# Introduction to Network Security

Chapter 1

Network Architecture



# Chapter Topics

- Introduction
- Layered architecture
- Key terms
- Protocol Functions
- OSI model
- TCP/IP Model

# Course Overview

- Protocols
- Protocol Implementations
- Security Issues
- Performance Issues
- Several programming assignments
  - packet sniffer
  - spam email



# History of Networking

1840
- 1844 First Telegraph line
- 1861 Over 2200 telegraph offices
- 1866 First transatlantic cable
- 1875 First words on a telephone

1900
- 1880 over 30,000 phones
- 1900 over 600,000 phones
- 1910 over 5,000,000 phones
- 1920 over 11,000,000 phones

1950s
- Point-to-point network to main frames

1960
- 1968 300 baud modem
- 1969 ARPA NET (4 nodes)

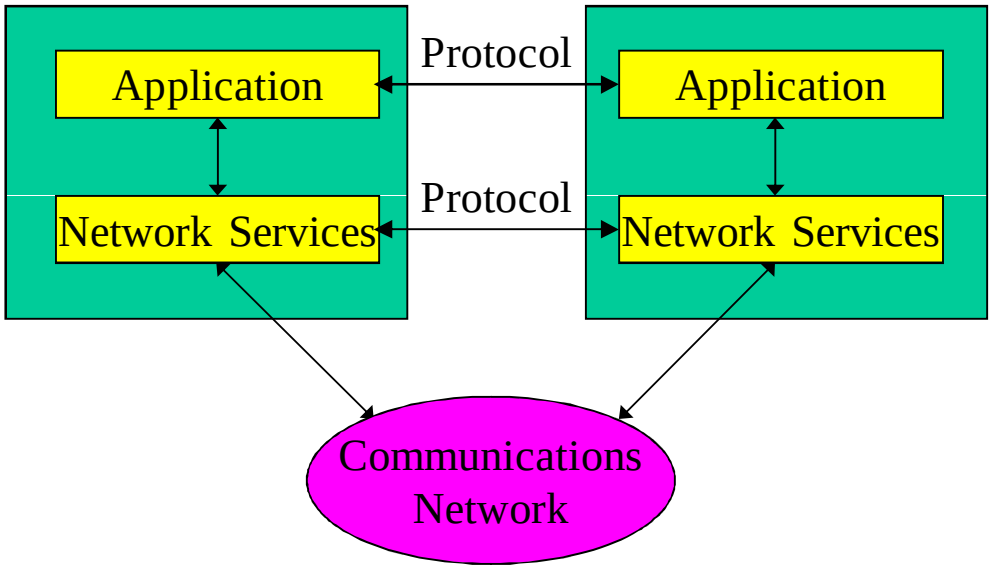
1970
- 1971 15 nodes in APRANET
- 1973 TCP/IP development
- 1973 Ethernet was proposal in a Ph.D. Dissertation
- 1977 TCP/IP test bed
- 1979 UUCPnet

1980
- 1980 ARPANET virus (accidental)
- 1983 TCP/IP becomes the protocol for ARPANET
- 1984 over 1000 hosts
- 1986 NSFNET is started
- 1987 over 10,000 hosts
- 1988 Internet worm infects over 6,000 hosts
- 1989 over 100,000 hosts

1990
- 1991 WWW released by CERN
- 1992 over 1,000,000 hosts
- 1995 First ISPs started
- 1996 over 10,000,000 hosts

