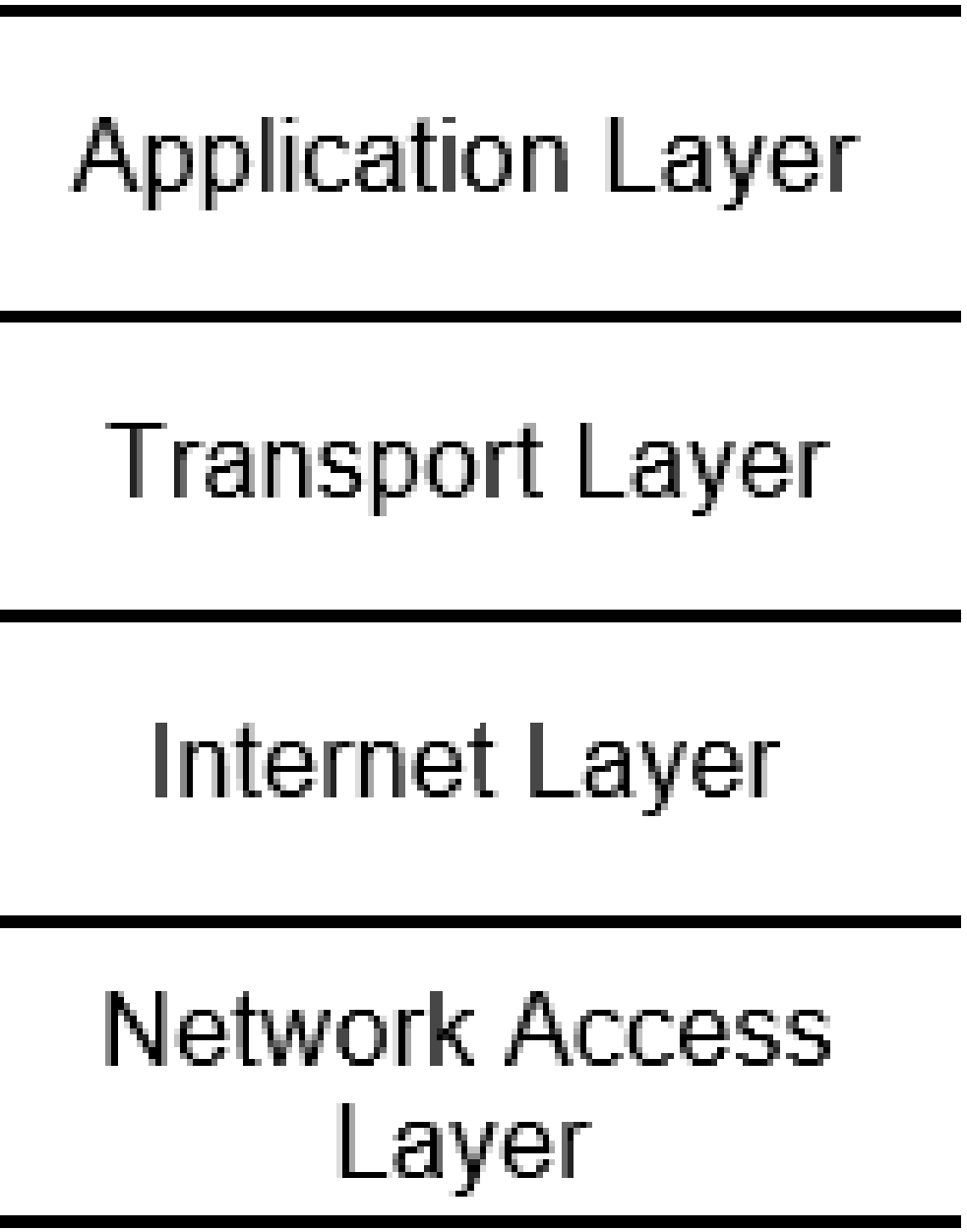
TCP/IP- Lecture 2

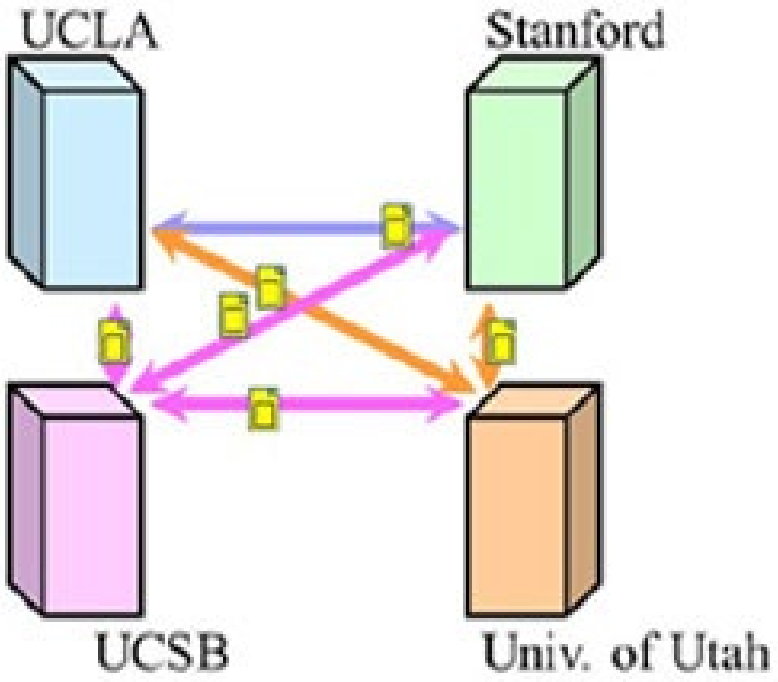
Andrei.doncescu@laas.fr



How TCP/IP Works

- The four-layer model is a common model for describing TCP/IP networking, but it isn't the only model.
- The ARPAnet model, for instance, as described in RFC 871, describes three layers: the Network Interface layer, the Host-to-Host layer, and the Process-Level/Applications layer.
- Other descriptions of TCP/IP call for a five-layer model, with Physical and Data Link layers in place of the Network Access layer (to match OSI). Still other models might exclude either the Network Access or the Application layer, which are less uniform and harder to define than the intermediate layers.
- The names of the layers also vary. The ARPAnet layer names still appear in some discussions of TCP/IP, and the Internet layer is sometimes called the Internetwork layer or the Network layer.

1969: The original ARPANET had 4 nodes



A message could start out at UCSB take many different routes to get to Stanford

TCP/IP Model

- **Network Access layer:** Provides an interface with the physical network. Formats the data for the transmission medium and addresses data for the subnet based on physical hardware addresses. Provides error control for data delivered on the physical network.

- **Internet layer:** Provides logical, hardware-independent addressing so that data can pass among subnets with different physical architectures. Provides routing to reduce traffic and support delivery across the internetwork. (The term **internetwork** refers to an interconnected, greater network of local area networks (LANs), such as what you find in a large company or on the Internet.) Relates physical addresses (used at the Network Access layer) to logical addresses.

- **Transport layer:** Provides flow-control, error-control, and acknowledgment services for the internetwork. Serves as an interface for network applications.

- **Application layer:** Provides applications for network troubleshooting, file transfer, remote control, and Internet activities. Also supports the network application programming interfaces (APIs) that enable programs written for a particular operating environment to access the network.

TCP/IP Model

When the TCP/IP protocol software prepares a piece of data for transmission across the network, each layer on the sending machine adds a layer of information to the data that is relevant to the corresponding layer on the receiving machine.

The Internet layer of the computer sending the data adds a header with some information that is significant to the Internet layer of the computer receiving the message.

This process is sometimes referred to as encapsulation.

At the receiving end these headers are removed as the data is passed up the protocol stack.

TCP/IP Model Architecture:

Layer #	Layer Name	Protocol	Protocol Data Unit	Addressing
5	Application	HTTP, SMTP, etc…	Messages	n/a
4	Transport	TCP/UDP	Segments/ Datagrams	Port #s
3	Network or Internet	IP	Packets	IP Address
2	Data Link	Ethernet, Wi-Fi	Frames	MAC Address
1	Physical	10 Base T, 802.11	Bits	n/a

