第6章 链路层和局域网

我们的目标:

- 理解链路层服务的主要功能:
  - 差错检查, 纠错
  - 共享一个广播信道: 多点接入问题(**multiple access**)
  - 链路层寻址(**link layer addressing**)
  - 局域网技术: **Ethernet, VLANs**
- 各种链路层技术的实现

主要内容

6.1 链路层概述

6.2 差错检测和纠正技术

6.3多路访问链路和协议

6.4 交换局域网

6.5 链路虚拟化(不讲)

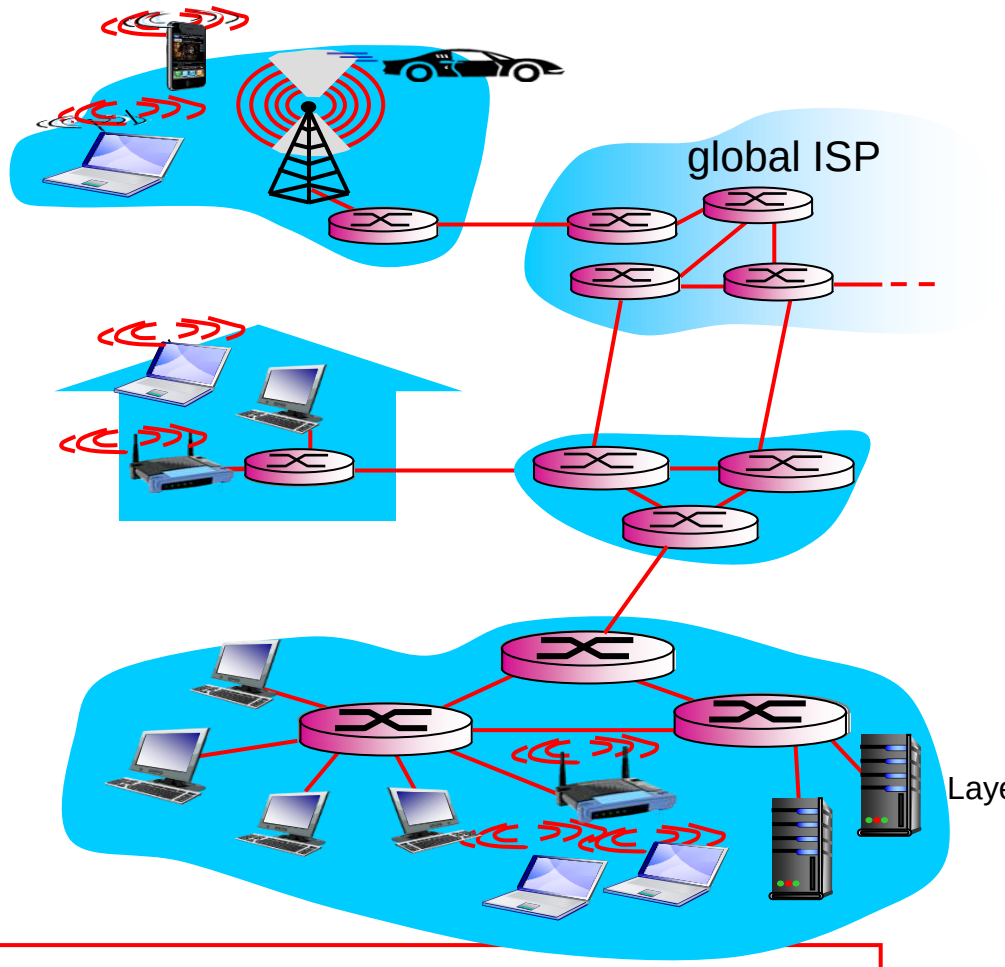
6.6 数据中心网络

6.7 回顾：web页面请求的历程

6.8 小结

6.1 链路层概述

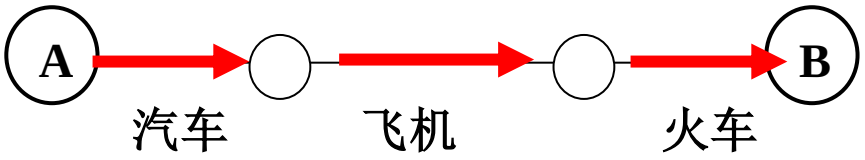
- 主机和路由器: **节点 (nodes)**
- 沿着通信路径连接相邻节点的通信信道： **链路 (links)**
  - 有线链路(wired links)
  - 无线链路(wireless links)
- 第二层的分组: **数据帧 (frame)**, 它是封装了的数据报



数据链路层的职责是将数据报从一个节点传送到与该节点直接有物理链路相连的另一个节点。

链路层的类比

- 数据报可以在不同的链路上，通过不同的链路层协议发送:
  - e.g., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- 每个链路层协议提供不同的服务:
  - e.g., 可以提供/也可以不提供可靠数据传输服务



运输的类比:

- 从学校到洛桑的旅程
  - 小汽车: 电子科大——双流机场
  - 飞机: 双流机场——日内瓦
  - 火车: 日内瓦——洛桑
- 游客 = datagram
- 分段旅程 = communication link
- 运输模式 = link layer protocol
- 旅行社代理 = routing algorithm

6.1.1 链路层提供的服务

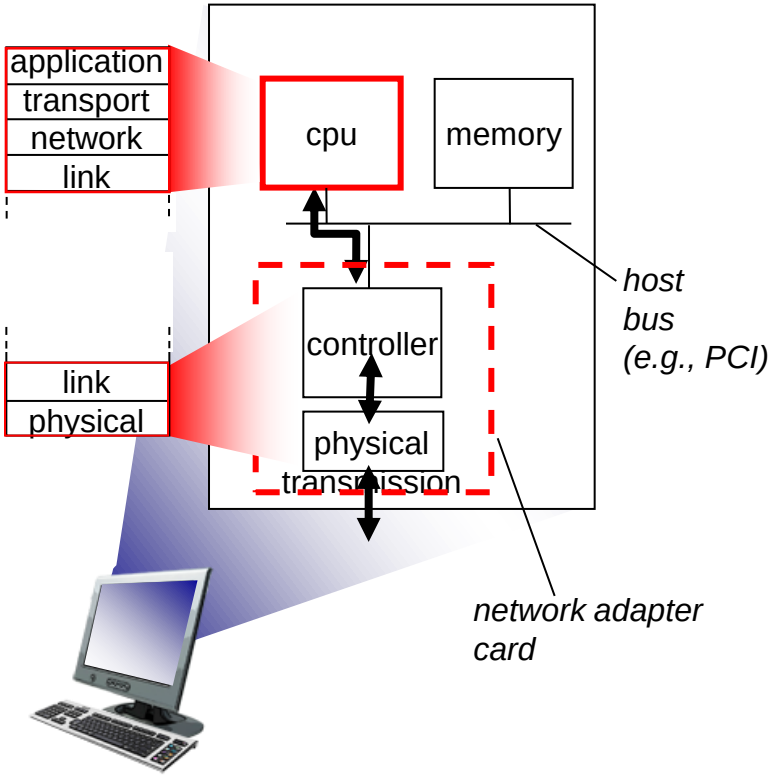
- **封装成帧，链路接入(framing, link access):**
  - 封装数据报为数据帧，增加头部，尾部信息
  - 如果是共享链路，接入链路
  - 在数据帧头部中，用MAC地址来标识源目的MAC地址
    - 不同于IP地址
- **在相邻节点之间可靠传输数据帧**
  - 我们在第3章已经学习了如何在运输层实现数据的可靠传输
  - 在比特错误率很低的链路(光纤、双绞线)很少使用
  - 无线链路：高比特错误率
    - 问题：为什么要在链路层和端到端都实现可靠传输?

6.1.1 链路层提供的服务

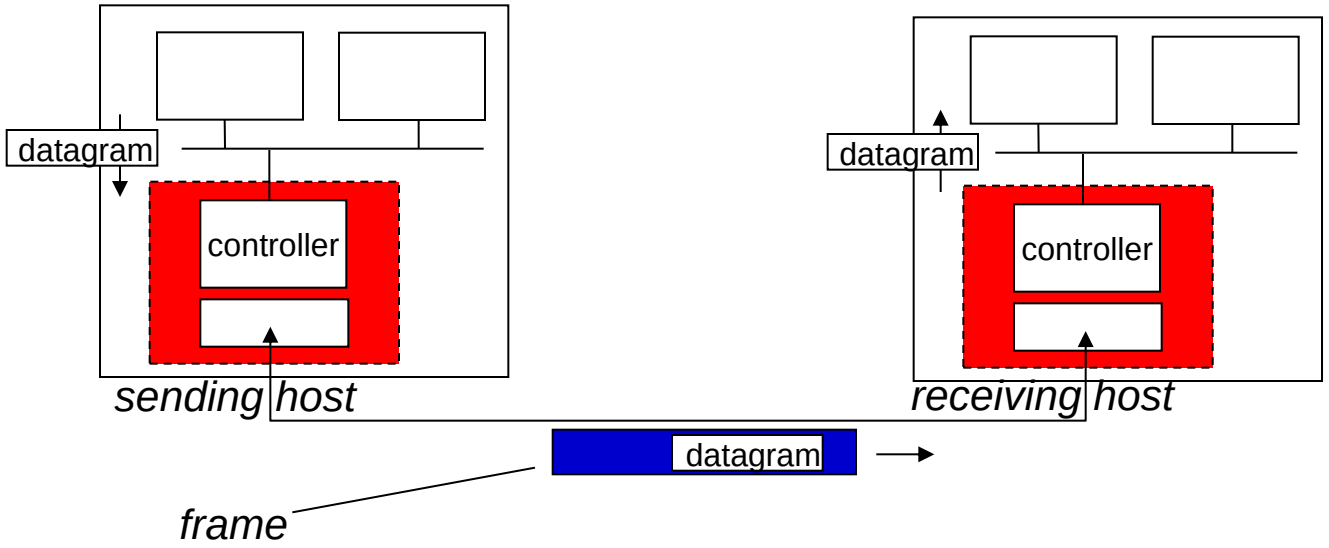
- **流量控制(flow control):**
  - 用于控制发送节点向直接相连的接收节点发送数据帧的频率
- **差错检查(error detection):**
  - 差错可能由信号衰减、噪声引入
  - 接收方检测是否出现错误:
    - 通知发送方重传或丢弃数据帧
- **错误纠正(error correction):**
  - 接收方标识和纠正比特错误，而不需要请求重传
- **半双工和全双工(half-duplex and full-duplex):**
  - 在半双工模式，链路的两个节点都可以发送数据，但是不能同时发送

6.1.2 链路层实现的位置

- 在主机和网络设备(路由器)上实现
- 在主机上，链路层的主体部分是在**网络适配器**上实现的(称为网卡)
  - 以太网卡，802.11卡；以太网芯片组
  - 实现链路层和物理层的功能
- 硬件、软件、固件的组合



网络适配器



发送方：

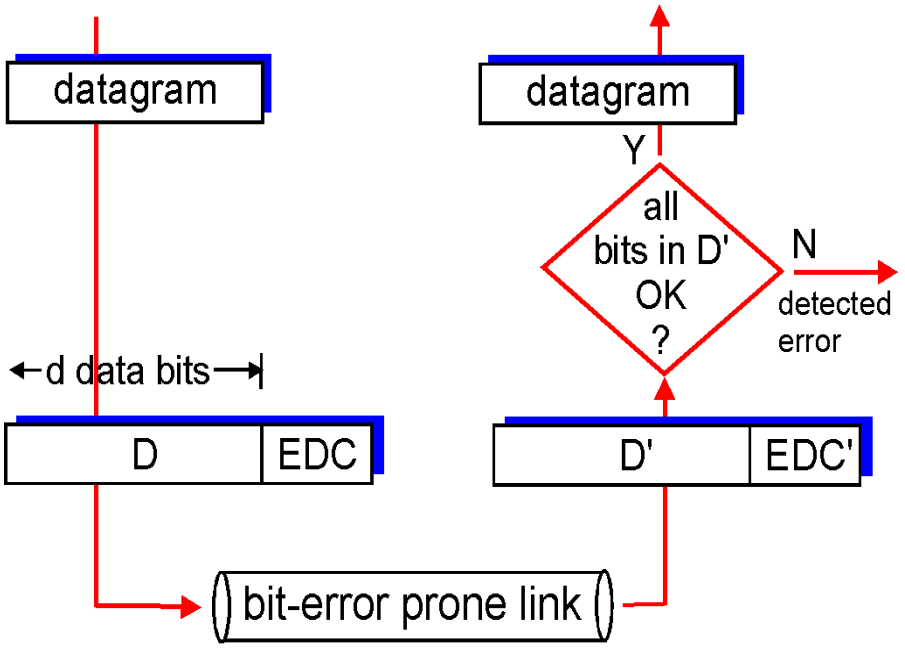
- 封装数据报为数据帧
- 增加差错检测比特，可靠数据传输，流量控制等机制。

接收方

- 执行检查错误、可靠数据传输、流量控制等机制
- 抽取数据报，将其递交给上层

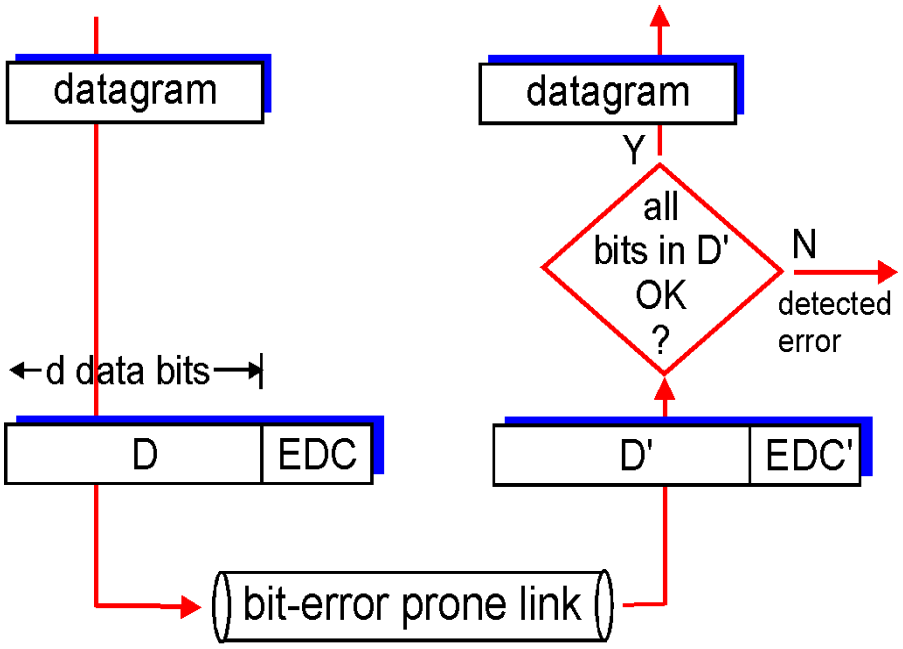
6.2 差错检测和纠错技术

- **比特级差错检测和纠错**
  - 对一个节点发送到一个相邻节点的帧，检测是否出现比特差错，并纠正。
  - 相关技术很多。
  - 差错检测和纠错的过程
- **差错检测并非100%可靠**
  - 协议可能丢失一些错误
  - 差错校验位越多，检测和纠正功能越好



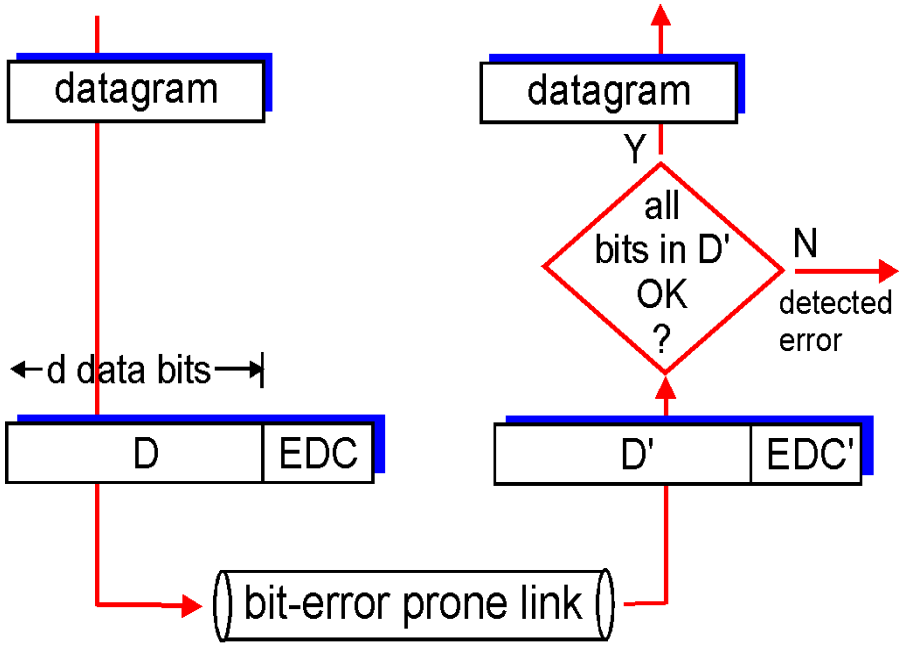
6.2 差错检测和纠错技术

- **发送节点**
  - 将数据D附加若干差错检测和纠错位EDC，一起发送到链路。
  - 数据D包括网络层传来的数据报，以及链路级寻址信息、序列号和其他字段。
  - 保护范围包括数据D的所有字段。



6.2 差错检测和纠错技术

- **接收节点**
  - 接收比特序列D'和EDC'。
  - 如果发生传输比特错误（0→1，1→0），D'和EDC'可能与发送的D和EDC不同。
  - 接收方根据D'和EDC'，判断D'是否和初始的D相同（D的传输是否正确）。
- 正确：解封取出数据报，交给网络层；
- 出错：差错处理。



说明

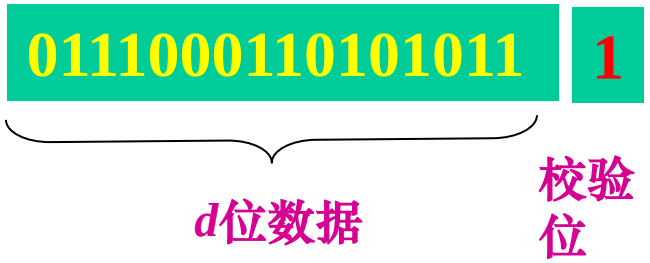
- 差错检测和纠正技术不能保证接收方检测到所有的比特差错，即*可能出现未检测到的比特差错*，而接收方并未发现。
- 选择一个合适的差错检测方案使未检测到的情况发生的概率很小即可。
- 差错检测和纠错技术越好，越复杂，开销更大。

三种主要差错检测技术

- **奇偶校验**：最基本的方法。
- **检查和方法**：常用于运输层。
- **循环冗余检测**：常用于链路层。

6.2.1 一比特奇偶校验

偶校验



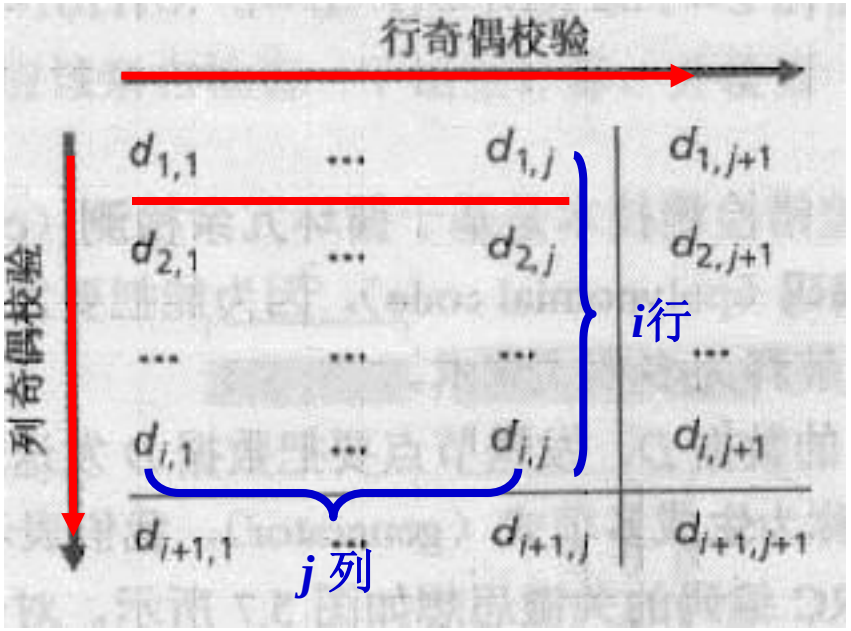
- **发送方：**
  - 在要发送的信息D（d位）后面附加一个奇偶校验位
  - 使“1”的个数是奇数（奇校验）或偶数（偶校验）
  - 一起传输发送（d+1位）。
- **接收方：**
  - 检测收到的信息（d+1位）中“1”的个数。
  - 偶校验：发现奇数个“1”，至少有一个比特发生差错（奇数个比特差错）。
  - 奇校验：发现偶数个“1”，至少有一个比特发生差错。

6.2.1 一比特奇偶校验

- 可以查出任意奇数个错误，但不能发现偶数个错误。
- 若比特差错概率很小，差错独立发生，一比特奇偶校验可满足要求。
- 若差错集中一起“突发”（突发差错），一帧中未检测到的差错的概率达到50%。

二维奇偶校验

- **基本思想：**
  - 将要传信息D（d比特）划分为i行j 列（i 个组，每组 j 位）；
  - 对每行和每列分别计算奇偶值；
  - 结果的i+j+1个奇偶比特构成了帧的差错检测比特。



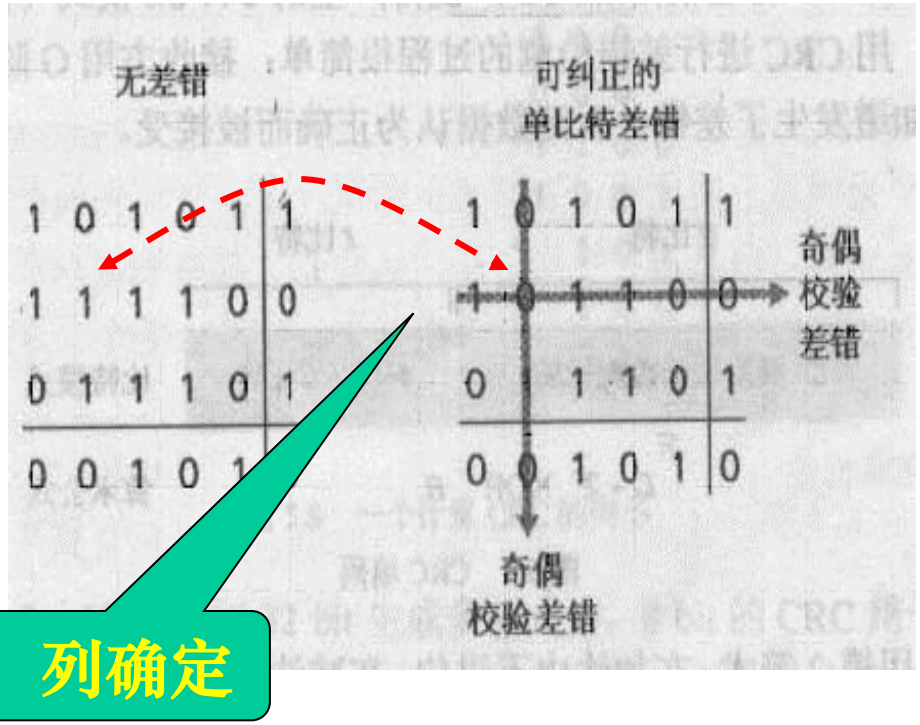
二维奇偶校验的例子

要发送的数据比特10101 11110 01110，

划分3组，每组5个比特。进行行、列偶校验

特点：

- 可以检测并纠正单个比特差错（数据或校验位中）。
- 能够检测(但不能纠正)分组中任意两个比特的差错。



6.2.2 检查和方法

发送方：

- 将数据的每两个字节当作一个16位的整数，可分成若干整数；
- 将所有16位的整数求和；
- 对得到的和逐位取反，作为检查和，放在报文段首部，一起发送。

接收方：

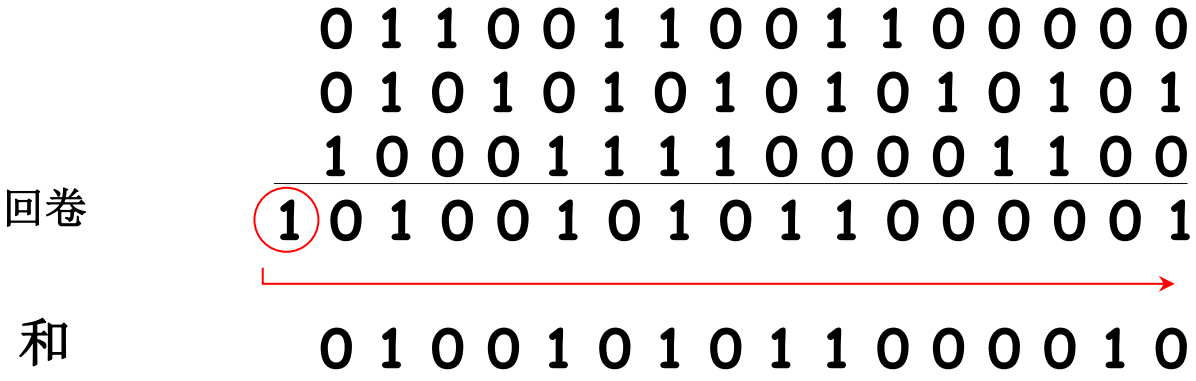
对接收到的信息（包括检查和）按与发送方相同的方法求和。

- 全“1”：收到的数据无差错；
- 其中有“0”：收到的数据出现差错。

或者核对计算的检查和是否等于检查和字段的值。

6.2.2 检查和的例子

- 注意
  - 当数字作加法时，最高位的进位要回加到结果中。
- 例，有三个16 比特的字：



检查和（取反） 1 0 1 1 0 1 0 1 0 0 1 1 1 1 0 1

无差错，和为： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1(3个16比特数据+检查和16比特)