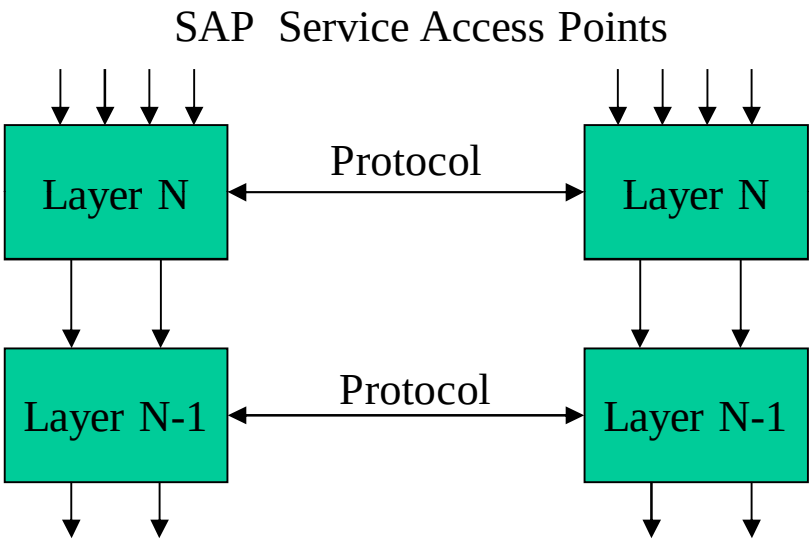
2000

# Layered Architecture






# Layered Architecture



# Layered Architecture

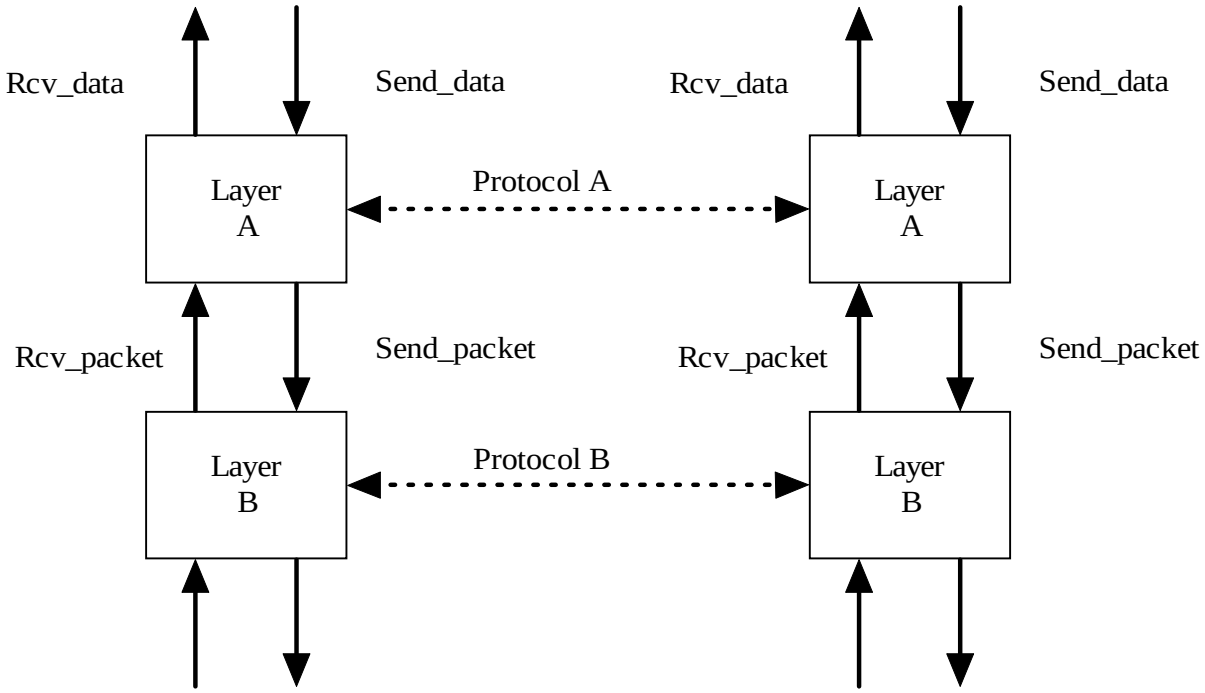
- Brought about because of a need for standards
- Layers:
  - take information from above (layer N-1)
  - and pass information below (layer N+1)
- The services are provided through the *service access points* (SAPs)
- Layer functionality is implemented through an *entity*
- Each layer contains one or more entities which are responsible for providing services to the N+1 layer



# Layered Architecture

- In order for layers to carry out functions, they need to communicate
- A layer N entity may need to communicate with another layer N entity, which does not reside on the same system, to provide the service.
- The layer N entity uses the layer N-1 services to communicate with the remote layer N entity.