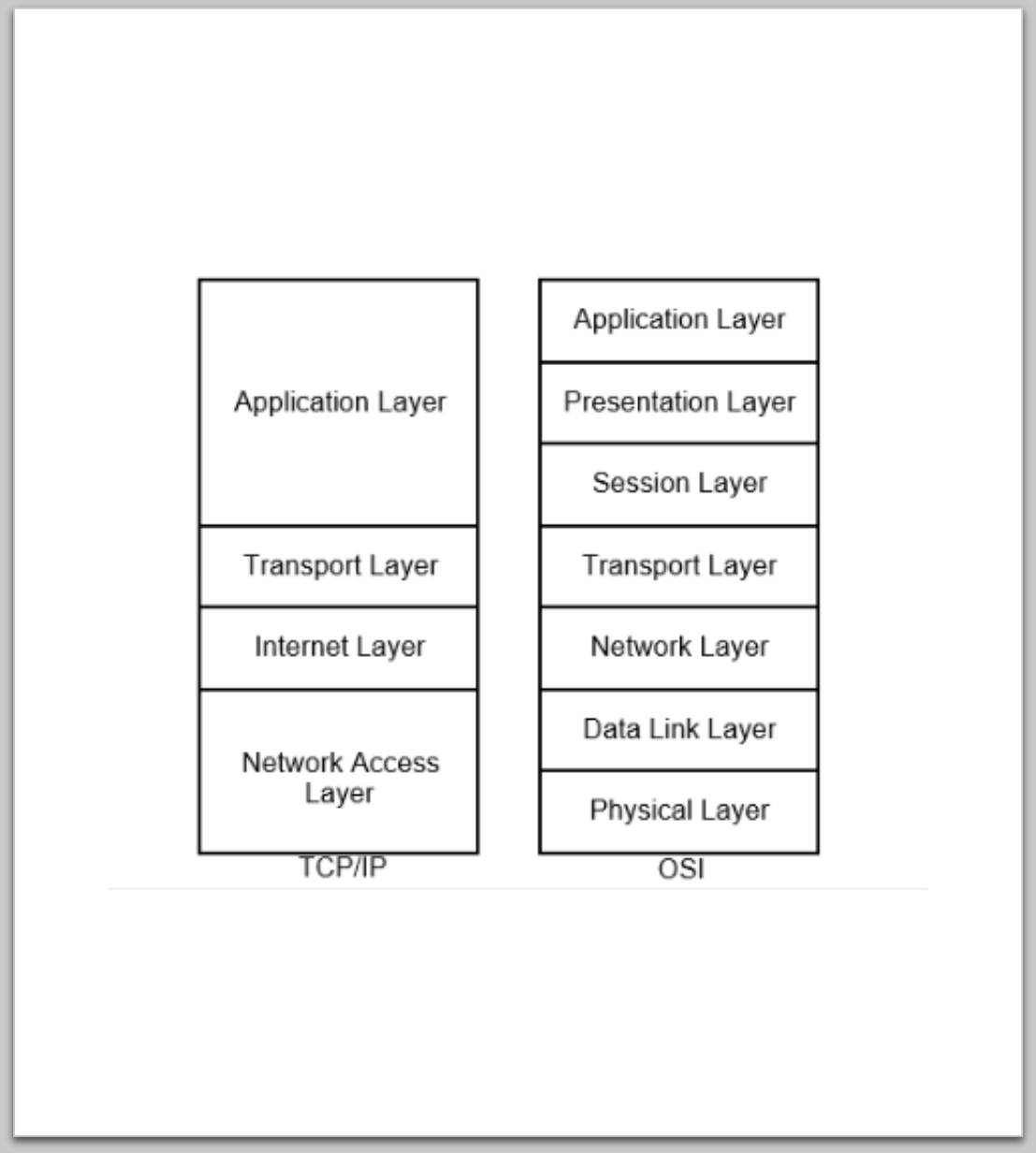
TCP/IP and the OSI Model

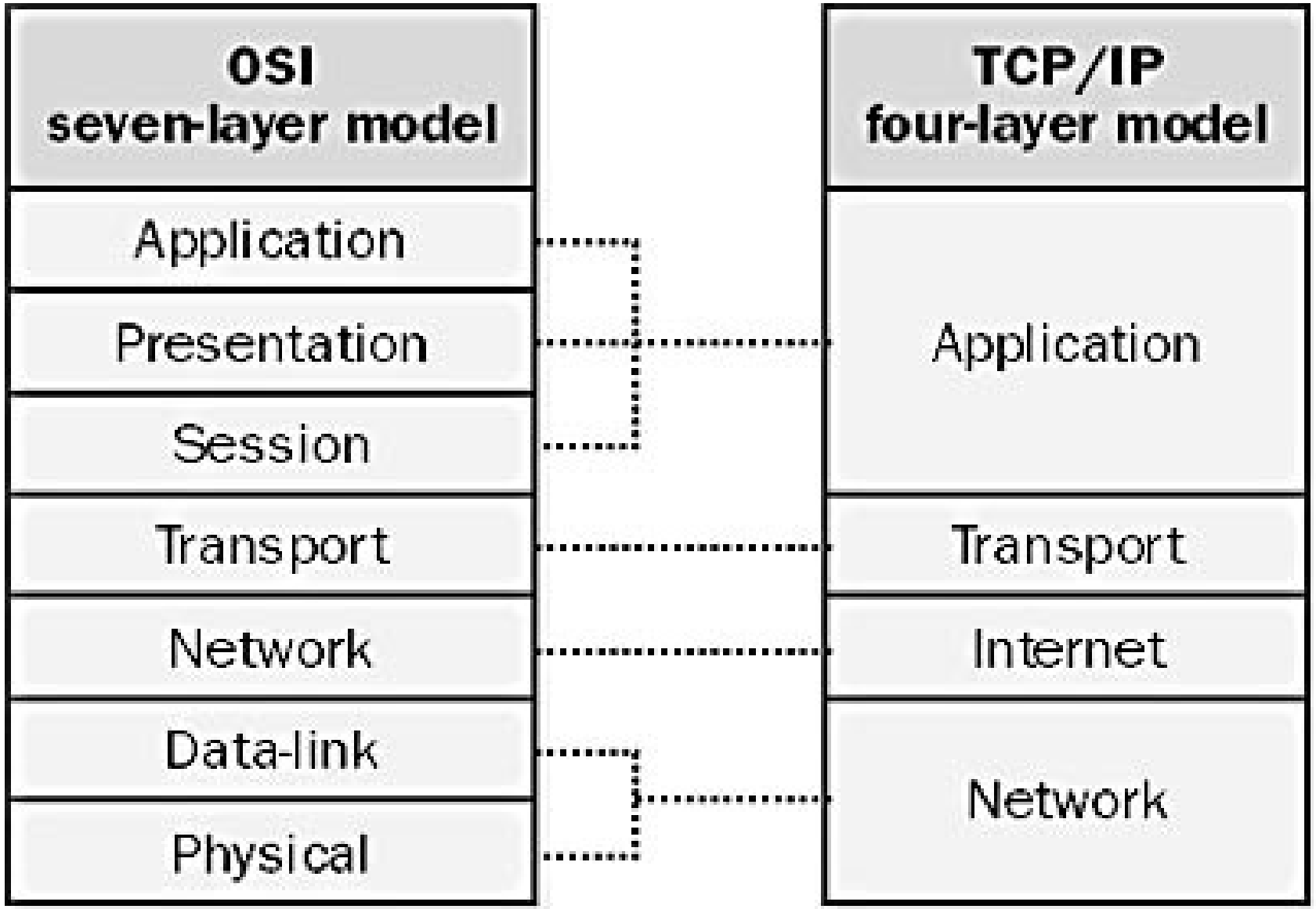
- The networking industry has a standard seven-layer model for network protocol architecture called the Open Systems Interconnection (OSI) model. The OSI model represents an effort by the International Organization for Standardization (ISO), an international standards organization, to standardize the design of network protocol systems to promote interconnectivity and open access to protocol standards for soft- ware developers.
- TCP/IP was already on the path of development when the OSI standard architecture appeared and, strictly speaking,
- TCP/IP does not conform to the OSI model.
- The two models did have similar goals, and enough interaction occurred among the designers of these standards that they emerged with a certain compatibility.
- The OSI model has been very influential in the growth and development of protocol implementations, and it is quite common to see the OSI terminology applied to TCP/IP.

