LAYERS AND PROTOCOLS

COMP 30650: NETWORKS AND INTERNET SYSTEMS

Dr. Gavin McArdle
Email: gavin.mcardle@ucd.ie
Office: A1.09 Computer Science

RECAP

Components of a network

- Nodes, hosts, routers, links

Network Topology

- ‘Shape’ of the network infrastructure
  - Advantages/disadvantages

Functions of Networks

Require modularity

- Achieved through protocols and layering

TODAY'S PLAN

Example Layers

Types of connections in a network

Services and Protocols

- Service Primitives

Guidelines for the designing the layers and functionality

- OSI Model
- TCP/IP Model
- The Model will we study

EXAMPLE LAYERS TCP/IP

Application: In this model, the application layer is responsible for **creating and transmitting user data** between applications. The applications can be on remote systems, and should appear to operate as if locally to the end user. The communication is said to take place between peers. HTTP

Transport: The transport layer is responsible for communication between processes. This level of networking utilizes ports to address different services. It can build up **unreliable** or **reliable connections** depending on the type of protocol used. TCP/UDP

Internet: The internet layer is used to **transport** data from node to node in a network. This layer is aware of the endpoints of the connections, but does not worry about the actual connection needed to get from one place to another. **IP addresses** are defined in this layer as a way of reaching remote systems in an addressable manner. IP

Link: The link layer implements the actual topology of the **local network** that allows the internet layer to present an addressable interface. It **establishes connections** between neighboring nodes to send data. ETHERNET/WIFI

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 Transport Layer, Link Layer, Network Layer, Application Layer.
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

CONNECTION-ORIENTED VS. CONNECTIONLESS

Service provided by a layer may be kinds of either:

- Connection-oriented, must be set up for ongoing use (and torn down after use), e.g., phone call
  - Acts like a tube : the sender pushes objects at one end and receiver takes them out at the other end
  - Generally in the same order they were sent.
  - Can be negotiation about parameters when connection is establishes
- Connectionless, messages are handled separately, e.g., postal delivery
  - Each message has the complete address of the receiver and it is routed thorough the network independently of all subsequent messages.

RELIABLE AND UNRELIABLE SERVICES

Reliable service - TCP

- Receiver **acknowledges** receipt of message
- Creates overheads which are not always desirable
- Suitable for file transfers not suitable for VOIP

	Service	Example
Connection-oriented	Reliable message stream	Sequence of pages
	Reliable byte stream	Movie download
Connectionless	Unreliable connection	Voice over IP
	Acknowledged datagram	Text messaging
	Request-reply	Database query

STREAMS AND DATAGRAMS

- **Byte streams - TCP**
  - No message boundaries
  - Receiver has no way to know if a 2048 is two or one messages
- **Datagram – IP, UDP**
  - Message boundaries are preserved
  - Two 1024 byte message will arrive as two messages

SERVICES AND PROTOCOLS

A service is a set of operations that a layer provides to the layer above. The service defines what operations the layer is prepared to perform on behalf of its users but says nothing at all about how the operations are implemented.