6.2.2 检查和的特点

- 分组开销小：检查和位数比较少；
- 差错检测能力弱：
- 适用于运输层（差错检测用软件实现，检查和方法简单、快速）。
- 链路层的差错检测由适配器中专用的硬件实现，采用更强的CRC方法。

6.2.3 循环冗余检测

10111 → x^4+x^2+x+1

计算机网络中广泛采用

- **循环冗余检测CRC (cyclic redundancy check)码：**
  - 即多项式编码，把要发送的比特串看作为系数是0或1的一个多项式，对比特串的操作看作为多项式运算。
- **基本思想：**
  - 设发送节点要把数据D（d 比特）发送给接收节点。
  - 发送方和接收方先共同选定一个生成多项式G（r+1比特），最高有效位(最左边)是1。

循环冗余检测的基本思想

- **发送方：**
  - 计算出一个r位附加比特R，添加到D的后面产生DR（d+r 比特）
  - DR能被生成多项式G模2运算整除，一起发送。
- **接收方：**
  - 用生成多项式G去除接收到的DR（d+r比特）
    - 余数非0：传输发生差错；
    - 余数为0：传输正确，去掉尾部r位，得所需数据D。

D：要发送的数据（d位）　R：CRC校验（r位）

DR（d+r位）

什么是模2运算

- 加法不进位，减法不借位，即操作数的按位异或 (XOR)

例

1011 XOR 0101=1110; 1011-0101=1110

1001 XOR 1101=0100; 1001-1101=0100

- 乘法和除法与二进制运算类似，其中加法或减法没有进位或借位
- 乘以2^r，即比特模式左移r个位置。

$$D\times 2^r \text{ XOR } R = D\,00\cdots00 \text{ XOR } R$$

$$= DR \quad (d+r \text{ 比特})$$

计算R（CRC编码）

- DR能被G模2运算整除：即

$$D \times 2^r \ \text{XOR} \ R = nG$$

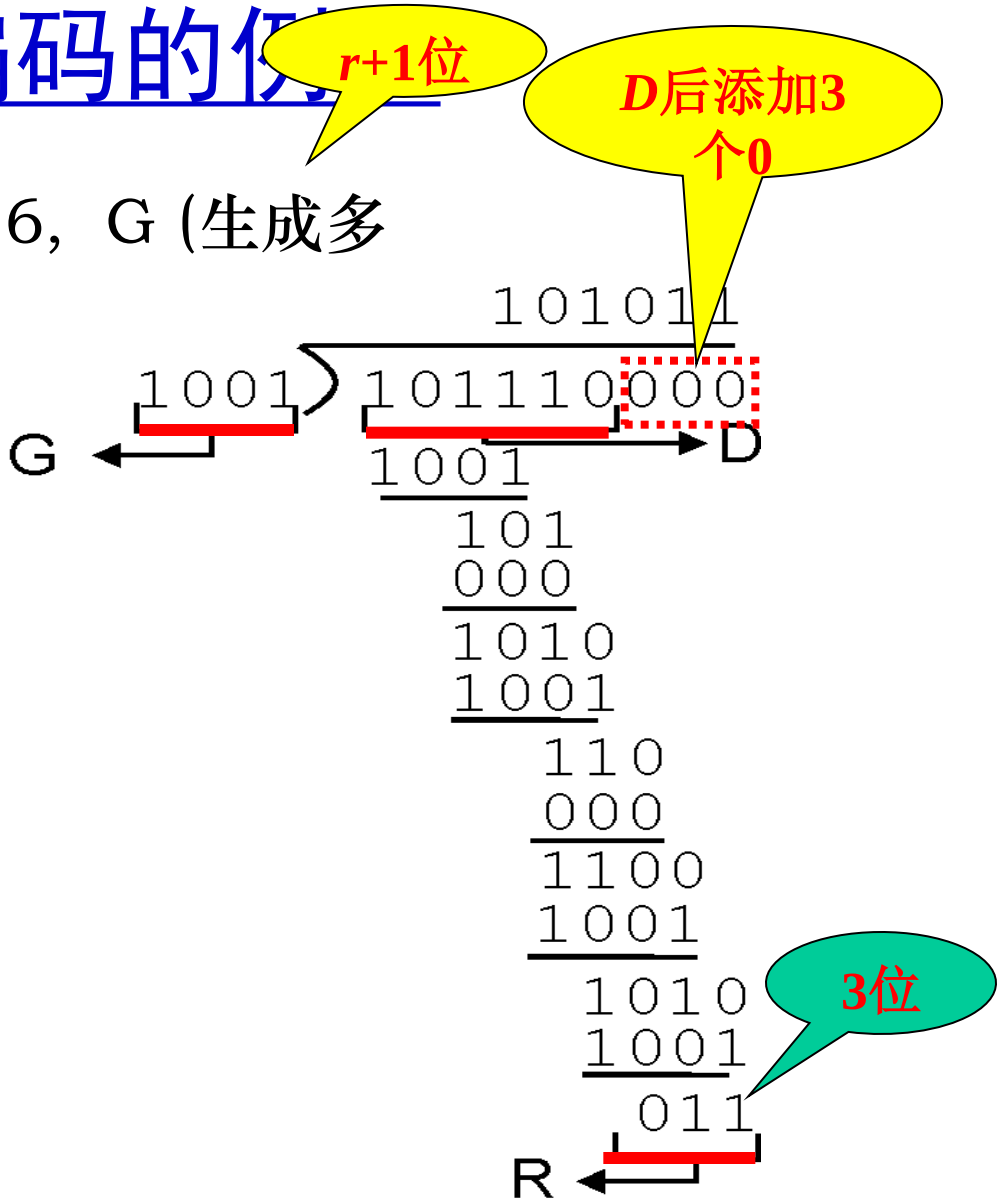
- 等式两边都用R异或，得到

$$D \times 2^r = nG \ \text{XOR} \ R$$

- 即用G来除$D \times 2^r$，余数值刚好为R。
- **R的计算：将数据D后面添加r个0，除以给定的生成多项式G，所得余数即为R（r位）。**

CRC编码的例

设 (数据) D = 101110,d = 6，G (生成多项式) = 1001，r = 3



实际传输的数据形式是： 101110011

循环冗余码CRC的特点

- 生成多项式G的选择：有8、12、16和32 比特生成多项式G。
  - 8 比特的CRC用于保护ATM信元首部；
  - 32 比特的标准CRC-32用于链路级协议：

  GCRC-32 = 100000100110000010001110110110111
- CRC特点：

  能检测小于 r+1 位的突发差错、任何奇数个差错。

课堂练习

- **教材P5**

P5. Consider the 5-bit generator, G=10011, and suppose that D has the value 1010101010. What is the value of R?

6.3 多路访问链路和协议

- 两种网络链路：
  - *点对点链路：*链路两端各一个节点。一个发送和一个接收。如，点对点协议PPP。
  - *广播链路：*多个节点连接到一个共享的广播信道。

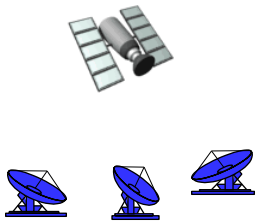
广播：任何一个节点传输一帧时，信号在信道上广播，其他节点都可以收到一个拷贝。

常用于局域网LAN中，如早期的以太网和无线局域网。

本节主要学习广播链路的信道共享技术。

shared wire (e.g., cabled Ethernet)

shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

广播信道要解决问题

- 传统的广播电视：是单向的广播，一个固定的节点向许多接收节点发送。
- 计算机网络：广播信道上的节点都能够发送和接收。

好比许多人聚集在一起交谈（空气是广播媒体）

解决“谁在什么时候获得说话权力”(向信道发送)

- 多路访问问题：如何协调多个发送和接收节点对共享广播信道的访问。相关技术是多路访问协议。

多路访问协议

- **目的：协调多个节点在共享广播信道上的传输。**
  - **避免多个节点同时使用信道，发生冲突（碰撞），产生互相干扰。**
- ***冲突（collide）：*两个以上的节点同时传输帧，使接收方收不到正确的帧（所有冲突的帧都受损丢失）。**
  - **造成广播信道时间的浪费。**
  - **多路访问协议可用于许多不同的网络环境，如有线和无线局域网、卫星网等。**

多路访问协议类型（三类）

- 信道划分协议
  - 把信道划分为小“片”（时隙）
  - 给节点分配专用的小“片”
- 随机访问协议
  - 不划分信道，允许冲突
  - 能从冲突中“恢复”
- 轮流协议
  - 通过轮流访问信道避免冲突，要发送的节点越多，轮流时间越长

多路访问协议的理想特性

设广播信道的速率为R(b/s)

- 只有一个节点发送数据时：该节点的吞吐量为R (b/s)；
- 有M个节点发送数据时：每个节点吞吐量为R/M (b/s)；
- 协议是分散的：不需要主节点协调传输；
- 协议是简单的：实现方便、价格适中。

6.3.1 信道划分协议

6.3.2 随机访问协议

6.3.3 轮流协议

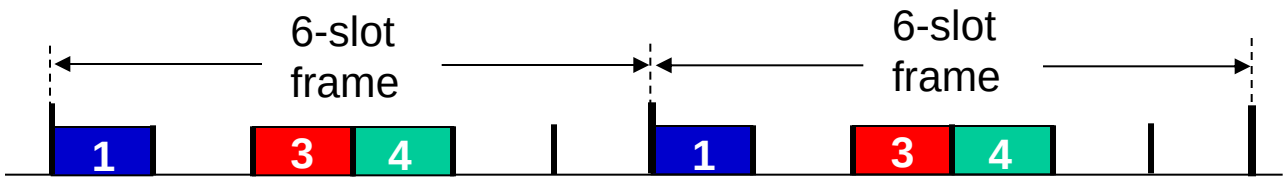
6.3.1 信道划分协议

主要有TDMA、FDMA、CDMA三种。

设信道支持*N*个节点，传输速率是*R*(b/s)。

- 时分多路接入 (time division multiple access):
  - 将时间划分为*时间帧*，每个时间帧再划分为*N个时隙*（长度保证发送一个分组），分别分配给*N*个节点。
  - 每个节点只在固定分配的时隙中传输。

例：6个站点的LAN, 时隙1、3、4 有分组, 时隙2、5、6 空闲



TDMA特点

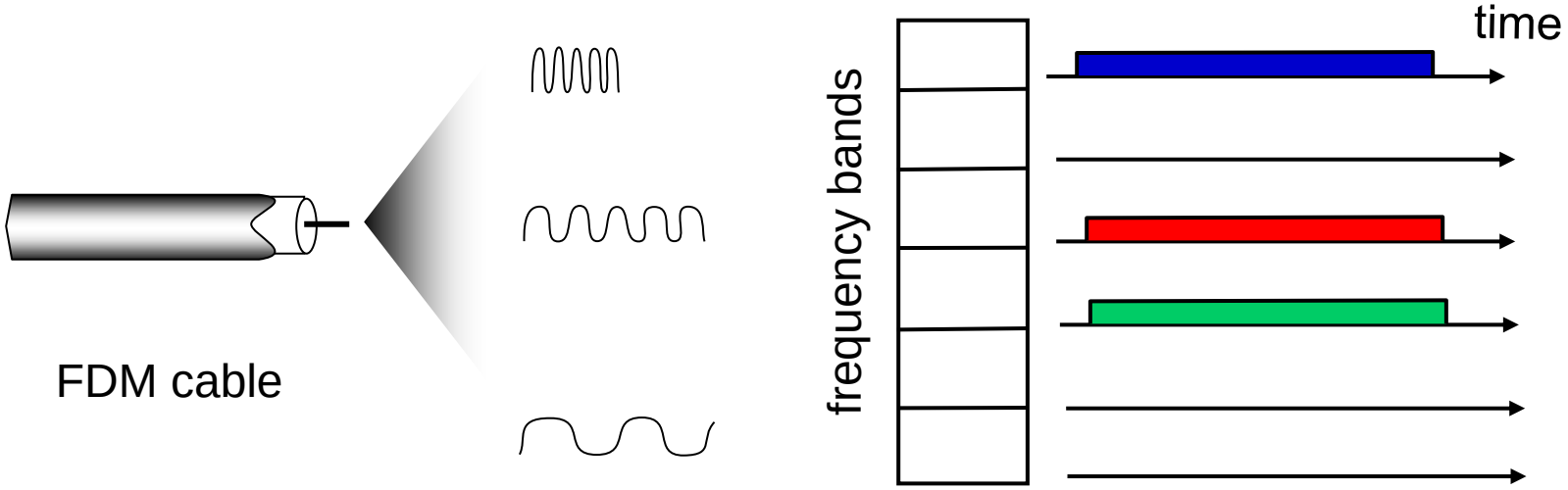
- 避免冲突、公平：每个节点专用速率R/N b/s。
- 节点速率有限：R/N b/s；
- 效率不高：节点必须等待它的传输时隙。

- 频分多路接入 (frequency division multiple access):

将总信道带宽R b/s划分为N个较小信道(频段，带宽为R/N)，分别分配给N个节点。

例：6个站点的LAN, 频带1、3、4 有分组，频带2、5、6 空闲

- 特点：与TDMA类似。
  - 避免冲突、公平：N个节点公平划分带宽；
  - 节点带宽有限、效率不高：节点带宽为R/N。

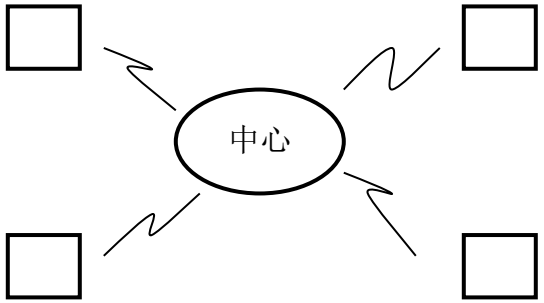


6.3.2 随机接入协议

- 基本思想:
  - 发送节点以信道全部速率（R b/s）发送;
  - 发生冲突时，冲突的每个节点分别等待一个随机时间，再重发，直到帧(分组)发送成功。
- 典型随机访问协议:
  - ALOHA协议
  - 载波监听多路访问CSMA协议
  - 带冲突检测载波监听多路访问CSMA/CD

ALOHA

- ALOHA：夏威夷大学研制的一个无线电广播通信网（20世纪70年代初）。采用星型拓扑结构，使地理上分散的用户通过无线电来使用中心主机。
  - 中心主机通过下行信道向二级主机广播分组；
  - 二级主机通过上行信道向中心主机发送分组（可能会冲突，无线电信道是一个公用信道）。
- 若干种形式：
  - 时隙ALOHA
  - 纯ALOHA



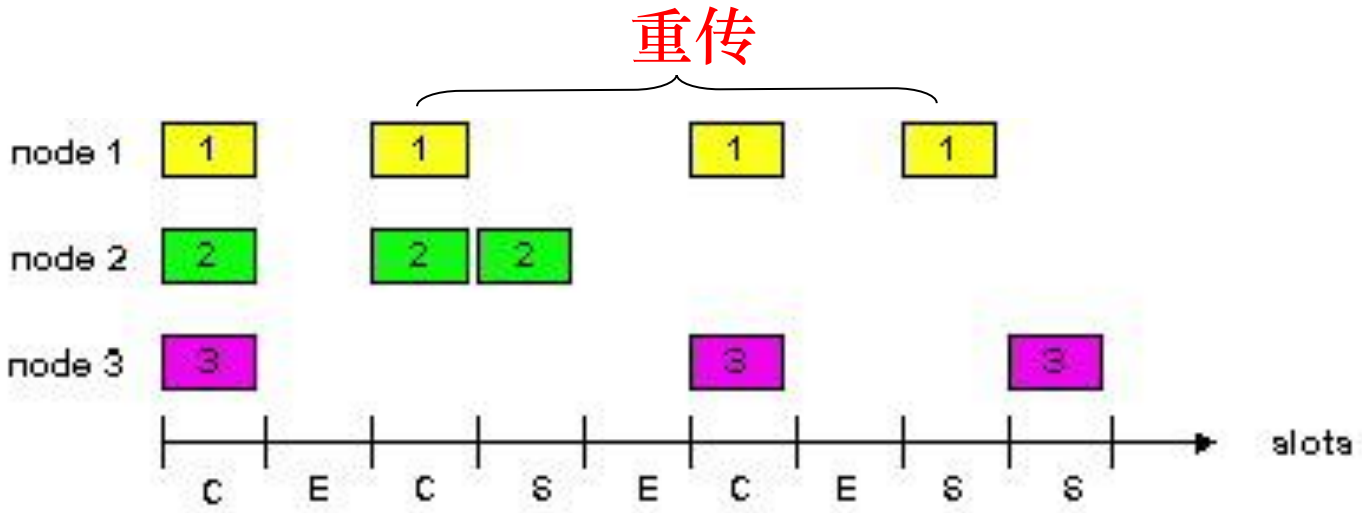
1、时隙ALOHA

假设：

- 所有的帧长L比特；
- 时间被划分为若干等长的时隙（长度为一帧的传输时间 L/R s）；
- 节点只在时隙的开始点传输帧；
- 所有节点同步传输，都知道时隙什么时候开始；
- 如果一个时隙有多个节点同时传送，所有节点都能检测到冲突。

时隙ALOHA操作过程

- 当节点有新的帧要发送，需等到下一个时隙开始，才传输整个帧。
- *无冲突：*节点成功传输帧。
- *有冲突：*节点检测到冲突后，以概率*p*在后续的每一个时隙重传该帧，直到成功。

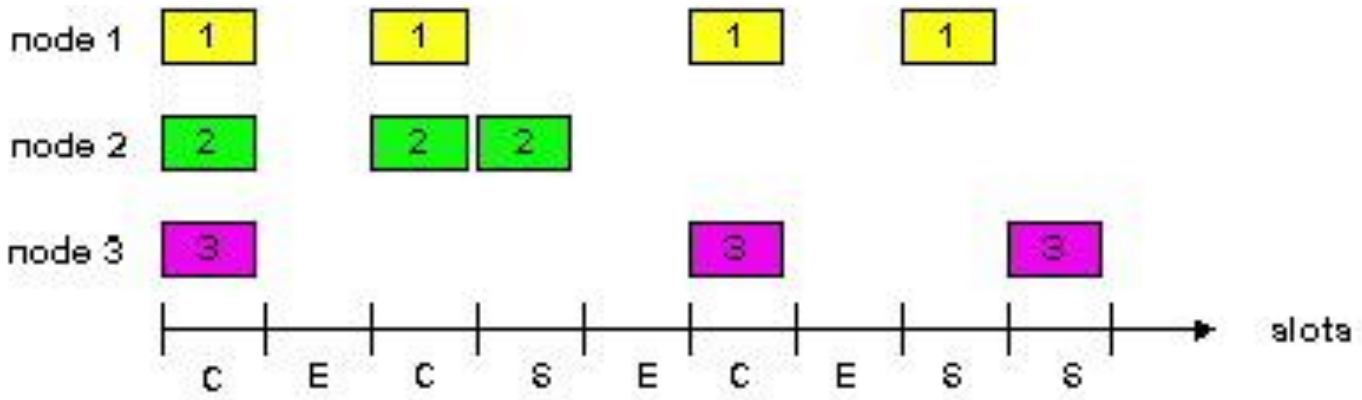


时隙ALOHA

- 特点：
  - 当只有一个活动节点（有帧要发送）时，以全速 R 连续传输；
  - 分散的：每个节点检测冲突并独立决定何时重传；
  - 发送控制简单；
  - 有多个活动节点时效率低。

效率计算

- 有三种可能时隙
  - *冲突时隙C：* 出现帧冲突，被“浪费”。
  - *空闲时隙E：* 所有活动节点停止传输，被“浪费”。
  - *成功时隙S：* 只有一个节点在传输的时隙。



效率计算

- 假设：
  - 有N个节点；
  - 每个节点都有一帧（新帧或重传帧）要发送，试图在每个时隙以概率 p 传输。
- 成功时隙的概率：只有一个节点传输而其他N-1个节点不传输的概率。
  - 若一个节点传输的概率是 p ，剩余的节点不传输的概率是$(1-p)^{N-1}$。
  - 一个给定的节点成功传送的概率是：$p(1-p)^{N-1}$

时隙ALOHA的效率

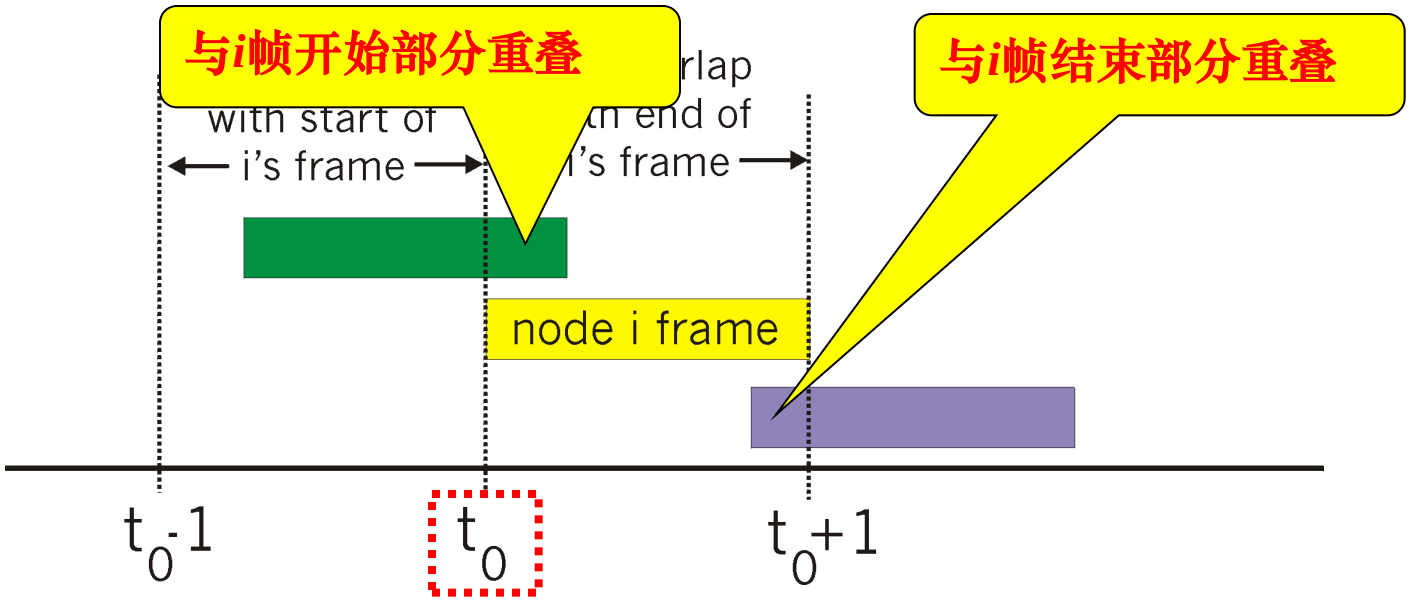
- N个节点中，任意节点成功传送的概率：$Np(1-p)^{N-1}$
- 取极限后，*最大效率为：$1/e = 0.37$*。
  - 即当许多节点都有很多帧要传输时，最多只有37%的时隙在成功传输，信道有效的传输速率是0.37*R (b/s)。
- 类似分析得出：37%的时隙空，26%的时隙有冲突。

2、纯ALOHA

- ALOHA的最初形式是一个非时隙、完全分散的协议。
- 工作过程：
  - 节点有数据帧要发送，就立即传输。
  - 如果与其他数据帧产生冲突，在该冲突帧传完之后
    - 以概率 p 立即重传该帧；
    - 或以概率 $(1-p)$ 等待一个帧的传输时间，再以概率 p 传输该帧。

纯ALOHA最大效率

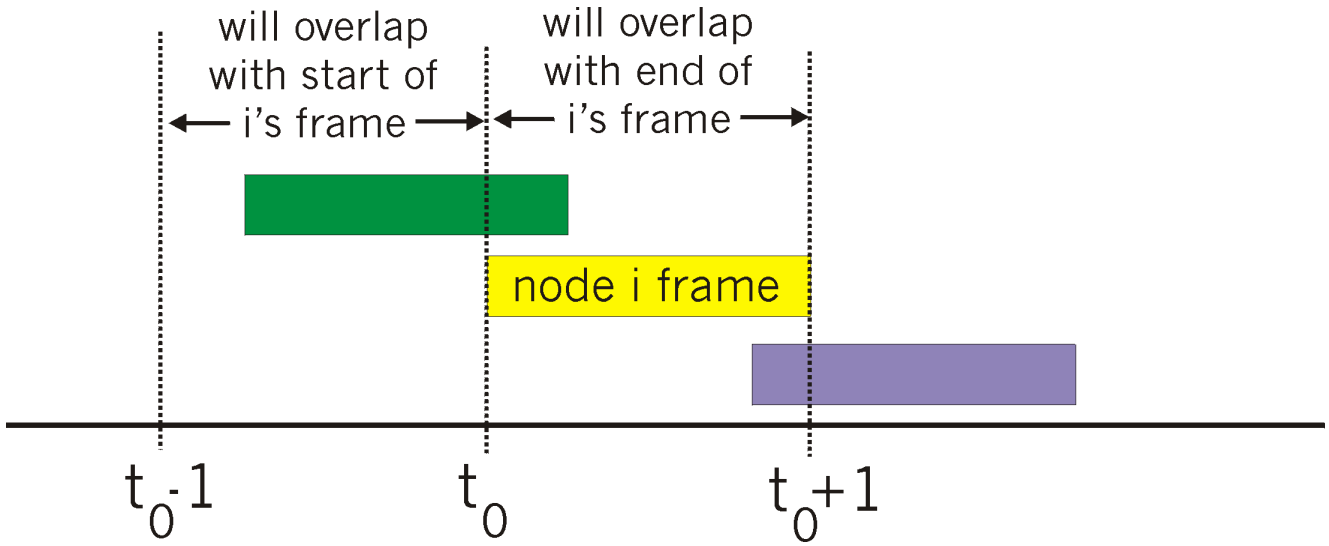
- 假设：
  - 帧传输时间为一个时间单元。
  - 任何给定时间，一个节点传输一帧的概率是 p。
  - 节点 i 在时间 t_0 开始传输帧，如图所示。
- 结果：在t_0发送的帧会和在 $[t_0-1,t_0+1]$的发送的其它帧冲突



