链路层和局域网

The link layer and LANs

本课内容

- 学习链路层服务背后的原理
- 学习不同的链路层技术、协议的细节

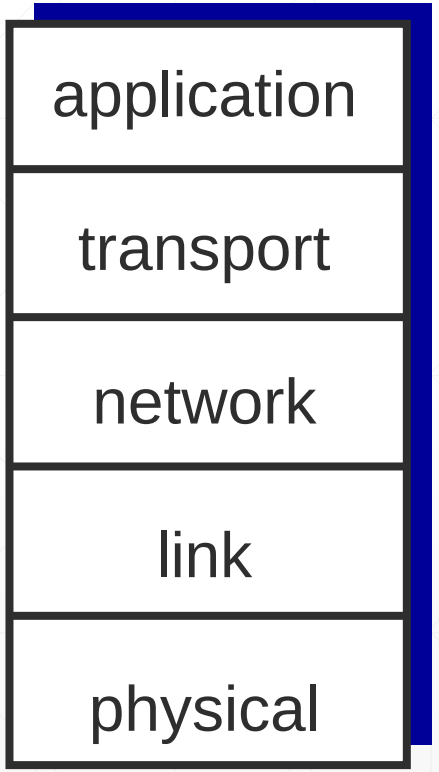
- 链路层概述
- 差错检测
- 广播信道多接入复用
- 链路层寻址
- 局域网：以太网、交换机、VLANs

链路层概述

Introduction to the link layer

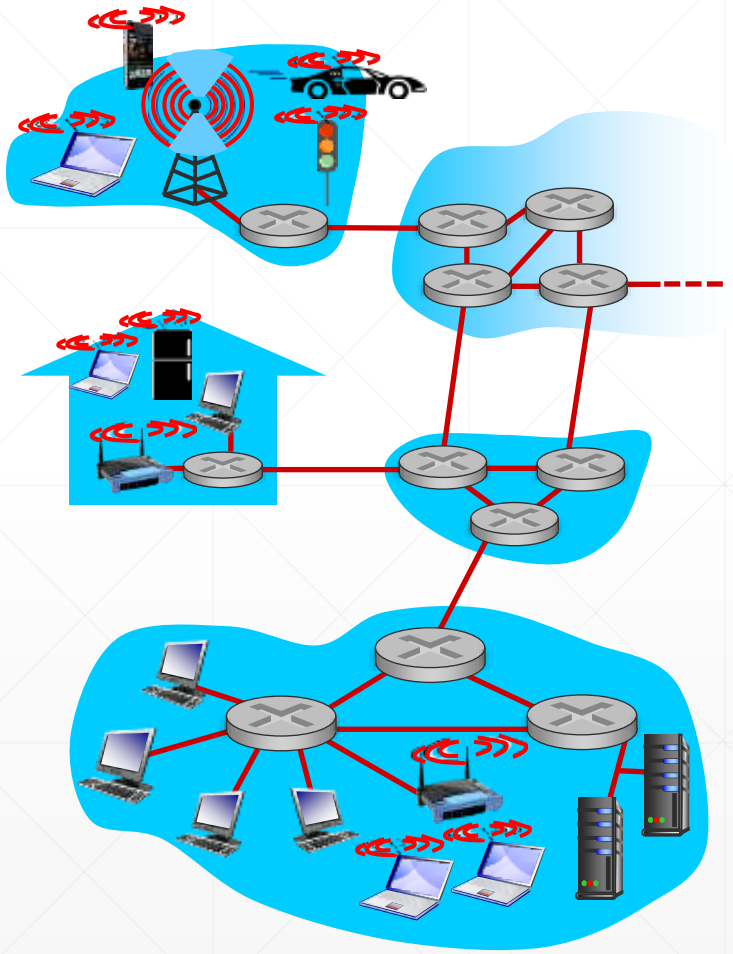
链路层的位置

- 链路层/数据链路层
- Link layer; data link layer
- 以后每一次课件都会有这张图!