TCP provides a reliable bytestream service at the Transport layer

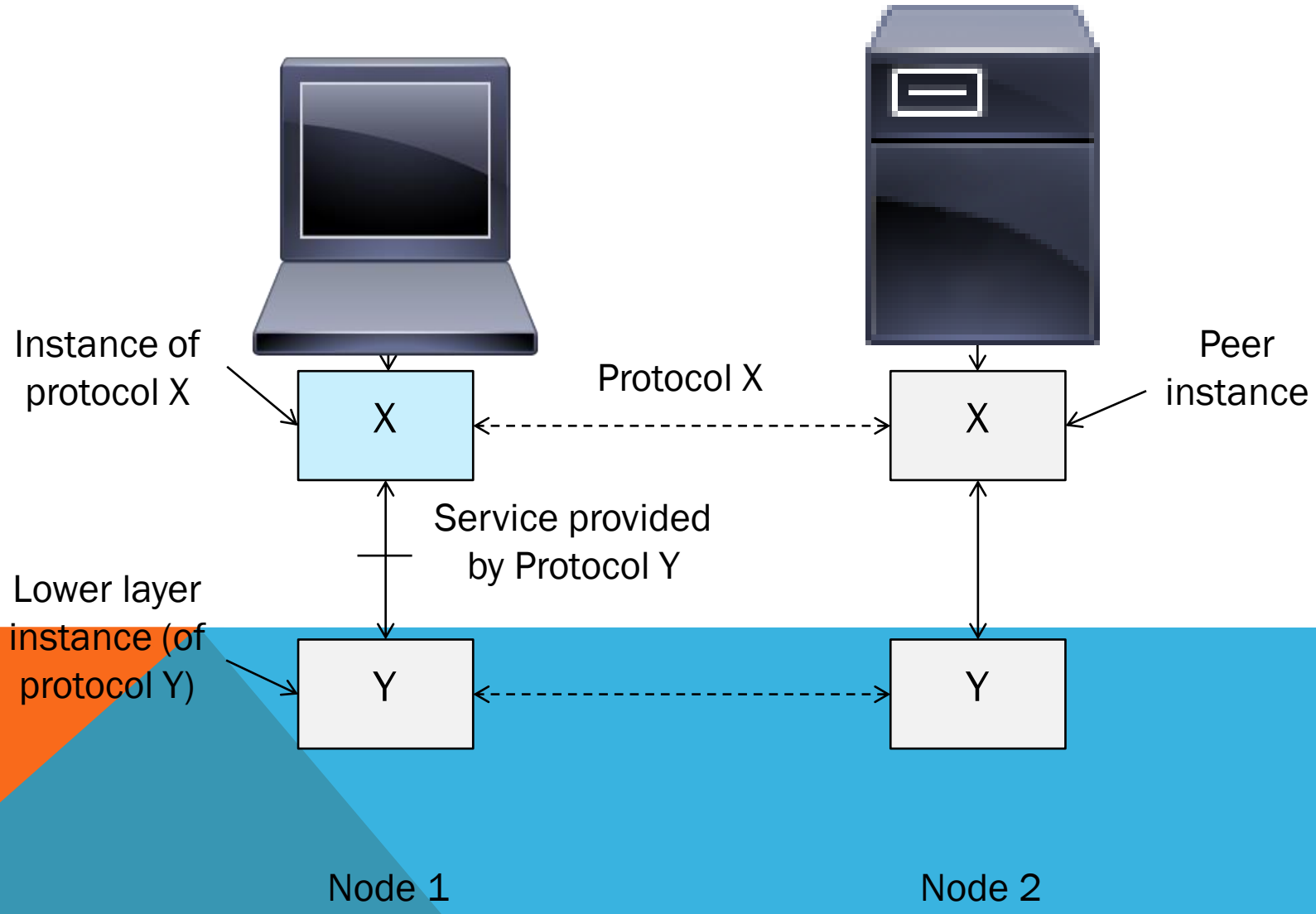
IP provides unreliable datagram service at the Network layer.

SERVICES AND PROTOCOLS

- A protocol is a set of rules governing the format and meaning of the packets, or messages that are exchanged by the peer entities within a layer.
- They are free to change their protocols, provided they do not change the service visible their users.
- Services relate to the interfaces between layers
- Protocols relate to packets sent between peer entities on different machines.

PROTOCOLS AND LAYERS

Protocols are horizontal, services are vertical



SERVICE PRIMITIVES (1)

