历史事件

Norm Abramson 和 ALOHAnet

Norm Abramson 是一名有博士学位的工程师，对冲浪运动很有激情，而且对分组交换很感兴趣。这些兴趣的结合使他在 1969 年到了夏威夷大学。夏威夷是由许多巨大的岛屿组成的，安装和运营基于陆地的网络是困难的。当不冲浪的时候，Abramson 思考如何设计一种在无线信道上完成分组交互的网络。他设计的网络有一个中心主机和几个分散在夏威夷各个岛上的二级节点。该网络有两个信道，每个信道使用不同的频段。下行链路信道从中心主机向二级主机广播分组；上行信道从二级主机向中心主机发送分组。除了发送信息分组，中心主机还在下行信道上对从二级主机成功接收到的每个分组发送确认。

因为二级主机以分散的方式传输分组，在上行信道上出现碰撞是不可避免的。这个观察导致 Abramson 设计了如本章所描述的那种纯 ALOHA 协议。在 1970 年，通过不断从 ARPA 获得的资助，Abramson 将他的 ALOHAnet 与 ARPAnet 相连。Abramson 的工作是很重要的，不仅因为它是无线分组网络的第一个例子，而且因为它激励了 Bob Metcalfe。几年之后，Metcalfe 修改了 ALOHA 协议，创造了 CSMA/CD 协议和以太网局域网。

3. 载波侦听多路访问（CSMA）

在时隙和纯 ALOHA 中，一个节点传输的决定独立于连接到这个广播信道上的其他节点的活动。特别是，一个节点不关心在它开始传输时是否有其他节点碰巧在传输，而且即使有另一个节点开始干扰它的传输也不会停止传输。在我们的鸡尾酒会类比中，ALOHA 协议非常像一个粗野的聚会客人，他喋喋不休地讲话而不顾是否其他人在说话。作为人类，我们有人类的协议，它要求我们不仅要更为礼貌，而且在谈话中要减少与他人“碰撞”的时间，从而增加我们谈话中交流的数据量。具体而言，有礼貌的人类谈话有两个重要的规则：

- 说话之前先听。如果其他人正在说话，等到他们说完话为止。在网络领域中，这被称为**载波侦听**（carrier sensing），即一个节点在传输前先听信道。如果来自另一个节点的帧正向信道上发送，节点则等待直到检测到一小段时间没有传输，然后开始传输。
- 如果与他人同时开始说话，停止说话。在网络领域中，这被称为**碰撞检测**（collision detection），即当一个传输节点在传输时一直在侦听此信道。如果它检测到另一个节点正在传输干扰帧，它就停止传输，在重复“侦听 - 当空闲时传输”循环之前等待一段随机时间。

这两个规则包含在**载波侦听多路访问**（Carrier Sense Multiple Access，CSMA）和**具有碰撞检测的 CSMA**（CSMA with Collision Detection，CSMA/CD）协议族中［Kleinrock 1975b；Metcalfe 1976；Lam 1980；Rom 1990］。人们已经提出了 CSMA 和 CSMA/CD 的许多变种。这里，我们将考虑一些 CSMA 和 CSMA/CD 最重要的和基本的特性。

关于 CSMA 你可能要问的第一个问题是，如果所有的节点都进行载波侦听了，为什么当初会发生碰撞？毕竟，某节点无论何时侦听到另一个节点在传输，它都会停止传输。对于这个问题的答案最好能够用时空图来说明［Molle 1987］。图 6-12 显示了连接到一个线

状广播总线的4个节点（A、B、C、D）的时空图。横轴表示每个节点在空间的位置；纵轴表示时间。

在时刻 t_0，节点B侦听到信道是空闲的，因为当前没有其他节点在传输。因此节点B开始传输，沿着广播媒体在两个方向上传播它的比特。图6-12中B的比特随着时间的增加向下传播，这表明B的比特沿着广播媒体传播所实际需要的时间不是零（虽然以接近光的速度）。在时刻 t_1（$t_1 > t_0$），节点D有一个帧要发送。尽管节点B在时刻 t_1 正在传输，但B传输的比特还没有到达D，因此D在 t_1 侦听到信道空闲。根据CSMA协议，从而D开始传输它的帧。一个短暂的时间之后，B的传输开始在D干扰D的传输。从图6-12中可以看出，显然广播信道的端到端**信道传播时延**（channel propagation delay）（信号从一个节点传播到另一个节点所花费的时间）在决定其性能方面起着关键的作用。该传播时延越长，载波侦听节点不能侦听到网络中另一个节点已经开始传输的机会就越大。

4. 具有碰撞检测的载波侦听多路访问（CSMA/CD）

在图6-12中，节点没有进行碰撞检测；即使已经出现了碰撞，B和D都将继续完整地传输它们的帧。当某节点执行碰撞检测时，一旦它检测到碰撞将立即停止传输。图6-13表示了和图6-12相同的情况，只是这两个节点在检测到碰撞后很短的时间内都放弃了它们的传输。显然，在多路访问协议中加入碰撞检测，通过不传输一个无用的、（由来自另一个节点的帧干扰）损坏的帧，将有助于改善协议的性能。