- 保证帧成功传输：
  - 在时间间隔[t_0 -1，t_0]中，不能有其他节点开始传输。

    其他节点没有开始传输的概率是$(1-p)^{N-1}$
  - 当节点i传输时，在时间间隔[t_0 ，t_0+1]中，其他节点不能开始传输。

    其他节点没有开始传输的概率是$(1-p)^{N-1}$。



纯ALOHA效率

P(给定节点成功传送) = P(节点传送) ·
P(没有其他节点在$[t_0-1,t_0]$内传送) ·
P(没有其他节点在$[t_0,t_0+1]$内传送)

$= p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1}$

$= p \cdot (1-p)^{2(N-1)}$

取极限为 $1/(2e) = 0.18$。

只有时隙ALOHA协议的一半。

ALOHA协议的特点

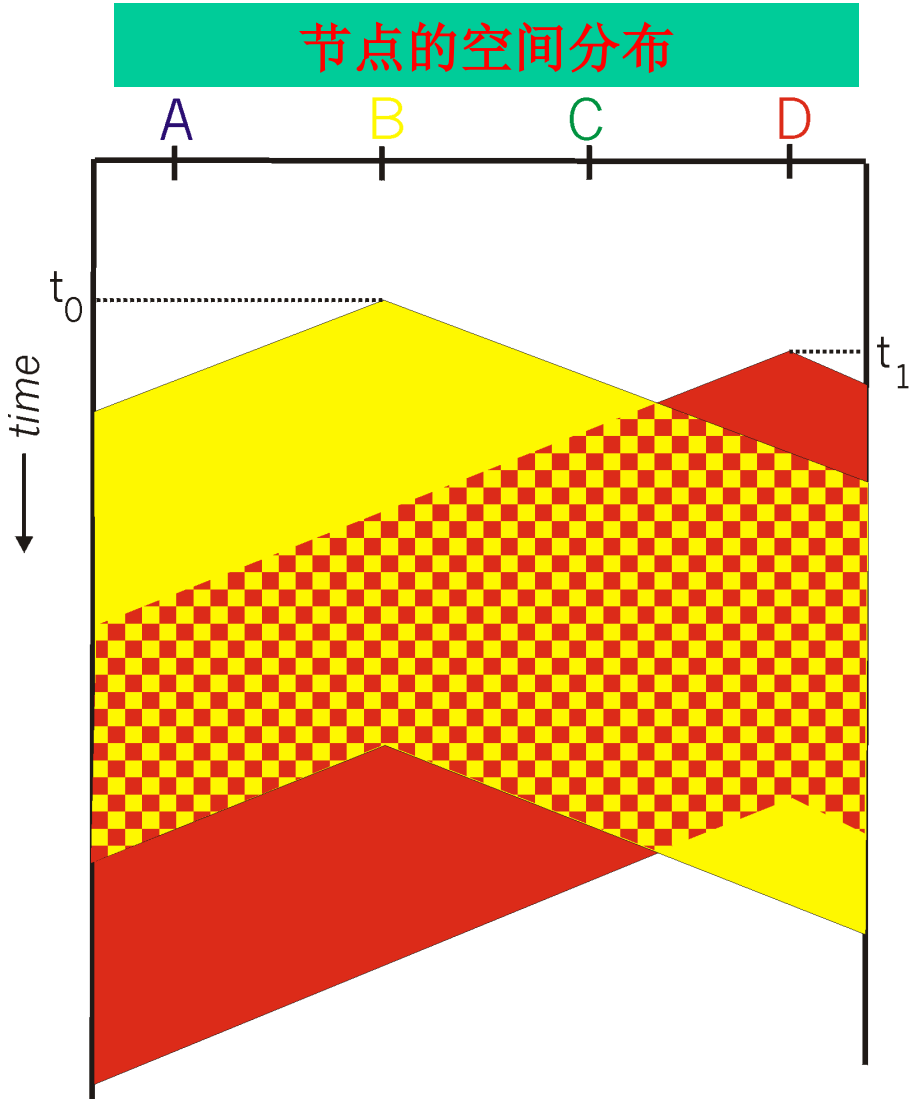
- 每个节点的传输与广播信道上其他节点的活动是相互独立的。
- 一个节点开始传输时并不知道是否有其他节点正在传输；
- 发生冲突时不会停止传输。
- 效率不高。

3、载波侦听多路访问CSMA

- **载波侦听**：某个节点在发送之前，先监听信道。
  - 信道忙：有其他节点正往信道发送帧，该节点随机等待（回退）一段时间，然后再侦听信道。
  - 信道空：该节点开始传输整个数据帧。
- **人类类比: 自己说话之前，先听一下有没有其他人正在说话，不要打断他人说话!**
- **CSMA 的特点：**
  - 发前监听，可减少冲突。
  - 由于传播时延的存在，仍有可能出现冲突，并造成信道浪费。

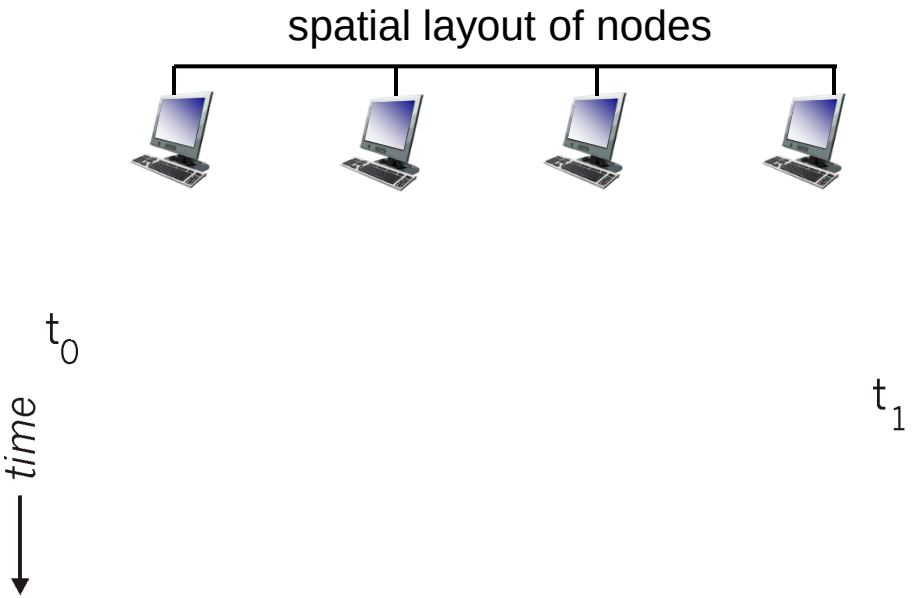
例子

- 一个广播总线连接4个节点(A、B、C、D)传输的时空图。



- 时间t_0：节点B侦听到信道空，开始传输帧，沿着媒体传播比特。
- 时间t_1（$t_1 > t_0$）：节点D有帧要发送。B的传输信号未到D，D检测到信道空，开始传输。很快，B的传输开始在D节点干扰D的传输(冲突)

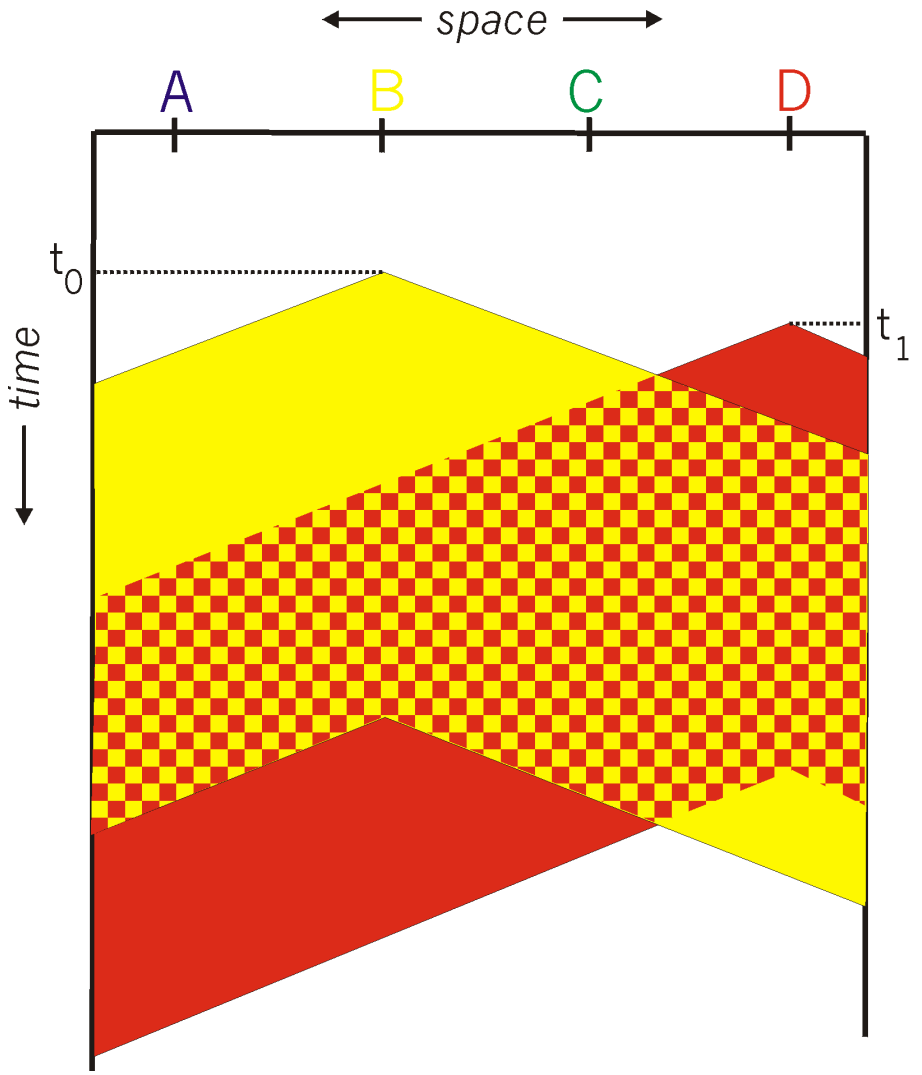
端到端信道传播时延：信号从一个节点到另一个节点所花费的传播时间。传播时延越长，节点不能侦听到另一个节点已经开始传输的可能性越大。



带来问题：信道浪费

- 节点没有进行冲突检测，既使发生了冲突，节点仍继续传输它们的帧。但该帧已经被破坏、是无用的帧，信道传输时间被浪费。

注意：距离与传播时延对碰撞概率的影响。



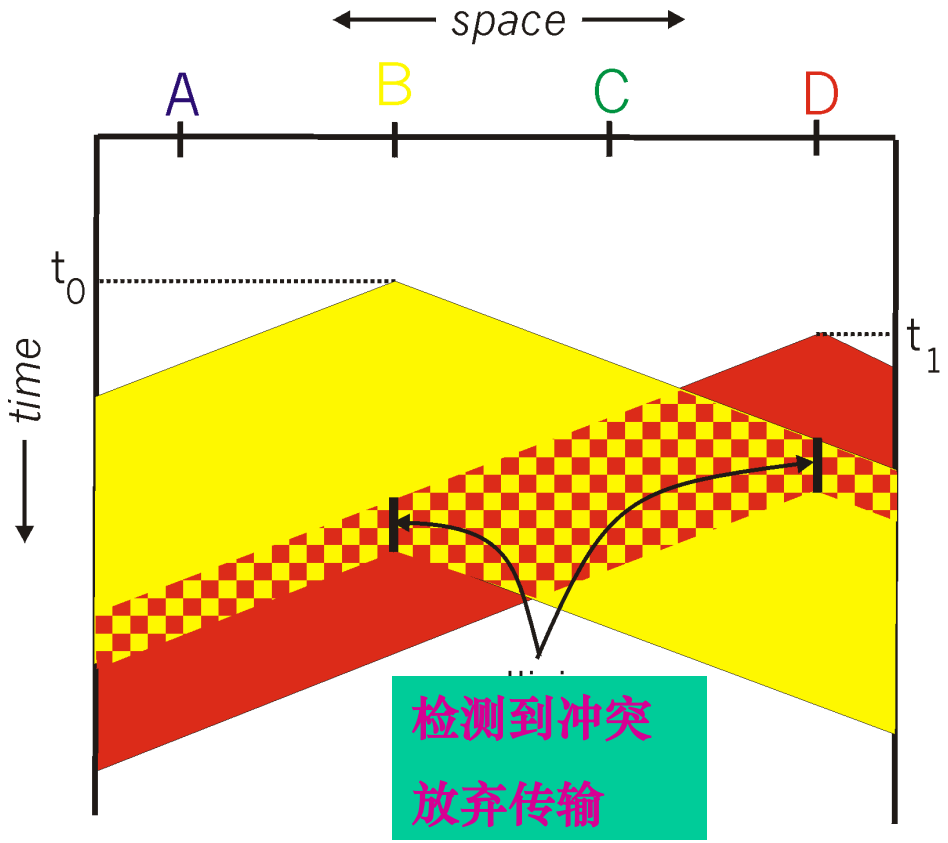
4、具有碰撞检测的载波侦听多路访问(CSMA/CD)

- 增加“载波侦听”和“冲突检测”两个规则。
  - “先听后说”(listen before talk)
  - “边说边听”(listen while talk)
- 基本原理：传送前侦听
  - 信道忙：延迟传送；
  - 信道闲：传送整个帧；
  - 发送同时进行冲突检测：一旦检测到冲突就立即停止传输，尽快重发。
- 目的：缩短无效传送时间，提高信道的利用率。

例子

两个节点B、D在检测到冲突之后很短的时间内都放弃传输

以太网即采用
CSMA/CD协议



以太网CSMA/CD的运行机制

- 适配器从网络层获得一个数据报，封装成帧，准备发送；
- 如果适配器侦听到信道空闲，开始传输帧；如果检测到信道繁忙，将等待一段时间，直到侦听到信道空闲，开始传输帧；
- 在传输过程中，适配器会同时监听是否有其他适配器的信号能量；
- 如果适配器在整个帧的传输过程中，没有监听到其他信号，则完成该帧的传输；如果监听到来自其他适配器的信号，则中止传输帧；
- 中止传输后，适配器会等待一个随机时间，重新执行步骤2

回退时间的讨论

- 以太网的**二进制指数回退**
  - 当传输一个给定帧时，在该帧经历了一连串的n次碰撞后，结点随机地从{0,1,2…. 2^n-1}中选择一个K值，NIC waits K·512 bit times
  - 假设是100Mbps的以太网，那么发送512bit的时间是5.12*10-6秒
    - 第一次碰撞：{0, 1}
    - 第二次碰撞：{0, 1, 2, 3}
    - 第三次碰撞：{0, 1, 2, 3, 4, 5, 6, 7}
    - ……

K是等概率选择

6.3.3 轮流协议

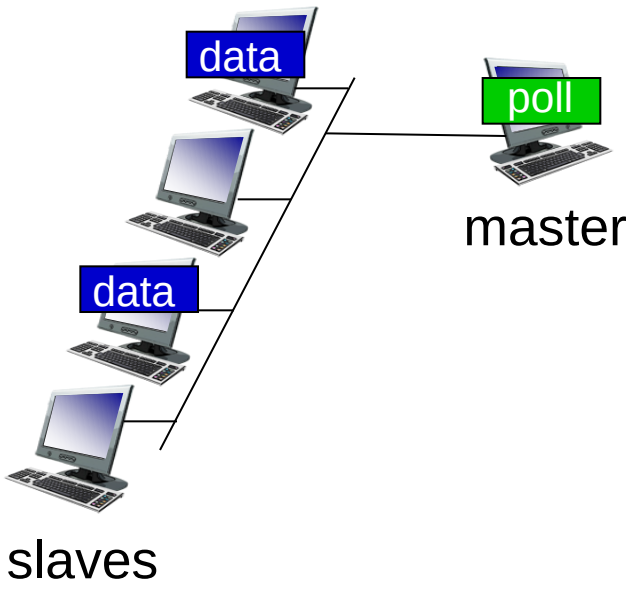
- 多路访问协议理想特性:
  - 只有一个节点活动时，吞吐量R b/ s;
  - 有M个节点活动时，吞吐量 R/M b/ s。
  - ALOHA和CSMA协议有第一个特性，但没有第二个特性(公平性)

1、轮询协议

2、令牌传递协议

轮询协议(polling)

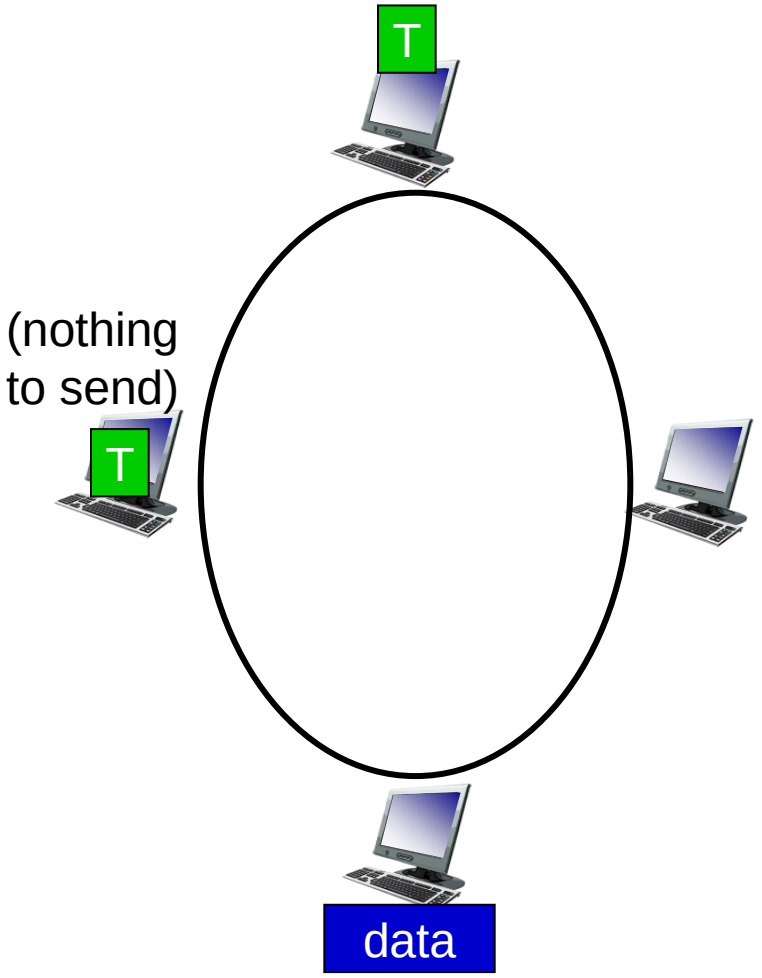
- 主节点“邀请”从节点依次传送
- 问题:
  - 轮询的开销
  - 延时
  - 单点故障(主节点)



令牌传递协议

令牌传递(token passing):

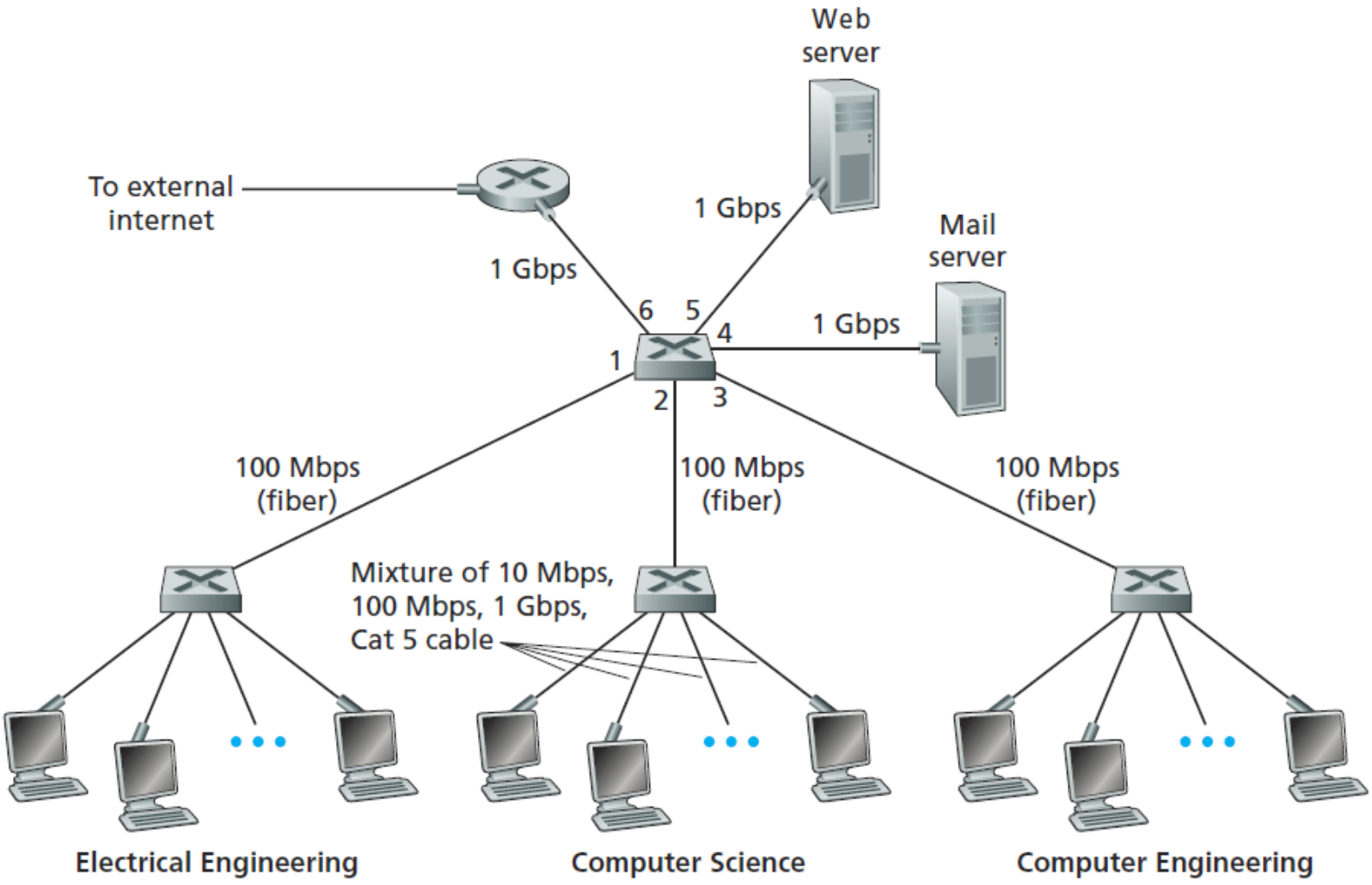
- 控制令牌顺序从一个节点传递到下一个节点。
- 问题:
  - 令牌开销
  - 延时
  - 单点失效(token)



多路访问控制协议的总结

- **信道划分**：时分，频分，码分
- **随机接入**：
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - 载波侦听：在某些技术中容易实现(有线)，在有些技术中比较困难(无线)
  - CSMA/CD used in Ethernet
  - CSMA/CA used in 802.11
- **轮流**
  - 来自中心站的轮询
  - 令牌传递

6.4 交换局域网


Web server
To external internet
1 Gbps
1 Gbps
Mail server
1 Gbps
6
5
4
1
2
3
100 Mbps (fiber)
100 Mbps (fiber)
100 Mbps (fiber)
Mixture of 10 Mbps, 100 Mbps, 1 Gbps, Cat 5 cable
Electrical Engineering
Computer Science
Computer Engineering

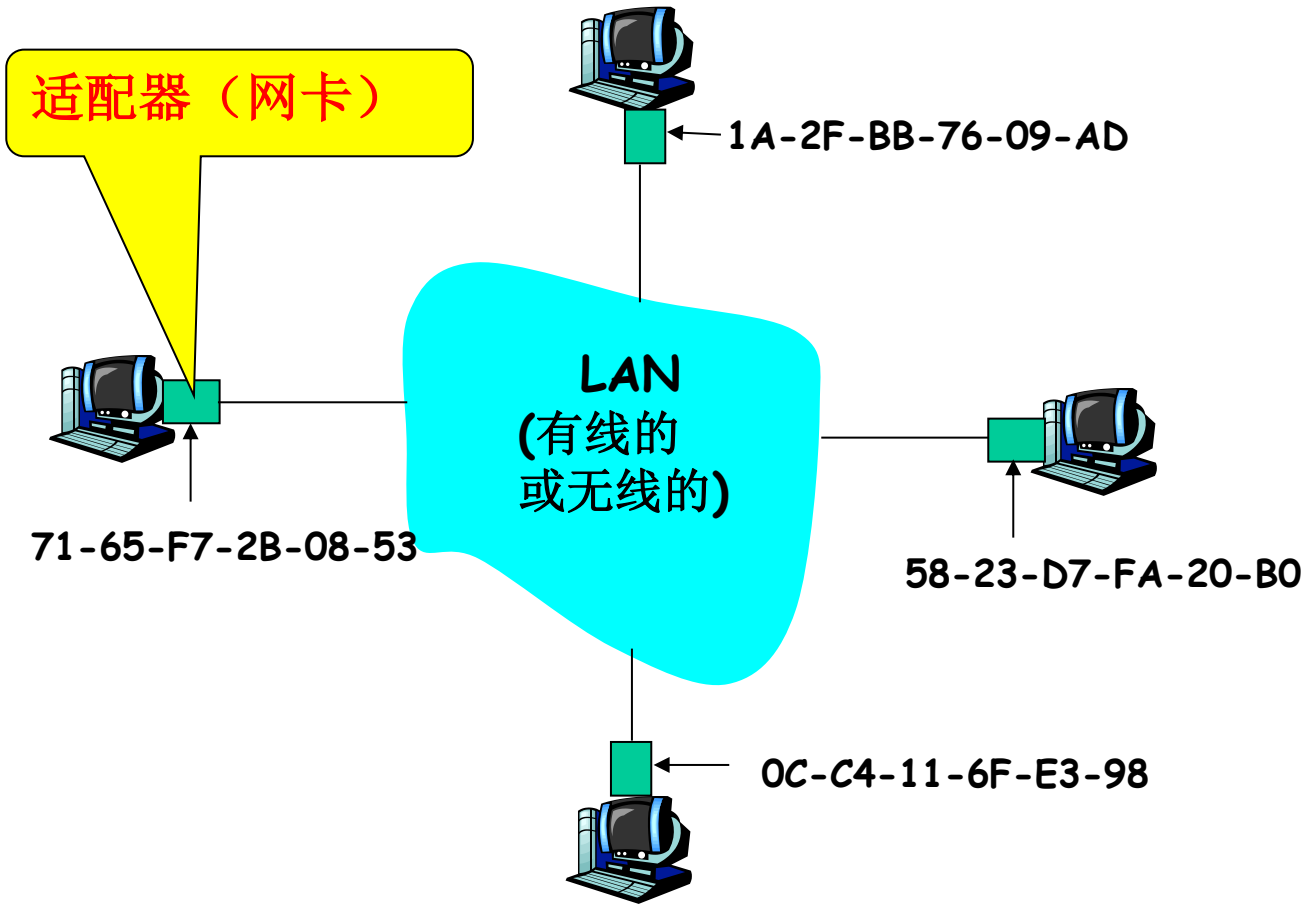
6.4.1 链路层寻址和ARP

每个节点有网络层地址和链路层地址。

- 网络层地址：节点在网络中分配的一个唯一地址(IP地址)。
  用于把分组送到目的IP网络，长度为32比特(IPv4)。
- 链路层地址：MAC地址或物理地址、局域网地址。
  - 用于把数据帧从一个节点传送到另一个节点(同一子网)。
- MAC地址（LAN地址、物理地址）：
  - 节点“网卡”本身所带的地址（唯一）。
  - MAC地址长度通常为6字节(48比特)，共2^{48}个。
  - 6字节地址用16进制表示，每个字节表示为一对16进制数
  - 网卡的MAC地址是永久的（生产时固化在其ROM里）