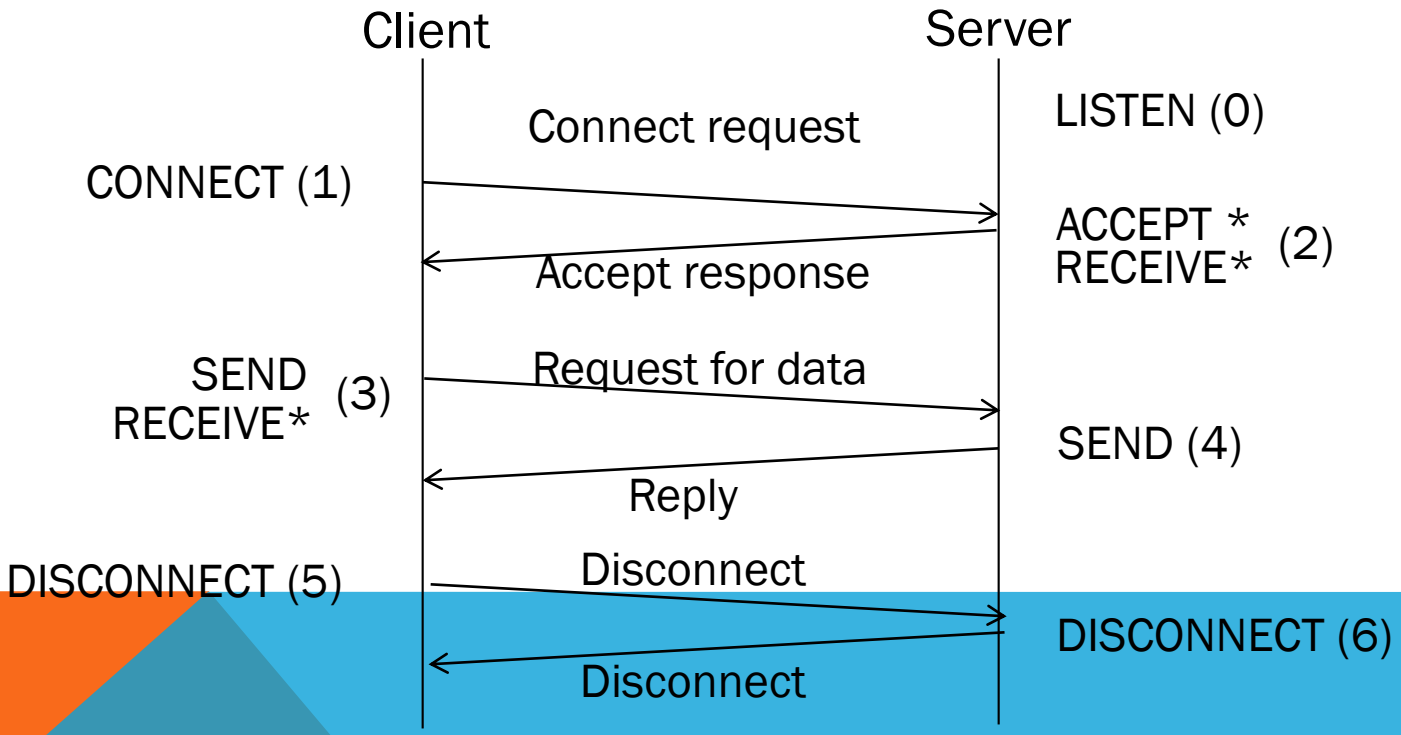
A service is provided to the layer above as primitives (operations)

Hypothetical example of service primitives that may provide a reliable byte stream (connection-oriented) service

Primitive	Meaning
LISTEN	Block waiting for an incoming connection
CONNECT	Establish a connection with a waiting peer
ACCEPT	Accept an incoming connection from a peer
RECEIVE	Block waiting for an incoming message
SEND	Send a message to the peer
DISCONNECT	Terminate a connection

SERVICE PRIMITIVES (2)

Hypothetical example of how these primitives may be used for a client-server interaction



SERVICE PRIMITIVES (3)

The primitives are called at the client and server by the higher layer using the service. The layer implements the primitives by sending messages using the services of the lower layer; these messages are assumed to be reliable for simplicity and the lower layer service is not otherwise described.

DESIGN GUIDANCE

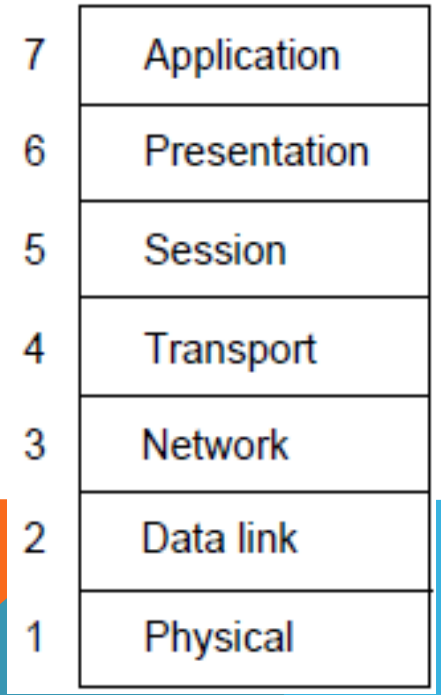
What functionality should we implement at which layer?