链路层基础术语

- **节点（nodes）**：主机、路由器等
- **链路（links）**：在通信路径上，连接相邻节点的通信线路
  - 有线（wired links）
  - 无线（wireless links）
  - 局域网（LANs）
- 链路层中的数据包（即layer-2 packet），我们称之为**帧（frame）**。它包含了网络层的数据报（datagram）。
- 链路层负责从一个节点，通过一条链路，将数据报传送到**物理相邻**的另一个节点。

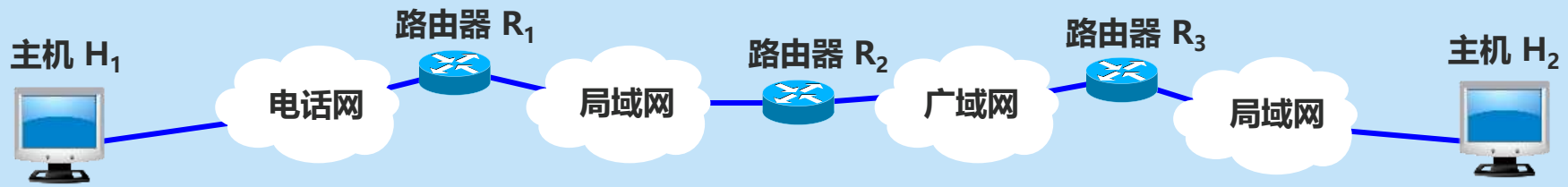


链路层基础术语

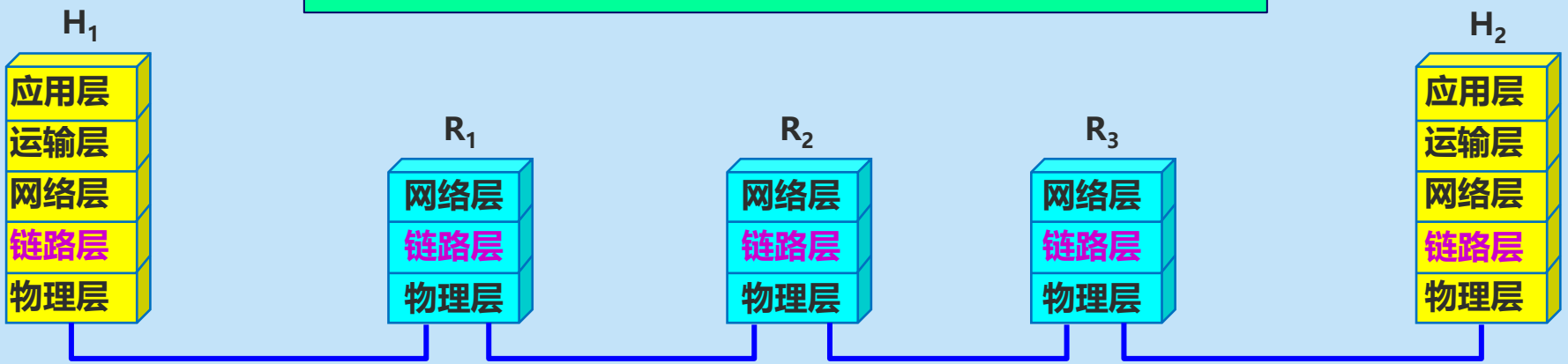
- 类比旅行过程：
- 假设有一趟旅行从暨南大学到清华大学
  - 打的：从暨大到白云机场
  - 飞机：从白云机场到首都机场
  - 大巴：从首都机场到清华大学

- 类比旅行过程：
- 游客 → **数据报（datagram）**
- 比较牵强的，带了票的游客 → **帧（frame）**
- 每段旅行 → **通信链路（communication link）**
- 旅行方式 → **链路层协议（link layer protocol）**
- 旅行社给出的旅行计划 → **路由算法（routing algorithm**，这个是网络层任务）