OSI Model

- The seven layers of the OSI model are as follows:
  - **Physical layer:** Converts the data into the stream of electric or analog pulses that will actually cross the transmission medium and oversees the trans- mission of the data
  - **Data Link layer:** Provides an interface with the network adapter; maintains logical links for the subnet
  - **Network layer:** Supports logical addressing and routing
  - **Transport layer:** Provides error control and flow control for the internetwork
  - **Session layer:** Establishes sessions between communicating applications on the communicating computers
  - **Presentation layer:** Translates data to a standard format; manages encryption and data compression
  - **Application layer:** Provides a network interface for applications; supports network applications for file transfer, communications, and so forth

- Note that the OSI model divides the duties of the Application layer into three layers: Application, Presentation, and Session.
- OSI splits the activities of the Network Access layer into a Data Link layer and a Physical layer. This increased subdivision adds some complexity, but it also adds flexibility for developers by targeting the protocol layers to more specific services.
- In particular, the division at the lower level into the Data Link and Physical layers separates the functions related to organizing communication from the functions related to accessing the communication medium.
- The three upper OSI layers offer a greater variety of alternatives for an application to interface with the protocol stack.




OSI
seven-layer model
Application
Presentation
Session
Transport
Network
Data-link
Physical
TCP/IP
four-layer model
Application
Transport
Internet
Network

Data Packages

The important thing to remember about the TCP/IP protocol stack is that each layer plays a role in the overall communication process.

Each layer invokes services that are necessary for that layer to perform its role. As an outgoing transmission passes down through the stack, each layer includes a bundle of relevant information called a **header** along with the actual data.

The little data package containing the header and the data then becomes the data that is repackaged at the next lower level with the next lower layer's header.

The reverse process occurs when data is received on the destination computer. As the data moves up through the stack, each layer unpacks the corresponding header and uses the information.

Data Packages

As the data moves down through the stack, the effect is a little like the nested Russian wooden dolls you might have seen; the innermost doll is enclosed in another doll, which is then enclosed in another doll, and so on.

