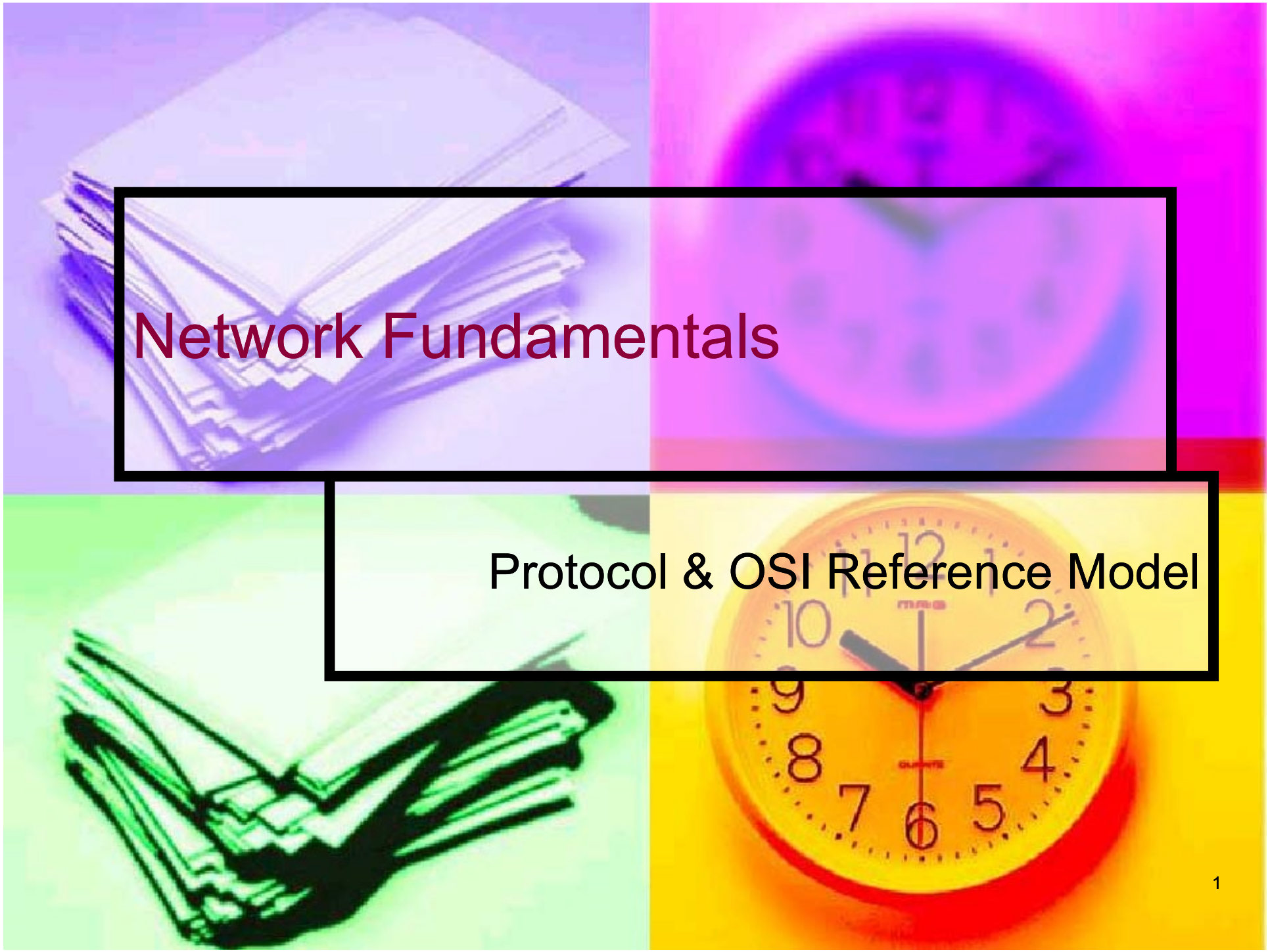


Network Fundamentals

Protocol & OSI Reference Model

NETWORK PROTOCOL

- A network protocol defines rules and conventions for communication between network devices. OR A Network protocol is a set of rules that govern data communication. Network protocols include mechanisms for devices to identify and make connections with each other, as well as formatting rules that specify how data is packaged into messages sent and received. Some protocols also support message acknowledgment and data compression designed for reliable and/or high-performance network communication.

Protocol Hierarchies

- To reduce the design complexity of the networks, most are organized as a series of ***layers*** or ***levels***, each one built upon the one below it.
- The number of layers, the name of each layer, the contents of each layer and the function of each layer differ from network to network.