- This is a key design question
- Reference models provide frameworks that guide us »

OSI "7 LAYER" REFERENCE MODEL

A principled, international standard, to connect systems

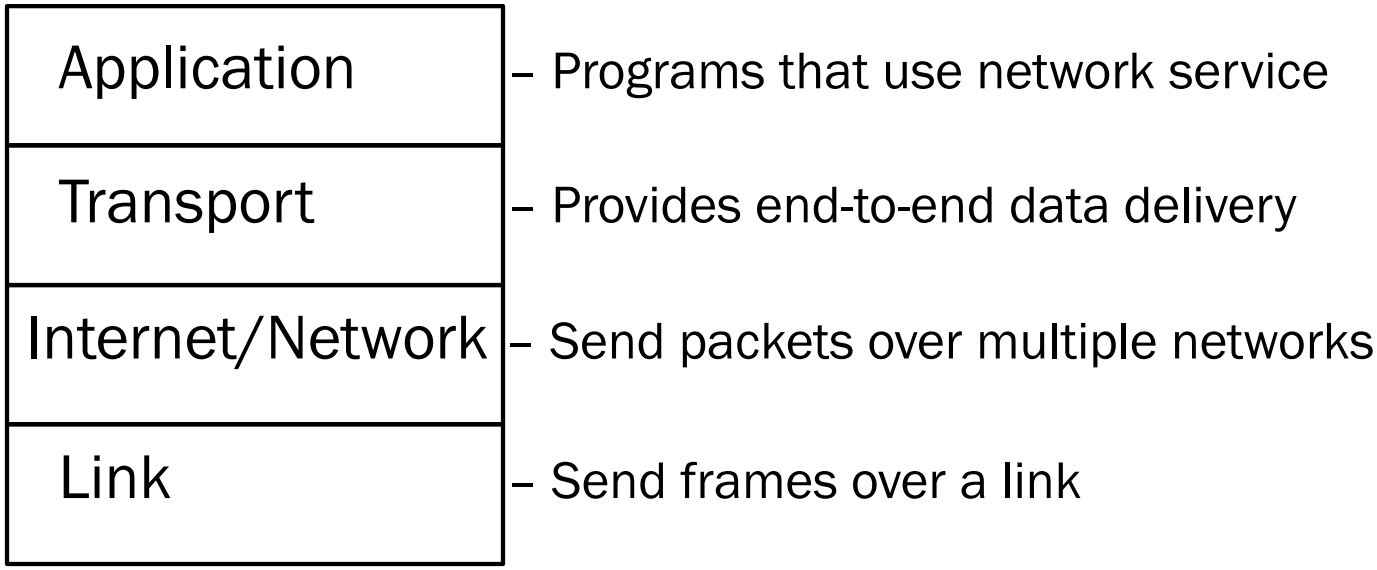
- Influential, but not used in practice.



