CASE STUDY NO :2

Date	/ /20
Page No.	12

Case Study On :- Study of layered protocol in Distributed system.

Layered Protocols :- Due to the absence of shared memory, all communication in distributed systems is based on message passing. When proces a wants to communicate with process B, it first builds a message in its own address space. Then it executs a system call that causes the operating system to fetch the message and send it over the network to B. Although the basic idea sound simple enough, in order to prevent chaos, A and B have to agree on the meaning of the bits being sent. If a sends a brilliant new novel written in French and encoded in IBM's EBCDIC character code, and B expects the inventory of a supermarket written in English and encoded in ASCII, communication will be less than optimal.
In the OSI model, there are not two layers, but seven, as we saw. The collection of protocols used in a particular system is called a protocol suite or protocol stack. In the session, we will briefly examine each of the layers in turn, starting at the bottom. Where appropriate, we will also ~~priad~~ point out some of the protocols used in each layer.

* The Physical Layer -

The physical layer is concerned with transmitting the 0s and 1s. How many volts to use for 0 and 1, how many bits per second can be sent, and whether transmission can take place in both directions simultaneously are key issues in the physical layer. In addition The size and shape of the network connector (plug), as well as the number of pins and meaning of each are of concern here.

The physical layer protocol deals with standardizing the electrical, mechanical, and signaling interfaces so that when one machine send a 0 bit it is actually received as a 0 bit and not a 1 bit. Many physical layer standards have been developed (for different media), for example, the RS-232-C standard for serial communication lines.

* The Data Link Layer -

The physical layer just send bits. As long as no errors occur all is well. However, real communication networks are subject to errors, so some mechanism is needed to detect and correct them. This mechanism is the main task of the data link layer. What it does is to group the bits into units, sometimes called frames and see that each frame is correctly received.

The data link layer its work by putting a special bit pattern on the start of each frame, to mark them, as well as computing a check sum by adding up all the bytes in the frame in a certain way. The data link layer append the checksum to the frame. When the frame arrives, the receiver recomputes the checksum from the data and compares the result to the checksum following the frame. If they agree, the frame is considered correct and is accepted. If they disagree, the receiver asks the sender to retransmit it. Frames are assigned sequence numbers (in the header), so everyone can tell which is which.

* The Network Layer -

On a LAN, there is usually no need for the sender to locate the receiver. It just puts the message out the network and the receiver takes it off. A wide area network, however, consists of a large number of machines, each with some number of lines to other machines, rather like a large-scale map showing major cities and roads connecting them. For a message to get from the sender to the receiver it may have to make a number of hops, at each one choosing an outgoing line to use. The question of how to choose the best path is called routing, and is the primary task of the network layer.

The problem is complicated by the fact

that the shortest route is not always the best route. What really matters is the amount of dealy on given route, which, in turn is related to the amount of traffic and the number of messages queued up for transmission over the various lines. The dealy can thus change over the course of time. Some routing algorithms try to adapt to changing loads, where as others are content to make decisions based on long-term averages.

* The Transport Layer -

Packets can be lost on the way from the receiver. ~~Ath~~ Although some application can handle their own error recovery, others prefer a reliable connection. The job of the transport layer is to provide this service. The idea is that the session layer should be able to deliver a message to the transport layer with the exception that it will be delivered without loss.

Upon receiving a message from the session layer, the transport layer breaks it into pieces small enough for each to fit in a single packet, assigns each one a sequence number, and then sends them all.

* The Session Layer -

The session layer is essentially an enhanced version of the transport layer. It provides dialog control, to keep track of which party is currently talking, and it provides synchronization facilities. The latter are usefull to allow users to insert checkpoints into long transfers, so that in the event of a crash it is only necessary to go back to the last checkpoint, rather than all the way back to the beginning. In practice, few applications are interested in the session layer and it is rarely supported. It is not even present in the DoD protocol suite.

* The Presentation Layer -

Unlike the lower layers, which are concerned with getting the bits from the sender to the receiver reliably and efficiently, the presentation layer is concerned with the meaning of the bits. Most message do not consist of random bit strings, but more structured information such as people's names, addresses, amounts of money, and so on.

* The Application Layer -

The application layer is really just a collection of miscellaneous protocols for common activities such as electronic mail, file transfer, and connecting remote terminals to computer over a networks. The best known of these are the X.400 electronic mail protocal and the X.500 directory server. Neither this layer nor the two layers directly under it will be interest to us in this book.