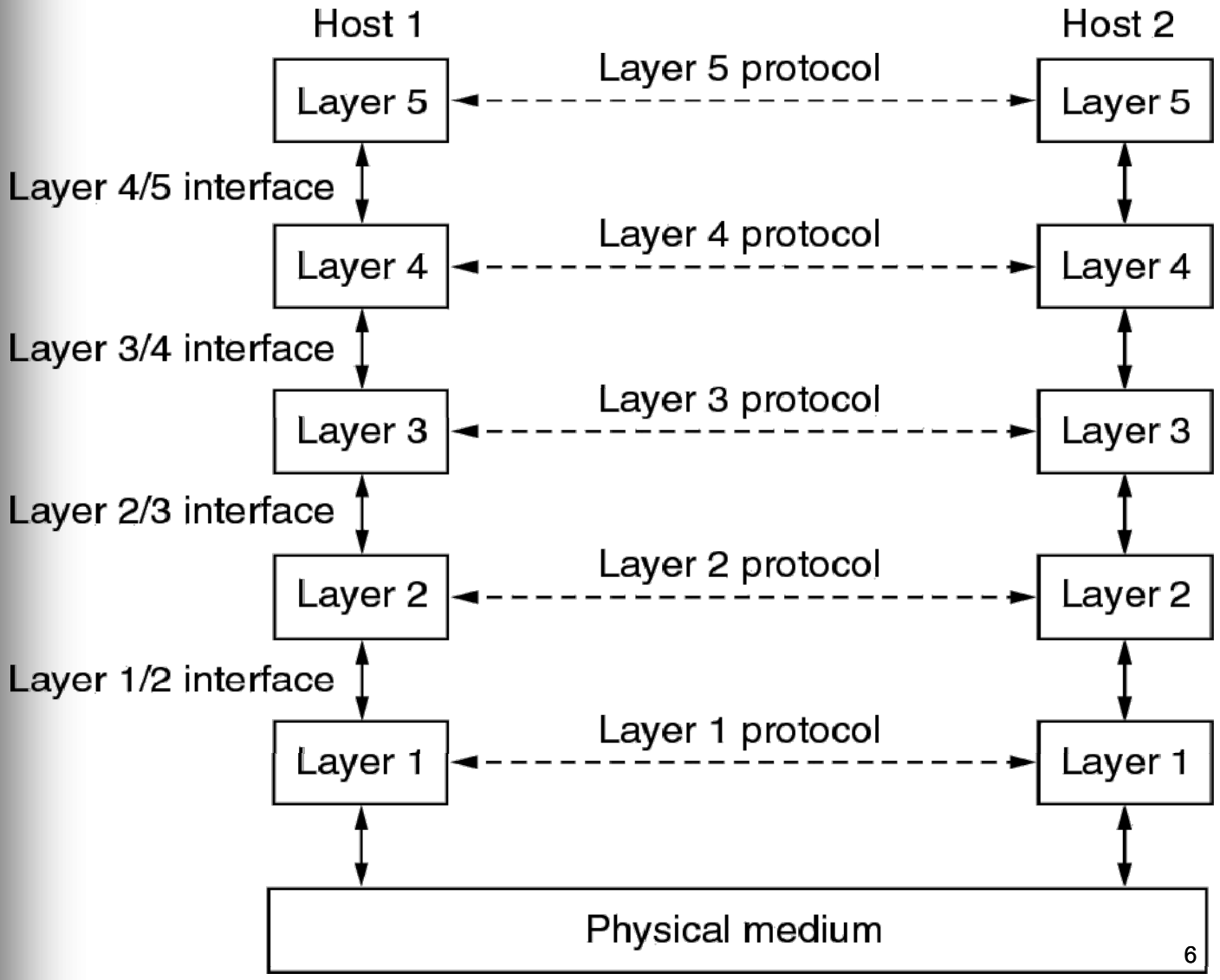
Protocol Hierarchies

- However, in all networks, the purpose of each layers is to offer certain services to the higher layers.
- Layer *n* on one machine carries on a conversation with layer *n* on another machine.
- The rules and conventions used in this conversation are collectively known as the layer *n* protocol.

Protocol Hierarchies

- A protocol is an agreement between the communication parties on how communication is to proceed.
- Violating the protocol will make communication more difficult, if not impossible.

Protocol Hierarchies



Protocol Hierarchies

- Here in previous figure five layers are there.
- The entities comprising the corresponding layers on different machines are called peers.
- In other words, it is the peers that communicate using the protocol.

Protocol Hierarchies

- No data are directly transferred from layer n on one machine to layer n on another machine.
- Each layer passes data and control information to the layer immediately below it, until the lower layer is reached.
- Below layer 1 is the physical medium through which actual communication occurs.