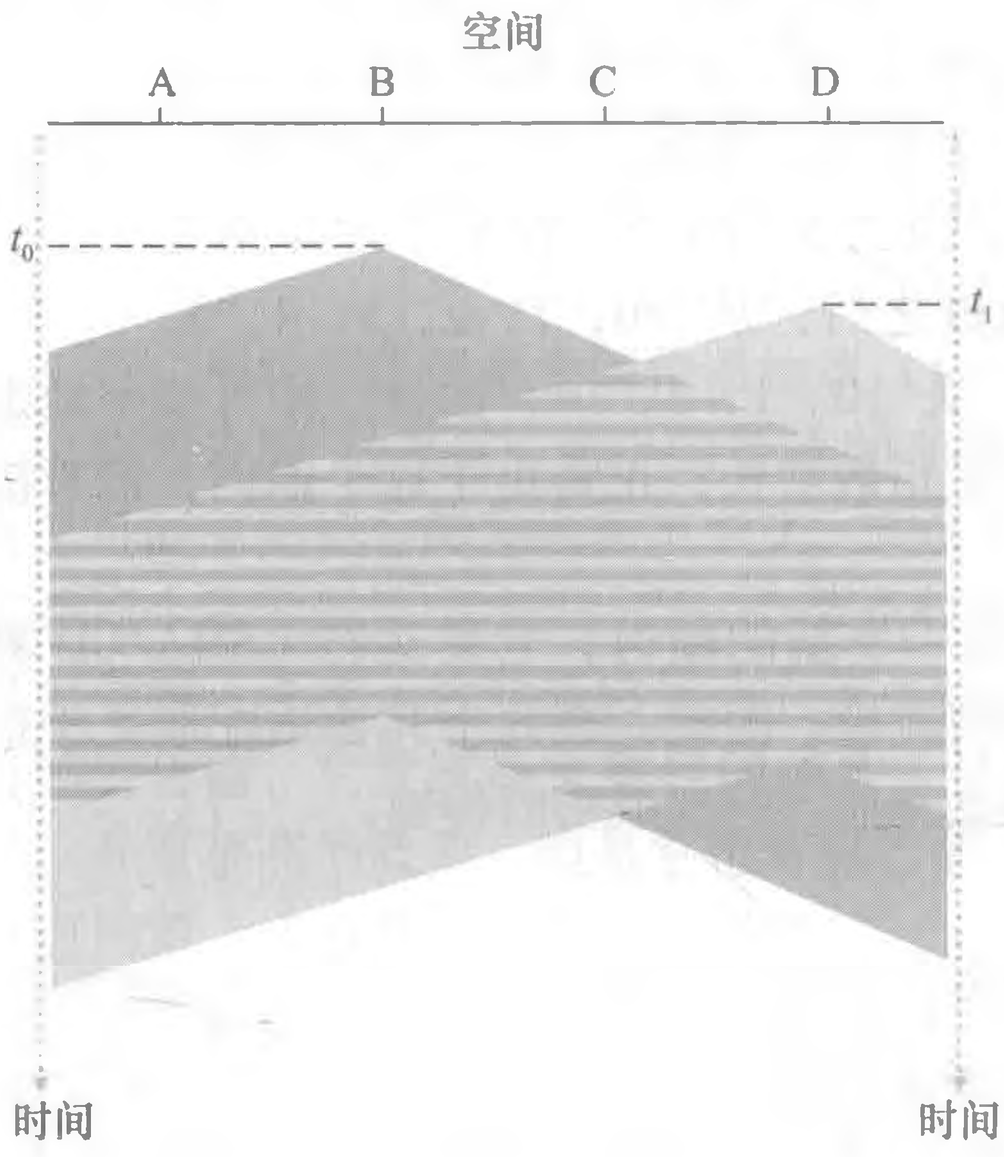

图6-12　发生碰撞传输的两个CSMA节点的时空图

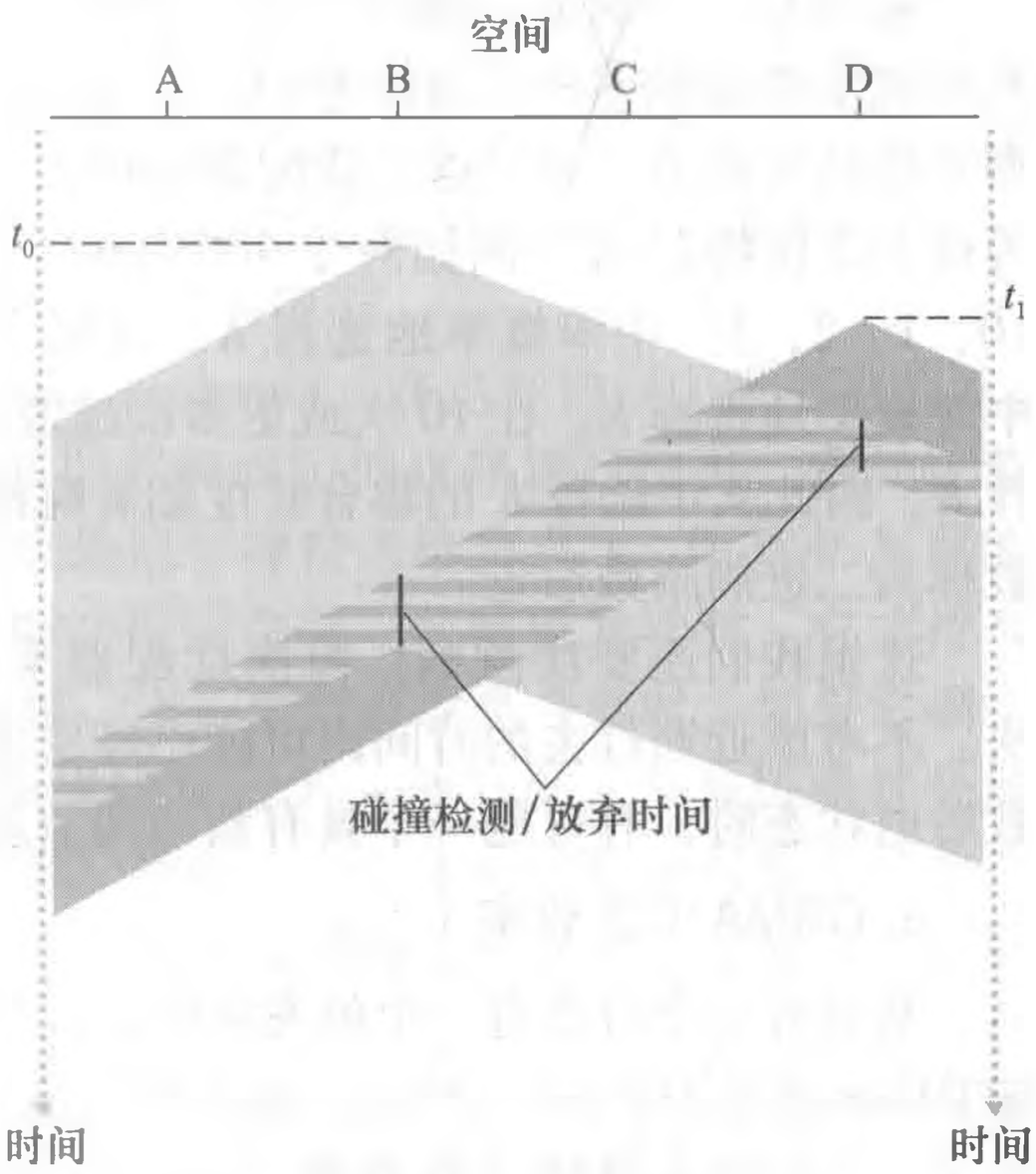

图6-13　具有碰撞检测的CSMA

在分析CSMA/CD协议之前，我们现在从与广播信道相连的适配器（在节点中）的角度总结它的运行：

1）适配器从网络层一条获得数据报，准备链路层帧，并将其放入帧适配器缓存中。

2）如果适配器侦听到信道空闲（即无信号能量从信道进入适配器），它开始传输帧。在另一方面，如果适配器侦听到信道正在忙，它将等待，直到侦听到没有信号能量时才开始传输帧。

3）在传输过程中，适配器监视来自其他使用该广播信道的适配器的信号能量的存在。

4）如果适配器传输整个帧而未检测到来自其他适配器的信号能量，该适配器就完成了该帧。在另一方面，如果适配器在传输时检测到来自其他适配器的信号能量，它中止传输（即它停止了传输帧）。

5）中止传输后，适配器等待一个随机时间量，然后返回步骤2。

等待一个随机（而不是固定）的时间量的需求是明确的——如果两个节点同时传输帧，然后这两个节点等待相同固定的时间量，它们将持续碰撞下去。但选择随机回退时间的时间间隔多大为好呢？如果时间间隔大而碰撞节点数量小，在重复“侦听－当空闲时传输”的步骤前，节点很可能等待较长的时间（使信道保持空闲）。在另一方面，如果时间间隔小而碰撞节点数量大，很可能选择的随机值将几乎相同，传输节点将再次碰撞。我们希望时间间隔应该这样：当碰撞节点数量较少时，时间间隔较短；当碰撞节点数量较大时，时间间隔较长。

用于以太网以及DOCSIS电缆网络多路访问协议［DOCSIS 2011］中的**二进制指数后退**（binary exponential backoff）算法，简练地解决了这个问题。特别是，当传输一个给定帧时，在该帧经历了一连串的 n 次碰撞后，节点随机地从 $\{0, 1, 2, \cdots, 2^n - 1\}$ 中选择一个 K 值。因此，一个帧经历的碰撞越多，K 选择的间隔越大。对于以太网，一个节点等待的实际时间量是 $K \cdot 512$ 比特时间（即发送512比特进入以太网所需时间量的 K 倍），n 能够取的最大值在10以内。

我们看一个例子。假设一个适配器首次尝试传输一个帧，并在传输中它检测到碰撞。然后该节点以概率0.5选择 $K=0$，以概率0.5选择 $K=1$。如果该节点选择 $K=0$，则它立即开始侦听信道。如果这个适配器选择 $K=1$，它在开始“侦听－当空闲时传输”。周期前等待512比特时间（例如对于100Mbps以太网来说为5.12ms）。在第2次碰撞之后，从 $\{0, 1, 2, 3\}$ 中等概率地选择 K。在第3次碰撞之后，从 $\{0, 1, 2, 3, 4, 5, 6, 7\}$ 中等概率地选择 K。在10次或更多次碰撞之后，从 $\{0, 1, 2, \cdots, 1023\}$ 中等概率地选择 K。因此从中选择 K 的集合长度随着碰撞次数呈指数增长；正是由于这个原因，该算法被称为二进制指数后退。

这里我们还要注意到，每次适配器准备传输一个新的帧时，它要运行CSMA/CD算法。不考虑近期过去的时间内可能已经发生的任何碰撞。因此，当几个其他适配器处于指数后退状态时，有可能一个具有新帧的节点能够立刻插入一次成功的传输。

5. CSMA/CD效率

当只有一个节点有一个帧发送时，该节点能够以信道全速率进行传输（例如10Mbps、100Mbps或者1Gbps）。然而，如果很多节点都有帧要发送，信道的有效传输速率可能会小得多。我们将**CSMA/CD效率**（efficiency of CSMA/CD）定义为：当有大量的活跃节点，且每个节点有大量的帧要发送时，帧在信道中无碰撞地传输的那部分时间在长期运行时间中所占的份额。为了给出效率的一个闭式的近似表示，令 d_{prop} 表示信号能量在任意两个适配器之间传播所需的最大时间。令 d_{trans} 表示传输一个最大长度的以太网帧的时间（对于10Mbps的以太网，该时间近似为1.2毫秒）。CSMA/CD效率的推导超出了本书的范围（见［Lam 1980］和［Bertsekas 1991］）。这里我们只是列出下面的近似式：

$$\text{效率} = \frac{1}{1 + 5d_{prop}/d_{trans}}$$

从这个公式我们看到，当 d_{prop} 接近0时，效率接近1。这和我们的直觉相符，如果传

播时延是0，碰撞的节点将立即中止而不会浪费信道。同时，当 d_{trans} 变得很大时，效率也接近于1。这也和直觉相符，因为当一个帧取得了信道时，它将占有信道很长时间；因此信道在大多数时间都会有效地工作。

6.3.3 轮流协议

前面讲过多路访问协议的两个理想特性是：①当只有一个节点活跃时，该活跃节点具有 R bps 的吞吐量；②当有 M 个节点活跃时，每个活跃节点的吞吐量接近 R/M bps。ALOHA 和 CSMA 协议具备第一个特性，但不具备第二个特性。这激发研究人员创造另一类协议，也就是**轮流协议**（taking-turns protocol）。和随机接入协议一样，有几十种轮流协议，其中每一个协议又都有很多变种。这里我们要讨论两种比较重要的协议。第一种是**轮询协议**（polling protocol）。轮询协议要求这些节点之一要被指定为主节点。主节点以循环的方式**轮询**（poll）每个节点。特别是，主节点首先向节点1发送一个报文，告诉它（节点1）能够传输的帧的最多数量。在节点1传输了某些帧后，主节点告诉节点2它（节点2）能够传输的帧的最多数量。（主节点能够通过观察在信道上是否缺乏信号，来决定一个节点何时完成了帧的发送。）上述过程以这种方式继续进行，主节点以循环的方式轮询了每个节点。

