

Computer Networks - CS 204

Rishit Saiya - 180010027, Assignment - 1

April 15, 2020

1

(a) **Virtual Circuit**

If connection-oriented service is offered, a path from the source router all the way to the destination router must be established before any data packets can be sent. This connection is defined as Virtual Circuit. The manifestation of which establishes Virtual Circuit Network.

(b) **Datagram Switching**

If connectionless-oriented service is offered, packets are injected into the network individually and routed independently of each other. No advance setup is needed. In this context, packets are frequently called as datagrams. This connection is defined as Datagram Network.

(c) **VC vs Datagram Network**

(d) **VC vs Physical Connection**

A virtual circuit naturally appears to be a dedicated physical circuit for a user.

However, other communications may also be sharing the parts of the same path.

Before the data transfer begins, the source and destination identify a suitable path for the virtual circuit. All intermediate nodes between the two points put an entry of the routing in their routing table for the call. Additional parameters, such as the maximum packet size, are also exchanged between the source and the destination during call setup. The virtual circuit is cleared after the data transfer is completed.

2

(a) **Logical Address**

An address which provides us the access to a network device by using an address that

Issue	Virtual Circuit Network	Datagram Network
Circuit setup	Required	Not needed
Addressing	Each packet contains a short VC number	Each packet contains the full source and destination address
State information	Each VC requires router table space per connection	Routers do not hold state information about connection
Routing	Route chosen when VC is setup; all packets follow it	Each packet is routed independently
Effect of router failures	All VCs that passed through the failed router are terminated	None, except for packets lost during the crash
Quality of service	Easy if enough resources can be allocated in advance for each VC	Difficult
Congestion control	Easy if enough resources can be allocated in advance for each VC	Difficult

Table 1: Answer to 1.c

we have assigned. These addresses are created and used in protocols like IP (Internet Protocol) and IPX (Internetwork Packet Exchange).

(b) **Physical Address**

A physical address is the hardware-level address used by the Ethernet interface to communicate on the network. Every device must have a unique physical address. It is often referred to as its MAC (Media Access Control) address.

(c) **Mapping**

Address binding is the process of mapping from one address space to another address space.

Logical address is address generated by CPU during execution whereas Physical Address refers to location in memory unit (the one that is loaded into memory).

It is to be noted that user deals with only logical address (Virtual address). The logical address undergoes translation by the MMU (Memory Management Unit) or ATU (Address Translation Unit) in particular. The output of this process is the appropriate physical address or the location of code/data in RAM.

Since the user only deals with the logical address, mapping is necessary so as to fetch the location of data in RAM (Random Access Memory).

(d) **Protocols for mapping**

The ARP protocol is a network-specific standard protocol. Its status is elective. The

address resolution protocol is responsible for converting the higher-level protocol addresses (IP addresses) to physical network addresses.

3

(a) **UDP over TCP**

In situations where we really want to get a simple answer to another server quickly, UDP works best. In general, we want the answer to be in one response packet, and we are prepared to implement our own protocol for reliability or to resend. DNS is the perfect description of this use case. The costs of connection setups are way too high (yet, DNS does support a TCP mode as well).

Another case is when we are delivering data that can be lost because newer data coming in will replace that previous data/state. Weather data, video streaming, a stock quotation service (not used for actual trading), or gaming data comes to mind.

Another case is when we are managing a tremendous amount of state and we want to avoid using TCP because the OS cannot handle that many sessions. This is a rare case today. In fact, there are now user-land TCP stacks that can be used so that the application writer may have finer grained control over the resources needed for that TCP state. Prior to 2003, UDP was really the only game in town.

One other case is for multicast traffic. UDP can be multicasted to multiple hosts whereas TCP cannot do this at all.

4

(a) **Time delay**

We just equate the time delays in both the cases.

$$\begin{aligned}\frac{10}{330} &= \frac{x}{2.3 \times 10^8} \\ x &= 6.9696 \times 10^6 m \\ x &\approx 7.0 \times 10^6 m\end{aligned}$$

5

(a) **Analogy to Packet Switching**

Let's assume the following analogy:

$$\textit{Container} \longrightarrow \textit{Constant - Size Packet}$$

We prefer the the length of the packet to be short.

Transmission systems of various types can be designed to transfer information of the given standardized size, much like trucks, trains, and ships can be designed to carry standard containers.

Packing and unpacking of fixed-size units is simpler than for variable-length units. Consequently, it is simpler to schedule the transfer of packets across switches that use constant-size packets than across switches that make use of variable-length packets.

6

(a) **Video Game Network Establishment**

We suppose that the game involves the interaction between a player and a server across a network. To support an interactive video game over a communications network, the network, whether connection-oriented or connectionless, must provide real-time delivery of the player's commands to the server, and of the server's responses to the player.

(b) **Requirements for Connection-Oriented Networks**

With a connection-oriented network, connections between the player and the servers transfer the sequence of commands and responses throughout the game with very little delay.

(c) **Requirements for Connectionless Networks**

In a connectionless network, user commands may be delivered to the other end with variable delay, out-of-sequence, or not at all. The user's network software is responsible for ensuring the ordered and correct delivery of game commands. In-time delivery of commands cannot be assured.

7

(a) **Importance of Port No. in Application Layer**

The TCP layer entity uses the port number to determine which application program the packets belong to. In the TCP connection setup process, it is very convenient to have a unique well-known port number, otherwise some protocol or procedure would be required to find the desired number. Also, if the port number matches then there will be a clash of client requests/operations pertaining to the same port.