Protocol Hierarchies

- Virtual communication is shown by dotted lines and physical communication by solid lines in figure.
- Between each pair of adjacent layers there is an ***interface***.
- The interface defines which primitive operations and services the lower layer offer to the upper one.

Protocol Hierarchies

- When network designers decide how many layers to include in a network and what each one should do, one of the most important considerations is defining clean interfaces between the layers.

Protocol Hierarchies

- A set of layers and protocols is called a **network architecture**.
- The specification of an architecture must contain enough information to allow an implementer to write the program or build the hardware for each layer so that it will correctly obey the appropriate protocol.

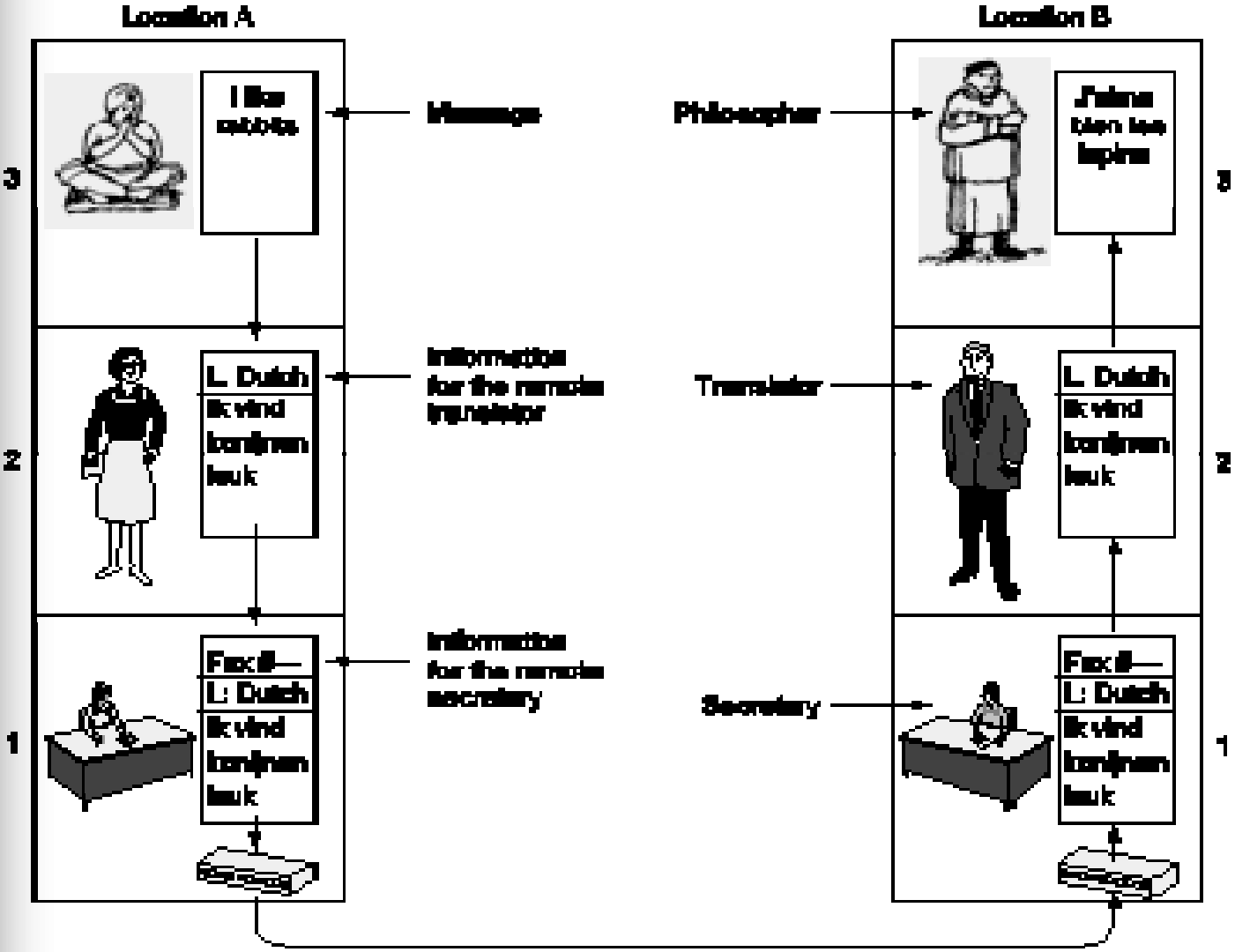
Protocol Hierarchies

- It is not even necessary that the interfaces on all machines in a network be the same, provided that each machine can correctly use all the protocols.
- A list of protocols used by a certain system, one protocol per layer, is called a **protocol stack**.

Protocol Hierarchies

- The following is the idea of multilayer communication.
- Imagine two philosophers, one of whom speaks Urdu and English and one of whom speaks Chinese and French.
- Since they have no common language, they each engage a translator.