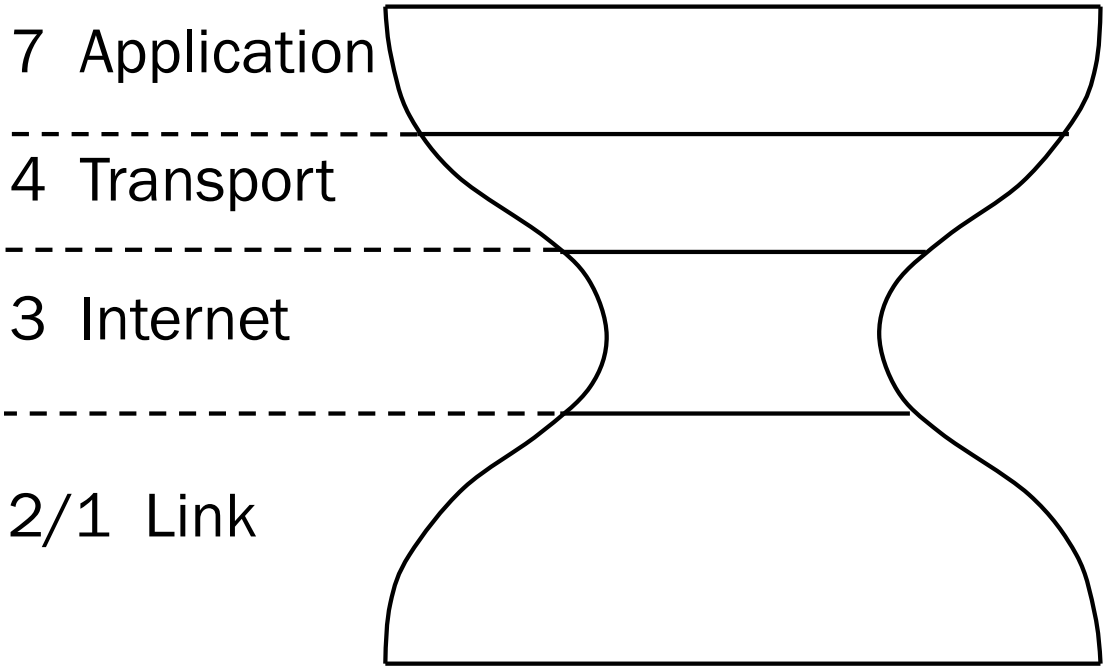
7 Provides functions needed by users
6 Converts different representations
5 Manages task dialogs
4 Provides end-to-end delivery
3 Sends **packets** over multiple links
2 Sends **frames** of information
1 Sends **bits** as signals

INTERNET REFERENCE MODEL

A four layer model based on experience; omits some OSI layers and uses IP as the network layer.