# Layered Architecture



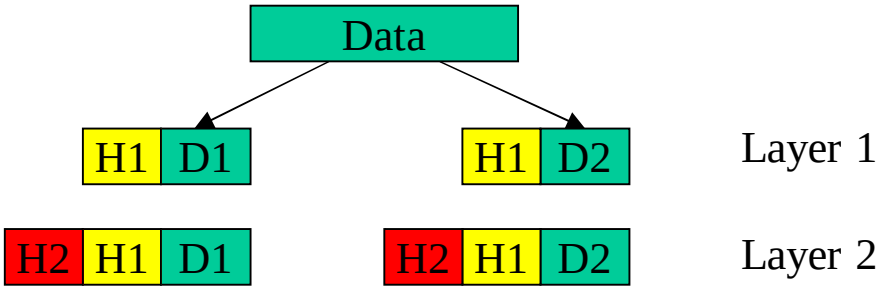



# Layered Architecture

- PROTOCOLS are the rules that have been defined for the layer N to layer N communication.
- They represent extra information
  - example: saying “hello” on the telephone is a protocol
- Protocols indicate when to send data, what language to use, etc.
- A *layer specification* defines
  - what protocol it uses
  - what it expects as input (SAPs)
  - what functions it provides
- Layer specifications allow multiple vendors to have the same functionality.
  - (ie: different ethernet card brands)

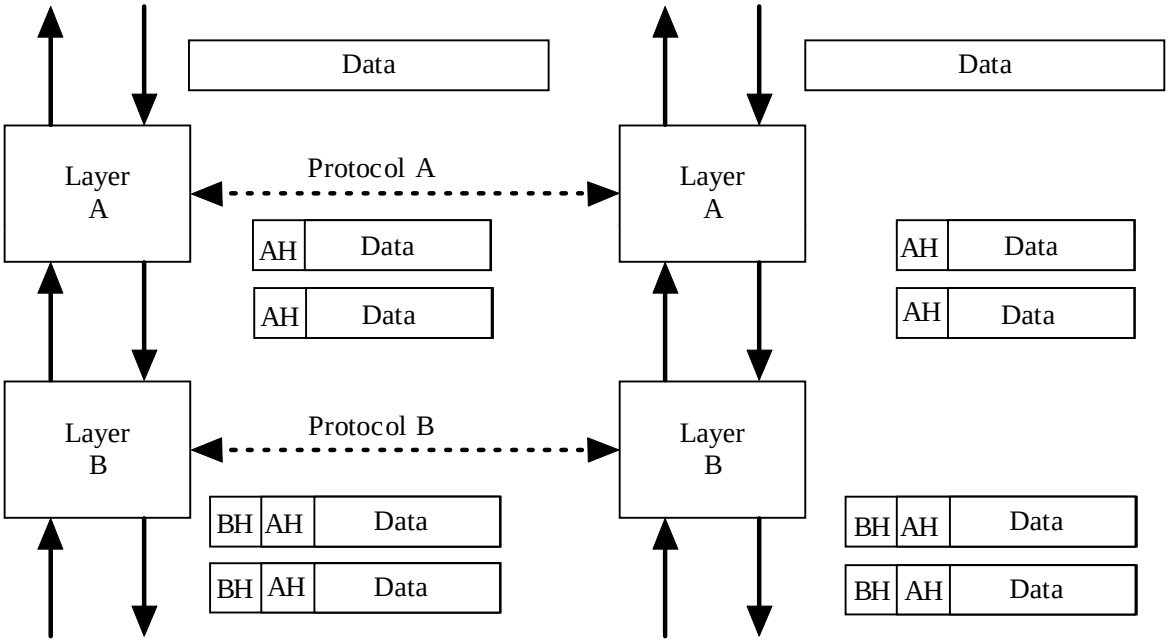
# Protocol Data Unit

- Protocol Data Unit (PDU) is the combination of data from the higher layer and the protocol or control information.
- The protocol or control information created by a layer is called the header.
- Each layer adds it's own header






# Control Information Encapsulation



# Key Terms

- The protocol defines the rules for PEER entity communication
- Service Access Points (SAP) specify how the N entity communicates with the N-1 entity.
- Services are provided by the N entity to the N+1 entity
- Functions are provided by the entity in coordination with the peer entity.



# Basic Functions of a Protocol

1. **Segmentation and reassembly:**
   - Often physical media or error control issues dictate a maximum data size
   - Therefore the data must be divided into smaller packets (**Segmentation**)
   - And eventually put back together (**Reassembly)**
   - Reassembly instructions are included in the header

# Basic Functions of a protocol

## 2. Encapsulation:

The addition of control information to the data element in the form of a header.

- **Address:** The address of the sender and/or receiver.
- **Error Detection Code:** Some sort of code is often included for error detection.
- **Protocol Control:** Additional information needed to implement the protocol.