轮询协议消除了困扰随机接入协议的碰撞和空时隙，这使得轮询取得高得多的效率。但是它也有一些缺点。第一个缺点是该协议引入了轮询时延，即通知一个节点“它可以传输”所需的时间。例如，如果只有一个节点是活跃的，那么这个节点将以小于 R bps 的速率传输，因为每次活跃节点发送了它最多数量的帧时，主节点必须依次轮询每一个非活跃的节点。第二个缺点可能更为严重，就是如果主节点有故障，整个信道都变得不可操作。我们在本节学习的802.15协议和蓝牙协议就是轮询协议的例子。

第二种轮流协议是**令牌传递协议**（token-passing protocol）。在这种协议中没有主节点。一个称为**令牌**（token）的小的特殊帧在节点之间以某种固定的次序进行交换。例如，节点1可能总是把令牌发送给节点2，节点2可能总是把令牌发送给节点3，而节点 N 可能总是把令牌发送给节点1。当一个节点收到令牌时，仅当它有一些帧要发送时，它才持有这个令牌；否则，它立即向下一个节点转发该令牌。当一个节点收到令牌时，如果它确实有帧要传输，它发送最大数目的帧数，然后把令牌转发给下一个节点。令牌传递是分散的，并有很高的效率。但是它也有自己的一些问题。例如，一个节点的故障可能会使整个信道崩溃。或者如果一个节点偶然忘记了释放令牌，则必须调用某些恢复步骤使令牌返回到循环中来。经过多年，人们已经开发了许多令牌传递协议，包括光纤分布式数据接口（FDDI）协议[Jain 1994] 和 IEEE 802.5 令牌环协议 [IEEE 802.5 2012]，每一种都必须解决这些和其他一些棘手的问题。

6.3.4 DOCSIS：用于电缆因特网接入的链路层协议

在前面3小节中，我们已经学习了3大类多路访问协议：信道划分协议、随机接入协议和轮流协议。这里的电缆接入网将作为一种很好的学习案例，因为在电缆接入网中我们将看到这三类多路访问协议中的每一种！

1.2.1节讲过，一个电缆接入网通常在电缆网头端将几千个住宅电缆调制解调器与一个**电缆调制解调器端接系统**（Cable Modem Termination System，CMTS）连接。**数据经电缆服务接口**（Data-Over-Cable Service Interface，CMTS）规范（DOCSIS）[DOCSIS 2011] 定

义了电缆数据网络体系结构及其协议。DOCSIS 使用 FDM 将下行（CMTS 到调制解调器）和上行（调制解调器到 CMTS）网络段划分为多个频率信道。每个下行信道宽 6MHz，每个信道具有大约 40Mbps 吞吐量（尽管这种数据率在实践中很少在电缆调制解调器中见到）；每个上行信道具有 6.4MHz 的最大信道带宽，并且最大的上行吞吐量约为 30Mbps。每个上行和下行信道均为广播信道。CMTS 在下行信道中传输的帧被所有在信道上做接收的电缆调制解调器接收到；然而因为仅有单一的 CMTS 在下行信道上传输，不存在多路访问问题。但在上行方向，存在着多个有趣的技术挑战，因为多个电缆调制解调器共享到 CMTS 的相同上行信道（频率），因此能够潜在地出现碰撞。

如图6-14 所示，每条上行信道被划分为时间间隔（类似于 TDM），每个时间间隔包含一个微时隙序列，电缆调制解调器可在该微时隙中向 CMTS 传输。CMTS 显式地准许各个电缆调制解调器在特定的微时隙中进行传输。CMTS 在下行信道上通过发送称为 MAP 报文的控制报文，指定哪个电缆调制解调器（带有要发送的数据）能够在微时隙中传输由控制报文指定的时间间隔。由于微时隙明确分配给电缆调制解调器，故 CMTS 能够确保在微时隙中没有碰撞传输。

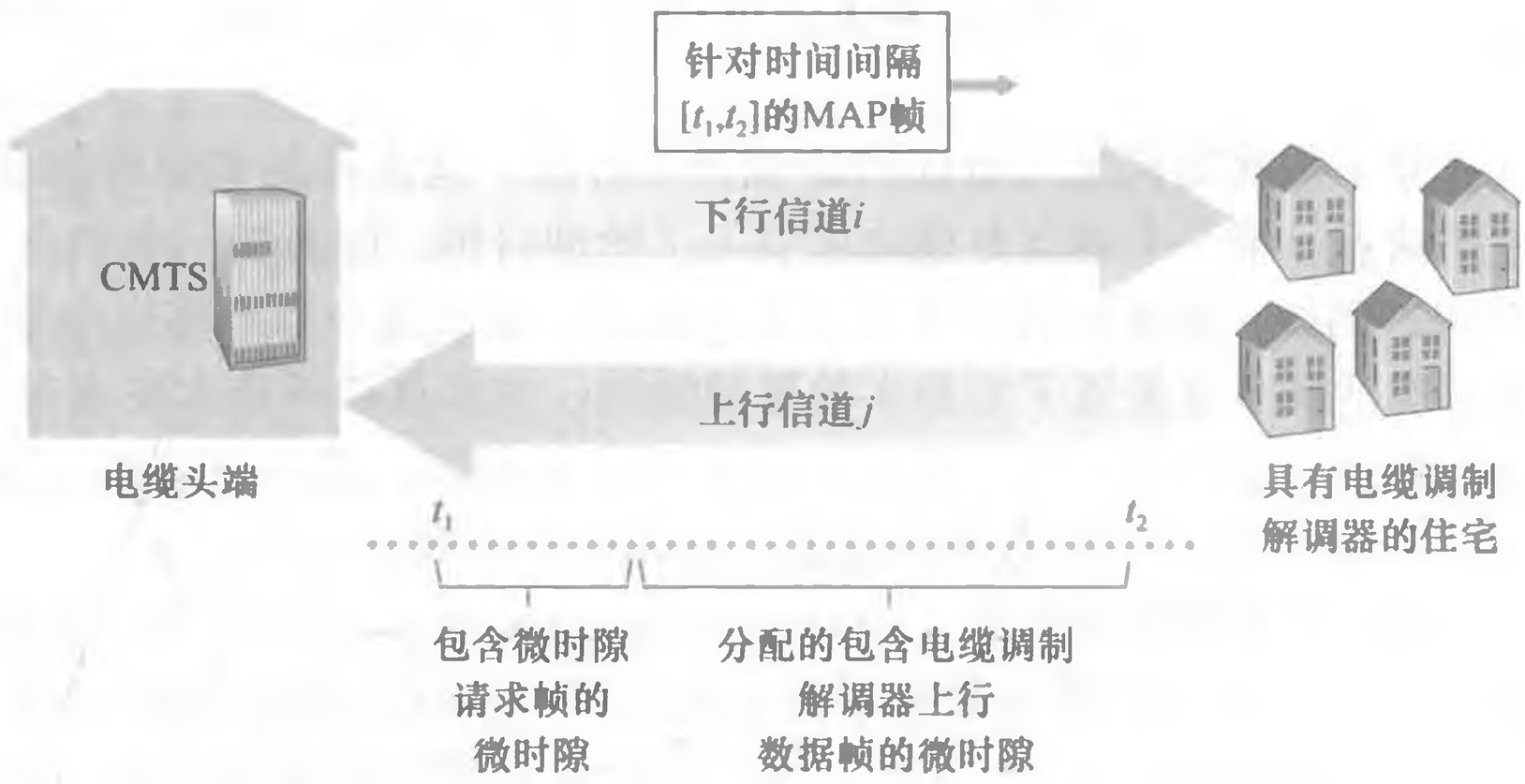

图 6-14　CMTS 和电缆调制解调器之间的上行和下行信道

但是 CMTS 一开始是如何知道哪个电缆调制解调器有数据要发送呢？通过让电缆调制解调器在专用于此目的的一组特殊的微时隙间隔内向 CMTS 发送微时隙请求帧来完成该任务，如图 6-14 所示。这些微时隙请求帧以随机接入方式传输，故可能相互碰撞。电缆调制解调器既不能侦听上行信道是否忙，也不能检测碰撞。相反，该电缆调制解调器如果没有在下一个下行控制报文中收到对请求分配的响应的话，就推断出它的微时隙请求帧经历了一次碰撞。当推断出一次碰撞，电缆调制解调器使用二进制指数回退将其微时隙请求帧延缓到以后的时隙重新发送。当在上行信道上有很少的流量，电缆调制解调器可能在名义上分配给微时隙请求帧的时隙内实际传输数据帧（因此避免不得不等待微时隙分配）。

因此，电缆接入网可作为应用多路访问协议（即 FDM、TDM、随机接入和集中分配时隙都用于一个网络中）的一个极好例子。

6.4　交换局域网

前面一节涉及了广播网络和多路访问协议，我们现在将注意力转向交换局域网。

图 6-15 显示了一个交换局域网连接了 3 个部门，两台服务器和一台与 4 台交换机连接的路由器。因为这些交换机运行在链路层，所以它们交换链路层帧（而不是网络层数据报），不识别网络层地址，不使用如 RIP 或 OSPF 这样的路由选择算法来确定通过第二层交换机网络的路径。我们马上就会看到，它们使用链路层地址而不是 IP 地址来转发链路层帧通过交换机网络。我们首先以讨论链路层寻址（6.4.1 节）来开始对交换机局域网的学习。然后仔细学习著名的以太网协议（6.4.2 节）。在仔细学习链路层寻址和以太网后，我们将考察链路层交换机的工作方式（6.4.3 节），并随后考察通常是如何用这些交换机构建大规模局域网的（6.4.4 节）。