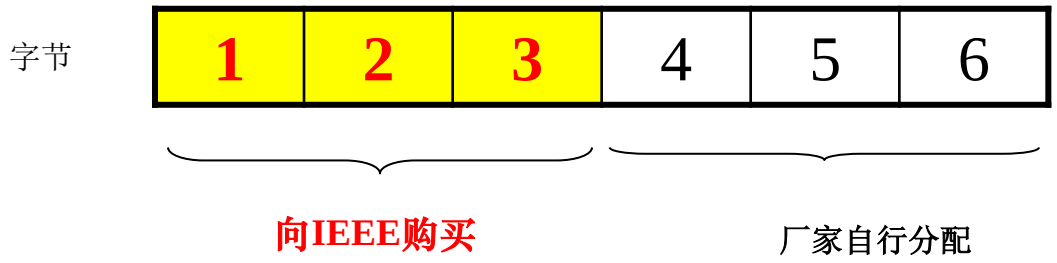
局域网地址

局域网中每个网卡都有唯一的局域网地址



MAC地址分配

- 由专门机构IEEE管理物理地址空间
  - 负责分配六个字节中的前三个字节（高24位，地址块）
- MAC 地址是平面结构
  - 同一网卡装在不同节点，在任何网络中都有同样的MAC地址。
- IP地址具有层次结构
  - 当节点移动到不同网络时，节点的IP地址发生改变。



MAC地址识别

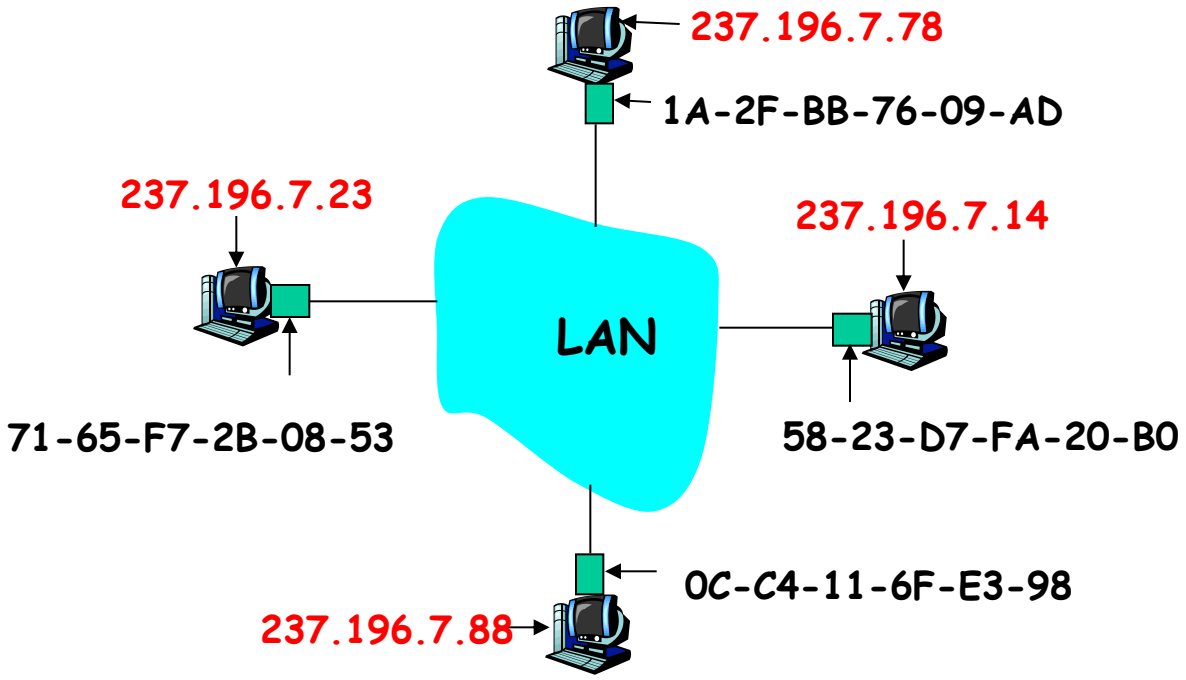
- 广播信道的局域网中，一个节点发送的帧，在信道上广播传输，其他节点都可能收到该帧。
- 大多数情况，一个节点只向某个特定的节点发送。
- 由“网卡”负责MAC地址的封装和识别。
- 发送适配器：将目的MAC地址封装到帧中，并发送。所有其他适配器都会收到这个帧。
- 接收适配器：检查帧的目的MAC地址是否与自己MAC地址相匹配：
  - 匹配：接收该帧，取出数据报，并传递给上层。
  - 不匹配：丢弃该帧。

特殊帧

- 广播帧：发送给所有节点的帧。
- MAC广播地址：全1地址。
- 如以太网和令牌传递LAN，其广播地址是48个连续的1组成的字符串，即：FF-FF-FF-FF-FF-FF

回顾：节点的3种不同地址表示

- 应用层的主机名、网络层的IP地址和链路层的MAC地址
- 实际在链路上传输时，根据MAC地址，确定相应的节点



地址之间的转换

通信时，需要进行地址转换：

主机名 → IP地址 → MAC地址

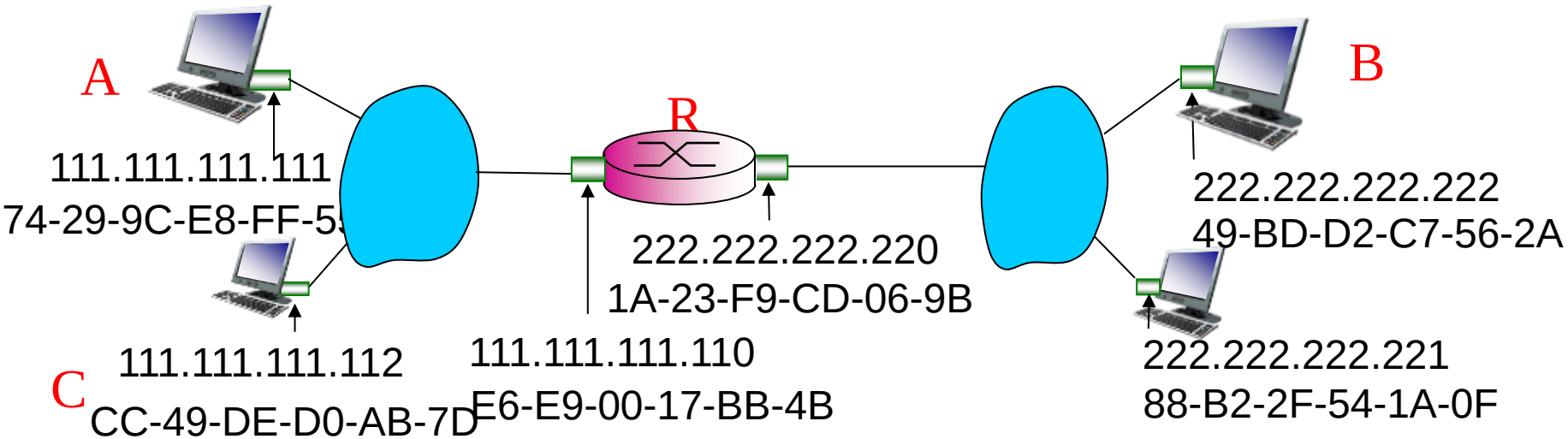
- *DNS域名系统：*将主机名解析到IP地址。

  DNS为在因特网中任何地方的主机解析主机名。

- *ARP地址解析协议：*将IP地址解析到MAC地址。

  ARP只为在同一个LAN上的节点解析IP地址。

位于同一局域网的两台主机通信



- 案例1：主机A发送数据报给主机C
  - 主机A的网络接口首先将数据报封装成数据帧

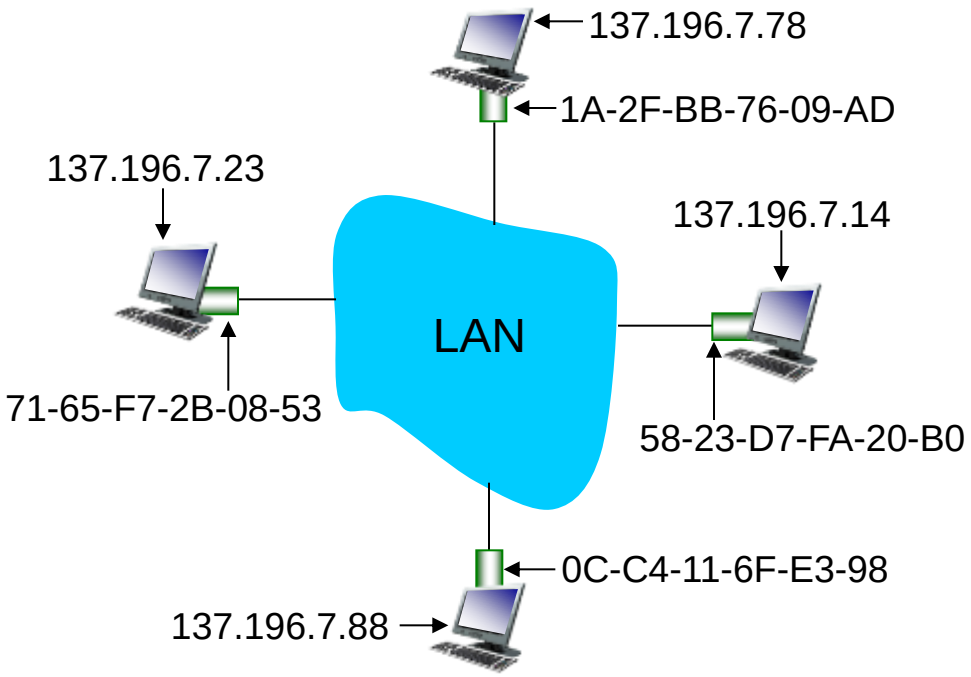
目的MAC地址：CC-49-DE-D0-AB-7D，源MAC地址：74-29-9C-E8-FF-55

  - 局域网内的网络接口都会收到该数据帧(A\C\R的左侧接口)
  - 只有主机C的MAC地址与该数据帧的目的MAC地址匹配，因此接收。

主机A怎么知道主机C的MAC地址?

ARP: 地址解析协议

问题：如何根据一个主机的IP地址，查找其MAC地址



ARP表: 局域网上的每个节点(主机、路由器)都有这个表

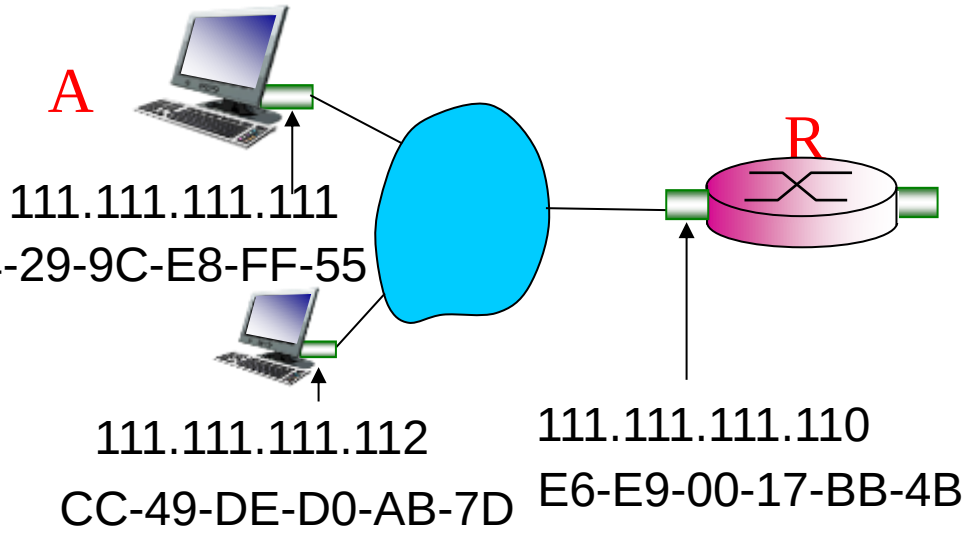
- 为某些局域网节点进行IP/MAC地址映射：

< IP address; MAC address; TTL>

- TTL (存活时间): 地址映射将被删除的时间（通常为20分钟）

ARP协议的工作过程

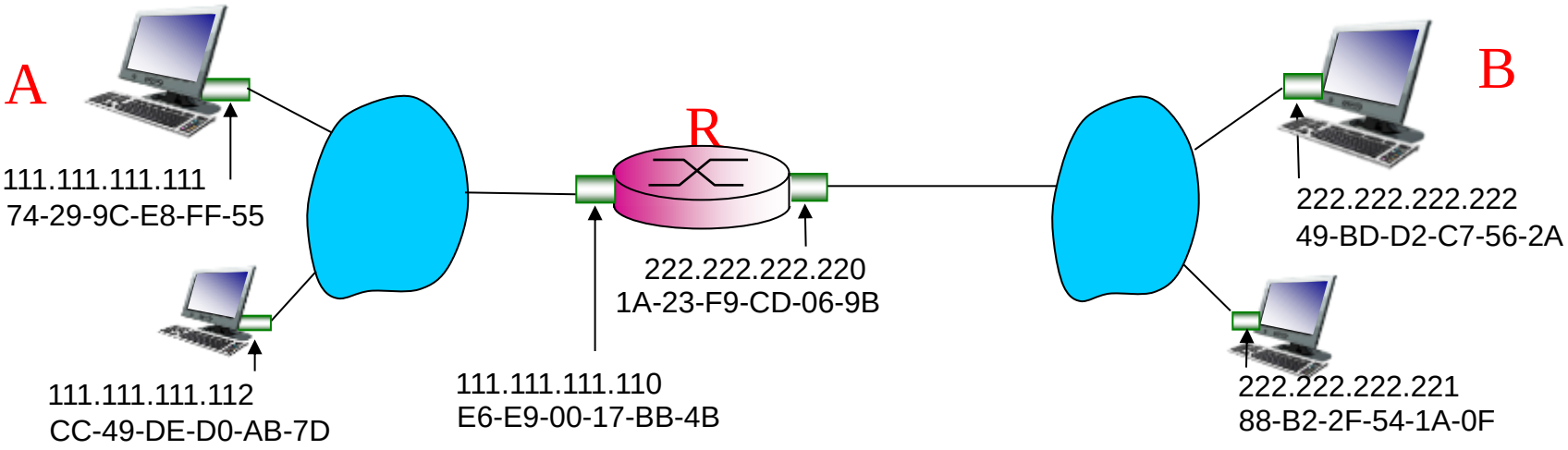
- 主机A希望发送数据报给主机C；
  - C的MAC地址不在A的ARP映射表中
- 主机A广播 ARP查询分组，其中包含C的IP地址
  - 目的MAC地址：FF-FF-FF-FF-FF-FF
  - 目的IP地址：111.111.111.112
  - 局域网中所有节点收到ARP查询分组
- 主机C收到ARP查询分组，返回响应分组给主机A，返回的数据帧包含有C的MAC地址 (单播)
- 主机A在它的ARP表中缓存 **IP-to-MAC 地址对，**直到信息超时



ARP协议只能查找位于局域网内部的网络接口的IP地址对应的MAC地址！

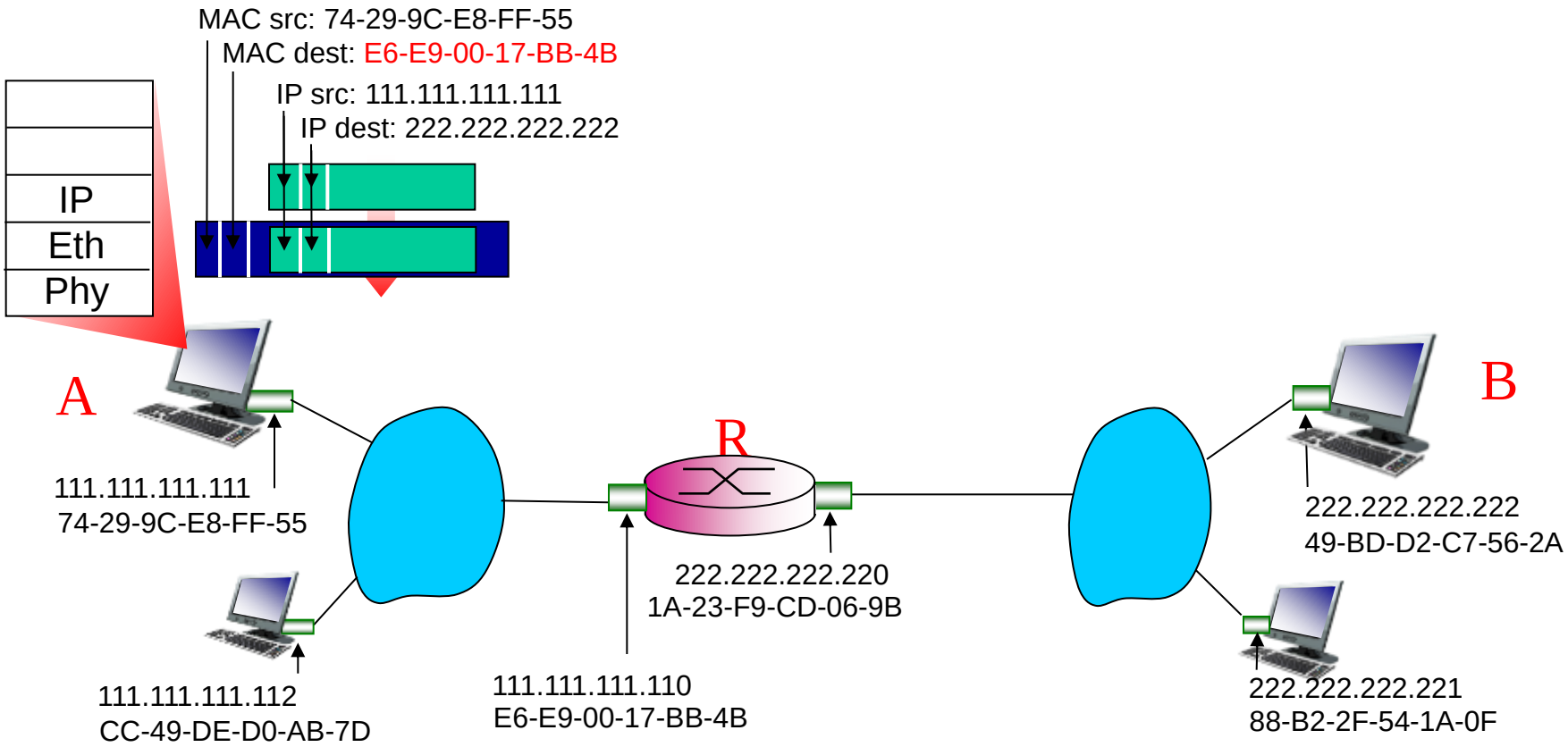
位于不同局域网的两台主机通信

- 案例：主机A经路由器R发送数据报给主机B
  - 集中在寻址上：IP层(数据报)和MAC层(数据帧)
  - 假设主机A知道主机B的IP地址
  - 假设主机A知道网关路由器R的IP地址(通过DHCP协议)
  - 假设主机A知道网关路由器R的MAC地址(通过ARP协议)