# Basic functions of a protocol

## 3. Connection Control:

– Connectionless Data Transfer

- Data is transferred without prior coordination
- No set path

– Connection-oriented Data Transfer

- A logical association, or **Connection**, is established between entities before any data is transferred
- Example: telephone

# Connection oriented

- The three phases of **Connection Control**
  - request/connect phase
  - data transfer phase
  - terminate phase



# Basic Functions of a protocol

## 4. Ordered Delivery

- Pieces arrive in the same order as sent
- Not provided by connectionless protocols
- Not required to be provided by Connection -oriented protocols, but it is common for most. (needed for file transfer)

# Basic Functions of a protocol

## 5. Flow Control:

- Technique for assuring that the transmitting entity does not overwhelm a receiving entity.
- Flow Control is typically implemented in several layers.
- Flow control is found in most connection-oriented protocols



# Basic Functions of a protocol

## 6. Error Control:

- Technique that allows a protocol to recover from lost or damaged PDUs.
- Three mechanisms:
  - Positive acknowledgment
  - Retransmit after timeout
  - Error detection

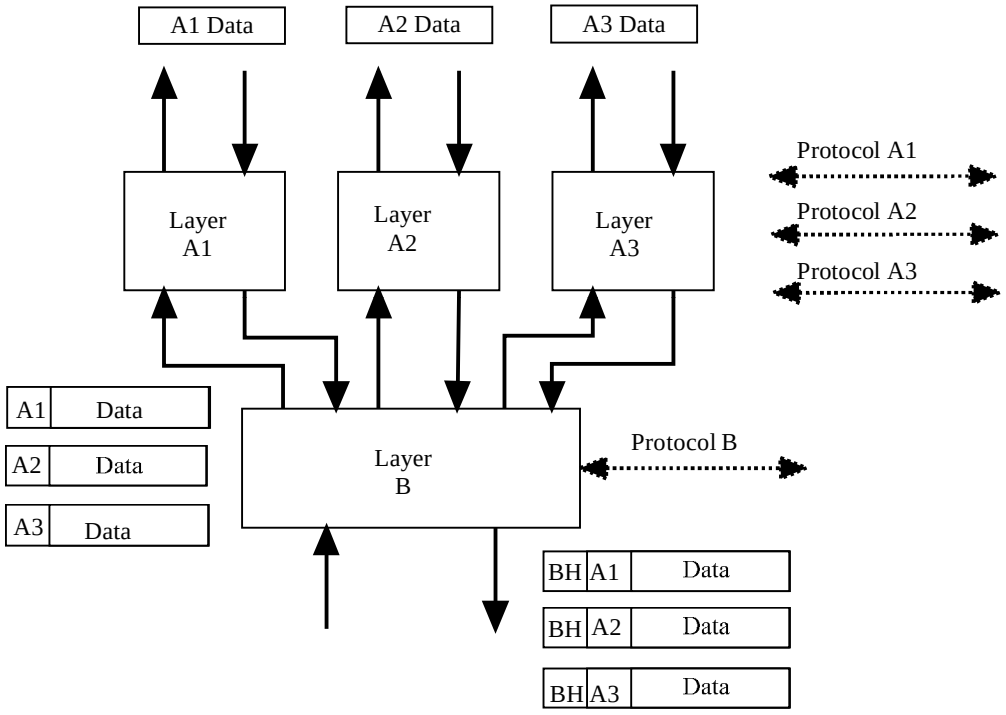
# Basic Functions of a protocol

## 7. Multiplexing:

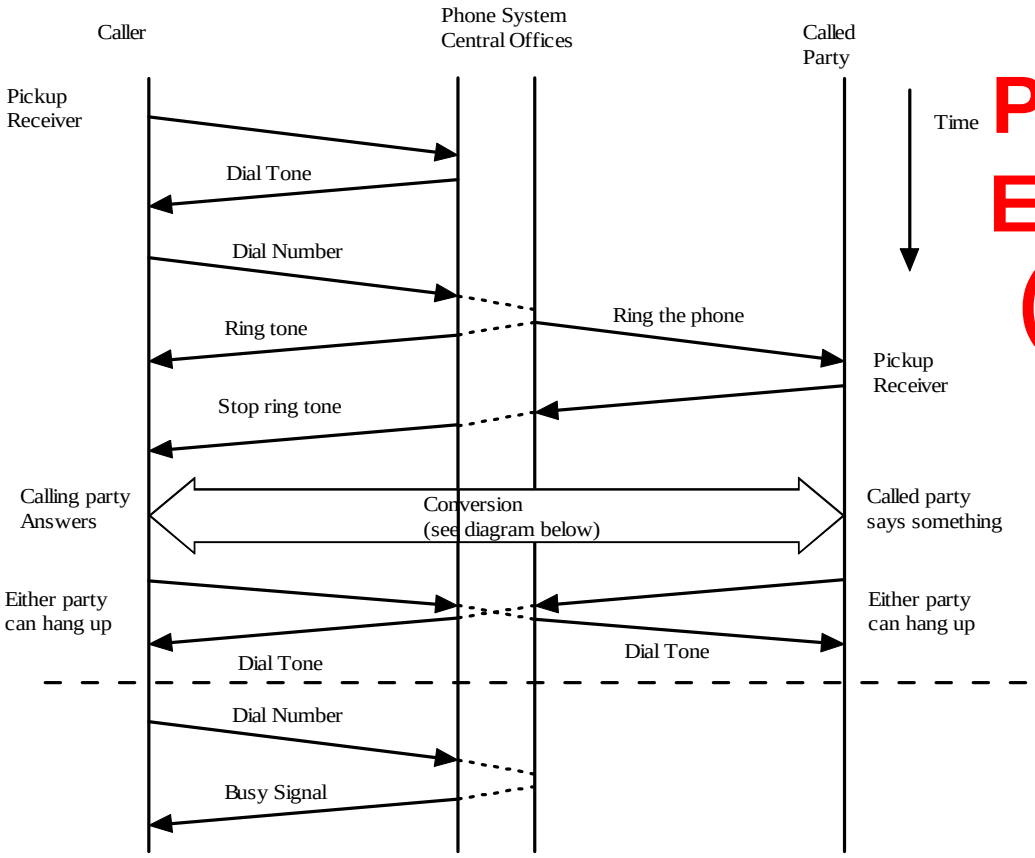
- **Upward Multiplexing** occurs when multiple higher level connections are multiplexed on a single lower level connection. Example: many applications utilize TCP (telnet, ftp, email)
- **Downward Multiplexing** occurs when a single higher level connection is multiplexed on multiple lower level connections. (not as common)
- Addressing is needed to support multiplexing



# Multiplexing

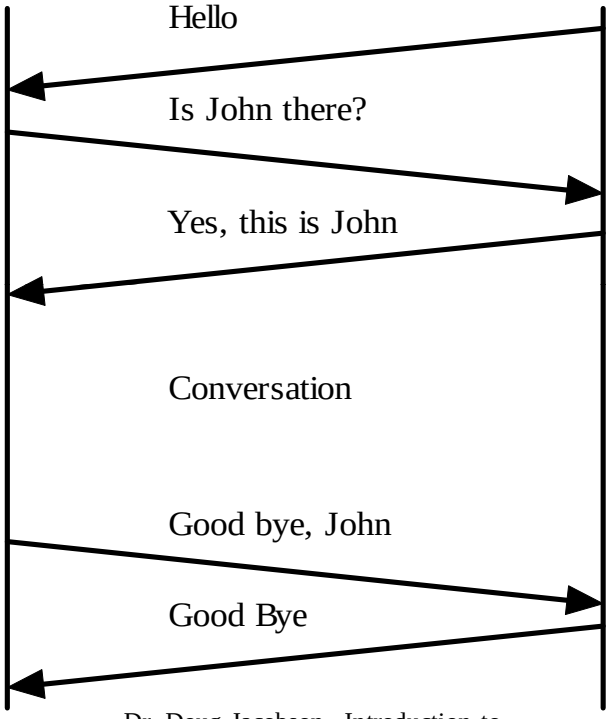


# Protocol Example (part 1)






# Protocol Example (part 2)



# OSI Model

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical



# Physical Layer

- Responsible for the transparent transmission of bit streams across the physical interconnection of systems
- Two configurations:
  - Point-to-point
  - Multipoint
- Physical layer must provide the data link entities with a means to identify the end point.
- Physical connection can be **Full Duplex** or **Half Duplex**
- Physical connection can be either bit serial or N bit parallel
- Physical layer must deliver the bits in the same order in which they were offered for transmission by the Data Link Layer.