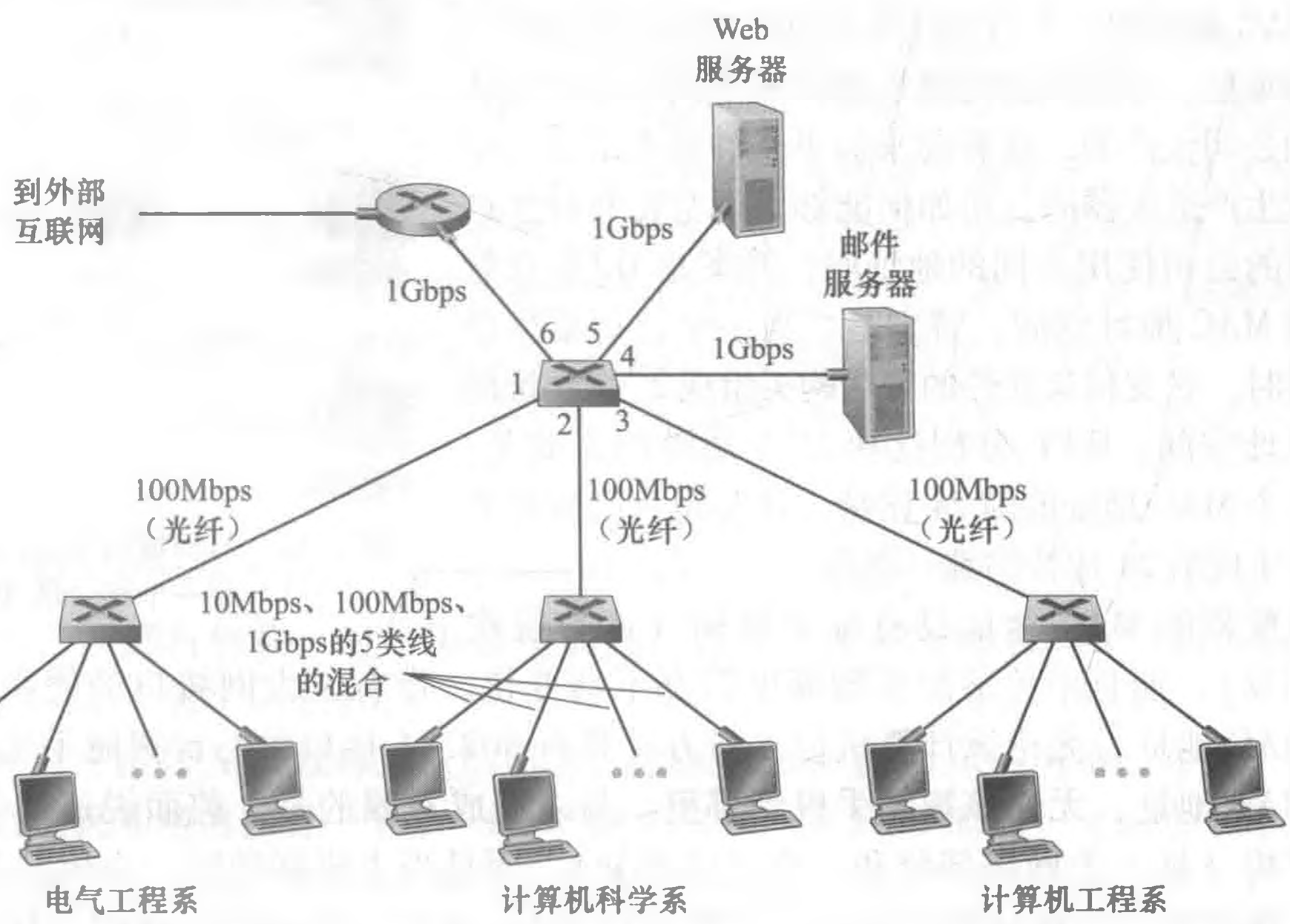

图 6-15　由 4 台交换机连接起来的某机构网络

6.4.1　链路层寻址和 ARP

主机和路由器具有链路层地址。现在你也许会感到惊讶，第 4 章中不是讲过主机和路由器也具有网络层地址吗？你也许会问：为什么我们在网络层和链路层都需要地址呢？除了描述链路层地址的语法和功能，在本节中我们希望明明白白地搞清楚两层地址都有用的原因，事实上这些地址是必不可少的。我们还将学习地址解析协议（ARP），该协议提供了将 IP 地址转换为链路层地址的机制。

1. MAC 地址

事实上，并不是主机或路由器具有链路层地址，而是它们的适配器（即网络接口）具有链路层地址。因此，具有多个网络接口的主机或路由器将具有与之相关联的多个链路层地址，就像它也具有与之相关联的多个 IP 地址一样。然而，重要的是注意到链路层交换机并不具有与它们的接口（这些接口是与主机和路由器相连的）相关联的链路层地址。这是因为链路层交换机的任务是在主机与路由器之间承载数据报；交换机透明地执行该项任务，这就是说，主机或路由器不必明确地将帧寻址到其间的交换机。图 6-16 中说明了这

种情况。链路层地址有各种不同的称呼：**LAN 地址**（LAN address）、**物理地址**（physical address）或 **MAC 地址**（MAC address）。因为 MAC 地址似乎是最为流行的术语，所以我们此后就将链路层地址称为 MAC 地址。对于大多数局域网（包括以太网和 802.11 无线局域网）而言，MAC 地址长度为 6 字节，共有 2^{48} 个可能的 MAC 地址。如图 6-16 所示，这些 6 个字节地址通常用十六进制表示法，地址的每个字节被表示为一对十六进制数。尽管 MAC 地址被设计为永久的，但用软件改变一块适配器的 MAC 地址现在是可能的。然而，对于本节的后面部分而言，我们将假设某适配器的 MAC 地址是固定的。

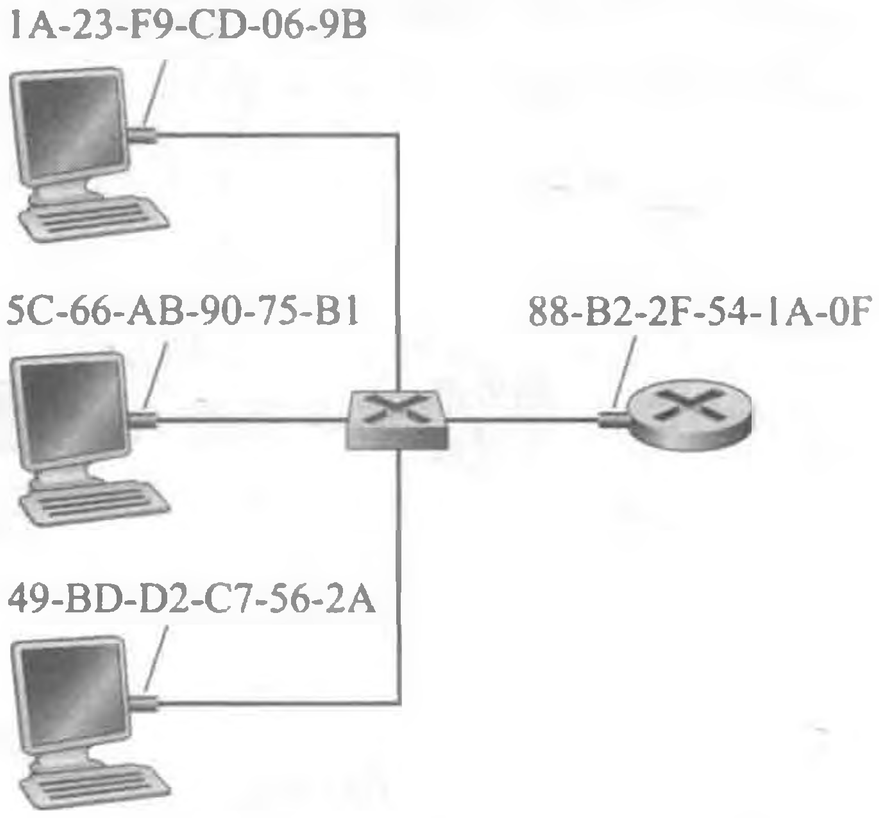

图 6-16　与局域网相连的每个接口都有一个唯一的 MAC 地址

MAC 地址的一个有趣性质是没有两块适配器具有相同的地址。考虑到适配器是由许多不同国家和地区的不同公司生产的，这看起来似乎是件神奇之事。中国台湾生产适配器的公司如何能够保证与比利时生产适配器的公司使用不同的地址呢？答案是 IEEE 在管理着该 MAC 地址空间。特别是，当一个公司要生产适配器时，它支付象征性的费用购买组成 2^{24} 个地址的一块地址空间。IEEE 分配这块 2^{24} 个地址的方式是：固定一个 MAC 地址的前 24 比特，让公司自己为每个适配器生成后 24 比特的唯一组合。