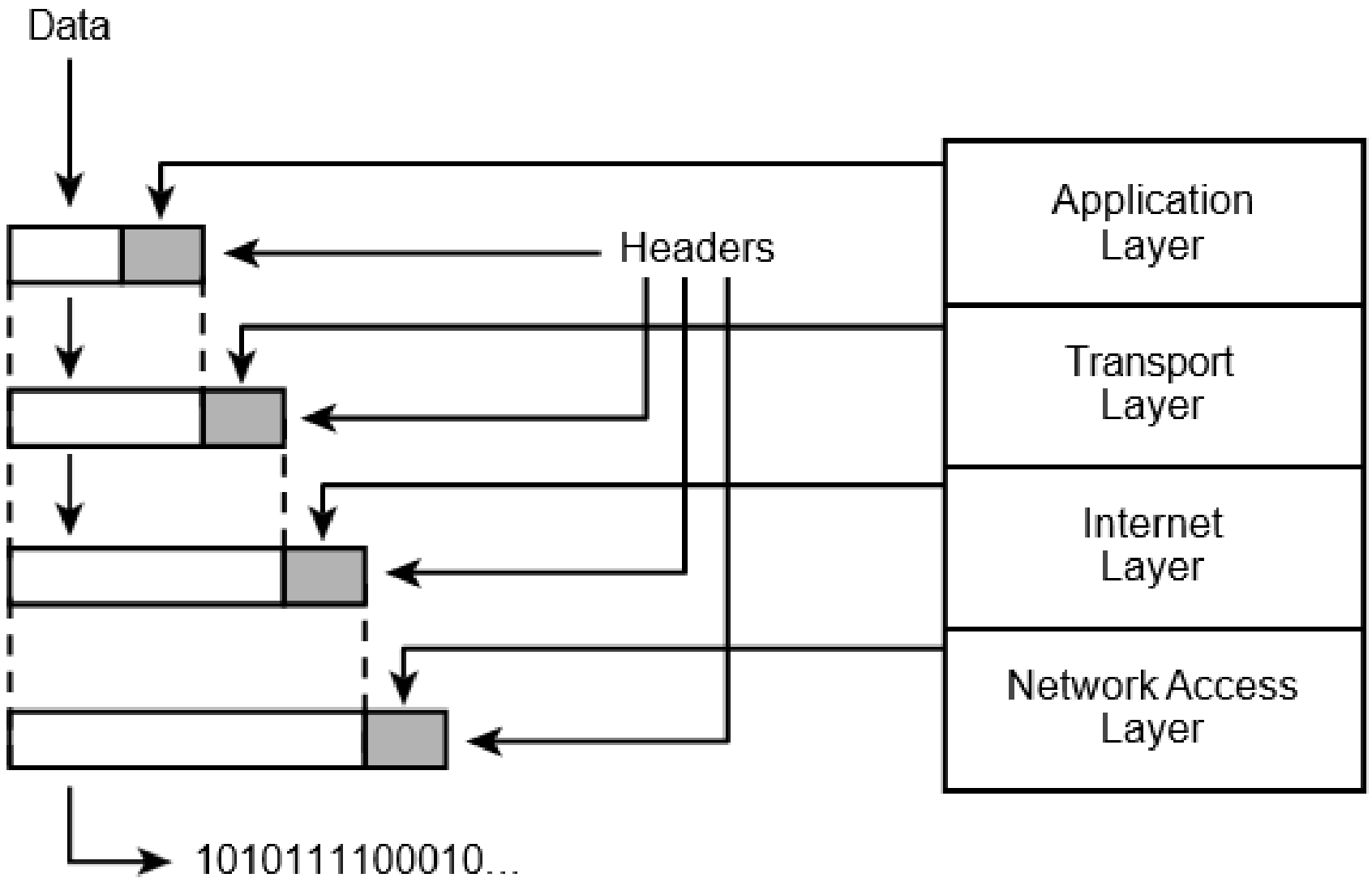
At the receiving end, the data packages are unpacked, one by one, as the data climbs back up the protocol stack.

The Internet layer on the receiving machine uses the information in the Internet layer header.

The Transport layer uses the information in the Transport layer header.