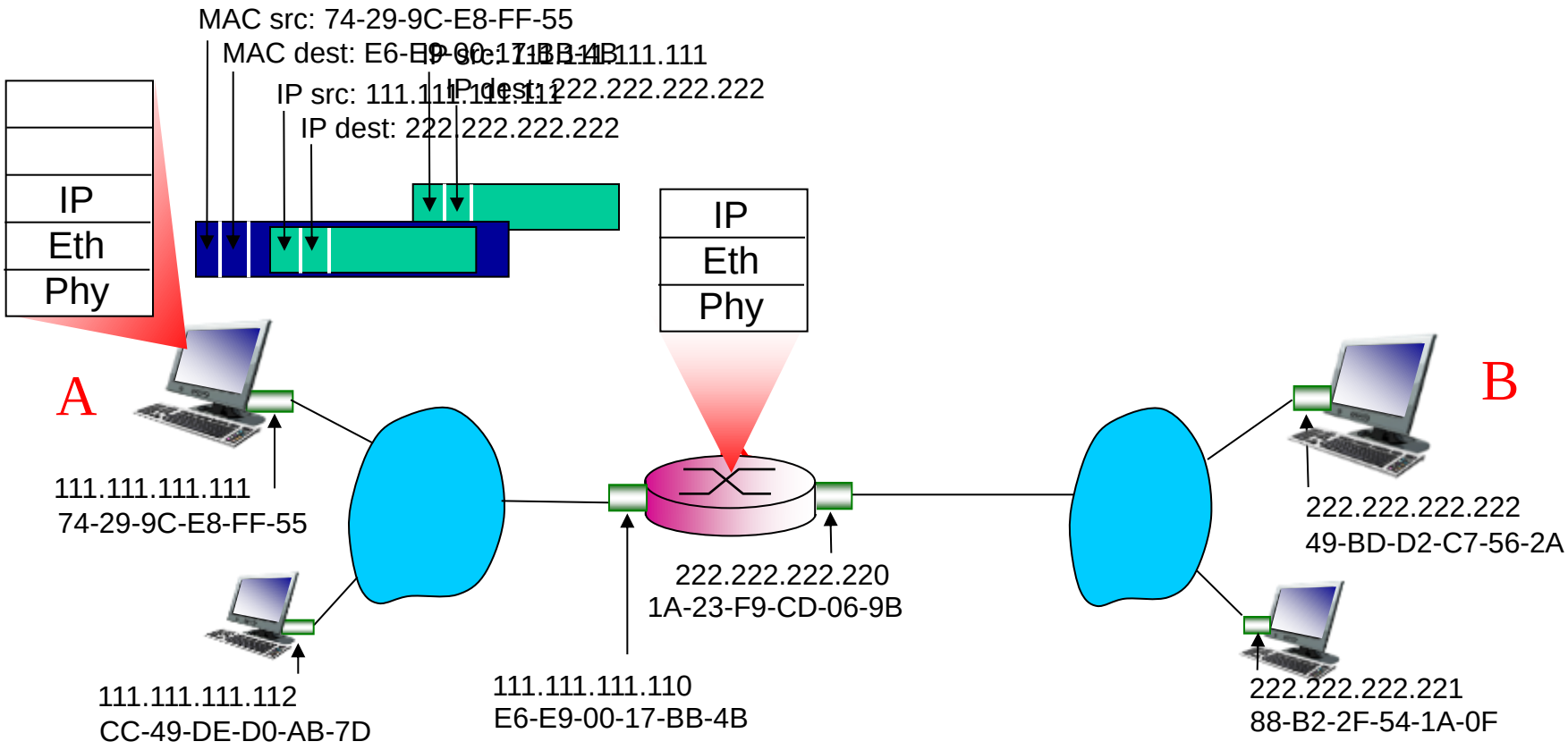
位于不同局域网的两台主机通信

- 主机A构建IP数据报，源地址是A的IP地址，目的地址是B的IP地址
- 主机A构建链路层数据帧，目的MAC地址是路由器左边端口的MAC地址



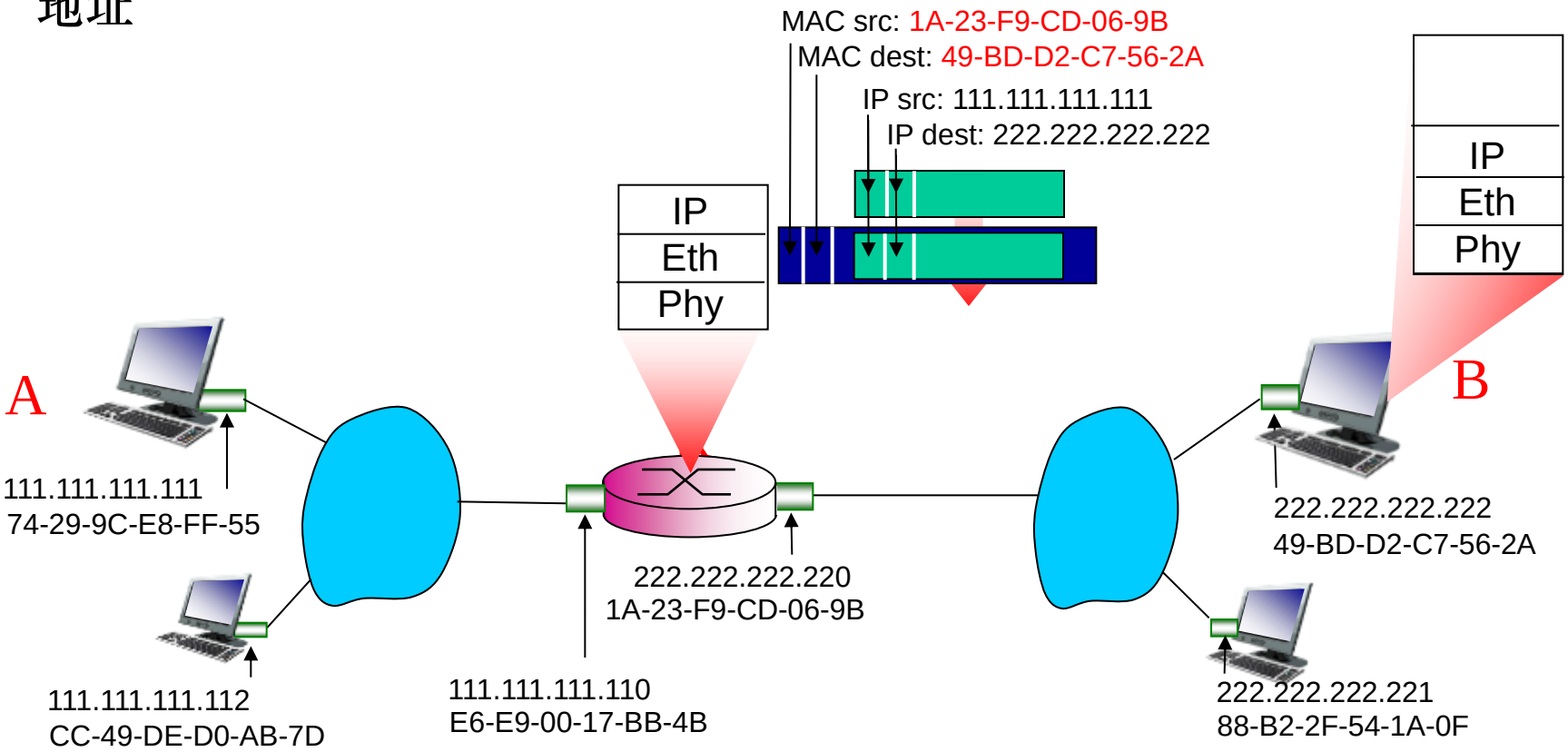
位于不同局域网的两台主机通信

- 数据帧从主机A发送到路由器R
- 路由器R收到数据帧，抽取出数据报递交到IP层



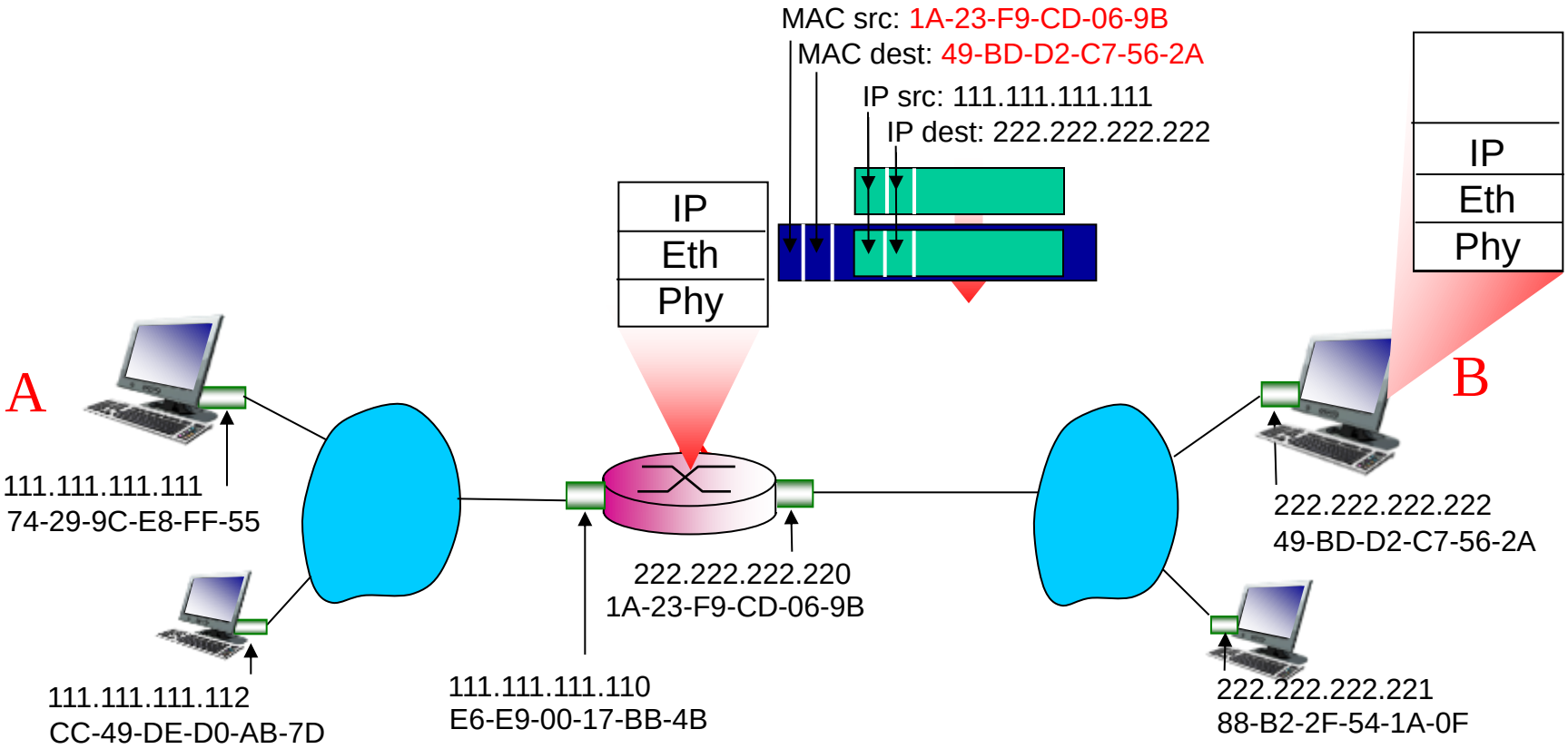
位于不同局域网的两台主机通信

- 路由器R转发数据报，源地址为A的IP地址，目的地址为B的IP地址
- 路由器R将该数据报封装成链路层帧，目的MAC地址为主机B的MAC地址



位于不同局域网的两台主机通信

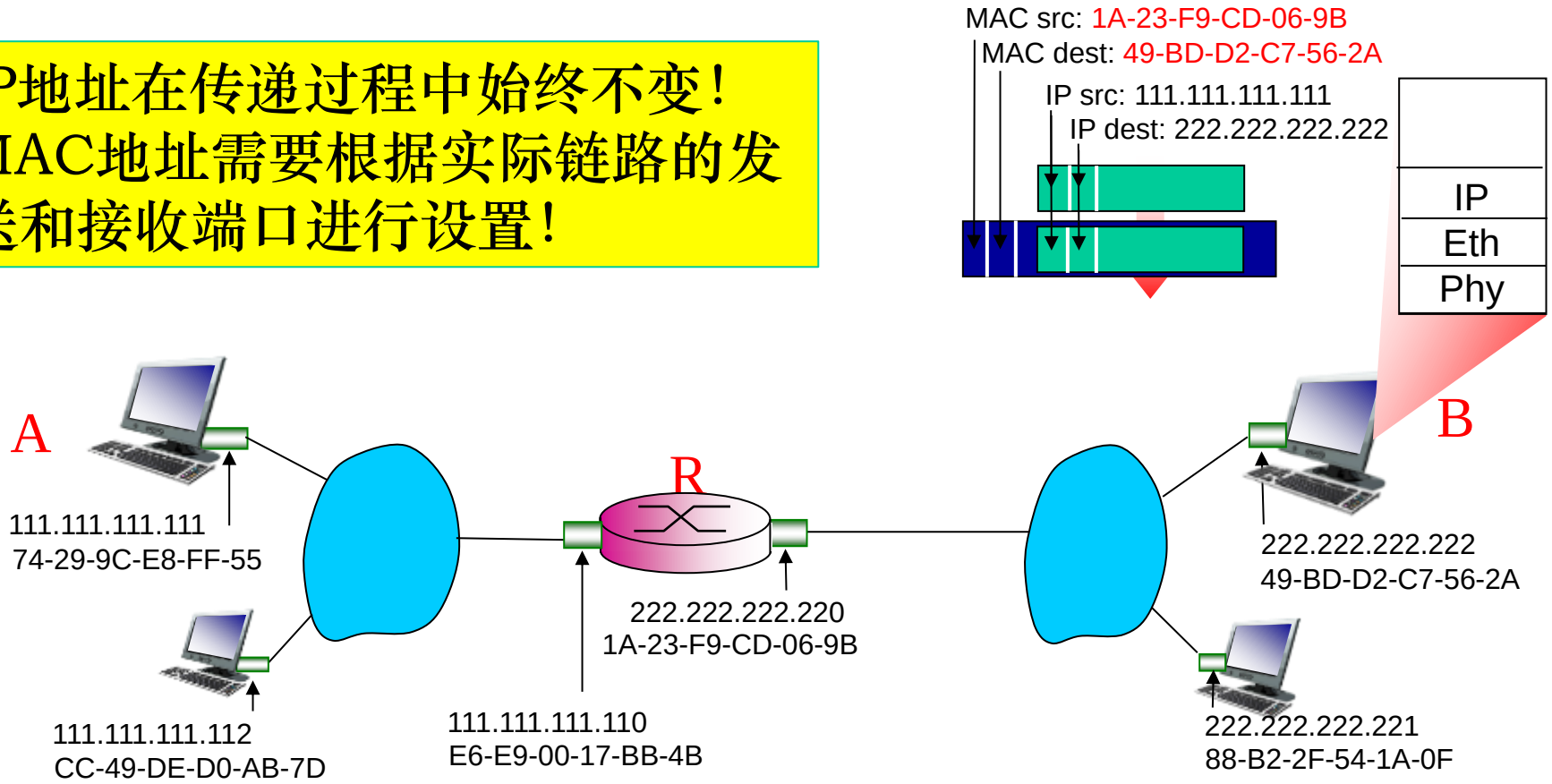
- 路由器R转发数据报，源地址为A的IP地址，目的地址为B的IP地址
- 路由器R将该数据报封装成链路层帧，目的MAC地址为主机B的MAC地址



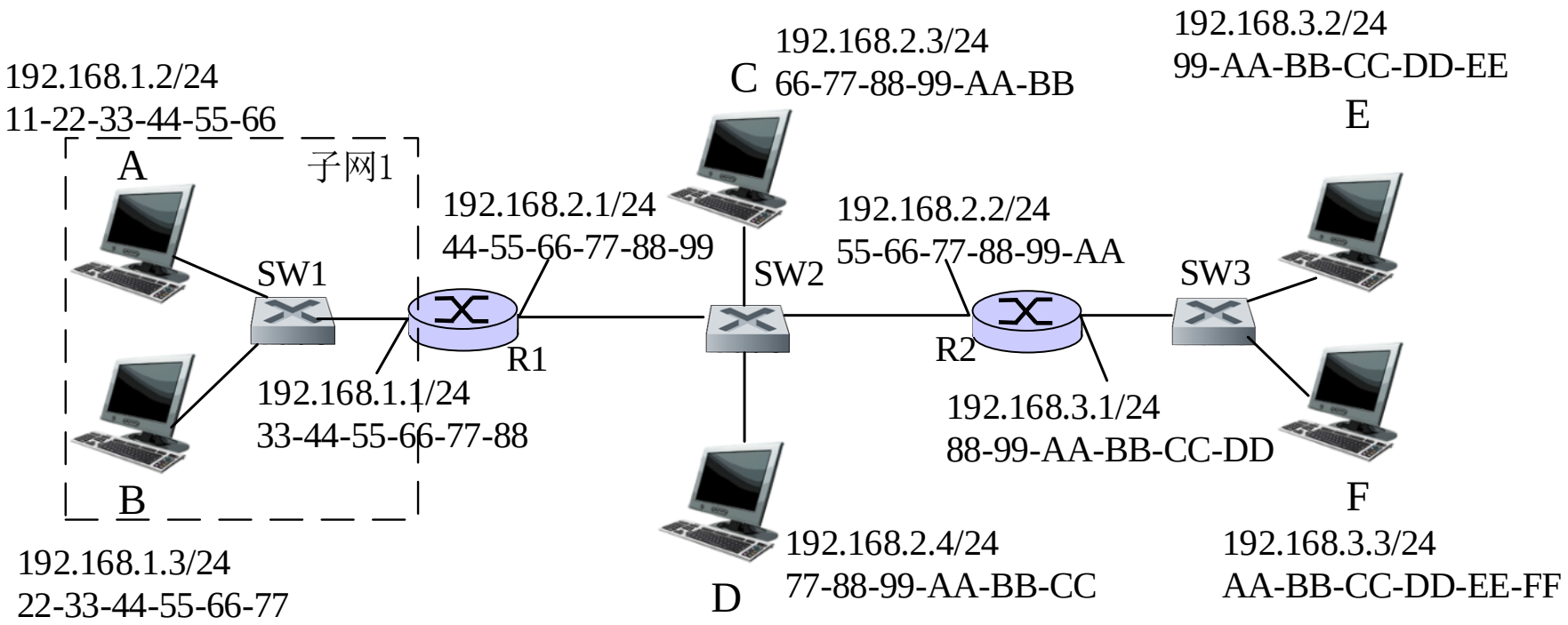
位于不同局域网的两台主机通信

- 路由器R转发数据报，源地址为A的IP地址，目的地址为B的IP地址
- 路由器R将该数据报封装成链路层帧，目的MAC地址为主机B的MAC地址

IP地址在传递过程中始终不变！
MAC地址需要根据实际链路的发送和接收端口进行设置！



更为复杂的链路层寻址案例



要求1：从主机E向主机B发送一个IP数据报，假设所有网络设备有最新的ARP缓存表，描述分组转发的步骤。

课后思考：假设E主机的ARP缓存表为空，其他节点的缓存表都是最新的，需要增加的步骤有哪些?

链路层寻址小结

- 链路层寻址：
  - 位于相同局域网的主机通信；
  - 位于不同局域网的主机通信；
- ARP协议：
  - 只针对相同局域网的网络接口查找MAC地址；
  - ARP协议的工作过程；
  - ARP缓存表是动态更新的。

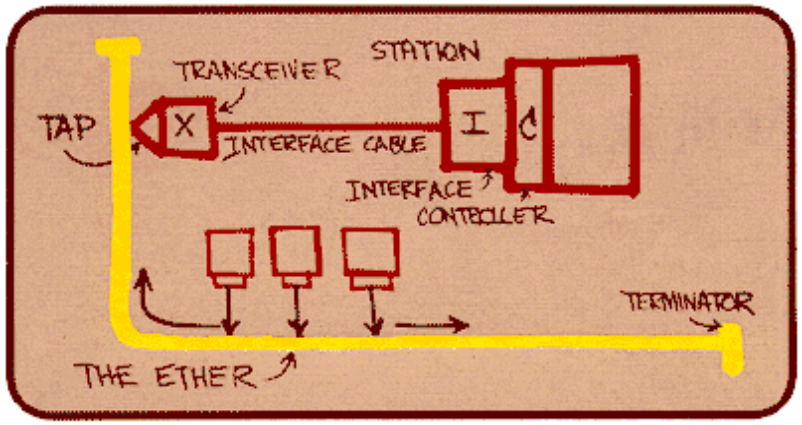
6.4.2 以太网

Ethernet

到目前为止，以太网是最为著名的有线局域网技术

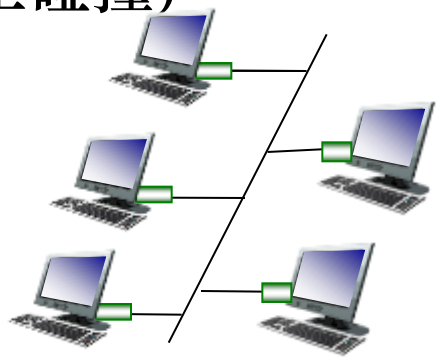
以太网成功的原因：

- 是第一个广泛使用的局域网技术；
- 简单、便宜；
- 版本不断更新，数据速率更高、成本更低。

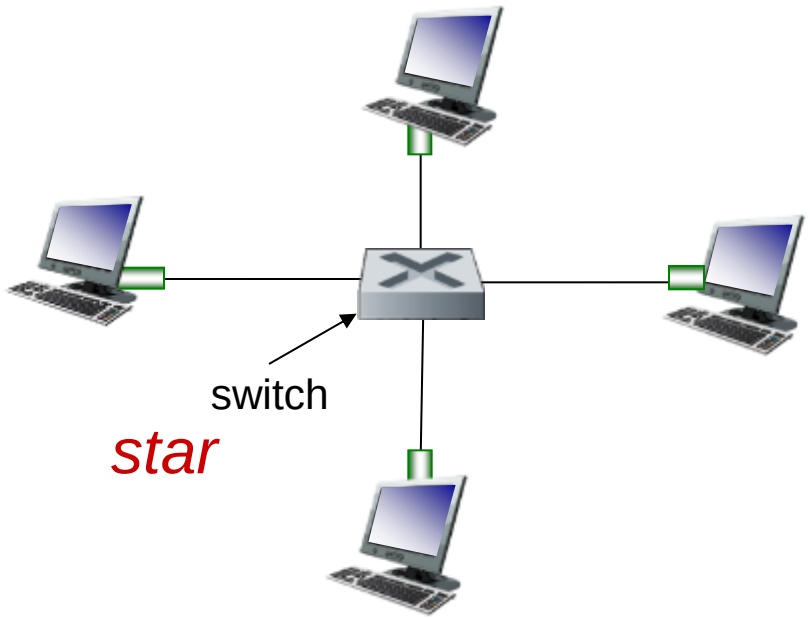


以太网物理拓扑结构

- 总线(bus): 一直流行到90年代中期
  - 所有节点都属于相同的冲突(碰撞)域
- 星形(star):目前流行
  - 中心是交换机
  - 每个端口运行一个独立的以太网协议(节点相互之间不发生碰撞)



bus: coaxial cable

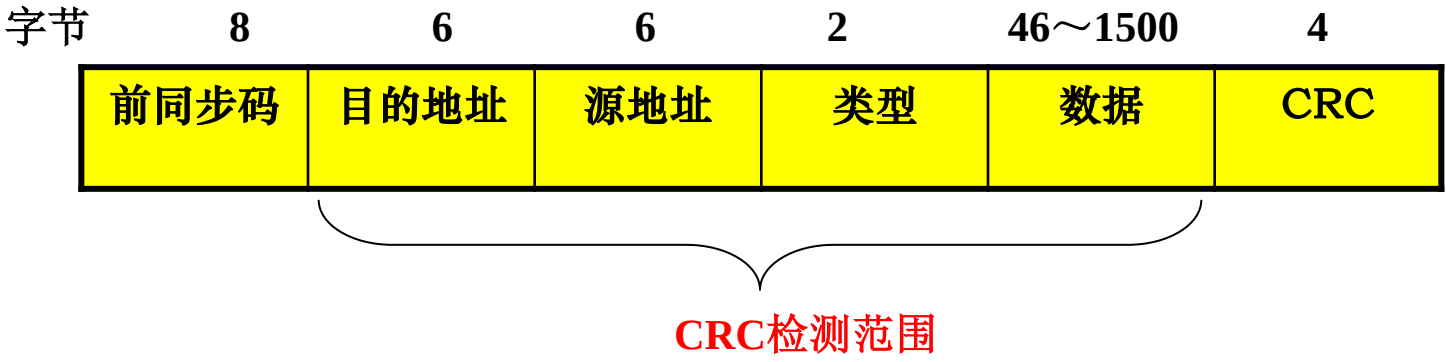


star

共享链路与点到点链路

- 早期采用同轴电缆作为传输介质，总线结构，因此是所有以太网的节点处于一条共享链路，整个以太网属于一个冲突域!
- 后期发展采用双绞线将节点连接到**集线器**，星形结构，这里集线器是一个**物理层设备**，直接将收到的每个比特拷贝到每个端口，因此对于集线器互联的以太网节点，所有节点仍然处于一个共享信道，因此仍然所有节点属于一个冲突域!
- 目前采用双绞线将节点连接到交换机，星形结构，这里交换机是一个**数据链路层设备**，具有针对数据帧的存储转发功能，因此对于每个交换机端口是一个独立的共享信道(冲突域)。如果一个以太局域网只有交换机互联节点，就不会发生碰撞，因此不需要MAC协议。

1、以太网帧结构



- *发送方：* 发送适配器将IP数据报封装成以太网帧，并传递到物理层。
- *接收方：* 接收适配器从物理层收到该帧，取出IP数据报，并传递给网络层。

前同步码(8 字节)

- 前7字节是“10101010”，最后一个字节是“10101011”。
- 使接收方和发送方的时钟同步，接收方一旦收到连续的8字节前同步码，可确定有帧传过来。
- 前同步码是“无效信号”，接收方收到后删除，不向上层传。
- CRC的校验范围不包括前同步码。

源、目的MAC地址(各6字节)

- 例，同一以太网LAN中两台主机通信。
- 主机A向主机B发送一个IP数据报。
  - 主机A适配器的MAC地址：AA-AA-AA-AA-AA-AA
  - 主机B适配器的MAC地址：BB-BB-BB-BB-BB-BB
- 适配器B只接收目的地址与其MAC地址匹配或广播地址的帧，并将数据字段的内容传递给网络层。否则，丢弃该帧。

类型字段(2 字节)

以太网可以“*多路复用*”（*支持*）多种网络层协议。通过“类型”字段区分。

- 发送方填入网络层协议“类型”编号；
- 接收适配器根据“类型”字段，将数据字段传递给相应的网络层协议。

数据字段(46～1500 字节)

携带网络层传来的IP数据报

- *以太网的最大传输单元MTU是1500字节:*
  - 若IP数据报超过1500字节，必须将该数据报分段。
- *最小长度是46字节:*
  - 如果IP数据报小于46字节，必须填充为46字节。接收方网络层去除填充内容。

循环冗余检测CRC(4字节)

检测数据帧中是否出现比特差错（翻转）。

- *发送主机计算CRC：*范围包括目的地址、源地址、类型、数据字段的比特，结果放入帧CRC字段。
- *接收主机进行CRC校验：*接收主机对收到的帧进行同样计算，并校验结果是否和CRC字段的内容相等。若计算结果不等于CRC字段的值(CRC校验失败)，该帧有差错。

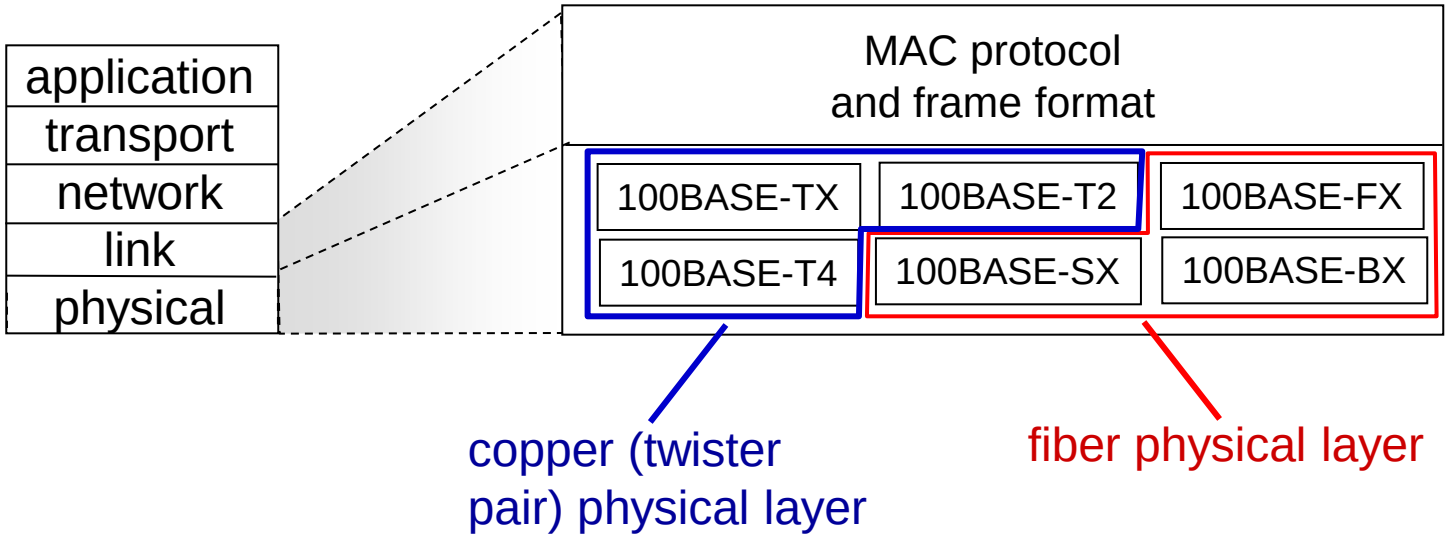
以太网: 不可靠的无连接服务

以太网向网络层提供的服务。

- 无连接服务：通信时，发送方适配器不需要先和接收方适配器“握手”。
- 不可靠的服务：接收到的帧可能包含比特差错。
  - 收到正确帧，不发确认帧；
  - 收到出错帧，丢弃该帧，不发否定帧。
  - 发送适配器不会重发出错帧。
  - 丢弃数据的恢复是通过终端传输层的可靠数据传输机制来实现的
- 以太网的MAC协议：无时隙的CSMA/CD协议（二进制指数回退）