适配器的 MAC 地址具有扁平结构（这与层次结构相反），而且不论适配器到哪里用都不会变化。带有以太网接口的便携机总具有同样的 MAC 地址，无论该计算机位于何方。具有 802.11 接口的一台智能手机总是具有相同的 MAC 地址，无论该智能手机到哪里。与之形成对照的是，前面说过的 IP 地址具有层次结构（即一个网络部分和一个主机部分），而且当主机移动时，主机的 IP 地址需要改变，即改变它所连接到的网络。适配器的 MAC 地址与人的社会保险号相似，后者也具有扁平寻址结构，而且无论人到哪里该号码都不会变化。IP 地址则与一个人的邮政地址相似，它是有层次的，无论何时当人搬家时，该地址都必须改变。就像一个人可能发现邮政地址和社会保险号都有用那样，一台主机具有一个网络层地址和一个 MAC 地址是有用的。

当某适配器要向某些目的适配器发送一个帧时，发送适配器将目的适配器的 MAC 地址插入到该帧中，并将该帧发送到局域网上。如我们马上要看到的那样，一台交换机偶尔将一个入帧广播到它的所有接口。我们将在第 7 章中看到 802.11 也广播帧。因此一块适配器可以接收一个并非向它寻址的帧。这样，当适配器接收到一个帧时，将检查该帧中的目的 MAC 地址是否与它自己的 MAC 地址匹配。如果匹配，该适配器提取出封装的数据报，并将该数据报沿协议栈向上传递。如果不匹配，该适配器丢弃该帧，而不会向上传递该网络层数据报。所以，仅当收到该帧时，才会中断目的地。

然而，有时某发送适配器的确要让局域网上所有其他适配器来接收并处理它打算发送的帧。在这种情况下，发送适配器在该帧的目的地址字段中插入一个特殊的 MAC **广播地址**（broadcast address）。对于使用 6 字节地址的局域网（例如以太网和 802.11）来说，广播地址是 48 个连续的 1 组成的字符串（即以十六进制表示法表示的 FF-FF-FF-FF-FF-FF）。

实践原则

保持各层独立

主机和路由器接口除了网络层地址之外还有 MAC 地址，这有如下几个原因。首先，局域网是为任意网络层协议而设计的，而不只是用于 IP 和因特网。如果适配器被指派 IP 地址而不是“中性的” MAC 地址的话，则适配器将不能够方便地支持其他网络层协议（例如，IPX 或者 DECnet）。其次，如果适配器使用网络层地址而不是 MAC 地址的话，网络层地址必须存储在适配器的 RAM 中，并且在每次适配器移动（或加电）时要重新配置。另一种选择是在适配器中不使用任何地址，让每个适配器将它收到的每帧数据（通常是 IP 数据报）沿协议栈向上传递。然后网络层则能够核对网络地址层是否匹配。这种选择带来的一个问题是，主机将被局域网上发送的每个帧中断，包括被目的地是在相同广播局域网上的其他节点的帧中断。总之，为了使网络体系结构中各层次成为极为独立的构建模块，不同的层次需要有它们自己的寻址方案。我们现在已经看到 3 种类型的地址：应用层的主机名、网络层的 IP 地址以及链路层的 MAC 地址。

2. 地址解析协议

因为存在网络层地址（例如，因特网的 IP 地址）和链路层地址（即 MAC 地址），所以需要在它们之间进行转换。对于因特网而言，这是**地址解析协议**（Address Resolution Protocol，ARP）[RFC 826] 的任务。

为了理解对于诸如 ARP 这样协议的需求，考虑如图 6-17 所示的网络。在这个简单的例子中，每台主机和路由器有一个单一的 IP 地址和单一的 MAC 地址。与以往一样，IP 地址以点分十进制表示法表示，MAC 地址以十六进制表示法表示。为了便于讨论，我们在本节中将假设交换机广播所有帧；这就是说，无论何时交换机在一个接口接收一个帧，它将在其所有其他接口上转发该帧。在下一节中，我们将更为准确地解释交换机操作的过程。

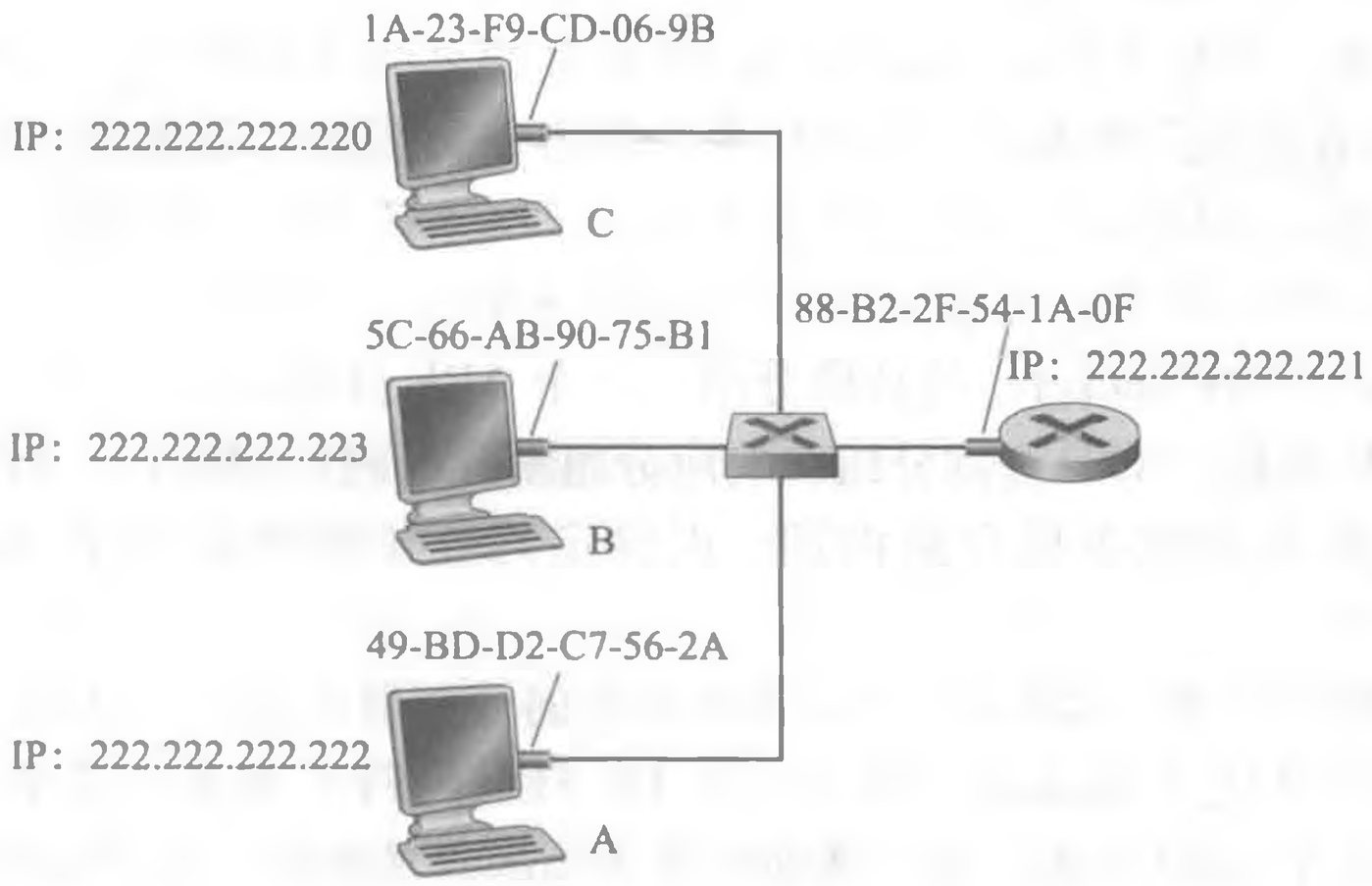

图 6-17 局域网上的每个接口都有一个 IP 地址和一个 MAC 地址

现在假设 IP 地址为 222. 222. 222. 220 的主机要向主机 222. 222. 222. 222 发送 IP 数据

报。在本例中，源和目的均位于相同的子网中（在4.3.3节中的寻址意义下）。为了发送数据报，该源必须要向它的适配器不仅提供IP数据报，而且要提供目的主机222.222.222.222的MAC地址。然后发送适配器将构造一个包含目的地的MAC地址的链路层帧，并把该帧发送进局域网。

在本节中要处理的重要问题是，发送主机如何确定IP地址为222.222.222.222的目的主机的MAC地址呢？正如你也许已经猜想的那样，它使用ARP。在发送主机中的ARP模块将取在相同局域网上的任何IP地址作为输入，然后返回相应的MAC地址。在眼下的这个例子中，发送主机222.222.222.220向它的ARP模块提供了IP地址222.222.222.222，并且其ARP模块返回了相应的MAC地址49-BD-D2-C7-56-2A。

因此我们看到了ARP将一个IP地址解析为一个MAC地址。在很多方面它和DNS（在2.4节中学习过）类似，DNS将主机名解析为IP地址。然而，这两种解析器之间的一个重要区别是，DNS为在因特网中任何地方的主机解析主机名，而ARP只为在同一个子网上的主机和路由器接口解析IP地址。如果美国加利福尼亚州的一个节点试图用ARP为美国密西西比州的一个节点解析IP地址，ARP将返回一个错误。