Protocol Hierarchies



The philosopher-translator-secretary architecture.

Protocol Hierarchies

- Note that each protocol is completely independent of the other ones as long as the interfaces are not changed.
- Now consider one another problem that how to provide communication to the top layer of the five-layer network shown in the figure.

Protocol Hierarchies

- Message M is generated by layer 5 and passed to layer 4 for transmission.
- Layer 4 adds header H4 in front of message to identify the message and send to layer 3.
- The header includes control information, such as sequence number.

Protocol Hierarchies

- This information allows layer 4 on the destination machine to deliver messages in the right order if the lower layers do not maintain sequence.
- In some layers, headers also contain sizes, times and other control fields.

Protocol Hierarchies

- There is no limit to the size of messages transmitted in the layer 4 protocol, but there is nearly always a limit imposed by the layer 3 protocol.
- Layer 3 must break up the incoming messages into smaller units, packets, appending a layer 3 header to each packet.

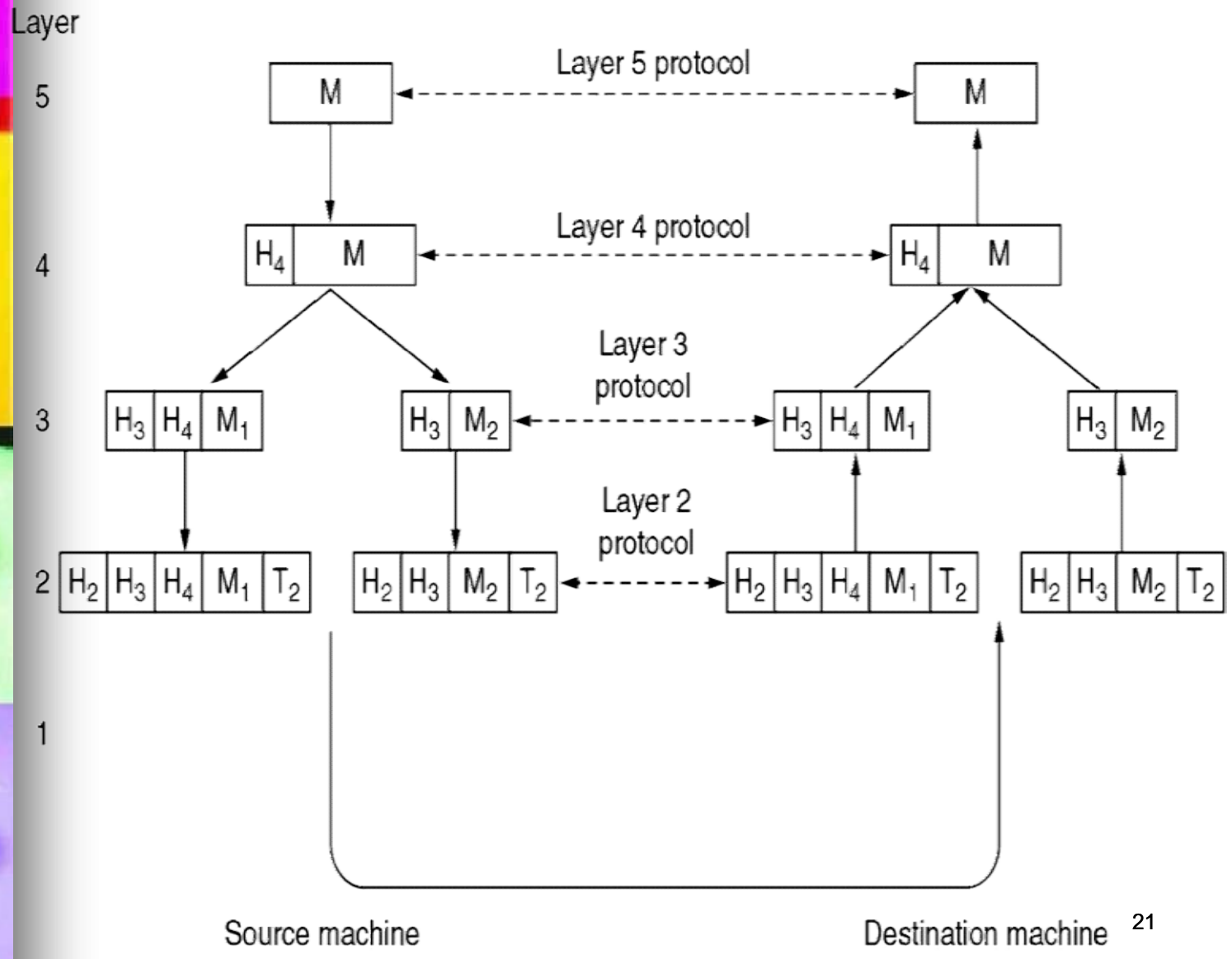
Protocol Hierarchies

- Layer 3 decides which of the outgoing lines to use and passes the packets to layer 2.
- Layer 2 adds not only a header to each piece, but also a trailer, and gives the resulting unit to layer 1 for physical transmission.