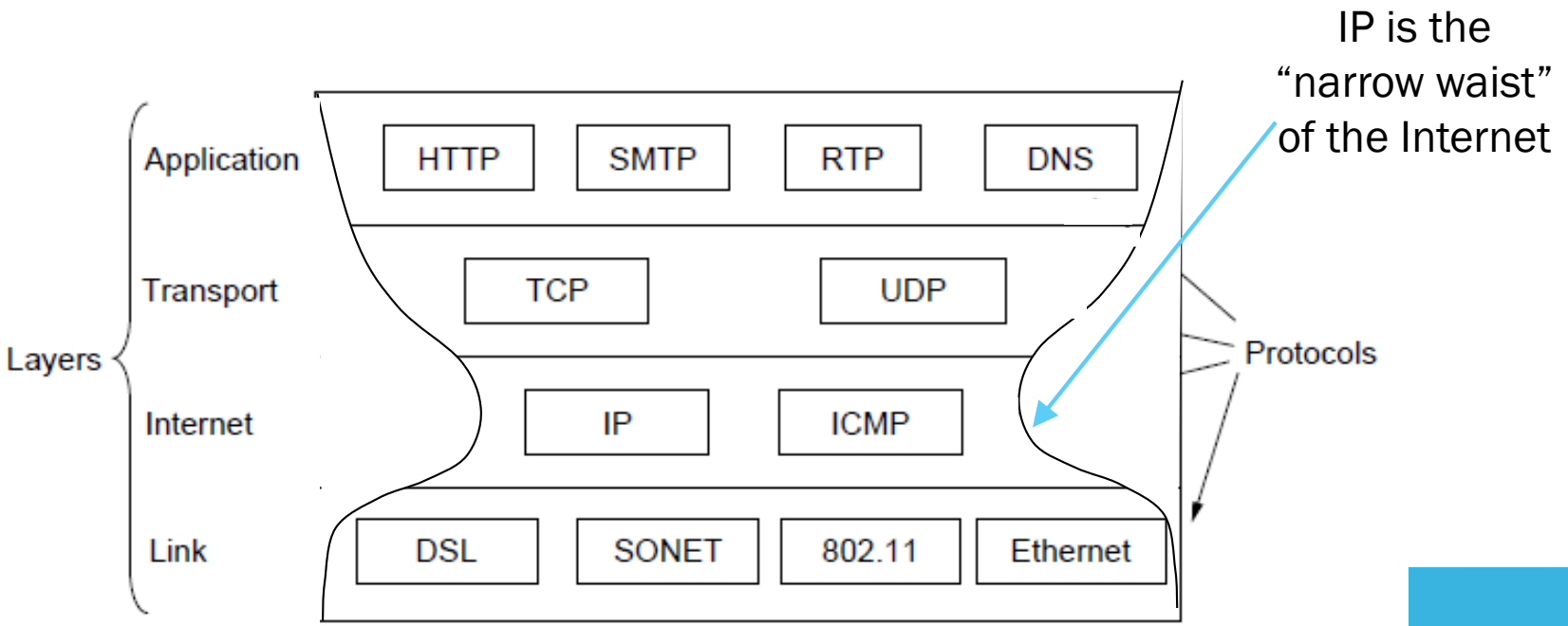


INTERNET REFERENCE MODEL (2)



TCP/IP REFERENCE MODEL

A four layer model derived from experimentation; omits some OSI layers and uses the IP as the network layer.



LAYER-BASED NAMES

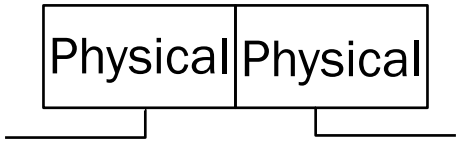
For units of data:

Layer	Unit of Data
Application	Message
Transport	Segment
Network	Packet
Link	Frame
Physical	Bit

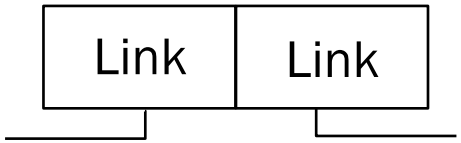
LAYER-BASED NAMES (2)

For devices in the network:

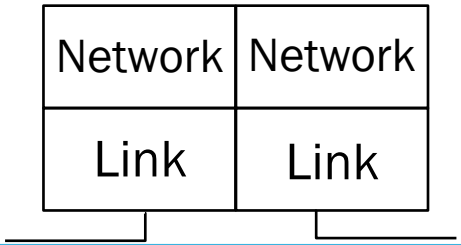
Repeater (or hub)



Switch (or bridge)



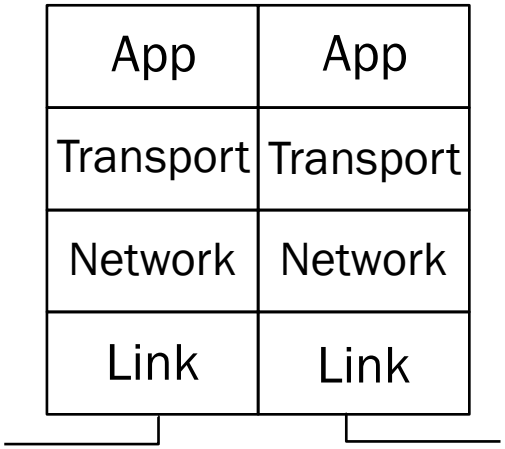
Router



LAYER-BASED NAMES (3)

For devices in the network:

Proxy or middlebox or gateway



But mostly they look like this!