8

(a) **Circuit switching over Packet Switching**

Packet switching network is not suitable for real time services. In circuit switched network it can give end to end bandwidth during a call. In the packet switched network

it cannot guarantee any end to end bandwidth. In circuit switched network Quality of Service (QoS) is guaranteed while in packet switched network it is not guaranteed and packet switched network might have delay and time insensitive.

(b) **TDM over FDM**

TDM has an advantage over FDM as it gives bandwidth saving and there is low interference between multiplexed signals.

9

(a) **Methodical transfer of big file**

Consider given data:

Suppose end system A wants to send a large file to end system B. The following steps to end System A creates packets from the file at very high level:

- Divide file into chunks.
- Create a packet by attach header to chunk.
- Each packet maintain an address of the destination.

(b) **Link to forwarding packet**

The following information in the packet does the switch use to determine the link onto which the packet is forwarded:

- Switch uses the destination address.
- It is easy to find which packet is forward to the header.

(c) **Packet Switching in Internet**

The following the packet switching in the Internet analogous to driving from one city to another and asking directions along the way:

- Each packet maintain an address of the destination.
- Reaching packet, packet display outgoing link which road to take to forwarded.

10

(a) **Packet Switching vs Circuit Switched Network**

A circuit-switched network would be well suited to the application, because the application involves long sessions with predictable smooth bandwidth requirements. Since the transmission rate is known, bandwidth can be reserved for each application session without significant waste. In addition, the overhead costs of setting up and tearing down connections are amortized over the lengthy duration of a typical application session.

(b) **Congestion Control requirement**

In the worst case, all the applications simultaneously transmit over one or more network links. However, since each link has sufficient bandwidth to handle the sum of all of the applications' data rates, no congestion (very little queuing) will occur. Given such generous link capacities, the network does not need congestion control mechanisms.

11

(a) **Importance of Port No.**

Here we will use Little's formula:

$$N = a \times d$$

Let N denote the average number of packets,

$a = \text{the rate of packets arriving at the link,}$

$d = \text{the average total delay } (d_{\text{queue}} + d_{\text{trans}})$

We have been given the following information

- The buffer contains 10 packets.
- The average packet queuing delay is 10 msec.
- The link's transmission rate is 100 packets/sec.

So,

$$d_{\text{queue}} = 10 \text{ msec},$$

$$R = 100 \frac{\text{packets}}{\text{sec}},$$

If the 1st packet is not considered,

$$N = 10$$

We want to find the average packet arrival rate assuming there is no packet loss. We are trying to solve this problem for a.

N and d_{queue} are given.

So we need to find d_{trans} and add that with d_{queue} to get d .

Hence,

$$N = a \times d$$

$$N = a \times (d_{queue} + d_{trans})$$

We know that

$$d_{trans} = L/R$$

, where L is for 1 packet.

$$d_{trans} = \frac{1 \text{ packet}}{100 \frac{\text{packets}}{\text{sec}}}$$

So,

$$d_{trans} = 10 \text{ msec}$$

Finally, calculating total d,

$$d = 10 \text{ msec} + 10 \text{ msec}$$

$$d = 20 \text{ msec}$$

Using Little's formula, now we solve for a, the average packet arrival rate:

$$\frac{10 \text{ packets}}{0.02 \text{ sec}} = 500 \frac{\text{packets}}{\text{sec}}$$

So average packet arrival rate is,

$$a = 500 \frac{\text{packets}}{\text{sec}}$$

If the 1st packet is considered,

$$N = 11$$

Therefore,

$$\frac{11 \text{ packets}}{0.02 \text{ sec}} = 550 \frac{\text{packets}}{\text{sec}}$$

So average packet arrival rate is,

$$a = 550 \frac{\text{packets}}{\text{sec}}$$

12

(a) **Difference between both approach**

The TCP/IP stack is responsible for the "chopping up" into packets of the data for transmission and for their acknowledgment. Depending on the transport protocol that is used (TCP or UDP) each packet will be acknowledged or not, respectively.

The strategy when the file is chopped up into packets, which are individually acknowledged by the receiver, but the file transfer as a whole is not acknowledged is OK in situations (Applications) that do not need the whole file to be sent, website for example: different parts of the website can arrive in different times.

The other strategy, in which the packets are not acknowledged individually, but the entire file is acknowledged when it arrives is suitable for FTP (mail transfer), we need whole mail, not parts of it.

13

(a) **Uniqueness of port in TCP connection**

No. A pair of ports are uniquely set up for ONE connection.

14

(a) **Delay between both packets**

If the bottleneck link is the first link, then packet B is queued at the first link waiting for the transmission of packet A. So the packet inter-arrival time at the destination is simply

$$t = \frac{L}{R_s}$$

(b) **Delay between both packets with given condition**

If we send the second packet T seconds later, we will ensure that there is no queuing delay for the second packet at the second link if we have:

$$\frac{L}{R_s} + d_{prop} + \frac{L}{R_s} + T \geq \frac{L}{R_s} + d_{prop} + \frac{L}{R_c}$$

Thus,

$$T \geq \frac{L}{R_c} - \frac{L}{R_s}$$
$$T_{min} = \frac{L}{R_c} - \frac{L}{R_s}$$