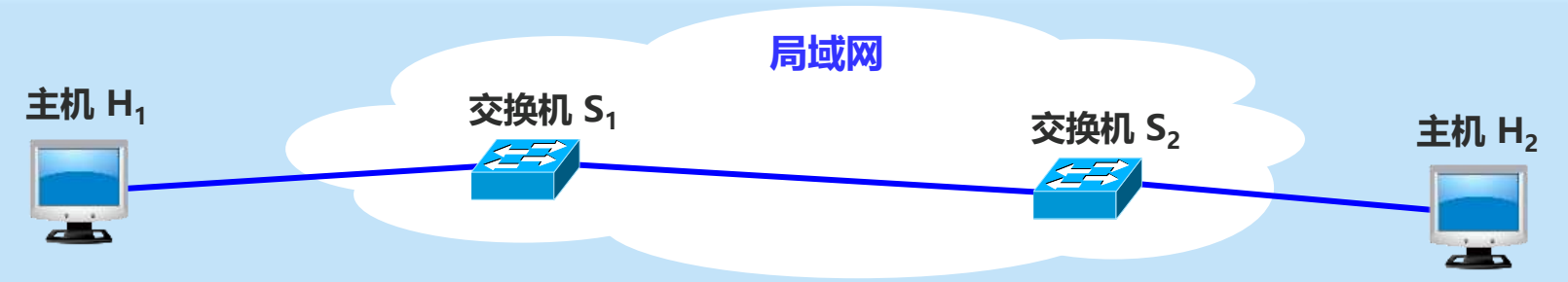
数据链路层是实现设备之间通信的非常重要的一层



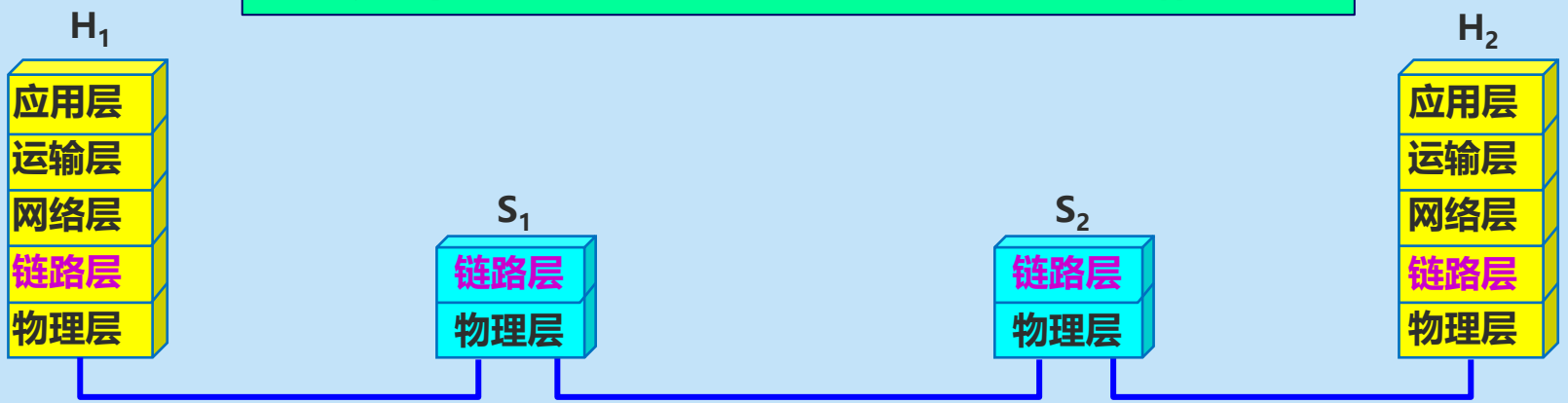
网络中的主机、路由器等都必须实现数据链路层



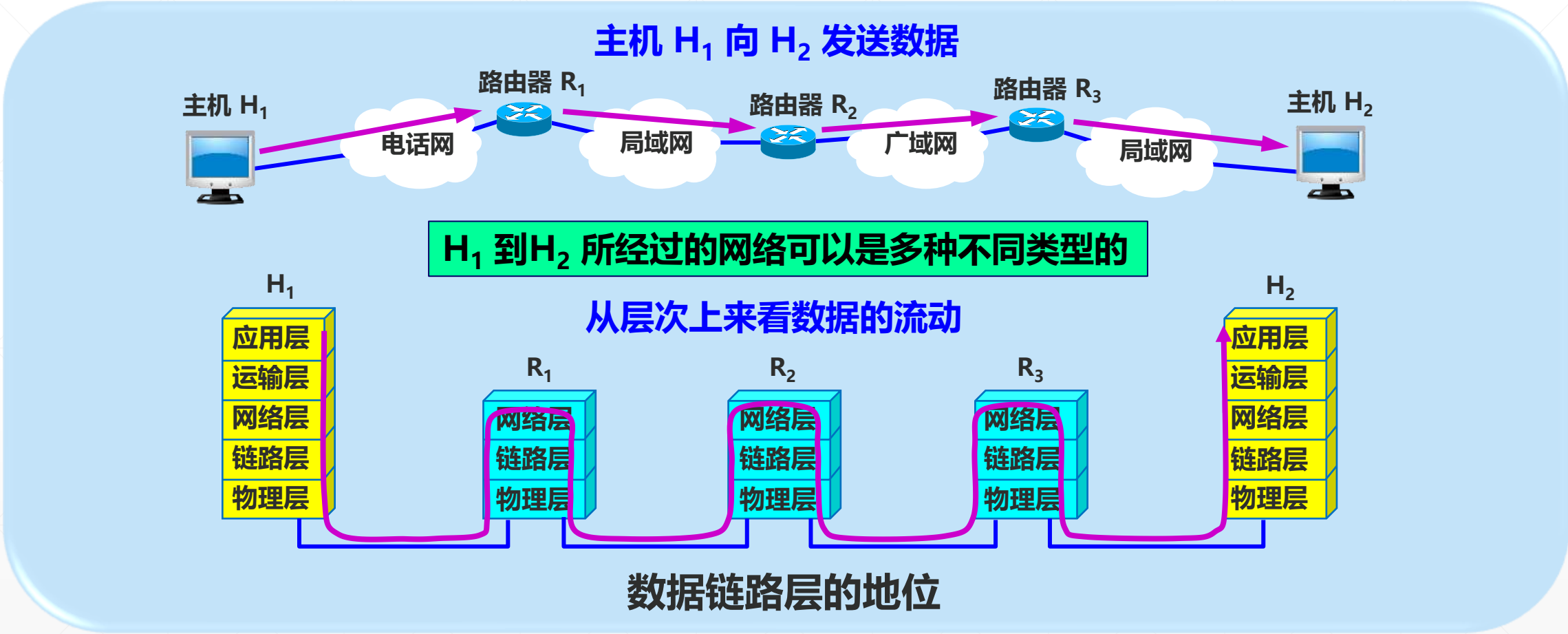
