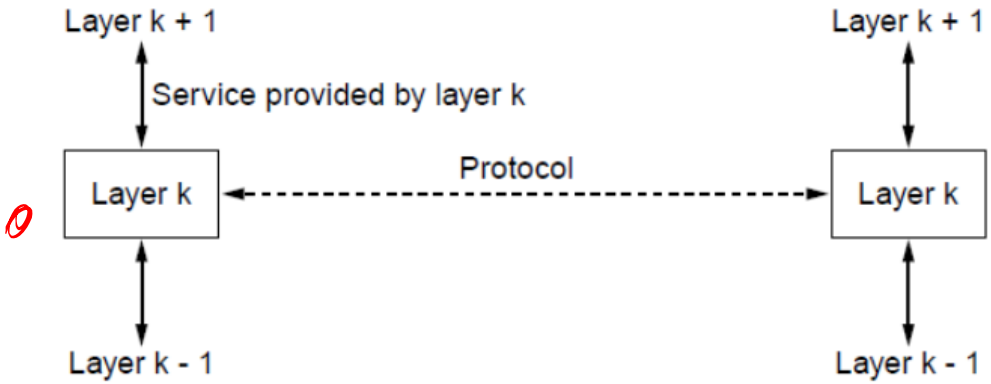


Relationship of Services to Protocols

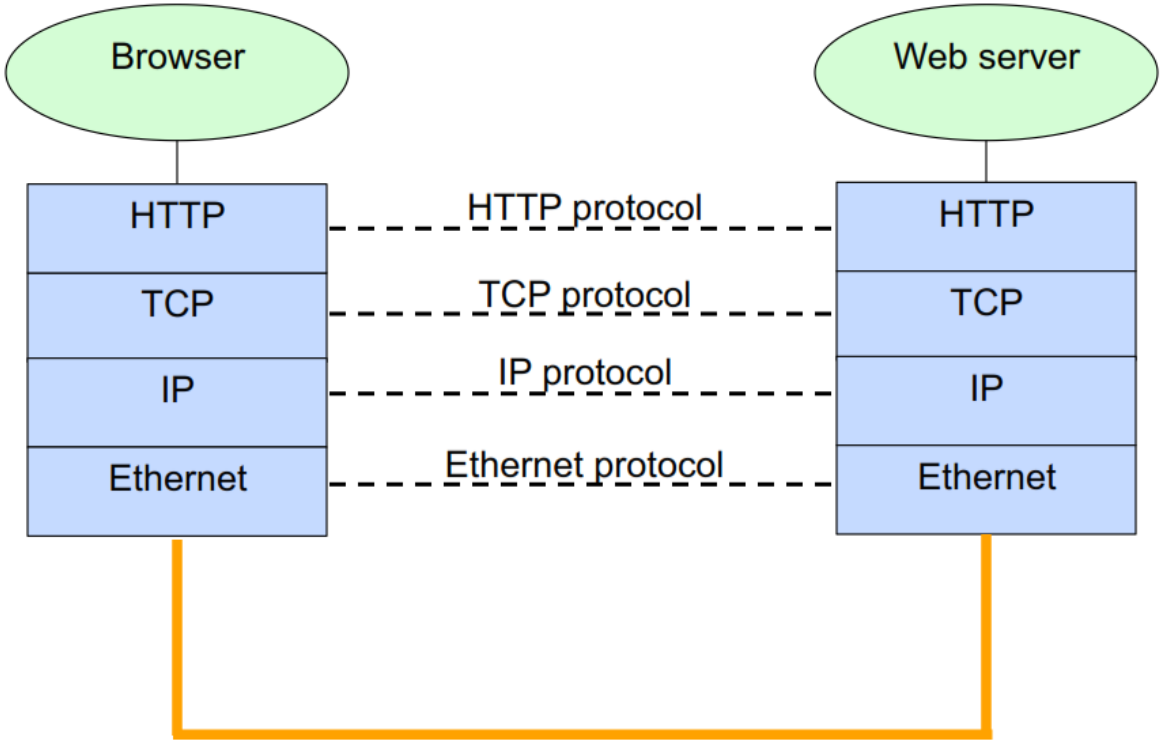
Recap:

- A layer provides a service to the one above (vertical)
- A layer talks to its peer using a protocol (horizontal)

LISTEN
CONNECT
ACCEPT



Example Protocol Stack



Encapsulation