

2-17:

【2-17】试比较 ADSL, HFC 以及 FTTx 接入技术的优缺点。

解答:使用 ADSL 最大的好处就是可以利用现有电话网中的用户线,不需要重新布线。用户可以根据自己的情况使用不同速率的宽带接入(按带宽付费)。这种接入的缺点是对用户线的质量有较高的要求。如果用户住宅距离电话交换局较远,或线路的噪声较大,那么宽带接入的速率就会适当地降低。

HFC 的优点是覆盖面很广,并且其带宽也很高,可以传送很高速率的数据;缺点是必须对现有单向传输的有线电视进行改造,使其变为可双向通信的电缆。用户家中需要增加一个机顶盒,用来观看电视和传送上行信号(在点播节目时使用)。此外,为了解决信号传输时有衰减的问题,在有线电视中每隔一定距离就要加入一个放大器。大量放大器的接入将使整个网络的可靠性下降。在我国利用 HFC 接入到互联网并未得到普遍使用。

光纤接入 FTTx 是解决宽带接入最理想的方案,因为光纤可传送的数据率很高,且通信质量最好。随着光纤接入的价格越来越便宜,现在我国宽带接入的主流已经是光纤接入了。尤其是新建造的高层建筑或居民小区,采用光纤接入已是用户实现高速宽带上网的首选。现在速率为 100 Mbit/s 的光纤接入已相当普遍,而 ADSL 在我国宽带接入中所占的比例已经非常小了。

3-03:

【3-03】网络适配器的作用是什么?网络适配器工作在每一层?

解答:适配器又称为网络接口卡或简称为“网卡”。在适配器上面装有处理器和存储器(包括 RAM 和 ROM)。适配器和局域网之间的通信是通过电缆或双绞线以串行传输方式进行的,而适配器和计算机之间的通信则是通过计算机主板上的 I/O 总线以并行传输方式进行的。因此,适配器的一个重要功能就是要进行数据串行传输和并行传输的转换。由于网络上的数据率和计算机总线上的数据率并不相同,因此在适配器中必须装有对数据进行缓存的存储器。若在主板上插入适配器时,还必须把管理该适配器的设备驱动程序安装在计算机的操作系统中。这个驱动程序以后就会告诉适配器,应当从存储器的什么位置把多长的数据块发送到局域网,或者应当在存储器的什么位置把局域网传送过来的数据块存储下来。适配器还要能够实现以太网协议。

适配器接收和发送各种帧时不使用计算机的 CPU。这时 CPU 可以处理其他任务。当适配器收到有差错的帧时,就把这个帧丢弃而不必通知计算机。当适配器收到正确的帧时,它就使用中断来通知该计算机并交付协议栈中的网络层。当计算机要发送 IP 数据报时,就由协议栈把 IP 数据报向下交给适配器,组装成帧后发送到局域网。

3-04:

【3-04】数据链路层的三个基本问题(封装成帧、透明传输和差错检测)为什么都必须加以解决?

解答:封装成帧就是在一段数据的前后分别添加首部和尾部(在首部和尾部里面有许多必要的控制信息),这样就构成了一个帧。接收端在收到物理层上交的比特流后,就能根据首部和尾部的标记,从收到的比特流中识别帧的开始和结束。

所谓“透明传输”就是上层交下来的数据,不管是什么形式的比特组合,都必须能够正确传送。由于帧的开始和结束标记使用专门指明的控制字符,因此,所传输的数据中的任何比特

组合一定不允许和用作帧定界的控制字符的比特编码一样，否则就会出现帧定界的错误。数据链路层不应当对要传送的数据提出限制，即不应当规定某种形式的比特组合不能够传送。

如果数据链路层没有差错检测，那么当目的主机收到其他主机发送来的数据时，在交给高层后，如果应用程序要求收到的数据必须正确无误，那么目的主机的高层软件可以对收到的数据进行差错检测。如果发现数据中有差错，就可以请求源主机重传这些数据。这样做就可以达到正确接收数据的目的。但这种工作方式有一个很大的缺点，就是一些在传输过程中出现了错误的帧（请注意，这些已经是没有用处的数据）还会继续在网络中传送，这样就浪费了网络的资源。例如，源主机到目的主机的路径中共有 20 个节点。在传送数据时，第一个节点就检测出了差错。如果数据链路层有差错检测的功能，就可以把这个有差错的帧丢弃，以后就不再传送了。否则这个没有用处的帧还要在网络上继续传送，还要陆续通过后面的 19 个节点，这就造成了网络资源的浪费。

3-07:

【3-07】 要发送的数据为 1101011011。采用 CRC 的生成多项式是 $P(X) = X^4 + X + 1$ 。试求应添加在数据后面的余数。

若要发送的数据在传输过程中最后一个 1 变成了 0，即变成了 1101011010，问接收端能否发现？

若要发送的数据在传输过程中最后两个 1 都变成了 0，即变成了 1101011000，问接收端能否发现？

采用 CRC 检验后，数据链路层的传输是否就变成了可靠的传输？

解答：采用 CRC 的生成多项式是 $P(X) = X^4 + X + 1$ ，用二进制表示就是 $P = 10011$ 。现在除数是 5 位，因此在数据后面添加 4 个 0 就得出被除数（如图 T-3-07(a)所示）。

$$\begin{array}{r}
 \text{除数 } P \rightarrow 10011 \overline{) 11010110110000} \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 000010110 \\
 \underline{10011} \\
 010100 \\
 \underline{10011} \\
 01110 \\
 \underline{10011} \\
 1110
 \end{array}$$

$1100001010 \leftarrow Q \text{ 商}$
 $11010110110000 \leftarrow 2^M \text{ 被除数}$
 $1110 \leftarrow R \text{ 余数}$

图 T-3-07(a) 计算余数

除法运算得出的余数 R 就是应当添加在数据后面的检验序列：1110。

现在要发送的数据在传输过程中最后一个 1 变成了 0，即 1101011010。把检验序列 1110 接在数据 1101011010 的后面，下一步就是进行 CRC 检验（如图 T-3-07(b)所示）。

$$\begin{array}{r}
 \text{除数 } P \rightarrow 10011 \overline{) 11010110101110} \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 000010101 \\
 \underline{00011} \\
 011011 \\
 \underline{01101} \\
 10011 \\
 \underline{10000} \\
 10011 \\
 \underline{0011} \\
 0011 \leftarrow R \text{ 余数}
 \end{array}$$

图 T-3-07(b) 出现一个错误的 CRC 检验

从图 T-3-07(b)可看出，余数 R 不为零，因此判定所接收的数据有差错。可见，这里的 CRC 检验可以发现这个差错。

若要发送的数据在传输过程中最后两个 1 都变成了 0，即 1101011000。把检验序列 1110 接在数据 1101011000 的后面，下一步就是进行 CRC 检验（如图 T-3-07(c)所示）。

现在余数 R 不为零，因此判定所接收的数据有差错。可见，这里的 CRC 检验可以发现这个差错。

采用 CRC 检验后，数据链路层的传输并非变成了可靠的传输。当接收方进行 CRC 检验时，如果发现有差错，就简单地丢弃这个帧。数据链路层并不能保证接收方接收到的和发送方发送的完全一样。

$$\begin{array}{r}
 \text{除数 } P \rightarrow 10011 \overline{) 11010110001110} \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 000010001 \\
 \underline{00011} \\
 001011 \\
 \underline{00000} \\
 10110 \\
 \underline{10011} \\
 0101 \leftarrow R \text{ 余数}
 \end{array}$$

图 T-3-07(c) 出现两个错误的 CRC 检验