# Data Link Layer

- Main task is to shield higher layers from the characteristics of the physical transmission medium.
- Should provide the higher layers with a reliable transmission which is basically **Error-Free**, although errors may occur in the transmission on the physical connection.
- Services provided should be independent of the data transmitted.
- Data link layer connects two network entities in adjacent systems called **Data link connection**.



# Data Link Layer

- Each data-unit from the network layer is mapped into the data link protocol data unit along with the data link protocol information, and is called a **Frame.**
- The data link layer must provide a method of recognizing the start and end of the **Frame**.
- Frames must be presented to the network in the same order they are received.
- The data link layer should also implement **Flow Control** to prevent data overrun.

# Network Layer

- The primary responsibility of the network layer is to provide the transparent transfer of all data submitted by the transport layer to any transport entity anywhere in the network.
- The network layer must handle the routing of data packets.
- The network layer can be the highest layer in a device such as a gateway or router.
- IP protocol



# Transport Layer

- Responsible for a **reliable** transparent data transfer between two session layer entities.
- Transport connection is provide to the session entities independent of their location.
- Transport layer must optimize resources while maintaining a guaranteed quality of service.
- Session layer requests a level of service and once the transport connection is provided with a certain quality of service it must be maintained unless notified of the change.
- TCP protocol

# Transport Layer

- The transport layer is only concerned with transfer of data between session layers. It is not aware of the structure of the underlying layers or the topology.
- The transport layer will use the network layer to get a network connection from one transport entity to another.
- Depending on the quality of the network the transport layer may have to perform additional functions to offer the service.
- The transport layer provides flow and error control.



# Session Layer

- The session layer is not concerned with the network.
- The session layer's goal is to coordinate the dialog between presentation layers
- The session layer must provide the establishment of a session connection and the management of the dialog on that connection.
- Example: An atm maintains a constant connection with a bank (transport service). The session starts when the user logs in.

# Presentation Layer

- The presentation layer provides the application layer with services related to the presentation of information in a form that is meaningful to the application entities.
- The presentation layer provides the mechanism for the application layer to translate its data into a common format that can be translated by the peer application entity.

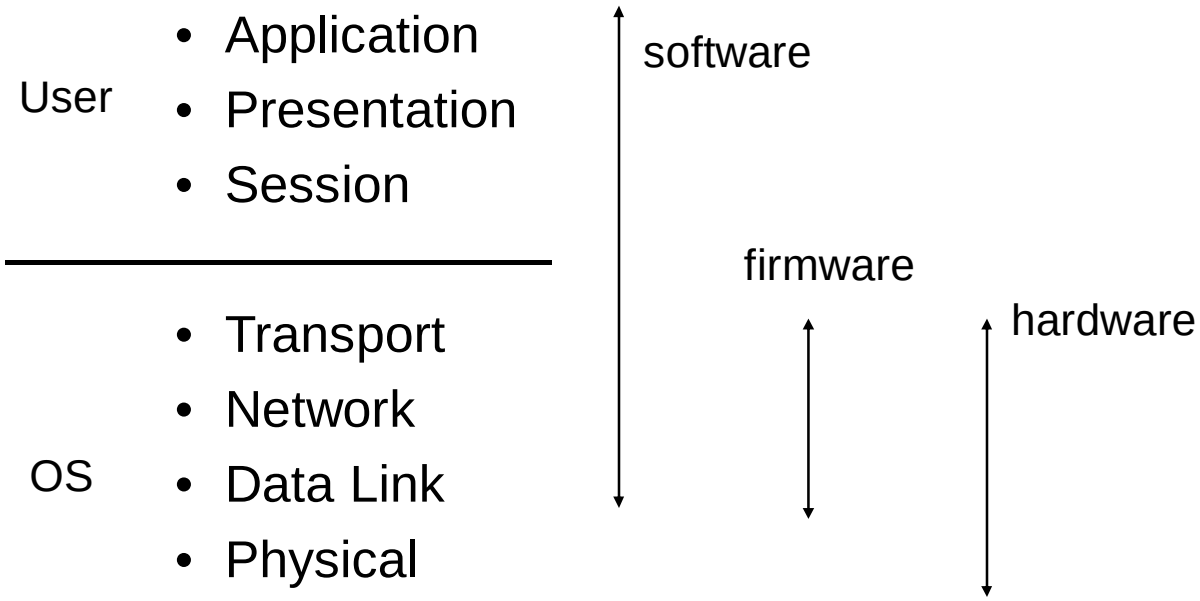
- data format M1 → network format → data format M2