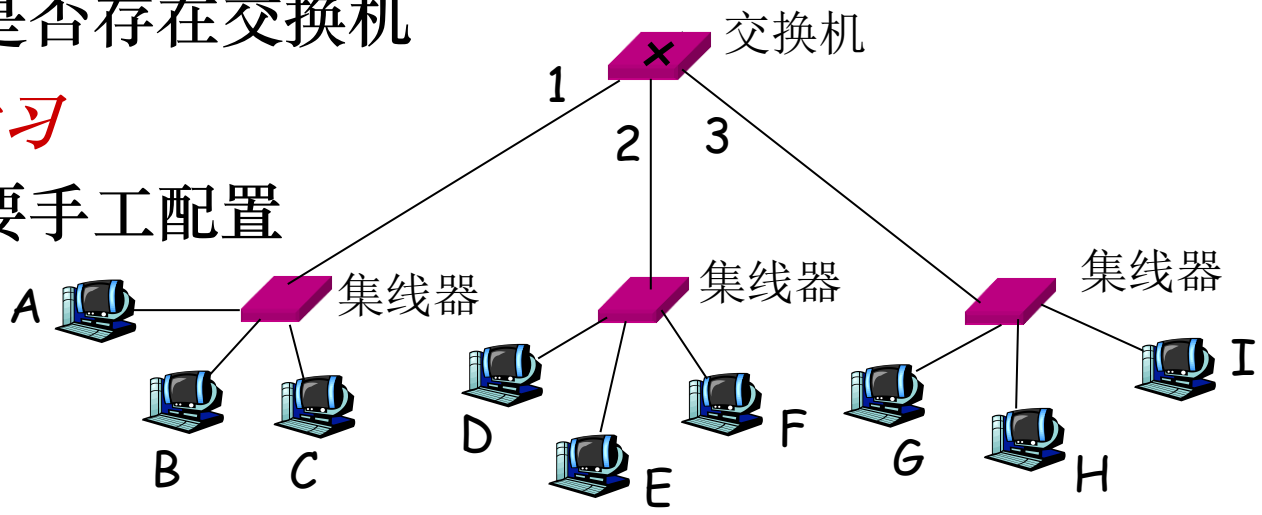
6.4.2 以太网

- 以太网的协议标准由IEEE 802.3 CSMA/CD工作组标准化：<传输速率><BASE><物理介质>
  - 第一部分传输速率：10(Mbps)、100(Mbps)、1000(Mbps)
  - 第二部分BASE：表示基带传输(只传输以太网流量)
  - 第三部分物理介质：2/5(两种同轴电缆，500米)、T(双绞铜线，100米)、FX/SX/BX(光纤，几千米)



6.4.3 链路层交换机

- 链路层设备
  - 存储转发数据帧
  - 检查达到的数据帧的MAC地址，有选择的转发数据帧到一个或多个输出链路，当数据帧被转发到一个共享网段时，使用CSMA/CD来访问共享链路
- 透明
  - 主机不关心是否存在交换机
- *即插即用和自学习*
  - 交换机不需要手工配置



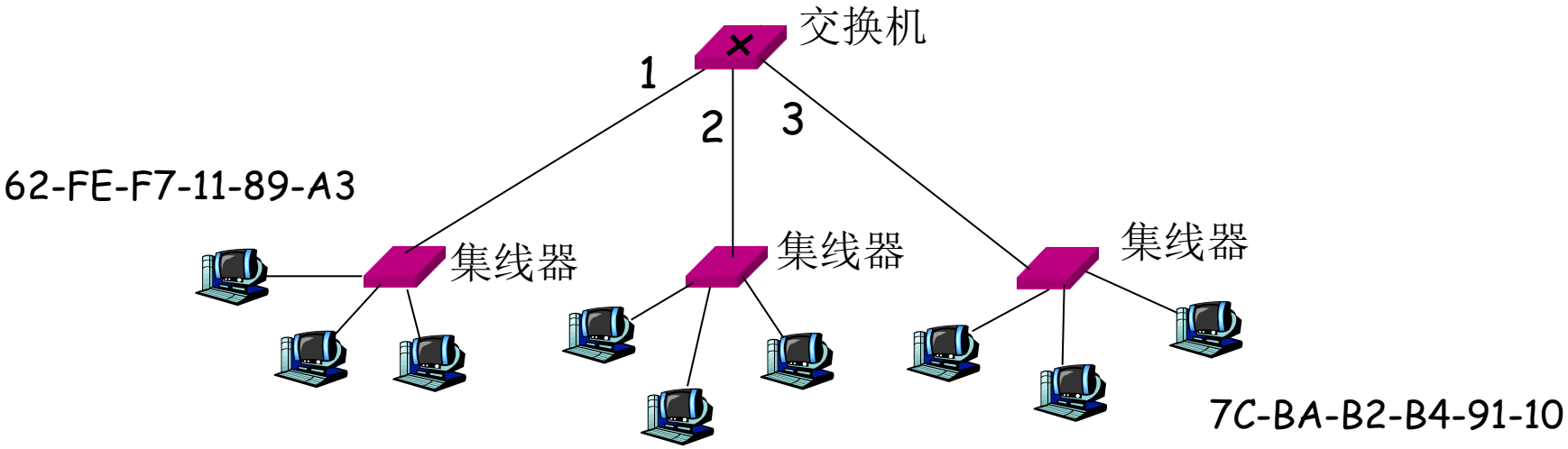
1、交换机转发和过滤

- 过滤(filtering)：交换机判断一个帧是应该转发到某个接口还是丢弃。
- 转发(forward)：交换机决定一个帧应该被指向哪个接口，并引导到该接口。
- 过滤和转发通过**交换机表(switch table)**完成。
- 交换机表：
  - 包含LAN上部分节点的表项；
  - 内容：节点的MAC地址、到达该节点的交换机接口、节点表项产生的时间。

交换表的例子

地址	接口	时间
62-FE-F7-11-89-A3	1	9:32
7C-BA-B2-B4-91-10	3	9:36
...	...	...

图 5.29　图 5.28 中所示的 LAN 的网桥表的一部分



过滤和转发的原理

当交换机收到数据帧:

1. 记录到达链路和发送主机的MAC地址
2. 使用数据帧的目的MAC地址，在转发表中检索
3. 如果在转发表条目中找到对应的MAC地址
4. 执行{

   如果目的MAC地址对应的端口与数据帧的达到端口相同

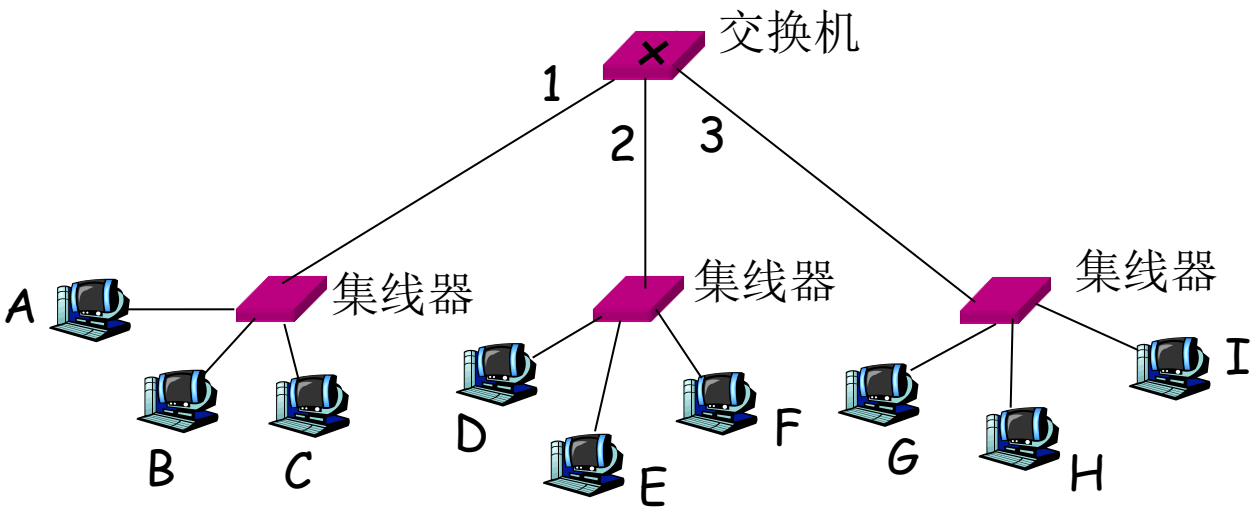
   则 丢弃该数据帧

   否则 转发该数据帧到条目指定的端口
5. }
6. 否则，向除到达端口之外的所有端口转发(flood)

例子

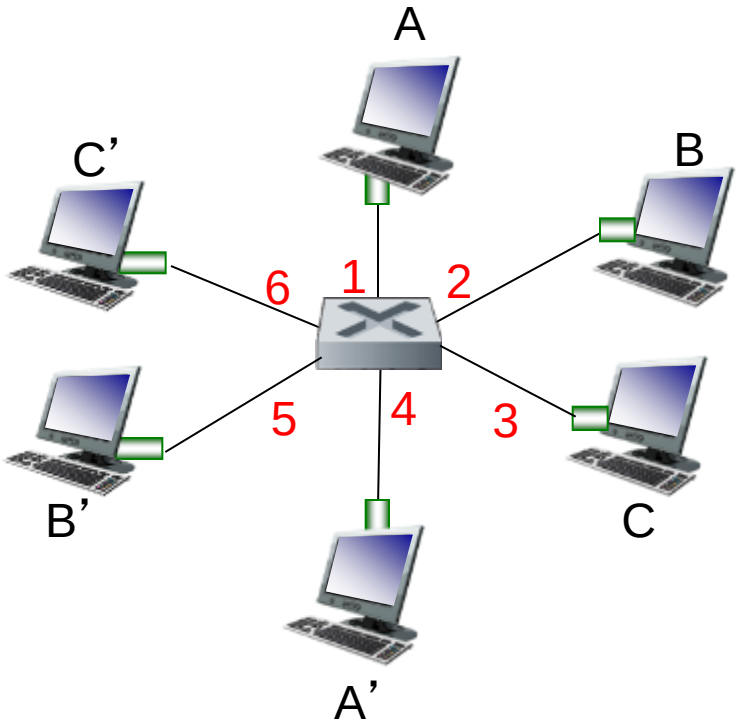
- 节点A向B发帧：1#收到→查表→1#转发→同一网段，丢弃此帧
- 节点D向C发帧：2#收到→查表→1#转发→不同网段，从1#转发
- 节点A向H发帧：1#收到→查表→没有H的对应表项→向2#和3#端口均转发该帧，G收到

地址	接口
A	1
B	1
C	1
D	2



交换机：支持多个节点同时传输

- 每个主机由单独的链路直接连到交换机端口
- 交换机可以缓存数据帧
- 以太网协议在每个输入链路使用，无碰撞，全双工
  - 每条链路自身是一个碰撞域
- 交换机：A-to-A’和B-to-B’可以同时传输，而不会发生碰撞



switch with six interfaces (1,2,3,4,5,6)

交换机转发表的建立

问题: 交换机是怎么知道A’可通过端口4达到，B’可通过端口5到达?

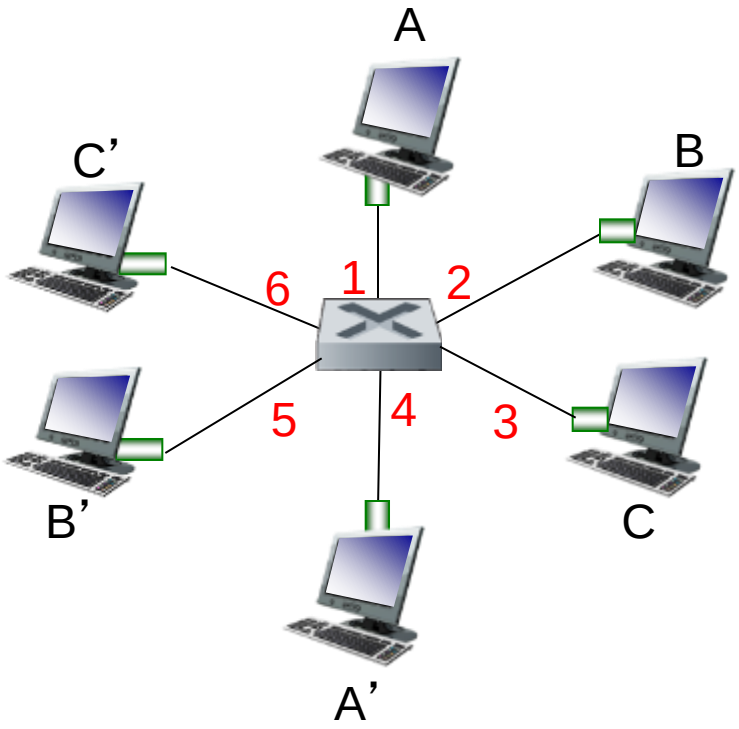
回答：每个交换机有一个交换机表，其中每个条目：

(主机的MAC地址，到达主机的端口，时戳)

类似于路由表

问题：转发表中的条目是怎么建立的呢？是否类似于路由协议呢?

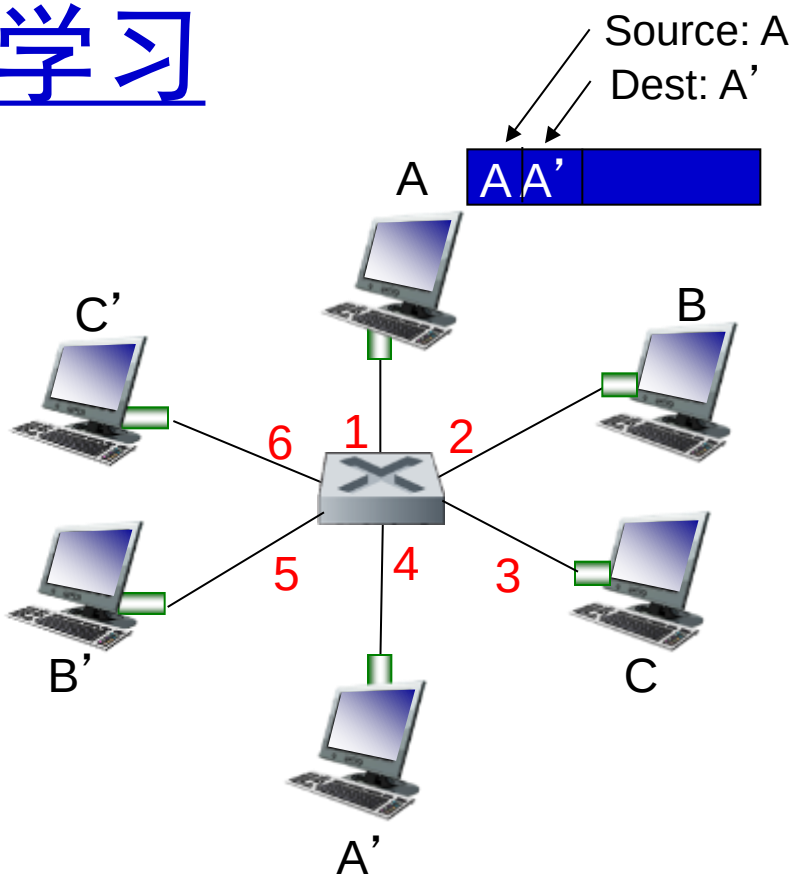
回答：通过自学习



switch with six interfaces (1,2,3,4,5,6)

2、自学习

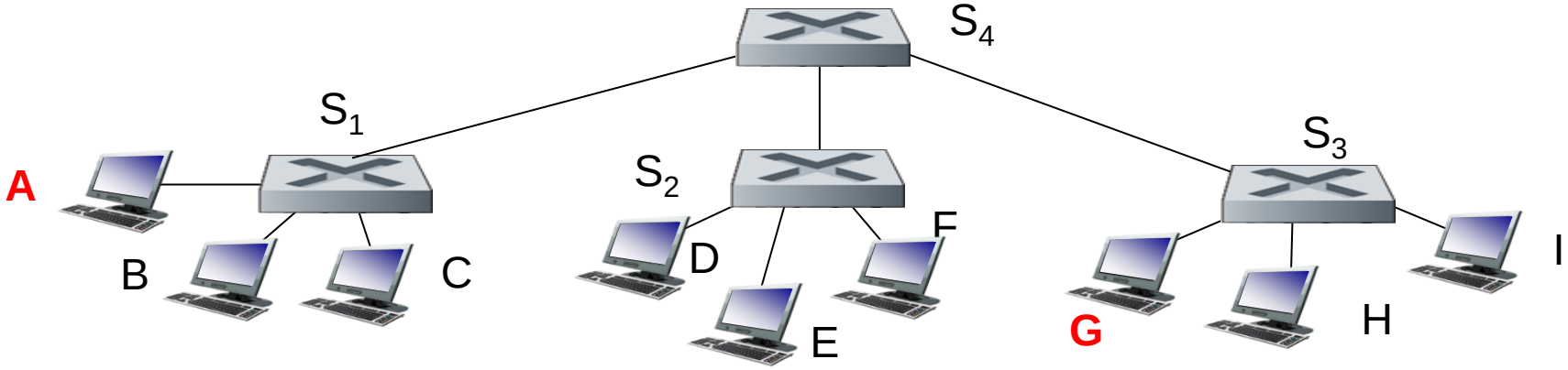
- 交换机会学习通过哪些端口可以到达哪些主机
  - 当收到数据帧时，交换机“学习”发送主机的位置：进入的局域网网段(到达端口)
  - 在转发表中记录发送主机/位置对



MAC addr	interface	TTL
A	*1*	*60*

Switch table (initially empty)

交换机互连

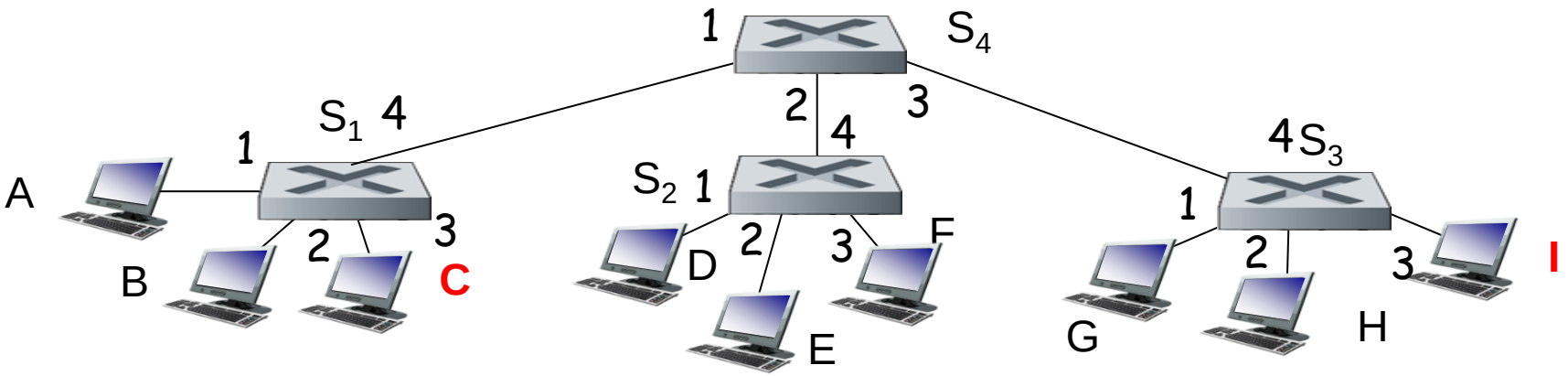


问题：A发送数据帧给G，S1是怎么知道要把数据帧先转发到S4和S3的?

回答：泛洪和自学习

多个交换机自学习的例子

假设主机C发送数据帧到主机I，主机I响应给主机C



问题: 请大家画出S_1，S_2，S_3，S_4交换机转发表和分组转发

S_1

MAC addr	interface	TTL
C	3	60
I	4	60

S_2

MAC addr	interface	TTL
C	4	60

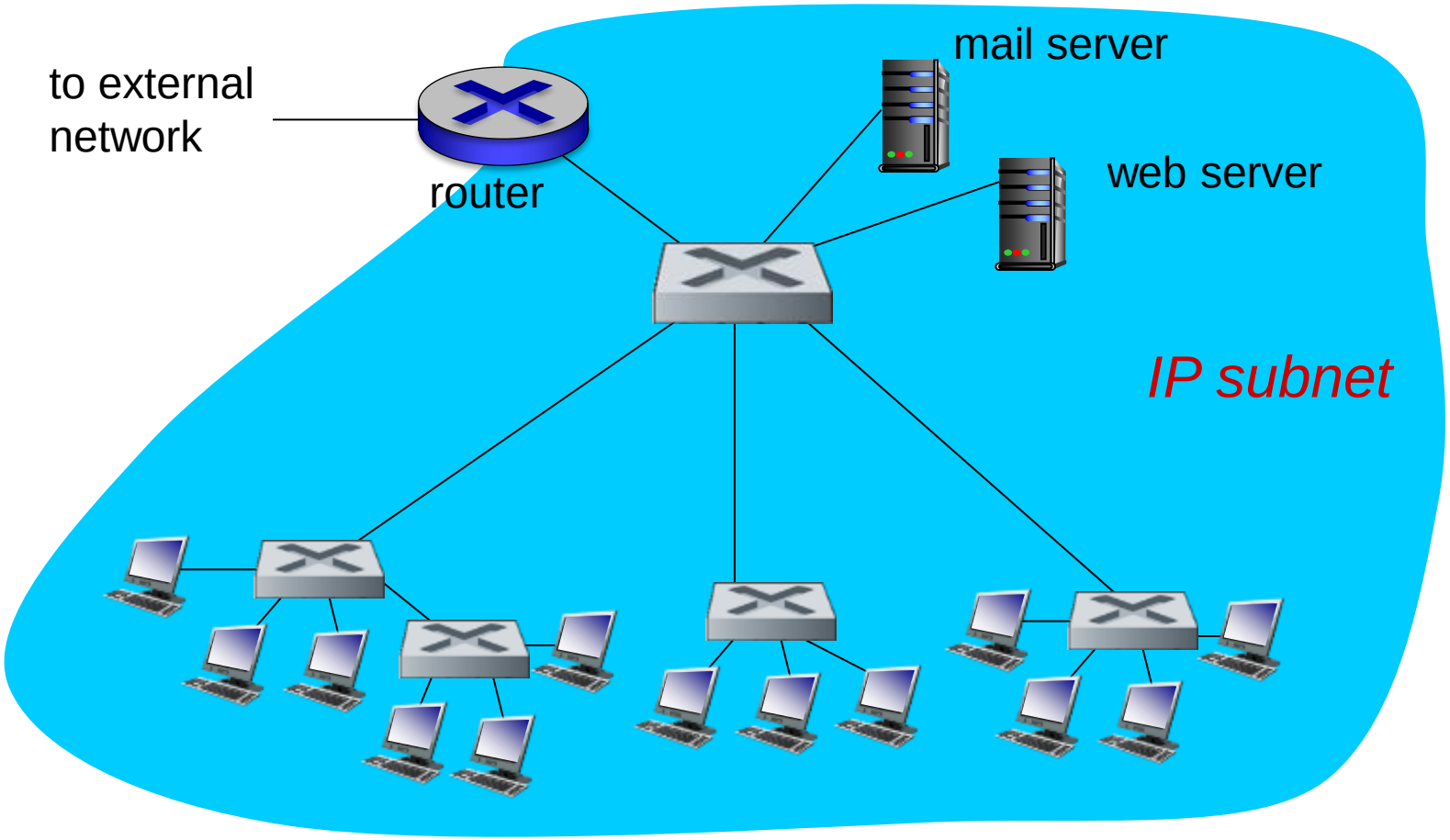
S_3

MAC addr	interface	TTL
C	4	60
I	3	60

S_4

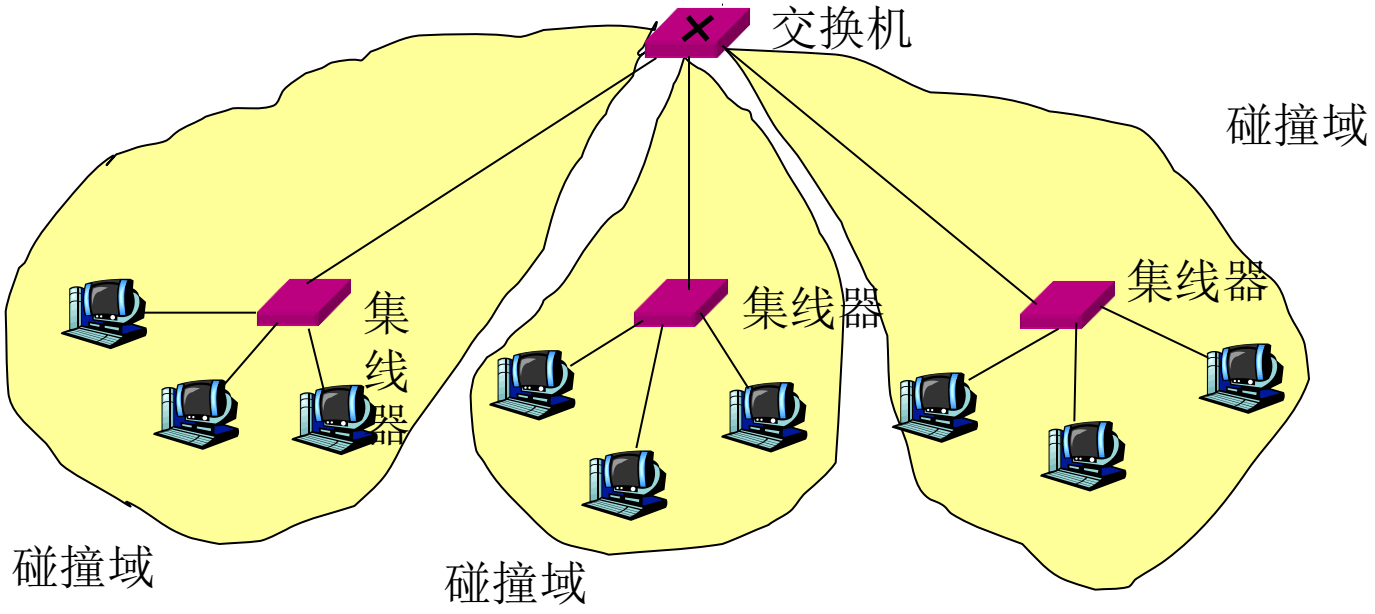
MAC addr	interface	TTL
C	1	60
I	3	60

机构的网络


to external
network
router
mail server
web server
IP subnet

交换局域网的优点

- 交换机所连接的不同的LAN网段保持独立的冲突域。
- 使不同LAN网段的两组节点同时通信而互不干扰。
- 流量过滤



交换机与集线器比较

- 转发功能
  - 集线器：转发帧时只是发送比特到链路上，并不侦听该链路是否忙；
  - 交换机：将帧转发到输出端口对应的链路上，在每个端口运行CSMA/CD；
    - 如果侦听到要转发的LAN网段上忙，停止传输；
    - 如果出现冲突，采用指数后退算法。
- 互联功能
  - 交换机：互联不同技术的以太网段，无地理范围限制。
  - 集线器：不具备该特性。