Protocol Hierarchies

- At the receiving machine the message moves upward, from layer to layer, with headers being stripped off as it progresses.
- None of the headers for layers below *n* are passed up to layer *n*.

Protocol Hierarchies



Protocol Hierarchies

- Here important thing is the relation between the virtual and actual communication and the difference between protocols and interfaces.

Design Issues for the Layers

- Every layer needs a mechanism for identifying senders and receivers.
- Since a network normally has many computers, some of which have multiple processes, a means is needed for a process on one machine to specify with whom it wants to talk.

Design Issues for the Layers

- Multiple destinations are also there so some form of addressing is needed in order to specify a specific destination.
- Another set of design concerns the rules for data transfer.
- In some system, data only travel in one direction (simplex communication).

Design Issues for the Layers

- In other they can travel in either direction, but not simultaneously (half-duplex communication).
- In still others they travel in both directions at once (full-duplex communication).

Design Issues for the Layers

- The protocol must also determine how many logical channels the connection corresponds to, and what their priorities are.
- Many networks provide at least two logical channels per connection, one for normal data and one for urgent data.

Design Issues for the Layers

- Error control is an important issue because physical communication circuits are not perfect.
- Many error-detecting and error-correcting codes are know, but both ends of the connection must agree on which one is being used.

Design Issues for the Layers

- Here, the receiver must have some way of telling the sender which messages have been correctly received and which have not.

Design Issues for the Layers

- Not all communication channels preserve the order of messages send on them.
- To deal with a possible loss of sequencing, the protocol must make explicit provision for the receiver to allow the pieces to be put back together properly.

Design Issues for the Layers

- An issue that occurs at every level is how to keep a fast sender from swamping a slow receiver with data.
- Various solutions have been proposed.

Design Issues for the Layers

- Some of them involve some kind of feedback from the receiver to the sender, either directly or indirectly, about the receiver's current situation.
- Others limit the sender to an agreed upon transmission rate.

Design Issues for the Layers

- Another problem must be solved at several levels is the inability of all processes to accept arbitrarily long messages.
- This property leads to mechanisms for disassembling, transmitting, and then reassembling messages.

Design Issues for the Layers

- A related issue is what to do when processes insist upon transmitting data in units that are so small that sending each one separately is inefficient.
- Solution of such kind of problem is to combine several small messages heading toward a common destination into a single large message.

Design Issues for the Layers

- When it is inconvenient or expensive to set up a separate connection for each pair of communicating processes, the underlying layer may decide to use the same connection for multiple, unrelated conversations.

Design Issues for the Layers

- When there are multiple paths between source and destination, a route must be chosen.
- Some times this decision must be split over two or more layers.

Interface and Services

- The function of each layer is to provide services to the layer above it.
- The active elements in each layer are often called ***entities***.
- An entity can be software entity (such as a process), or a hardware entity (such as intelligent I/O chip).

Interface and Services

- Entities in the same layer on different machines are called ***peer entities***.
- The entities in layer *n* implement a service used by layer *n* + 1.
- In this case layer *n* is called the ***service provider*** and layer *n* + 1 is called the ***service user***.

Interface and Services

- Layer *n* may use the services of layer *n* – 1 in order to provide its service.
- It may offer several classes of service, for example, fast, expensive communication and slow, cheap communication.

Interface and Services

- Services are available at SAPs (Service Access Points), The layer *n* SAPs are the places where layer *n* + 1 can access the services offered.
- Each SAP has an address that uniquely identifies it.

Interface and Services

- In order for two layers to exchange information, there has to be an agreed upon set of rules about the interface.
- At a typical interface, the layer *n* + 1 entity passes an IDU (Interface Data Unit) to the layer *n* entity through the SAP.

Interface and Services

- The IDU consists of an SDU (Service Data Unit) and some control information.
- The SDU is the information passed across the network to the peer entity and then up to layer *n* + 1.
- The control information is needed to help the lower layer to do its job.