既然已经解释了ARP的用途，我们再来看看它是如何工作的。每台主机或路由器在其内存中具有一个**ARP表**（ARP table），这张表包含IP地址到MAC地址的映射关系。图6-18显示了在主机222.222.222.220中可能看到的ARP表中的内容。该ARP表也包含一个寿命（TTL）值，它指示了从表中删除每个映射的时间。注意到这张表不必为该子网上的每台主机和路由器都包含一个表项；某些可能从来没有进入到该表中，某些可能已经过期。从一个表项放置到某ARP表中开始，一个表项通常的过期时间是20分钟。

IP地址	MAC地址	TTL
222.222.222.221	88-B2-2F-54-1A-0F	13:45:00
222.222.222.223	5C-66-AB-90-75-B1	13:52:00

图6-18　在主机222.222.222.220中的一个可能的ARP表

现在假设主机222.222.222.220要发送一个数据报，该数据报要IP寻址到本子网上另一台主机或路由器。发送主机需要获得给定IP地址的目的主机的MAC地址。如果发送方的ARP表具有该目的节点的表项，这个任务是很容易完成的。但如果ARP表中当前没有该目的主机的表项，又该怎么办呢？特别是假设222.222.222.220要向222.222.222.222发送数据报。在这种情况下，发送方用ARP协议来解析这个地址。首先，发送方构造一个称为**ARP分组**（ARP packet）的特殊分组。一个ARP分组有几个字段，包括发送和接收IP地址及MAC地址。ARP查询分组和响应分组都具有相同的格式。ARP查询分组的目的是询问子网上所有其他主机和路由器，以确定对应于要解析的IP地址的那个MAC地址。

回到我们的例子上来，222.222.222.220向它的适配器传递一个ARP查询分组，并且指示适配器应该用MAC广播地址（即FF-FF-FF-FF-FF-FF）来发送这个分组。适配器在链路层帧中封装这个ARP分组，用广播地址作为帧的目的地址，并将该帧传输进子网中。回想我们的社会保险号/邮政地址的类比，一次ARP查询等价于一个人在某公司（比方说AnyCorp）一个拥挤的房间大喊："邮政地址是加利福尼亚州帕罗奥图市AnyCorp公司112房间13室的那个人的社会保险号是什么?"包含该ARP查询的帧能被子网上的所有其他

适配器接收到，并且（由于广播地址）每个适配器都把在该帧中的 ARP 分组向上传递给 ARP 模块。这些 ARP 模块中的每个都检查它的 IP 地址是否与 ARP 分组中的目的 IP 地址相匹配。与之匹配的一个给查询主机发送回一个带有所希望映射的响应 ARP 分组。然后查询主机 222.222.222.220 能够更新它的 ARP 表，并发送它的 IP 数据报，该数据报封装在一个链路层帧中，并且该帧的目的 MAC 就是对先前 ARP 请求进行响应的主机或路由器的 MAC 地址。

关于 ARP 协议有两件有趣的事情需要注意。首先，查询 ARP 报文是在广播帧中发送的，而响应 ARP 报文在一个标准帧中发送。在继续阅读之前，你应该思考一下为什么这样。其次，ARP 是即插即用的，这就是说，一个 ARP 表是自动建立的，即它不需要系统管理员来配置。并且如果某主机与子网断开连接，它的表项最终会从留在子网中的节点的表中删除掉。

学生们常常想知道 ARP 是一个链路层协议还是一个网络层协议。如我们所看到的那样，一个 ARP 分组封装在链路层帧中，因而在体系结构上位于链路层之上。然而，一个 ARP 分组具有包含链路层地址的字段，因而可认为是链路层协议，但它也包含网络层地址，因而也可认为是为网络层协议。所以，可能最好把 ARP 看成是跨越链路层和网络层边界两边的协议，即不完全符合我们在第 1 章中学习的简单的分层协议栈。现实世界协议就是这样复杂！

3. 发送数据报到子网以外

现在应该搞清楚当一台主机要向相同子网上的另一台主机发送一个数据报时 ARP 的操作过程。但是现在我们来看更复杂的情况，即当子网中的某主机要向子网之外（也就是跨越路由器的另一个子网）的主机发送网络层数据报的情况。我们在图 6-19 的环境中来讨论这个问题，该图显示了一个由一台路由器互联两个子网所组成的简单网络。

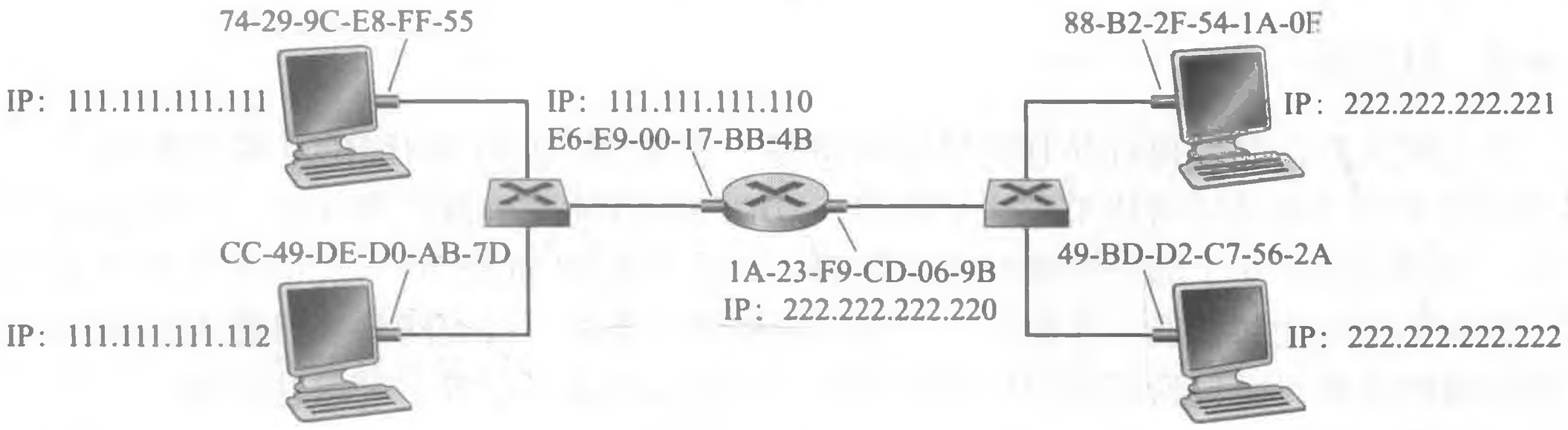

图 6-19　由一台路由器互联的两个子网

有关图 6-19 需要注意几件有趣的事情。每台主机仅有一个 IP 地址和一个适配器。但是，如第 4 章所讨论，一台路由器对它的每个接口都有一个 IP 地址。对路由器的每个接口，（在路由器中）也有一个 ARP 模块和一个适配器。在图 6-19 中的路由器有两个接口，所以它有两个 IP 地址、两个 ARP 模块和两个适配器。当然，网络中的每个适配器都有自己的 MAC 地址。

还要注意到子网 1 的网络地址为 111.111.111/24，子网 2 的网络地址为 222.222.222/24。因此，与子网 1 相连的所有接口都有格式为 111.111.111.xxx 的地址，与子网 2 相连的所有接口都有格式为 222.222.222.xxx 的地址。

现在我们考察子网 1 上的一台主机将向子网 2 上的一台主机发送数据报。特别是，假

设主机 111.111.111.111 要向主机 222.222.222.222 发送一个 IP 数据报。和往常一样，发送主机向它的适配器传递数据报。但是，发送主机还必须向它的适配器指示一个适当的目的 MAC 地址。该适配器应该使用什么 MAC 地址呢？有人也许大胆猜测，这个适当的 MAC 地址就是主机 222.222.222.222 的适配器地址，即 49-BD-D2-C7-56-2A。然而，这个猜测是错误的！如果发送适配器要用那个 MAC 地址，那么子网 1 上所有的适配器都不会费心将该 IP 数据报传递到它的网络层，因为该帧的目的地址与子网 1 上所有适配器的 MAC 地址都将不匹配。这个数据报将只有死亡，到达数据报天国。

如果我们仔细地观察图 6-19，我们发现为了使一个数据报从 111.111.111.111 到子网 2 上的主机，该数据报必须首先发送给路由器接口 111.111.111.110，它是通往最终目的地路径上的第一跳路由器的 IP 地址。因此，对于该帧来说，适当的 MAC 地址是路由器接口 111.111.111.110 的适配器地址，即 E6-E9-00-17-BB-4B。但发送主机怎样获得 111.111.111.110 的 MAC 地址呢？当然是通过使用 ARP！一旦发送适配器有了这个 MAC 地址，它创建一个帧（包含了寻址到 222.222.222.222 的数据报），并把该帧发送到子网 1 中。在子网 1 上的路由器适配器看到该链路层帧是向它寻址的，因此把这个帧传递给路由器的网络层。万岁！该 IP 数据报终于被成功地从源主机移动到这台路由器了！但是我们的任务还没有结束。我们仍然要将该数据报从路由器移动到目的地。路由器现在必须决定该数据报要被转发的正确接口。如在第 4 章中所讨论的，这是通过查询路由器中的转发表来完成的。转发表告诉这台路由器该数据报要通过路由器接口 222.222.222.220 转发。然后该接口把这个数据报传递给它的适配器，适配器把该数据报封装到一个新的帧中，并且将帧发送进子网 2 中。这时，该帧的目的 MAC 地址确实是最终目的地 MAC 地址。路由器又是怎样获得这个目的地 MAC 地址的呢？当然是用 ARP 获得的！