数据链路层是实现设备之间通信的非常重要的一层



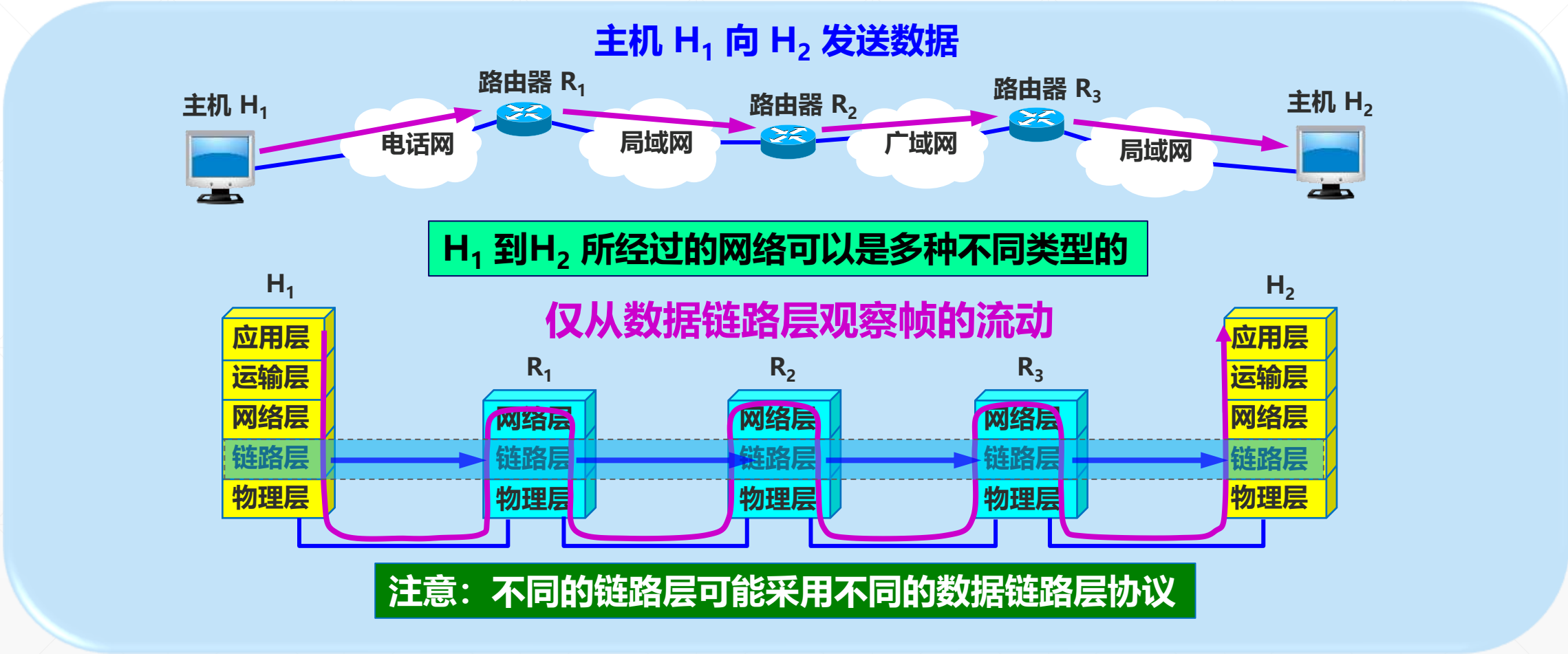
局域网中的主机、交换机等都必须实现数据链路层



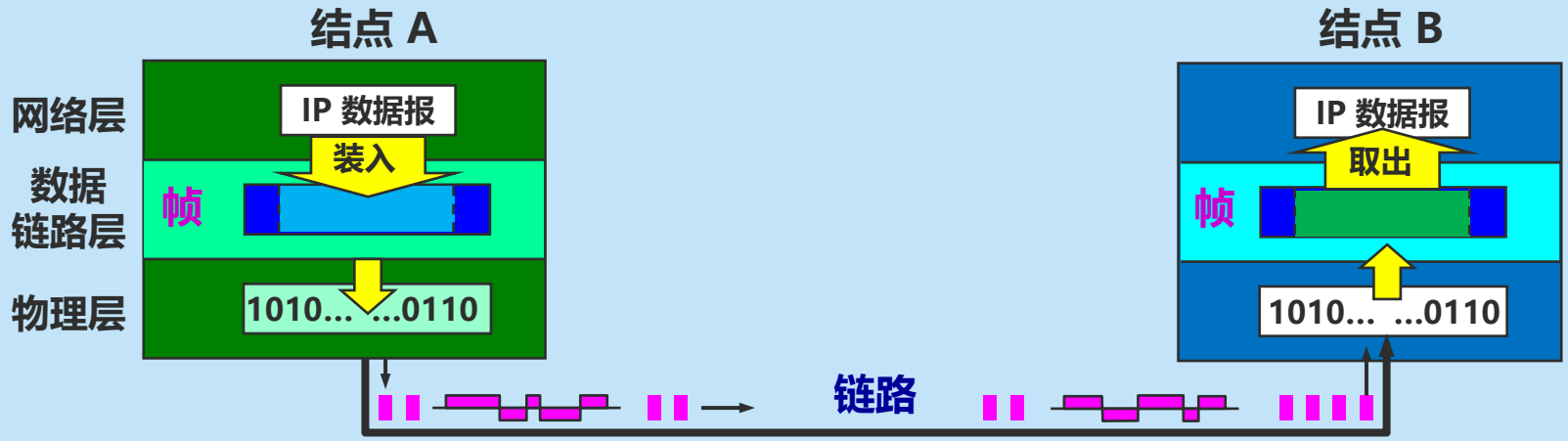
数据链路层的作用


主机 H1 向 H2 发送数据
主机 H1
路由器 R1
路由器 R2
路由器 R3
主机 H2
电话网
局域网
广域网
局域网
H1 到H2 所经过的网络可以是多种不同类型的
H1
R1
R2
R3
H2
从层次上来看数据的流动
应用层
运输层
网络层
链路层
物理层
数据链路层的地位