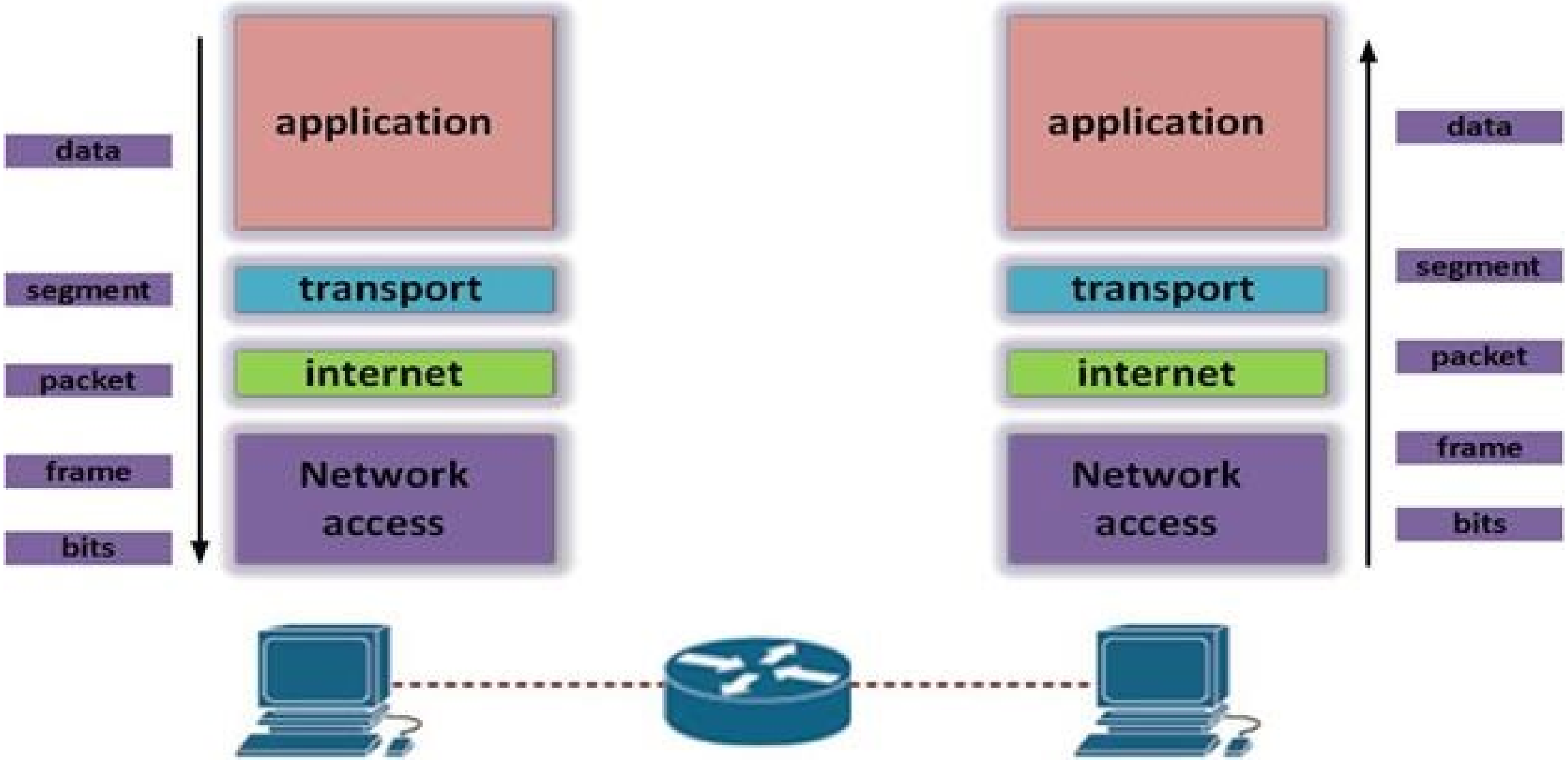
At each layer, the package of data takes a form that provides the necessary information to the corresponding layer on the receiving machine.

Because each layer is responsible for different functions, the form of the basic data package is very different at each layer.