用于以太网的 ARP 定义在 RFC 826 中。在 TCP/IP 指南 RFC 1180 中对 ARP 进行了很好的介绍。我们将在课后习题中更为详细地研究 ARP。

6.4.2　以太网

以太网几乎占领着现有的有线局域网市场。在 20 世纪 80 年代和 90 年代早期，以太网面临着来自其他局域网技术包括令牌环、FDDI 和 ATM 的挑战。多年来，这些其他技术中的一些成功地抓住了部分局域网市场份额。但是自从 20 世纪 70 年代中期发明以太网以来，它就不断演化和发展，并保持了它的支配地位。今天，以太网是到目前为止最流行的有线局域网技术，而且到可能预见的将来它可能仍保持这一位置。可以这么说，以太网对本地区域联网的重要性就像因特网对全球联网所具有的地位那样。

以太网的成功有很多原因。首先，以太网是第一个广泛部署的高速局域网。因为它部署得早，网络管理员非常熟悉以太网（它的奇迹和它的奇思妙想），并当其他局域网技术问世时，他们不愿意转而用之。其次，令牌环、FDDI 和 ATM 比以太网更加复杂、更加昂贵，这就进一步阻碍了网络管理员改用其他技术。第三，改用其他局域网技术（例如 FDDI 和 ATM）的最引人注目的原因通常是这些新技术具有更高数据速率；然而以太网总是奋起抗争，产生了运行在相同或更高数据速率下的版本。20 世纪 90 年代初期引入了交换以太网，这就进一步增加了它的有效数据速率。最后，由于以太网已经很流行了，所以以太网硬件（尤其是适配器和交换机）成了一个普通商品，而且极为便宜。

Bob Metcalfe 和 David Boggs 在 20 世纪 70 年代中期发明初始的以太局域网。初始的以太局域网使用同轴电缆总线来互联节点。以太网的总线拓扑实际上从 20 世纪 80 年代到 90

年代中期一直保持不变。使用总线拓扑的以太网是一种广播局域网，即所有传输的帧传送到与该总线连接的所有适配器并被其处理。回忆一下，我们在 6.3.2 节中讨论了以太网的具有二进制指数回退的 CSMA/CD 多路访问协议。

到了 20 世纪 90 年代后期，大多数公司和大学使用一种基于集线器的星形拓扑以太网安装替代了它们的局域网。在这种安装中，主机（和路由器）直接用双绞对铜线与一台集线器相连。**集线器**（hub）是一种物理层设备，它作用于各个比特而不是作用于帧。当表示一个 0 或一个 1 的比特到达一个接口时，集线器只是重新生成这个比特，将其能量强度放大，并将该比特向其他所有接口传输出去。因此，采用基于集线器的星形拓扑的以太网也是一个广播局域网，即无论何时集线器从它的一个接口接收到一个比特，它向其所有其他接口发送该比特的副本。特别是，如果某集线器同时从两个不同的接口接收到帧，将出现一次碰撞，生成该帧的节点必须重新传输该帧。

在 21 世纪初，以太网又经历了一次重要的革命性变化。以太网安装继续使用星形拓扑，但是位于中心的集线器被**交换机**（switch）所替代。在本章后面我们将深入学习交换以太网。眼下我们仅知道交换机不仅是“无碰撞的”，而且也是名副其实的存储转发分组交换机就可以了；但是与运行在高至第三层的路由器不同，交换机仅运行在第二层。

1. 以太网帧结构

以太网帧如图 6-20 所示。通过仔细研究以太网的帧，我们能够学到许多有关以太网的知识。

图 6-20　以太网帧结构

为了将对以太网帧的讨论放到切实的环境中，考虑从一台主机向另一台主机发送一个 IP 数据报，且这两台主机在相同的以太局域网上（例如，如图 6-17 所示的以太局域网）。（尽管以太网帧的负载是一个 IP 数据报，但我们注意到以太网帧也能够承载其他网络层分组。）设发送适配器（即适配器 A）的 MAC 地址是 AA-AA-AA-AA-AA-AA，接收适配器（即适配器 B）的 MAC 地址是 BB-BB-BB-BB-BB-BB。发送适配器在一个以太网帧中封装了一个 IP 数据报，并把该帧传递到物理层。接收适配器从物理层收到这个帧，提取出 IP 数据报，并将该 IP 数据报传递给网络层。我们现在在这种情况下考察如图 6-20 所示的以太网帧的 6 个字段：

- *数据字段*（46～1500 字节）。这个字段承载了 IP 数据报。以太网的最大传输单元（MTU）是 1500 字节。这意味着如果 IP 数据报超过了 1500 字节，则主机必须将该数据报分片，如 4.3.2 节所讨论。数据字段的最小长度是 46 字节。这意味着如果 IP 数据报小于 46 字节，数据报必须被填充到 46 字节。当采用填充时，传递到网络层的数据包括 IP 数据报和填充部分。网络层使用 IP 数据报首部中的长度字段来去除填充部分。
- *目的地址*（6 字节）。这个字段包含目的适配器的 MAC 地址，即 BB-BB-BB-BB-BB-BB。当适配器 B 收到一个以太网帧，帧的目的地址无论是 BB-BB-BB-BB-BB-BB，还是 MAC 广播地址，它都将该帧的数据字段的内容传递给网络层；如果它收到了具有任何其他 MAC 地址的帧，则丢弃之。

- 源地址（6字节）。这个字段包含了传输该帧到局域网上的适配器的MAC地址，在本例中为AA-AA-AA-AA-AA-AA。
- 类型字段（2字节）。类型字段允许以太网复用多种网络层协议。为了理解这点，我们需要记住主机能够使用除了IP以外的其他网络层协议。事实上，一台给定的主机可以支持多种网络层协议，以对不同的应用采用不同的协议。因此，当以太网帧到达适配器B，适配器B需要知道它应该将数据字段的内容传递给哪个网络层协议（即分解）。IP和其他链路层协议（例如，Novell IPX或AppleTalk）都有它们各自的、标准化的类型编号。此外，ARP协议（在上一节讨论过）有自己的类型编号，并且如果到达的帧包含ARP分组（即类型字段的值为十六进制的0806），则该ARP分组将被多路分解给ARP协议。注意到该类型字段和网络层数据报中的协议字段、运输层报文段的端口号字段相类似；所有这些字段都是为了把一层中的某协议与上一层的某协议结合起来。
- CRC（4字节）。如6.2.3节中讨论的那样，CRC（循环冗余检测）字段的目的是使得接收适配器（适配器B）检测帧中是否引入了差错。
- 前同步码（8字节）。以太网帧以一个8字节的前同步码（Preamble）字段开始。该前同步码的前7字节的值都是10101010；最后一个字节是10101011。前同步码字段的前7字节用于“唤醒”接收适配器，并且将它们的时钟和发送方的时钟同步。为什么这些时钟会不同步呢？记住适配器A的目的是根据以太局域网类型的不同，分别以10Mbps、100Mbps或者1Gbps的速率传输帧。然而，没有什么是完美无缺的，因此适配器A不会以精确的额定速率传输帧；相对于额定速率总有一些漂移，局域网上的其他适配器不会预先知道这种漂移的。接收适配器只需通过锁定前同步码的前7字节的比特，就能够锁定适配器A的时钟。前同步码的第8个字节的最后两个比特（第一个出现的两个连续的1）警告适配器B，“重要的内容”就要到来了。

所有的以太网技术都向网络层提供无连接服务。这就是说，当适配器A要向适配器B发送一个数据报时，适配器A在一个以太网帧中封装该数据报，并且把该帧发送到局域网上，没有先与适配器B握手。这种第二层的无连接服务类似于IP的第三层数据报服务和UDP的第四层无连接服务。

以太网技术都向网络层提供不可靠服务。特别是，当适配器B收到一个来自适配器A的帧，它对该帧执行CRC校验，但是当该帧通过CRC校验时它既不发送确认帧；而当该帧没有通过CRC校验时它也不发送否定确认帧。当某帧没有通过CRC校验，适配器B只是丢弃该帧。因此，适配器A根本不知道它传输的帧是否到达了B并通过了CRC校验。（在链路层）缺乏可靠的传输有助于使得以太网简单和便宜。但是它也意味着传递到网络层的数据报流能够有间隙。

如果由于丢弃了以太网帧而存在间隙，主机B上的应用也会看见这个间隙吗？如我们在第3章中学习的那样，这只取决于该应用是使用UDP还是使用TCP。如果应用使用的是UDP，则主机B中的应用的确会看到数据中的间隙。另一方面，如果应用使用的是TCP，则主机B中的TCP将不会确认包含在丢弃帧中的数据，从而引起主机A的TCP重传。注意到当TCP重传数据时，数据最终将回到曾经丢弃它的以太网适配器。因此，从这种意义上来说，以太网的确重传了数据，尽管以太网并不知道它是正在传输一个具有全新数据的全新数据报，还是一个包含已经被传输过至少一次的数据的数据报。