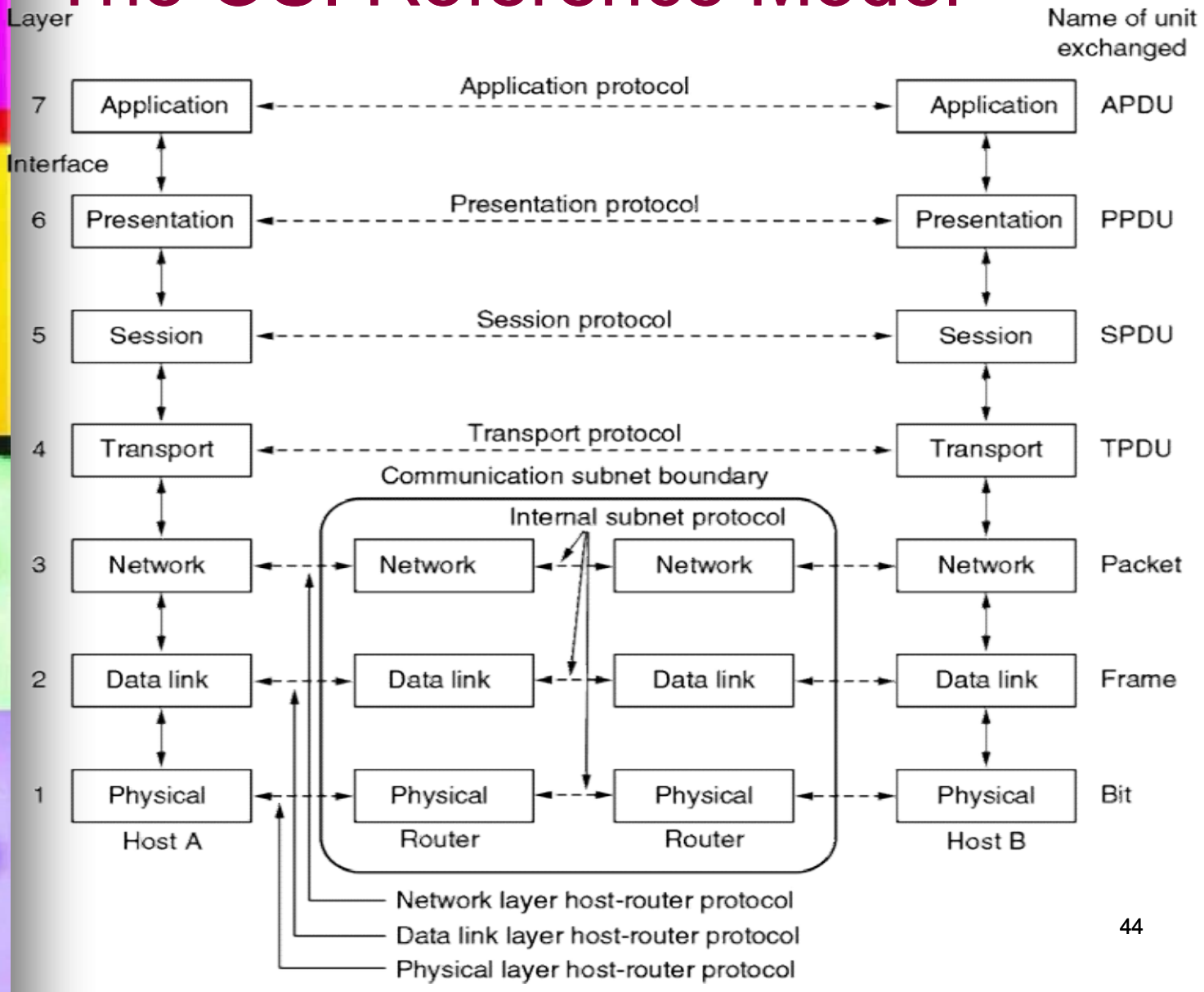
Interface and Services

- In order to transfer the SDU, the layer n entity may have to fragment it into several pieces, each of which is given a header and sent as a separate PDU (Protocol Data Unit) such as a packet.
- The PDU headers are used by the peer entities.

Interface and Services

- They identify which PDUs contain data and which contain control information, provide sequence numbers and counts, and so on.

The OSI Reference Model



The OSI Reference Model

- This model is based on proposal developed by the International Standards Organization (ISO) as a first step toward international standardization of the protocols used in the various layers.
- The model is called ISO OSI (Open System Interconnection) Reference Model.

The OSI Reference Model

- This name is given because it deals with connection open system – that is, systems that open for communication with other systems.
- The OSI model has seven layers.
- The principles that were applied to arrive at the seven layers are discussed in next slide.

The OSI Reference Model

- A layer should be created where a different level of abstraction is needed.
- Each layer should perform a well defined function.
- The function of each layer should be chosen with an eye toward defining internationally standardized protocols.

The OSI Reference Model

- The layer boundaries should be chosen to minimize the information flow across the interfaces.
- The number of layers should be large enough that distinct functions need not be thrown together in the same layer out of necessity.