A NOTE ABOUT LAYERS

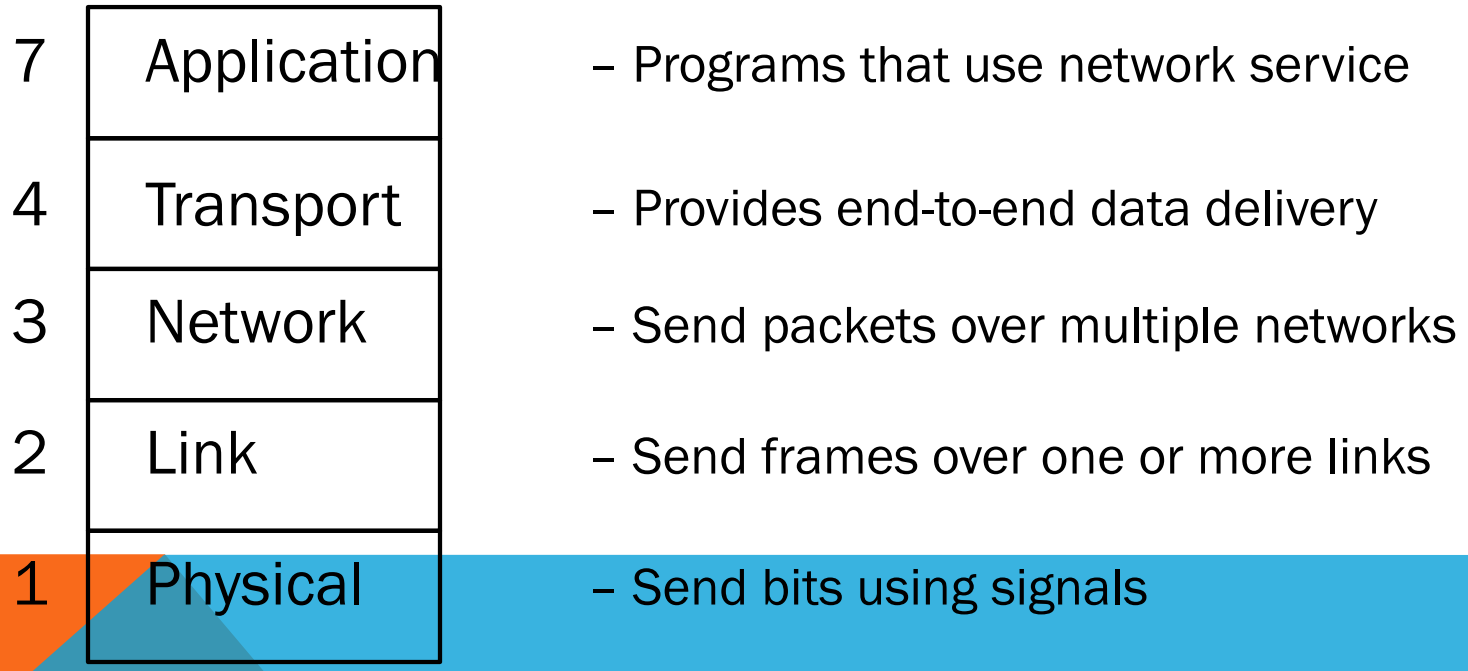
They are guidelines, not strict

- May have multiple protocols working together in one layer
- May be difficult to assign a specific protocol to a layer

COURSE REFERENCE MODEL

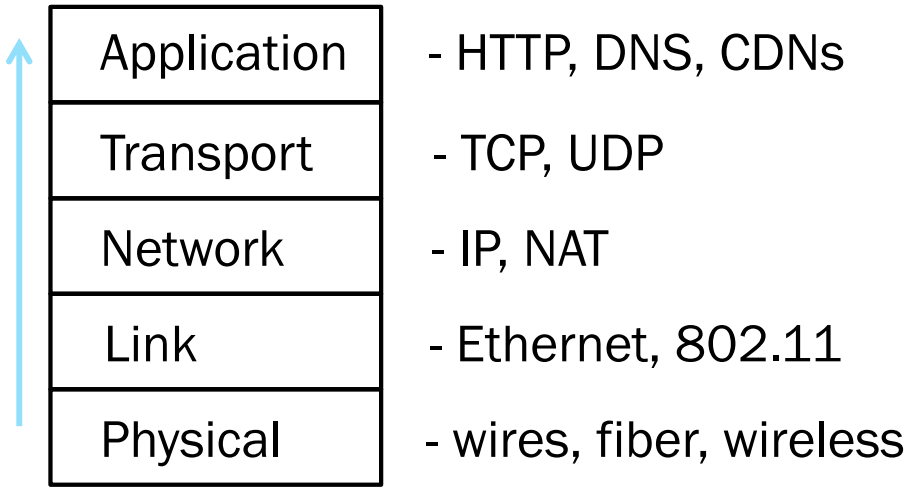
We mostly follow the Internet Reference Model

- PLUS Physical layer



LECTURE PROGRESSION

Bottom-up through the layers:



STANDARDS BODIES

Where all the protocols come from!

- Focus is on interoperability

Body	Area	Examples
ITU	Telecom	G.992, ADSL H.264, MPEG4
IEEE	Communications	802.3, Ethernet 802.11, WiFi
IETF	Internet	RFC 2616, HTTP/1.1 RFC 1034/1035, DNS
W3C	Web	HTML5 standard CSS standard