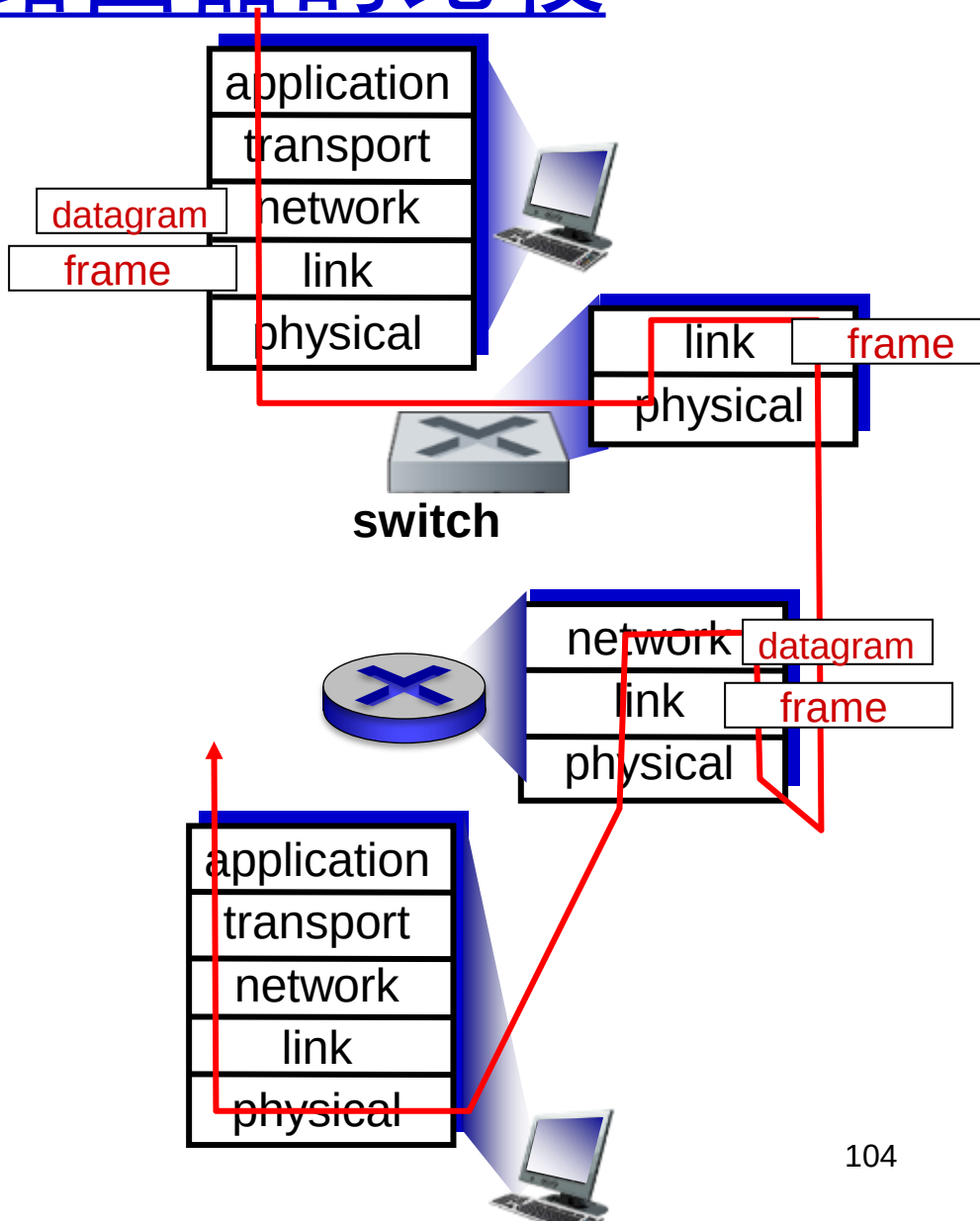
4、交换机和路由器的比较

两者都是存储转发设备:

- 路由器: 网络层设备(检查网络层头部)
- 交换机：链路层设备(检查链路层头部)

两者都有转发表

- 路由器：使用路由算法计算转发表，基于IP地址转发
- 交换机：通过泛洪、自学习来学习转发表，基于MAC地址转发



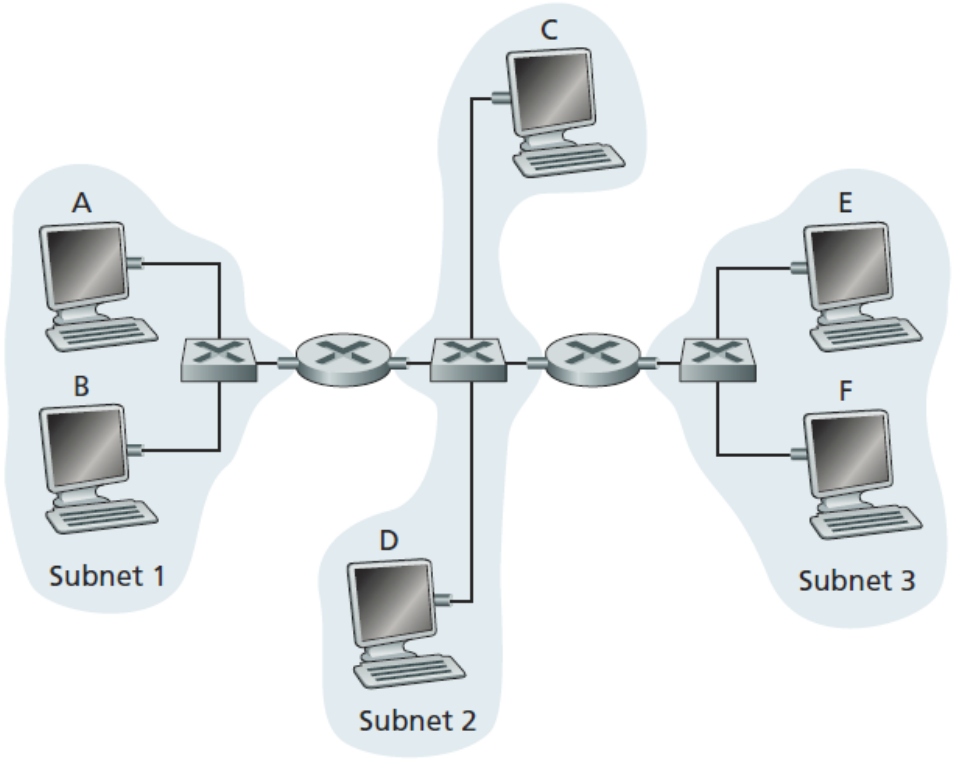
对广播帧的处理

- 集线器
  - 收到数据帧，会将其向自己的所有端口转发；
  - 不能隔离冲突域和广播域；
- 交换机
  - 收到广播帧，会将其向自己的所有端口转发；
  - 能够隔离冲突域，不能隔离广播域；
- 路由器
  - 收到广播帧，会根据目的IP地址向对应端口转发；
  - 能隔离冲突域，能隔离广播域

	Hubs	Routers	Switches
Traffic isolation	No	Yes	Yes
Plug and play	Yes	No	Yes
Optimal routing	No	Yes	No

课堂练习

15. Consider Figure 5.33. Now we replace the router between subnets 1 and 2 with a switch S1, and label the router between subnets 2 and 3 as R1.



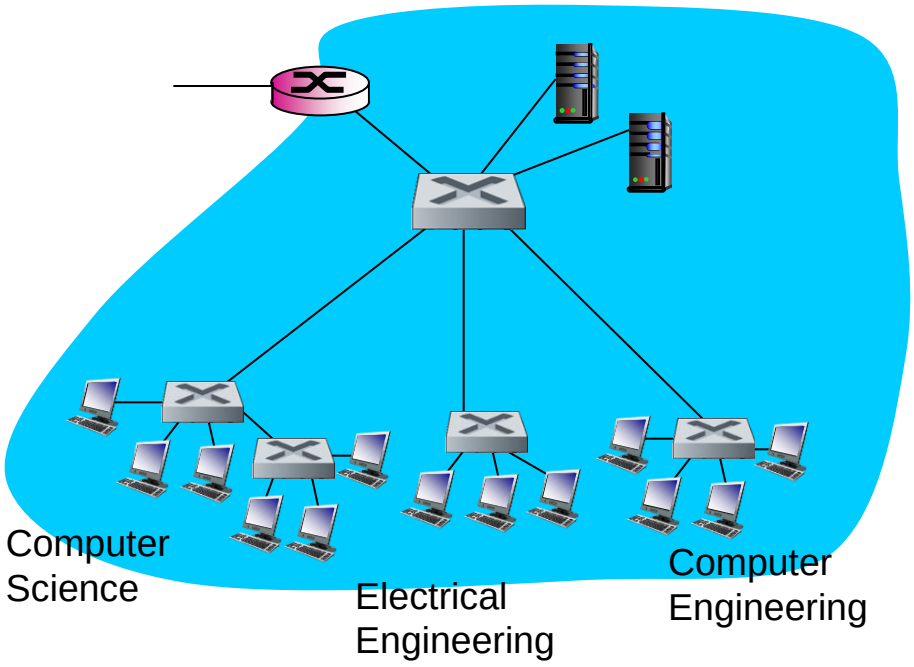
课堂练习

a. Consider sending an IP datagram from Host E to Host F. Will Host E ask router R1 to help forward the datagram? Why? In the Ethernet frame containing the IP datagram, what are the source and destination IP and MAC addresses?

b. Suppose E would like to send an IP datagram to B, and assume that E's ARP cache does not contain B's MAC address. Will E perform an ARP query to find B's MAC address? Why? In the Ethernet frame (containing the IP datagram destined to B) that is delivered to router R1, what are the source and destination IP and MAC addresses?

c. Suppose Host A would like to send an IP datagram to Host B, and neither A's ARP cache contains B's MAC address nor does B's ARP cache contain A's MAC address. Further suppose that the switch S1's forwarding table contains entries for Host B and router R1 only. Thus, A will broadcast an ARP request message. What actions will switch S1 perform once it receives the ARP request message? Will router R1 also receive this ARP request message? If so, will R1 forward the message to Subnet 3? Once Host B receives this ARP request message, it will send back to Host A an ARP response message. But will it send an ARP query message to ask for A's MAC address? Why? What will switch S1 do once it receives an ARP response message from Host B?

6.4.4 虚拟局域网



这些问题都可以通过虚拟局域网(VLAN)来解决!

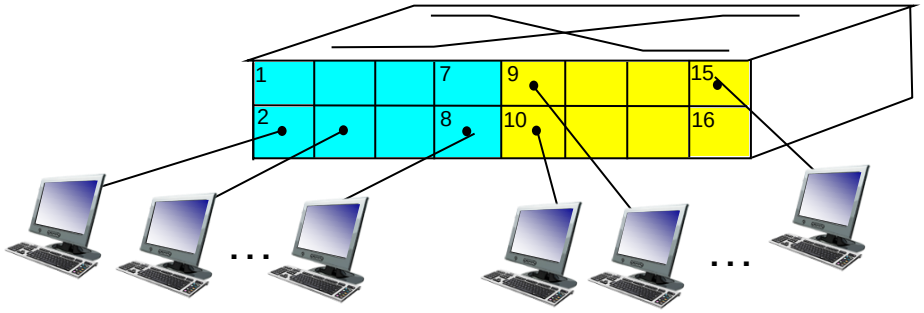
考虑:

- CS用户将办公地点转移到了EE，但是仍然希望连接CS的交换机?
- 如左图这样的连接方式存在的问题，这是一个单一的链路层广播域:
  - 所有的2层广播流量(ARP\DHCP\目的MAC地址未知未知的数据帧)都会穿越整个局域网
  - 安全/隐私问题

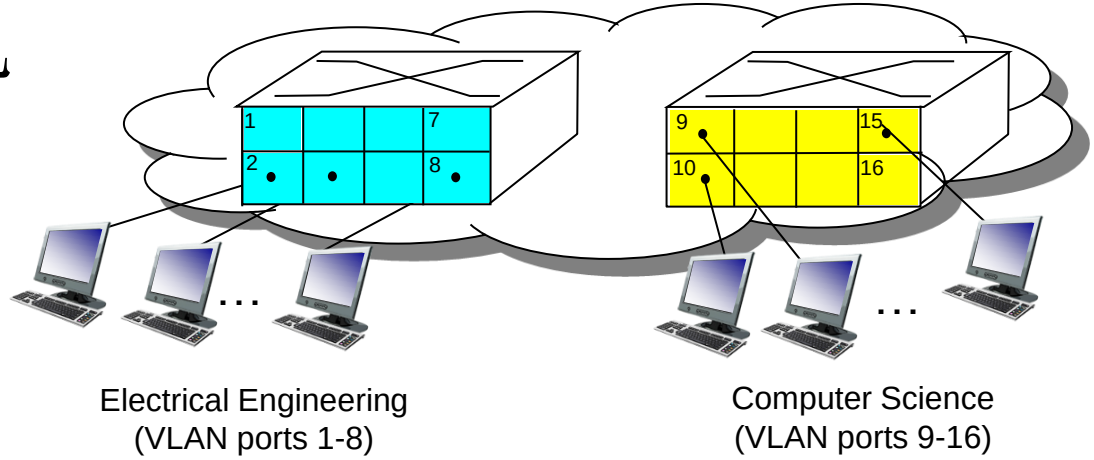
VLAN

虚拟局域网：

- 可以在一个支持VLAN的交换机上配置多个VLANs
- 在右图所示的例子中，两个VLANs可以看作相对独立运行的交换机，从而实现流量隔离

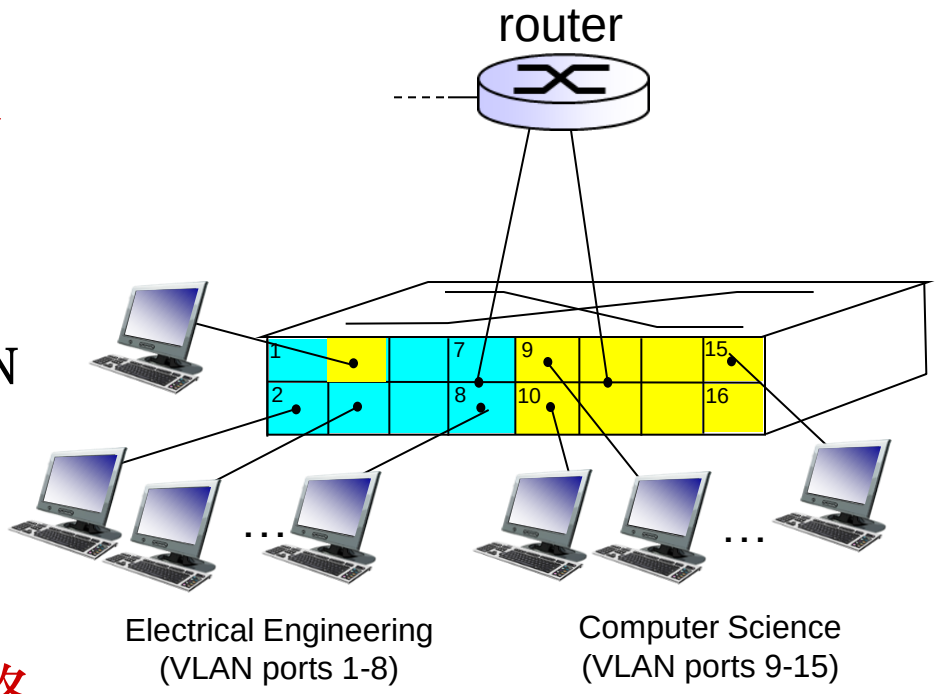


… operates as *multiple* virtual switches

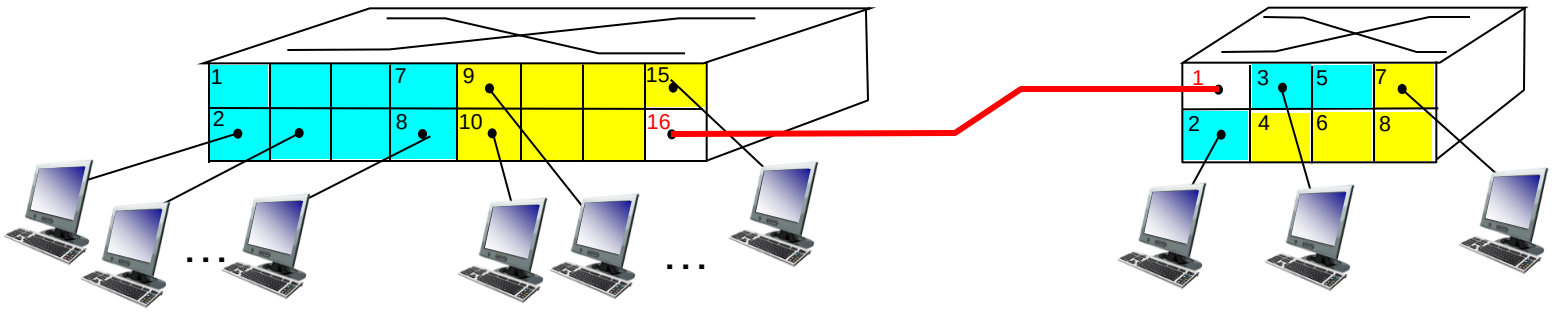


VLAN的功能

- 流量隔离：使得来自1-8号端口的数据帧只能被转发到1-8号端口
  - 既可通过端口号定义VLAN，也可通过MAC地址定义VLAN
- 动态关系：端口可以在不同VLANs之间动态分配
- VLANS之间的转发：通过路由(就像两个独立的交换机)
  - 实际上设备锁通常将交换机和路由器捆绑销售

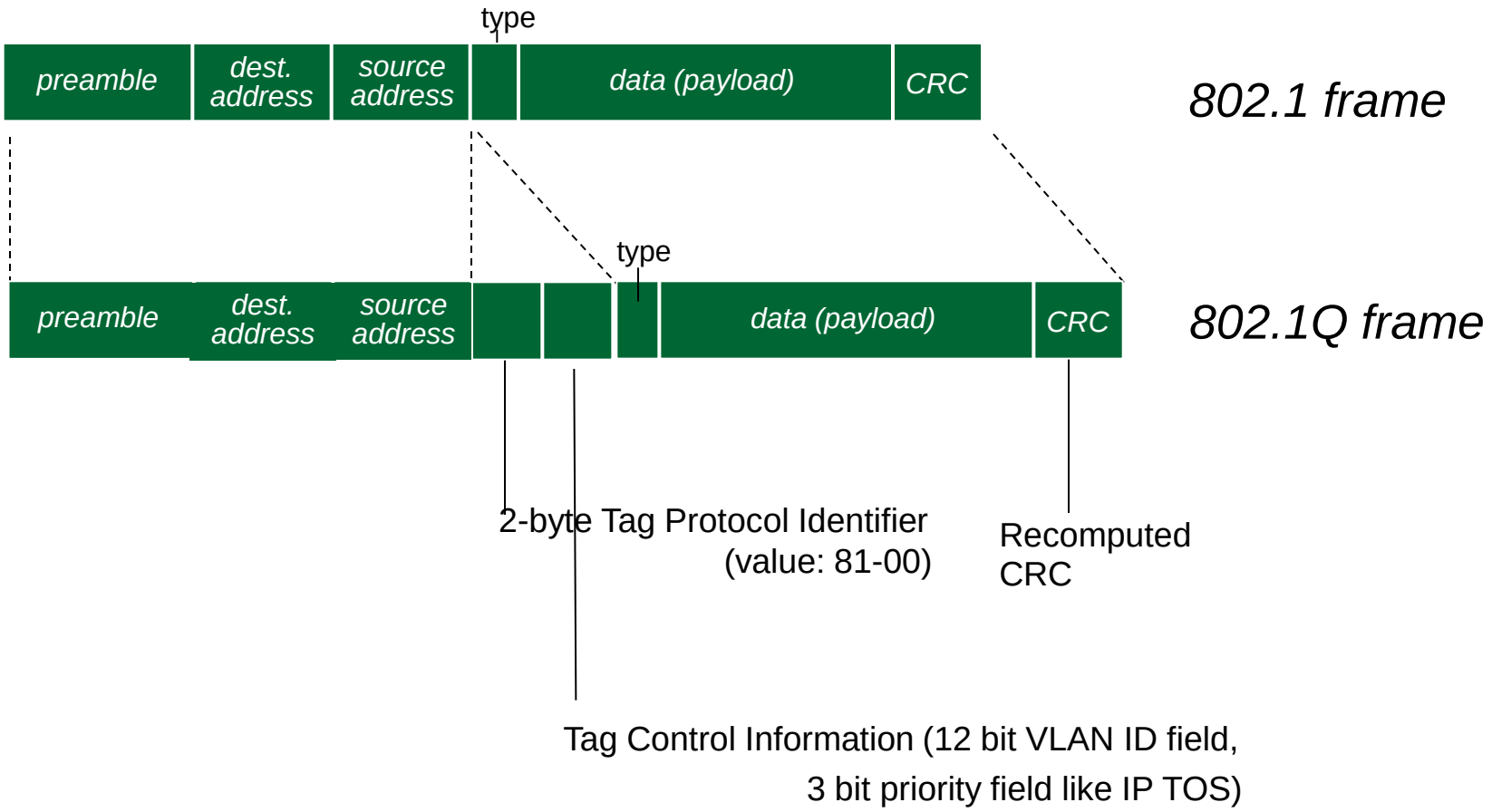


VLANs 可以跨越多个交换机



- 主干端口: VLAN可以跨越多个物理交换机，主干端口用于在不同交换机之间转发数据帧
  - 交换机之间VLAN的数据帧的转发不能采用原来的802.1以太帧的结构(必须要携带VLAN信息)
  - 802.1q 协议用于在数据帧中增加/删除附加的头部信息，当数据帧在主干端口之间转发时。

802.1Q VLAN数据帧格式



VLAN的划分方法

- 按照端口号划分
- 按照MAC地址划分
- 按照网络层地址划分(IPv4)

同学们自己通过VLAN仿真器搭建不同的VLAN

6.6 数据中心网络

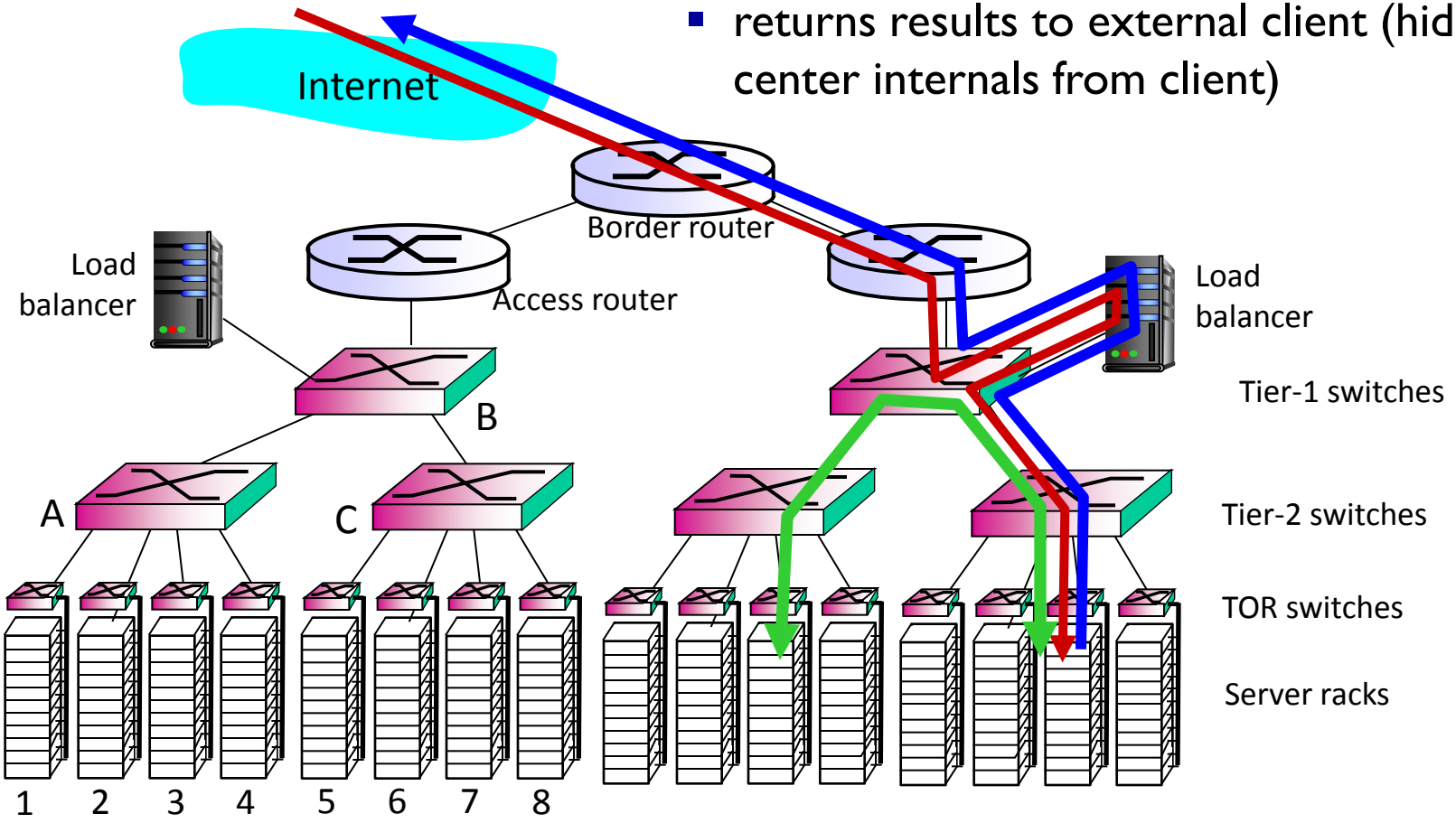
- 数万台至数十万台主机紧密地放置在一起:
  - 电子商务 (e.g. Amazon)
  - 内容服务 (e.g., YouTube, Akamai, Apple, Microsoft)
  - 搜索引擎, 数据挖掘 (e.g., Google)
- 挑战:
  - 多个应用程序，每个应用程序为大量客户机提供服务
  - 管理/平衡负载，避免处理瓶颈、网络瓶颈和数据瓶颈