The OSI Reference Model

- Note that the OSI model itself is not a network architecture because it does not specify the exact service and protocols to be used in each layer.
- It just tells what each layer should do.

The OSI Reference Model

- However, ISO has also produced standards for all the layers, although these are not part of the reference model itself.
- Each one has been published as a separate international standard.

The OSI Reference Model

1. Physical Layer
2. Data link Layer
3. Network Layer
4. Transport Layer
5. Session Layer
6. Presentation Layer
7. Application Layer

Physical Layer

- The physical layer is concerned with transmitting raw bits over a communication channel.
- The design issues have to do with making sure that when one side sends a 1 bit, it is received by other side as a 1 bit, not as a 0 bit.

Physical Layer

- Typical questions can be asked here are
- How many volts should be used to represent a 1 and how many for a 0?
- How the initial connection is established and how it is turn down when both sides are finished?
- How many pins the network connector has and what each pin is used for?

Physical Layer

- The design issues here largely deal with mechanical, electrical and procedural interfaces, and the physical transmission medium, which lies below the physical layer.

The Data Link Layer

- The main task of the data link layer is to take a raw transmission facility and transform it into a line that appears free of undeleted transmission errors to the network layer.

The Data Link Layer

- It accomplishes this task by having the sender break the input data up into data frames (typically a few hundred or a few thousand bytes), transmit the frames sequentially, and process the acknowledgement frames sent back by the receiver.

The Data Link Layer

- Physical layer merely accepts and transmits a stream of bits without any regard to meaning or structure, it is up to the data link layer to create and recognize frame boundaries.
- This can be accomplished by attaching special bit patterns to the beginning and end of the frame.

The Data Link Layer

- A noise burst on the line can destroy a frame completely.
- In this case, the data link layer software on the source machine can retransmit the frame.
- However, multiple transmission of the same frame introduce the possibility of duplicate frames.

The Data Link Layer

- It is up to this layer to solve the problems caused by damaged, lost, and duplicate frames.
- Another issue that arises in this layer is how to keep a faster transmitter from drowning a slow receiver in data.

The Data Link Layer

- Some traffic regulation mechanism must be employed to let the transmitter know how much buffer space the receiver has at the moment.
- Broadcast networks have an additional issue in the data link layer: how to control access to the shared channel.

The Data Link Layer

- A special sublayer of the data link layer, the medium access sublayer, deals with this problem.

The Network Layer

- The network layer is concerned with controlling the operation of the subnet (logical division of a network).
- A key deign issue is determining how packets are routed from source to destination.
- Routes can be based on static tables that are “wired into” the network and rarely changed.

The Network Layer

- They can also be determined at the start of each conversation, for example a terminal session.
- If too many packets are present in the subnet at the same time, they will get in each other's way, forming bottlenecks.
- The control of such congestion also belongs to the network layer.

The Network Layer

- There is also some accounting function built into network layer.
- At the very least, the software must count how many packets or characters or bits are sent by each customer, to produce billing information.

The Network Layer

- When a packet crosses a national border, with different rates on each side, the accounting can become complicated.
- When a packet has to travel from one network to another to get to its destination, many problems can arise.

The Network Layer

- The addressing used by the second network may be different from the first one.
- The second one may not accept the packet at all because it is too large.
- The protocols may differ, and so on.

The Network Layer

- It is up to network layer to overcome all these problems to allow heterogeneous networks to be interconnected.
- In broadcast networks, the routing problem is simple, so the network layer is often thin or even nonexistent.

The Transport Layer

- The basic function of the transport layer is to accept data from the session layer, split it up into smaller units if need be, pass these to the network layer, and ensure that the pieces all arrive correctly at the other end.

The Transport Layer

- Under normal conditions, the transport layer creates a distinct network connection for each transport connection required by the session layer.
- If the transport connection requires a high throughput, the transport layer might create multiple network connections.

The Transport Layer

- On the other hand, if creating or maintaining a network connection is expensive, the transport layer might multiplex several transport connections onto the same network connection to reduce the cost.

The Transport Layer

- The transport layer also determines what type of service to provide the session layer, and the users of the network.
- The most popular type of transport connection is an error-free point-to-point channel that delivers messages or bytes in the order in which they were sent.

The Transport Layer

- Other possible kinds of transport service are transport of isolated messages with no guarantee about the order of delivery, and broadcasting of messages to multiple destinations.
- The type of service is determined when the connection is established.

The Transport Layer

- The transport layer is a true end-to-end layer, from source to destination.
- In other words, a program on the source machine carries on a conversation with a similar program on the destination machine, using the message headers and control messages.

The Transport Layer

- In the lower layers, the protocols are between each machine and its immediate neighbors, and not by the ultimate source and destination machines, which may be separated by many routers.