数据链路层的作用


主机 H1 向 H2 发送数据
主机 H1
路由器 R1
路由器 R2
路由器 R3
主机 H2
电话网
局域网
广域网
局域网
H1 到H2 所经过的网络可以是多种不同类型的
H1
H2
仅从数据链路层观察帧的流动
R1
R2
R3
应用层
运输层
网络层
链路层
物理层
网络层
链路层
物理层
网络层
链路层
物理层
网络层
链路层
物理层
应用层
运输层
网络层
链路层
物理层
注意：不同的链路层可能采用不同的数据链路层协议

数据链路层传送的是帧



(a) 三层的简化模型

(b) 只考虑数据链路层

使用点对点信道的数据链路层

链路层在哪里实现?

- 链路层在每一个主机/路由器/交换机都必须实现
- 通常，链路层实现为“适配器”的形式（adapter，或者称之为network interface card，NIC）或者在芯片上实现：
  - 例如：有线网卡、无线网卡、以太网芯片组等等
  - 往往同时包含链路层和物理层的实现
- 适配器/NIC再接入到主机的系统总线（例如，通过PCIE总线连接电脑主板）
- 链路层实现为适配器，包括了硬件、软件和固件/驱动程序的实现。

差错检测

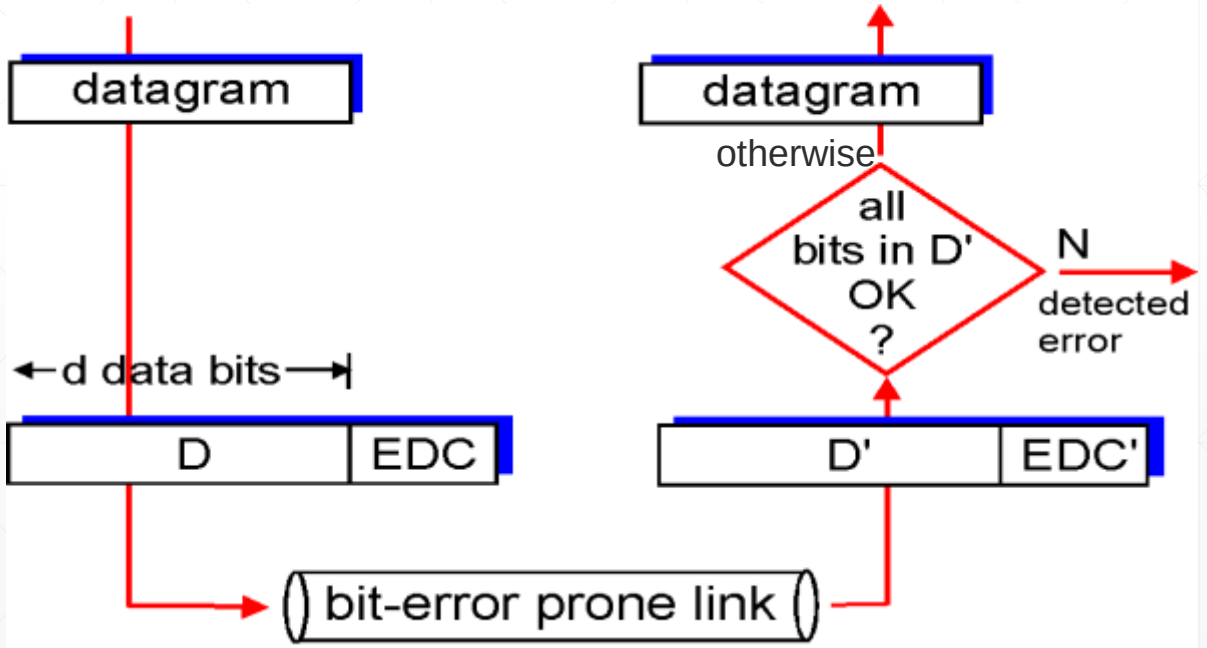
Error detection and correction

为什么需要差错检测?