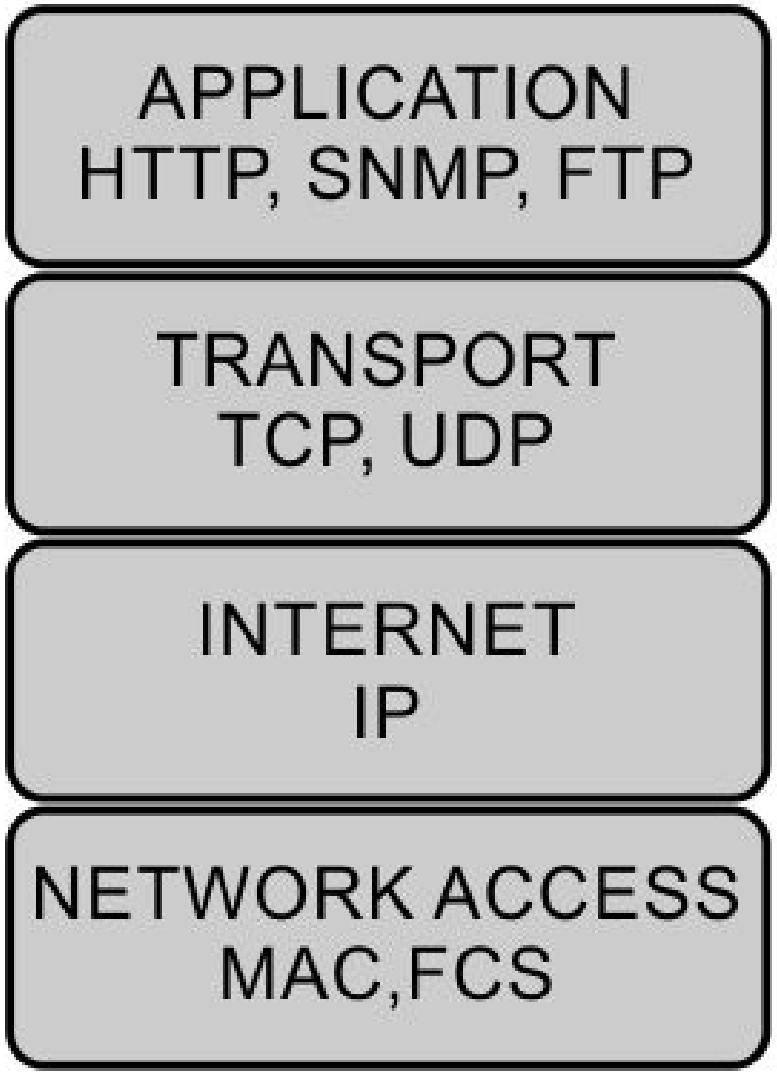

Data
Headers
Application Layer
Transport Layer
Internet Layer
Network Access Layer
10101111100010…

Data Packages

- The data packet looks different at each layer, and at each layer it goes by a different name. The names for the data packages created at each layer are as follows:
  - The data package created at the Application layer is called a **message**.
  - The data package created at the Transport layer, which encapsulates the Application layer message, is called **a segment** if it comes from the Transport layer's TCP protocol. If the data package comes from the Transport layer's **User Datagram Protocol (UDP)** protocol, it is called a **datagram**.
  - The data package at the Internet layer, which encapsulates the Transport layer segment, is called a datagram.
  - The data package at the Network Access layer, which encapsulates and may subdivide the datagram, is called a **frame**. This frame is then turned into a bitstream at the lowest sublayer of the Network Access layer.


application
transport
internet
Network access
application
transport
internet
Network access
data
segment
packet
frame
bits
data
segment
packet
frame
bits

TCP/IP Model



Interface for end point service
Examples are web browsing and email

Host to host communications
Segments and Diagrams

Source and destination IP addresses
www.google.com = IP address
Packets

Source and destination MAC addresses
Ethernet Frames
Physical Media

AD-net

TCP/IP Networking

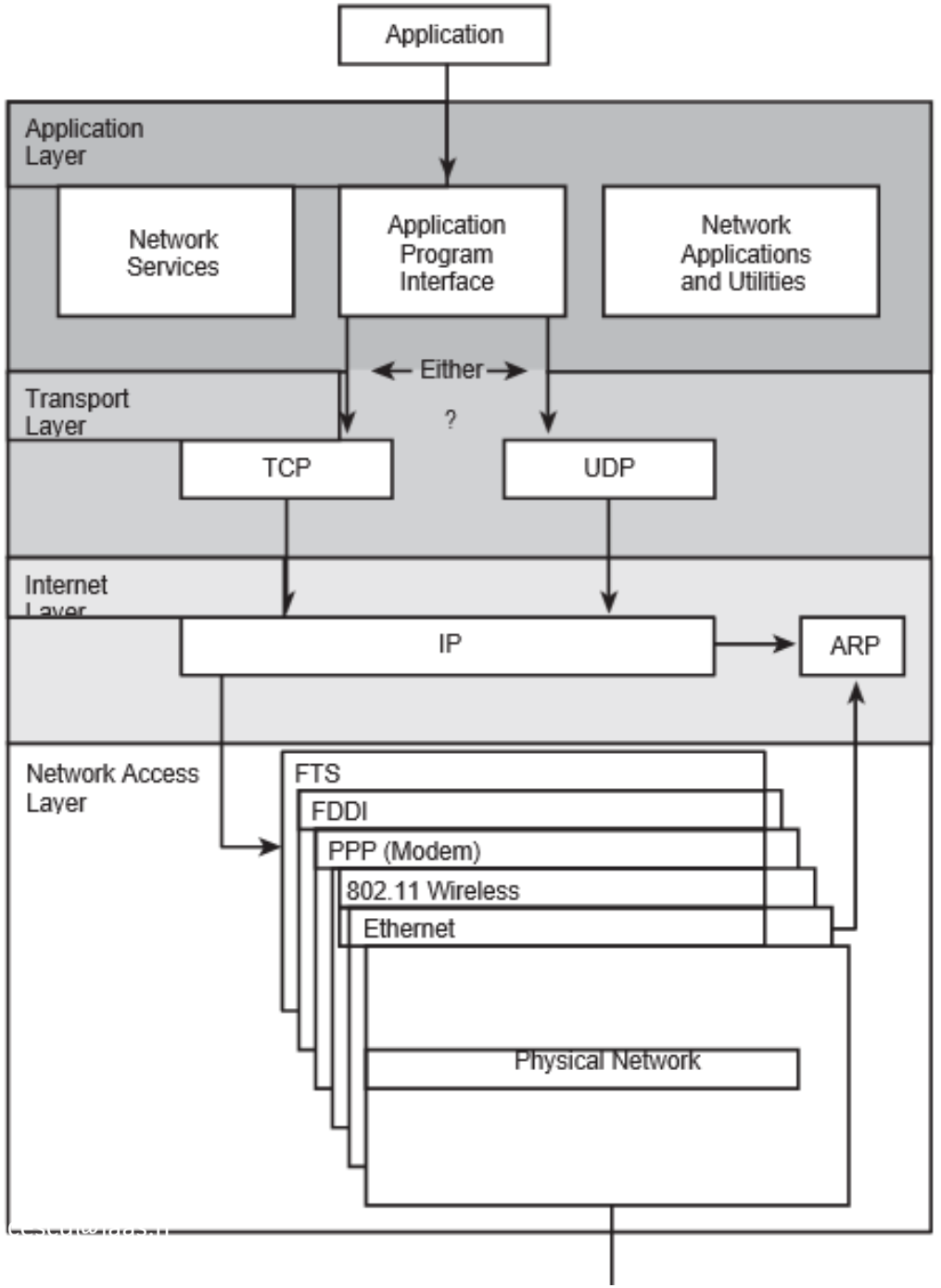
The basic scenario is as follows:

- Data passes from a protocol, network service, or application programming interface (API) operating at the Application layer through a TCP or UDP port to either of the two Transport layer protocols (TCP or UDP). Programs can access the network through either TCP or UDP, depending on the pro- gram's requirements:
  - **TCP** is a connection-oriented protocol. TCP goes to great effort to guarantee the delivery of the data. TCP is more reliable than UDP, but the additional error checking and flow control mean that TCP is slower than UDP.
  - **UDP** is a connectionless protocol. It is faster than TCP, but it is not as reliable. UDP offloads more of the error control responsibilities to the application.

TCP/IP Networking

- The data segment passes to the Internet level, where the **IP** protocol provides logical-addressing information and encloses the data into a datagram.
- The IP datagram enters the Network Access layer, where it passes to software components designed to interface with the physical network.
  - The Network Access layer creates one or more data frames designed for entry onto the physical network. In the case of a LAN system such as ethernet, the frame may contain physical address information obtained from lookup tables maintained using the Internet layer **ARP** protocol. (ARP, Address Resolution Protocol, translates IP addresses to physical addresses.)

TCP/IP Networking



Q&A

- Q. *What are the principal advantages of TCP/IP's modular design?*
- A. Because of TCP/IP's modular design, the TCP/IP protocol stack can adapt easily to specific hardware and operating environments. Breaking the networking software into specific, well designed components also makes it easier to write programs that interact with the protocol system.

- Q*. What functions are provided at the Network Access layer?*
- A. The Network Access layer provides services related to the specific physical net- work. These services include preparing, transmitting, and receiving the frame over a particular transmission medium, such as an ethernet cable.

- Q. *Which OSI layer corresponds to the TCP/IP Internet layer?*
- A. TCP/IP's Internet layer corresponds to the OSI Network layer.

- Q. *Why is header information enclosed at each layer of the TCP/IP protocol stack?*
- A. Because each protocol layer on the receiving machine needs different informa- tion to process the incoming data, each layer on the sending machine encloses header information.

Quiz

What two OSI layers map into the TCP/IP Network Access layer?

What TCP/IP layer is responsible for routing data from one computer to another?

What are the advantages and disadvantages of UDP as compared to TCP?

Which layer deals with frames?

What does it mean to say that a layer encapsulates data?

Exercises

List	List the functions performed by each layer in the TCP/IP stack.
List	List the layer(s) that deal with datagrams.
Explain	Explain how TCP/IP would have to change to use a newly invented type of network.
Explain	Explain what it means to say that TCP is a reliable protocol.

Key Terms

- **Address Resolution Protocol (ARP):** A protocol that resolves logical IP addresses to physical addresses.
- **Application layer:** The layer of the TCP/IP stack that supports network applications and provides an interface to the local operating environment.
- **Datagram:** The data package passed from the Internet layer to the Network Access layer, or a data package passed from UDP at the Transport layer to the Internet layer.
- **Frame:** The data package created at the Network Access layer.
- **Header:** A bundle of protocol information attached to the data at each layer of the protocol stack.
- **Internet layer:** The layer of the TCP/IP stack that provides logical addressing and routing.
- **IP (Internet Protocol):** The Internet layer protocol that provides logical addressing and routing capabilities.
- **Message:** In TCP/IP networking, a message is the data package passed from the Application layer to the Transport layer. The term is also used generically to describe a message from one entity to another on the net- work. The term doesn't always refer to an Application layer data package.
- **Network Access layer:** The layer of the TCP/IP stack that provides an interface with the physical network.
- **Segment:** The data package passed from TCP at the Transport layer to the Internet layer.
- **TCP (Transmission Control Protocol):** A reliable, connection-oriented protocol of the Transport layer.
- **Transport layer:** The layer of the TCP/IP stack that provides error control and acknowledgment and serves as an interface for network applications.
- **UDP (User Datagram Protocol):** An unreliable, connectionless protocol of the Transport layer.