The Transport Layer

- The difference between layers 1 through 3, which are chained and layers 4 through 7, which are end-to-end.
- Many hosts are multiprogrammed, which implies that multiple connections will be entering and leaving each host.

The Transport Layer

- There needs to be some way to tell which message belongs to which connection.
- The transport header is one place this information can be put.

The Transport Layer

- In addition to multiplexing several message streams onto one channel, the transport layer must take care of establishing and deleting connections across the network.

The Transport Layer

- This requires some kind of naming mechanism, so that a process on one machine has a way of describing with whom it wishes to talk.
- There must also be a mechanism to regulate the flow of information, so that a fast host cannot overrun a slow one.

The Transport Layer

- Such a mechanism is called ***flow control*** and plays a key role in the transport layer.
- Flow control between hosts is distinct from flow control between router.

The Session Layer

- The session layer allows users on different machines to establish sessions between them.
- A session allows ordinary data transport, as does the transport layer, but it also provides enhanced services useful in some applications.

The Session Layer

- A session might be used to allow a user to log into a remote timesharing system or to transfer a file between two machines.
- One of the services of the session layer is to manage dialogue control.

The Session Layer

- Session can allow traffic to go in both directions at the same time, or in only one direction at a time.
- If traffic can only go one way at a time , the session layer can help keep track of whose turn it is.

The Session Layer

- The related service is ***token management***.
- For some protocols, it is essential that both sides do not attempt the same operation at the same time.
- To manage these activities, the session layer provides tokens that can be exchanged.

The Session Layer

- Only the side holding the token may perform the critical operation.
- Another session service is ***synchronization***.
- Consider the problems that might occur when trying to do a 2-hour file transfer between two machines with a 1-hour mean time between crashes.

The Session Layer

- After each transfer was aborted, the whole transfer would have to start over again and would probably fail again the next time as well.

The Session Layer

- To eliminate this problem, the session layer provides a way to insert checkpoints into the data stream, so that after a crash, only the data transferred after the last checkpoint have to be repeated.

The Presentation Layer

- The presentation layer performs certain functions that are requested sufficiently often to warrant finding a general solution for them, than letting each user solve the problems.
- The presentation layer is concerned with the syntax and semantics of the information transmitted.

The Presentation Layer

- A typical example of a presentation service is encoding data in a standard agreed upon way.
- Most user programs do not exchange random binary bit strings.
- They exchange things such as people's names, dates, amounts of money, invoices.

The Presentation Layer

- These items are represented as character strings, integers, floating point numbers and data structures composed of several simpler items.
- Different computers have different codes for representing character strings and so on.

The Presentation Layer

- In order to make it possible for computers with different representations to communicate, the data structures to be exchanged can be defined in an abstract way, along with a standard encoding to be used “on the wire”.

The Presentation Layer

- The presentation layer manages these abstract data structures and converts from the representation used inside the computer to the network standard representation and back.

The Application Layer

- The application layer contains a variety of protocols that are commonly needed.
- For example, there are hundreds of incompatible terminal types in the word.
- Each terminals are having different screen layouts, escape sequences for inserting and deleting text, moving the cursor etc.

The Application Layer

- One way to solve this problem is to define an abstract network virtual terminal that editors and other programs can be written to deal with.
- To handle each terminal type, a piece of software must be written to map the functions of the network virtual terminal onto the real terminal.

The Application Layer

- For example, when the editor moves the virtual terminal's cursor to the left-hand corner of the screen, this software must issued the proper command sequence to the real terminal to get its cursor there too.
- All the virtual terminal software is in the application layer.

The Application Layer

- Another application layer function is file transfer.
- Different file systems have different file naming conventions, different ways of representing text lines, and so on.
- Transferring a file between two different systems requires handling these and other incompatibilities.

The Application Layer

- This is also the work of the application layer, as do electronic mail, remote job entry, directory lookup and various other general purpose- and special-purpose facilities.

Layer	Responsible For:
7.) Application	Provides Services to User Apps
6.) Presentation	Data Representation
5.) Session	Communication Between Hosts
4.) Transport	Flow Ctrl, Error Detection/Correction
3.) Network	End to End Delivery, Logical Addr
2.) Data Link	Media Access Ctrl, Physical Addr
1.) Physical	Medium, Interfaces, Puts Bits on Med.

Layer	Example
7.) Application	HTTP, FTP, SMTP
6.) Presentation	ASCII, JPEG, PGP
5.) Session	BOOTP, DNS
4.) Transport	TCP, UDP, SPX
3.) Network	IP, IPX, ICMP
2.) Data Link	Ethernet, Token Ring
1.) Physical	Bits, Interfaces, Hubs