# Application Layer

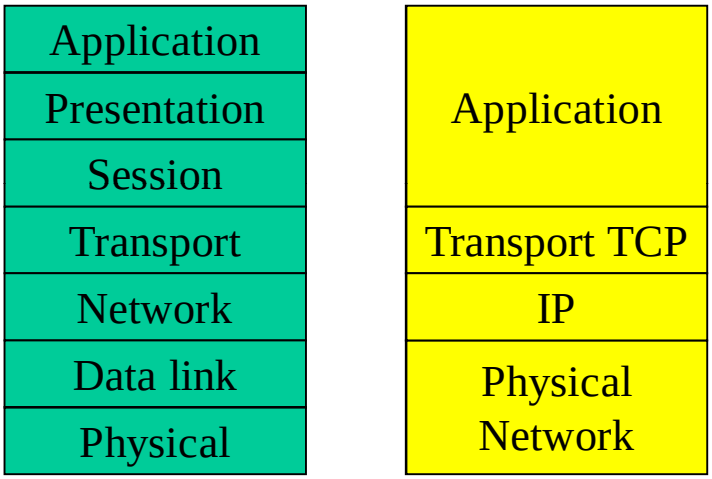
- The highest layer and it provides a means for application processes to access the OSI stack.
- Provides both general services and application specific services.
- This is what the user sees
- Examples: telnet, ftp, web

# Layered Network Model

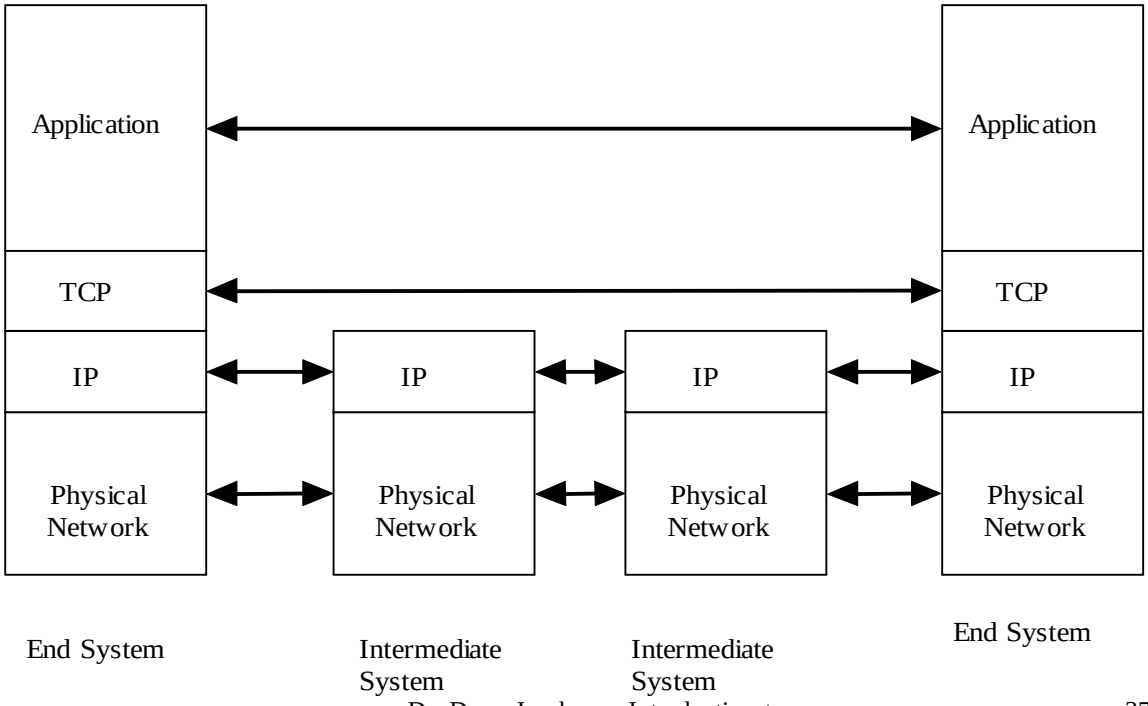





# TCP/IP vs OSI



# TCP/IP Network






# Non-layered Services