历史事件

Bob Metcalfe 和以太网

作为20世纪70年代早期哈佛大学的一名博士生，Bob Metcalfe 在 MIT 从事 ARPAnet 的研究。在他学习期间，他还受到了 Abramson 有关 ALOHA 和随机接入协议工作的影响。在完成了他的博士学位，并在开始 Xerox Palo Alto 研究中心（Xerox PARC）的工作之前，他用3个月访问了 Abramson 和他在夏威夷大学的同事，获得了 ALOHAnet 的第一手资料。在 Xerox PARC，Metcalfe 受到了 Alto 计算机的影响，这种计算机在很多方面是20世纪80年代个人计算机的先驱。Metcalfe 看到了对这些计算机以一种不昂贵的方式组网的需求。因此，基于他在 APRAnet、ALOHAnet 和随机接入协议方面的知识，Metcalfe 和他的同事 David Boggs 一起发明了以太网。

Metcalfe 和 Boggs 的初始以太网运行速度为 2.94Mbps，连接长达一英里范围的多达256台主机。Metcalfe 和 Boggs 成功地使得 Xerox PARC 的大多数研究人员通过他们的 Alto 计算机互相通信。然后 Metcalfe 推进了 Xerox、Digital 和 Intel 联盟，创建了以太网作为一种 10Mbps 的以太网标准，该标准后被 IEEE 认可。Xerox 对以太网商业化没有表现出太多的兴趣。1979 年，Metcalfe 建立了自己的公司 3Com，它发展和商业化包括以太网技术在内的联网技术。特别是，3Com 在20世纪80年代早期为非常流行的 IBM PC 开发了以太网网卡并使之市场化。

2. 以太网技术

在以上的讨论中我们已经提到以太网，仿佛它有单一的协议标准似的。但事实上，以太网具有许多不同的特色，具有某种令人眼花缭乱的首字母缩写词，如 10BASE-T、10BASE-2、100BASE-T、1000BASE-LX 和 10GBASE-T。这些以及许多其他的以太网技术在多年中已经被 IEEE 802.3 CSMA/CD（Ethernet）工作组标准化了［IEEE 802.3 2012］。尽管这些首字母缩写词看起来眼花缭乱，实际上其中非常有规律性。首字母缩写词的第一部分指该标准的速率：10、100、1000 或 10G，分别代表 10Mbps、100Mbps、1000Mbps（或 1Gbps）和 10Gbps 以太网。“BASE”指基带以太网，这意味着该物理媒体仅承载以太网流量；几乎所有的 802.3 标准都适用于基带以太网。该首字母缩写词的最后一部分指物理媒体本身；以太网是链路层也是物理层的规范，并且能够经各种物理媒体（包括同轴电缆、铜线和光纤）承载。一般而言，“T”指双绞铜线。

从历史上讲，以太网最初被构想为一段同轴电缆。早期的 10BASE-2 和 10BASE-5 标准规定了在两种类型的同轴电缆之上的 10Mbps 以太网，每种标准都限制在 500 米长度之内。通过使用**转发器**（repeater）能够得到更长的运行距离，而转发器是一种物理层设备，它能在输入端接收信号并在输出端再生该信号。同轴电缆很好地对应于我们将作为一种广播媒体的以太网视图，即由一个接口传输的所有帧可在其他接口收到，并且以太网的 CSMA/CD 协议很好地解决了多路访问问题。节点直接附着在电缆上，万事大吉，我们有了一个局域网了！

多年来以太网已经经历了一系列演化步骤，今天的以太网非常不同于使用同轴电缆的初始总线拓扑的设计。在今天大多数的安装中，节点经点对点的由双绞铜线或光纤线缆构成的线段与一台交换机相连，如图 6-15 至图 6-17 所示。

在20世纪90年代中期，以太网被标准化为100Mbps，比10Mbps以太网快10倍。初始的以太网MAC协议和帧格式保留了下来，但更高速率的物理层被定义为用铜线（100BASE-T）和用光纤（100BASE-FX、100BASE-SX、100BASE-BX）。图6-21显示了这些不同的标准和共同的以太网MAC协议和帧格式。100Mbps以太网用双绞线距离限制为100米，用光纤距离限制为几千米，允许把不同建筑物中的以太网交换机连接起来。

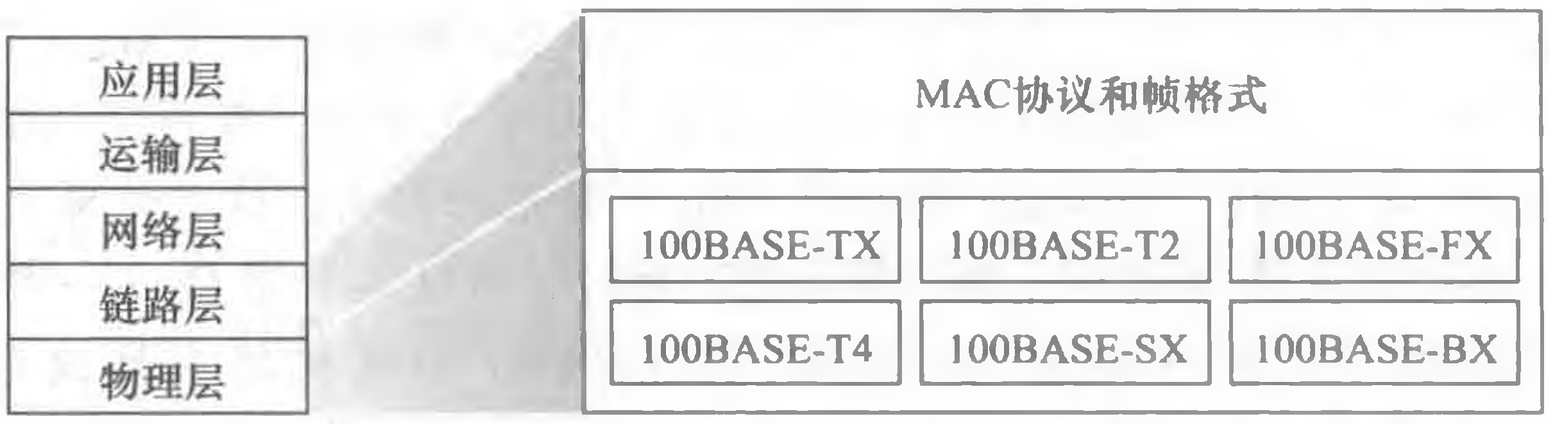

图6-21　100Mbps以太网标准：共同的链路层，不同的物理层

吉比特以太网是对极为成功的10Mbps和100Mbps以太网标准的扩展。40Gbps以太网提供40 000Mbps的总数据速率，与大量已经安装的以太网设备基础保持完全兼容。吉比特以太网的标准称为IEEE 802.3z，它完成以下工作：

- 使用标准以太网帧格式（参见图6-20），并且后向兼容10BASE-T与100BASE-T技术。这使得吉比特以太网和现已安装的以太网设备基础很容易集成。
- 允许点对点链路以及共享的广播信道。如前所述，点对点链路使用交换机，而广播信道使用集线器。在吉比特以太网术语中，集线器被称为“带缓存的分配器”。
- 使用CSMA/CD来共享广播信道。为了得到可接受的效率，节点之间的最大距离必须严格限制。
- 对于点对点信道，允许在两个方向上都以40Gbps全双工操作。

吉比特以太网最初工作于光纤之上，现在能够工作在5类UTP线缆上。

我们通过提出一个问题来结束有关以太网技术的讨论，这个问题开始可能会难倒你。在总线拓扑和基于集线器的星形拓扑技术时代，以太网很显然是一种广播链路（如6.3节所定义），其中多个节点同时传输时会出现帧碰撞。为了处理这些碰撞，以太网标准包括了CSMA/CD协议，该协议对于跨越一个小的地理半径的有线广播局域网特别有效。但是对于今天广为使用的以太网是基于交换机的星形拓扑，采用的是存储转发分组交换，是否还真正需要一种以太网MAC协议呢？如我们很快所见，交换机协调其传输，在任何时候决不会向相同的接口转发超过一个帧。此外，现代交换机是全双工的，这使得一台交换机和一个节点能够在同时向对方发送帧而没有干扰。换句话说，在基于交换机的以太局域网中，不会有碰撞，因此没有必要使用MAC协议了！

如我们所见，今天的以太网与Metcalfe和Boggs在30多年前构想的初始以太网有非常大的不同，即速度已经增加了3个数量级，以太网帧承载在各种各样的媒体之上，交换以太网已经成为主流，此时甚至连MAC协议也经常是不必要的了！所有这些还真正是以太网吗？答案当然是：“是的，根据定义如此。”然而，注意到下列事实是有趣的：通过所有这些改变，的确还有一个历经30年保持未变的持久不变量，即以太网帧格式。也许这才是以太网标准的一个真正重要的特征。

6.4.3　链路层交换机

到目前为止，我们有意对交换机实际要做的工作以及它是怎样工作的含糊其辞。交换