Inside a 40-ft Microsoft container, Chicago data center

6.6 数据中心网络

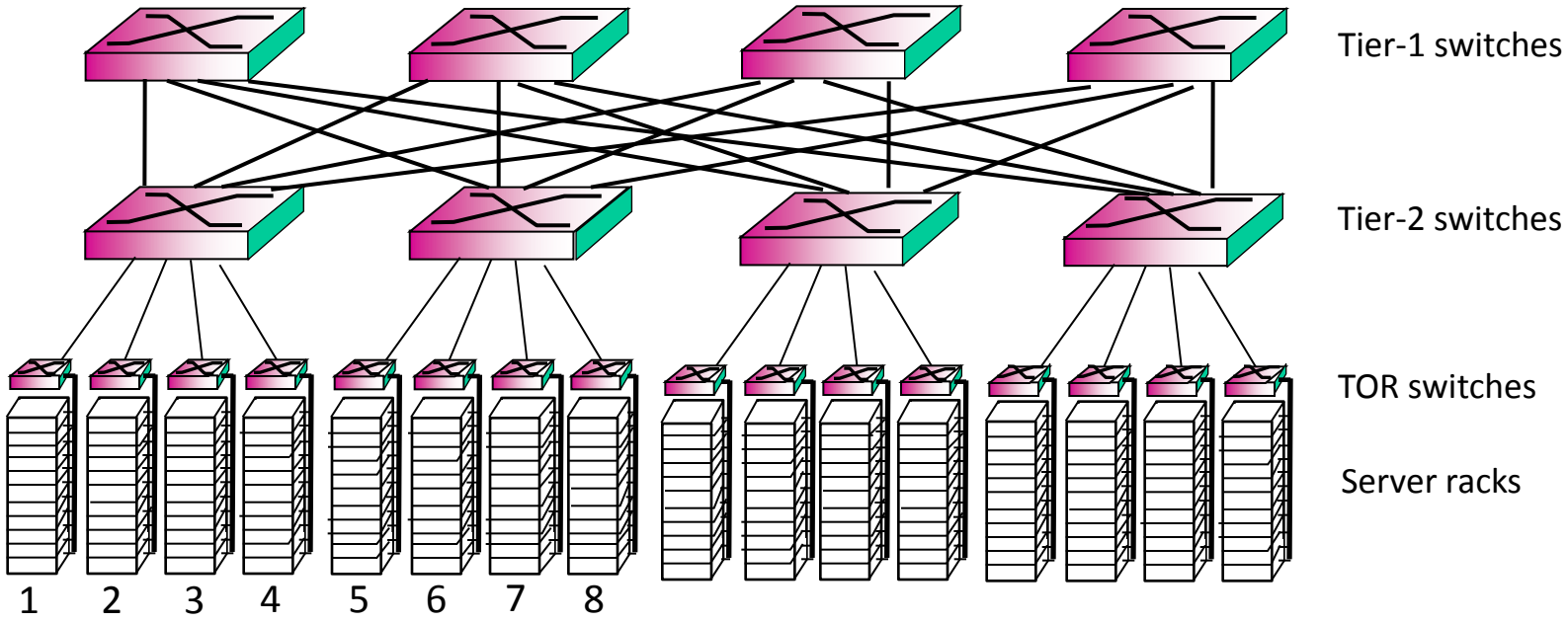
load balancer: application-layer routing

- receives external client requests
- directs workload within data center
- returns results to external client (hiding data center internals from client)



6.6 数据中心网络

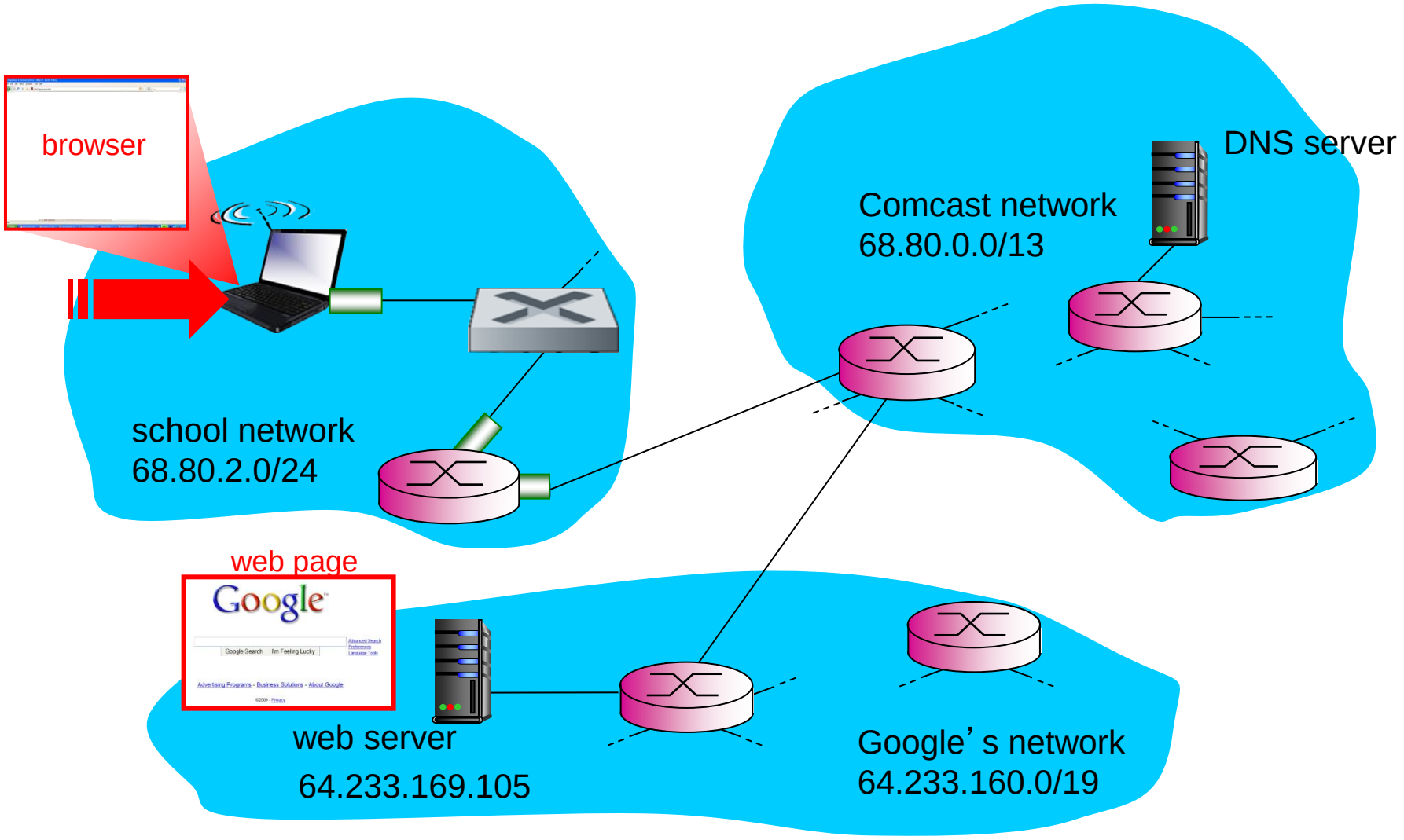
- rich interconnection among switches, racks:
  - increased throughput between racks (multiple routing paths possible)
  - increased reliability via redundancy



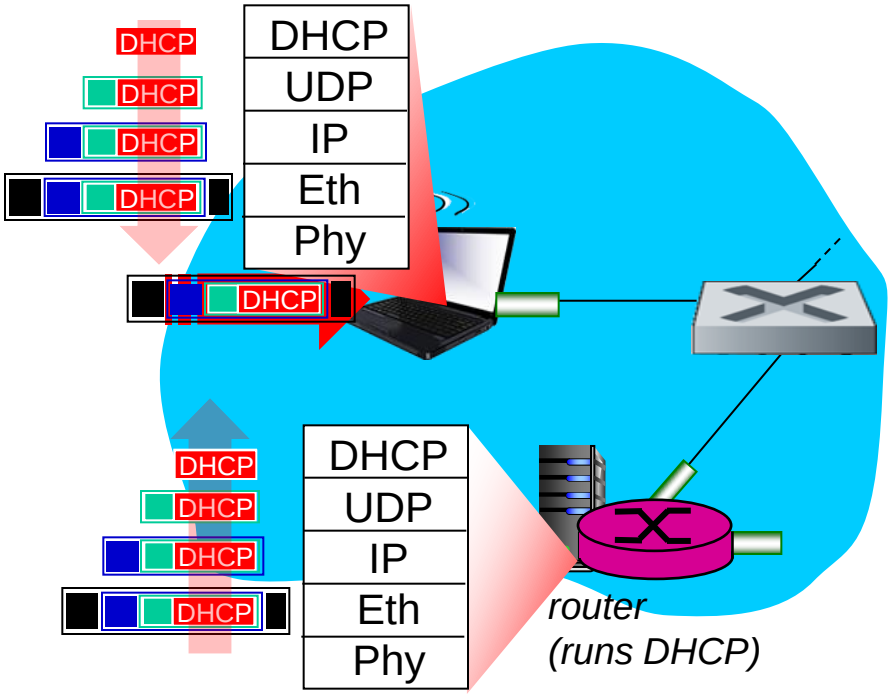
6.7 回顾：web页面请求的历程

- TCP/IP协议栈自顶向下协议层的学习已经完成!
  - 应用层，运输层，网络层，链路层
- 我们把这些协议放在一起: 综合的案例!
  - 目标：识别、回顾、理解看似简单的场景中涉及的协议：访问www页面；
  - 方案：学生使用笔记本接入校园网，请求/接收www.google.com

A day in the life: scenario

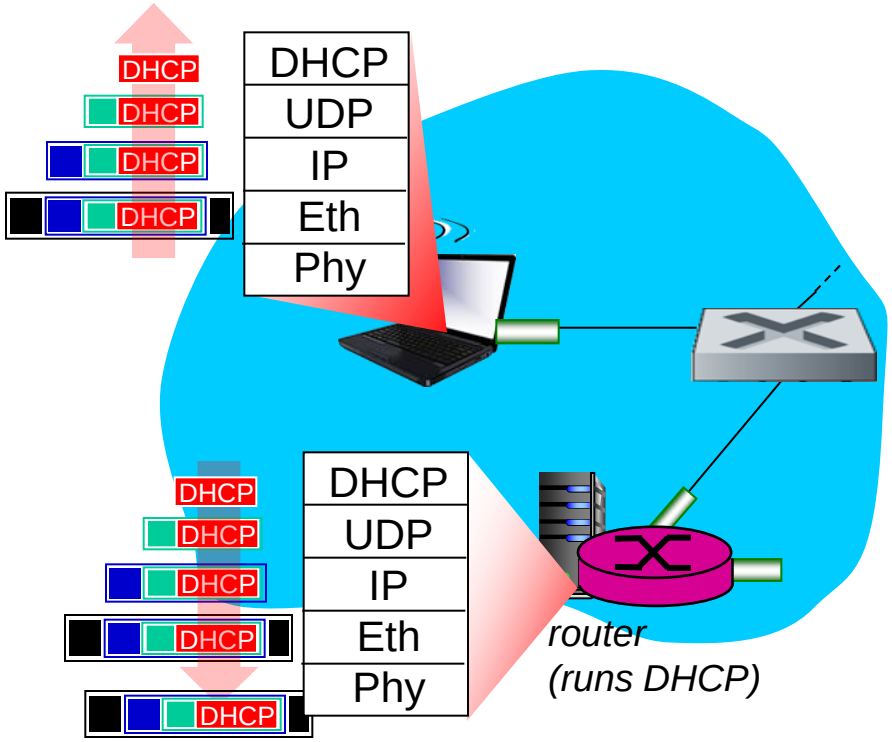


A day in the life… connecting to the Internet



- connecting laptop needs to get its own IP address, addr of first-hop router, addr of DNS server: use *DHCP*
- DHCP request *encapsulated* in *UDP*, encapsulated in *IP*, encapsulated in *802.3* Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

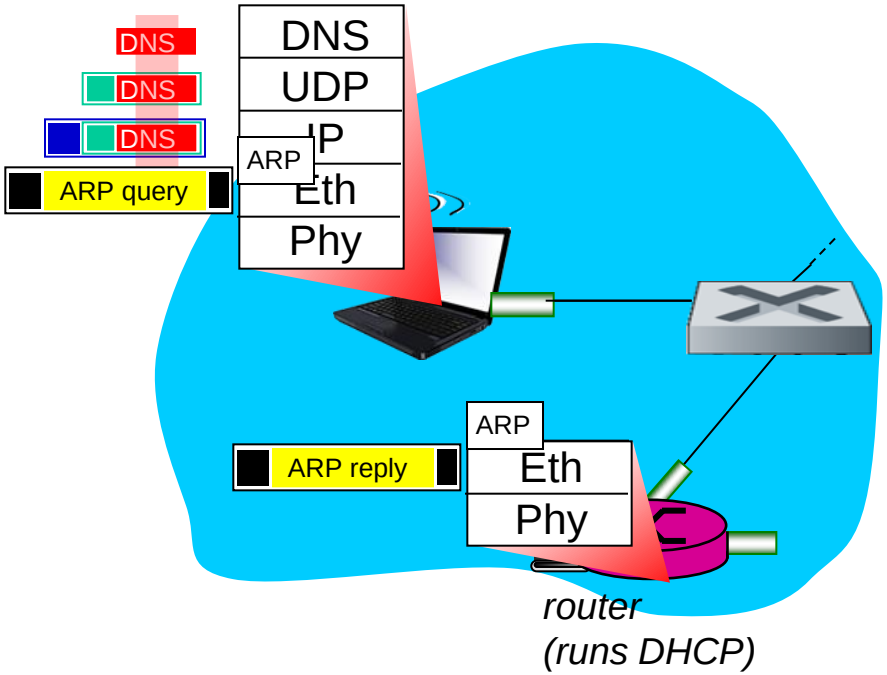
A day in the life… connecting to the Internet



- DHCP server formulates *DHCP ACK* containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
  - encapsulation at DHCP server, frame forwarded (*switch learning*) through LAN, demultiplexing at client
  - DHCP client receives DHCP ACK reply

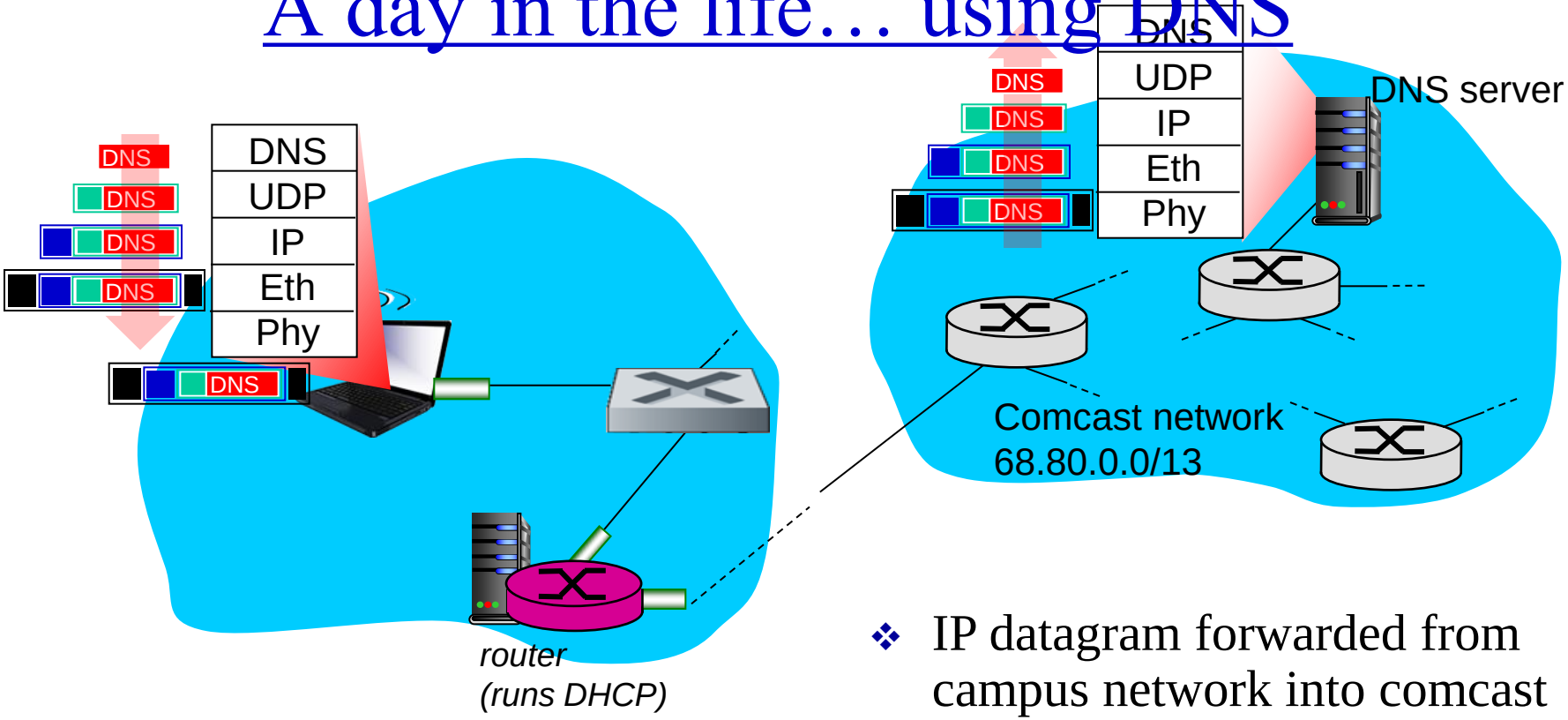
Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

A day in the life… ARP (before DNS, before HTTP)



- before sending *HTTP* request, need IP address of www.google.com: *DNS*
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: *ARP*
- ARP query broadcast, received by router, which replies with ARP reply giving MAC address of router interface
- client now knows MAC address of first hop router, so can now send frame containing DNS query

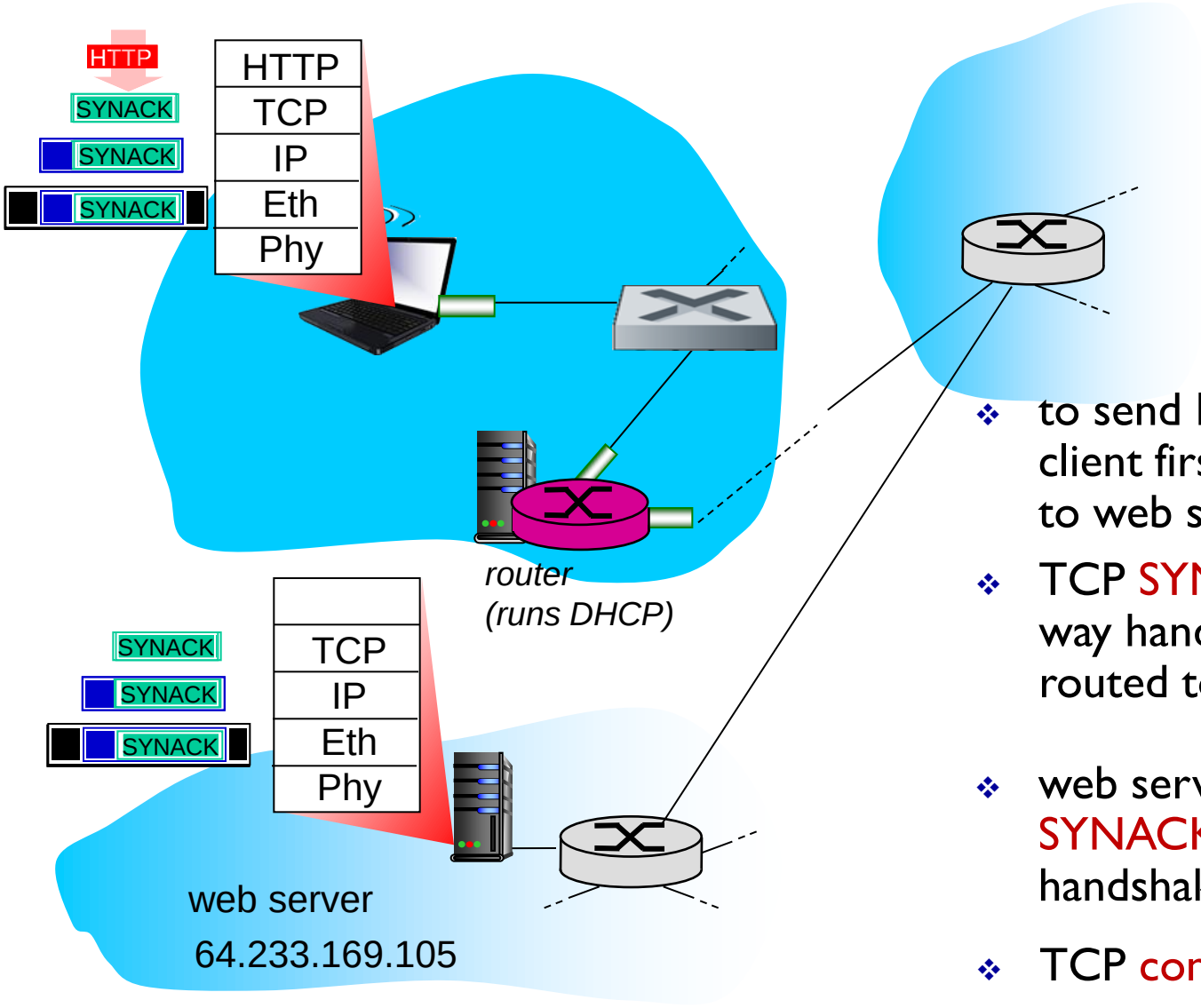
A day in the life… using DNS



- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router

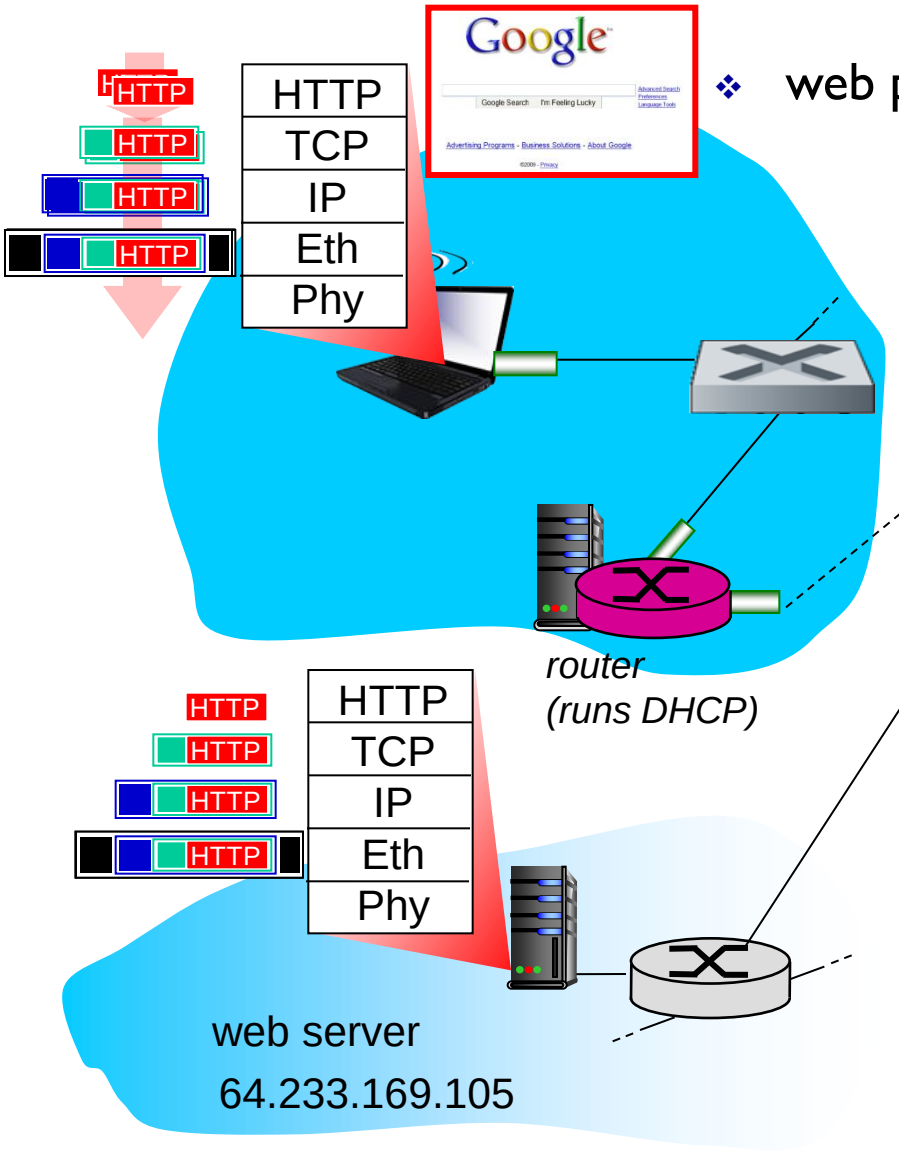
- IP datagram forwarded from campus network into comcast network, routed (tables created by RIP, OSPF, IS-IS and/or BGP routing protocols) to DNS server
- demux'ed to DNS server
- DNS server replies to client with IP address of www.google.com

A day in the life…TCP connection carrying HTTP



- to send HTTP request, client first opens *TCP socket* to web server
- TCP SYN segment (step 1 in 3-way handshake) inter-domain routed to web server
- web server responds with TCP SYNACK (step 2 in 3-way handshake)
- TCP connection established!

A day in the life… HTTP request/reply



- web page finally (!!!) displayed

- HTTP request sent into TCP socket
- IP datagram containing HTTP request routed to www.google.com
- web server responds with HTTP reply (containing web page)
- IP datagram containing HTTP reply routed back to client

第6章 小结

- 链路层提供的主要服务:
  - 差错检查, 纠错
  - 共享广播信道的接入: multiple access
  - 链路层的寻址
- 各种链路层技术
  - Ethernet(以太网)
  - switched LANS, VLANs
- 回归: web请求的历程

对课程建设的意见

- 课程内容
- 课程实验
- 课程设计
- 授课形式
- 互动环节

请写出真实想法!
特别是改进意见!

个人的一些感想与大家共勉

- 勿于浮沙筑高台
- 坚持目标
- 心态的重要性
- 本科成绩的重要性(保研)
- 本科的大学生竞赛和科研
- 重视同学(未来的职业圈)
- 能力和勤奋(职业选择)