We will now continue to look at the seven layers of computer protocol, as defined by the ISO standard model. To refresh:

- The physical layer is concerned with the direct transmission of information. It includes fiber optic cables, wireless transmission, copper wires, and any medium used to transfer information from one machine to the other.
- The data layer verifies that information arrives in the proper format. It appends a "checksum" to the end of the message it sends; if the message is altered in even a small way, the checksum will change dramatically, indicating that something is amiss. If the message arrives without a matching checksum, it is re-sent.
    - Logical Link Control (LLC) defines how the information is sent over the transmission layer. It also provides a data link to the higher layers.
    - Medium Access Control (MAC) decided when the information is sent over the transmission layer. If many computers want to send information at once, it decides which machine may transmit at any given time.

The network layer is concerned with the transmission of "packets", which are small units of information. There are many potential paths that a computer could take when it is sending information; the network layer specifies which paths a computer actually uses when connecting to another.

A network becomes congested when too many packets are present at the same time. In order to deal with congestion, the network layer may re-route the packets on a longer path, in order to prevent a bottleneck from occurring at one machine.

In a broadcast network, the routing problem is simple: send the information to all computers, and it will arrive at its destination quickly. Therefore, in broadcast networks, the network later may be small, or even nonexistent. However, if the network is large, like for the internet, a larger network layer is used.

The transport layer is what ensures reliable services and deals with lost messages.

The transport layer takes large messages and breaks them up into many smaller messages. It also labels each message, so that they can be reordered when they arrive at their destination. It usually does this with the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP).

The session layer is a rarely-used layer. It acts more like an enhanced transport layer, by adding dialog control and token management.

- Dialog control is a way to control which computers are "talking" at any one time. This prevents too many computers from trying to transmit at the same time.
- Token management prevents two computers from trying to perform the same important action at the same time.
- Synchronization is a way to make sure that long transmissions are arriving at the same time.

The presentation layer is a rarely-used layer. It is concerned with the syntax and semantics of the information it transmits. Frequently, two computers with different internal representations (such as different operating systems) will try to communicate. If they do, they many interpret the same set of

numbers in a different way, causing confusion. The presentation layer lets the sender tell the receiver about the format of the message, so that the receiver can interpret it.

The application layer is a collection of miscellaneous protocols for high-level applications. These are typically used for email, file transfer, web browsing, connecting to remote terminals, etc.

Typically, the protocols on the application layer include HTTP, FTP, telnet, and others.

The TCP/IP reference model is a very common type of reference model. It consists of five layers: physical, network interface, internet, transport, and application layers.

- TCP/IP's physical later specifies details about the actual transmission medium itself, such as the hardware it is using, the electrical properties of the wires being used, the radio frequencies it is transmitting and receiving on, and other such information.
- TCP/IP's network interface layer (sometimes called the MAC layer) specifies details about communication over a single network; it specifies the maximum size of packets that can be sent and received, network addresses, and other related information.
- TCP/IP's internet layer specifies communication between two computers on the internet. It also specifies the internet addressing structure, the format of the internet packets, the method for dividing packets into smaller forms for transmission, and a mechanism for reporting errors.
- TCP/IP's transport layer provides communication from one application to another. It controls the maximum rate a receiver can accept data. It also helps avoid network congestion and ensures that data is received in the correct order.
- TCP/IP's application layer specifies the format and the meaning of the messages that the applications exchange.

We will now begin learning about clients and servers. The client-server connection is not the actual computer itself, it is the process running inside the computer that provides the information transfer. The client sends a request to the server. The server waits passively, and then responds to clients that contact it. The server then sends information to the client. In two-way communication, such as instant messaging, the computer is both a client (listening to the other computer's messages) and a server (sending messages to the other computer) at the same time.

Clients and servers can both run multiple applications at the same time. Clients can listen to multiple servers at once, and servers can send information to multiple clients at once. Additionally, clients can listen to different types of servers – for example, an internet server and an instant-messaging server. This is what enables us to multitask on computers.

The client-server interaction differs for clients and servers.

- Server
    - Must start first, so that the client has something to listen to.
    - Does not need to know which client will contact it; the client will provide its contact information when it opens a connection.
    - Waits passively and arbitrarily long for client contact, since it only exists to wait and then send information.

    - Communicates with the client both by sending and receiving data.
    - Stays running after servicing one client, and sits to wait for another client when it is done.
- Client
    - Must start second, or else it will not have a server to listen to.
    - Must know which server to contact, and must open a connection with it to maintain contact.
    - Initiates a communication when it needs to.
    - Communicates with a server by sending and receiving data.
    - May end after it is done with the server; does not necessarily remain active when it no longer needs to stay connected.

A client connects with a server via two numbers: the IP address and the port number.

- An IP address is an internet protocol address. Every computer has a unique identifier that a client must specify before it contacts the server. In order to make this easier, a name is also assigned, such as www.google.com
- Every internet service is also assigned a unique 16-bit number known as a port protocol number. For example, email uses port number 25, while the World Wide Web uses port number 80.

A client must know the server's IP and port number. Once it connects, it tells the client its own IP and port number, in order to form a connection. The client must therefore be the first to initiate contact, since the server does not initially know the client's address.

A server can handle more than one interaction at the same time. It does not have to wait for one client to finish its download before it starts another; for example, if one client is downloading a long movie and another wants to download a short file, the server can do both at once. It does this by creating more threads. A thread is an instance of a program used to service one client.

When a client connects to a computer, it connects via a socket. A socket is an abstraction that an application can use to send and receive data. A socket is an integer number that the client "sends" information to. It is not a real piece of hardware; rather, it is a way for the server to split up resources and provide ways for more than one computer to connect to it at once.

TCP is Transmission Control Protocol – it is a reliable, connection-oriented service.
UDP is User Datagram Protocol – it is an unreliable, connectionless service.
The TCP and UDP communication protocols are two different types of communication protocols; they handle data communication between terminals in an IP network (this network is the internet.)