- 在传输过程中，可能因为信号放大、噪声等问题，导致比特差错：
  - 欲发送1，变成了0；欲发送0，变成了1。
  - 可能是1位出错，也可能是多位比特差错
- 在一段时间内，传输错误的比特占所传输比特总数的比率称为**误码率 BER (Bit Error Rate)**。
- 误码率与信噪比有很大的关系。
- 为了保证数据传输的可靠性，在计算机网络传输数据时，必须采用各种差错检测措施。
- 在数据链路层传送的帧中，广泛使用了**循环冗余检验 CRC** 的检错技术。由接收者检查或更正比特差错，而不需要进行重传。

差错检测

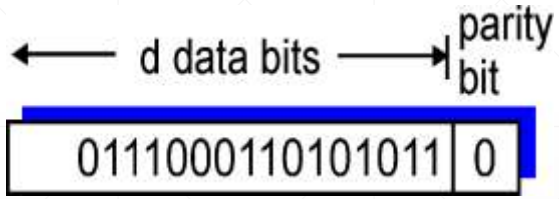
- 用EDC表示Error detection and correction bits，用D表示需要保护的数据（可能包括具体的报文和首部等).
- 差错检测并不是100%可靠的：
  - 可能会漏掉某些错误
  - 可能无法改正错误
  - 通常，EDC越长，查错、改错的能力越强。



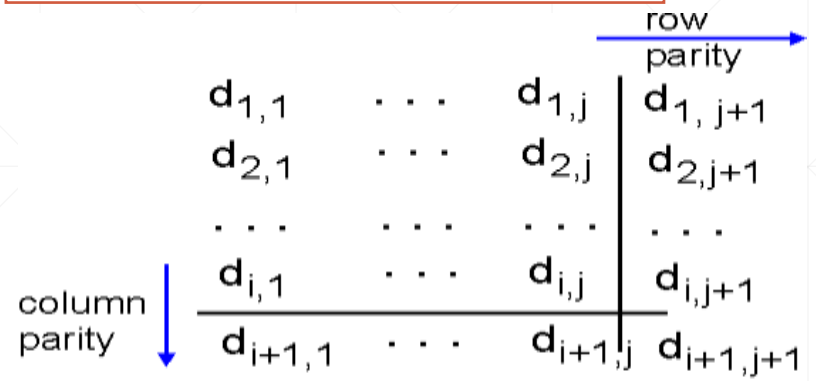
简单的例子 – 奇偶校验

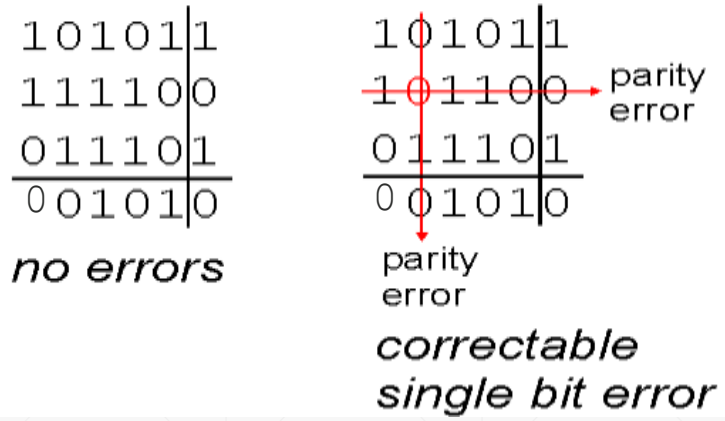
- 若数据中由奇数个1，则奇偶校验位为1，反之为0.

单个位奇偶校验:
- 检测单个位的错误



二维奇偶校验:
- 检测并更正单个位的错误



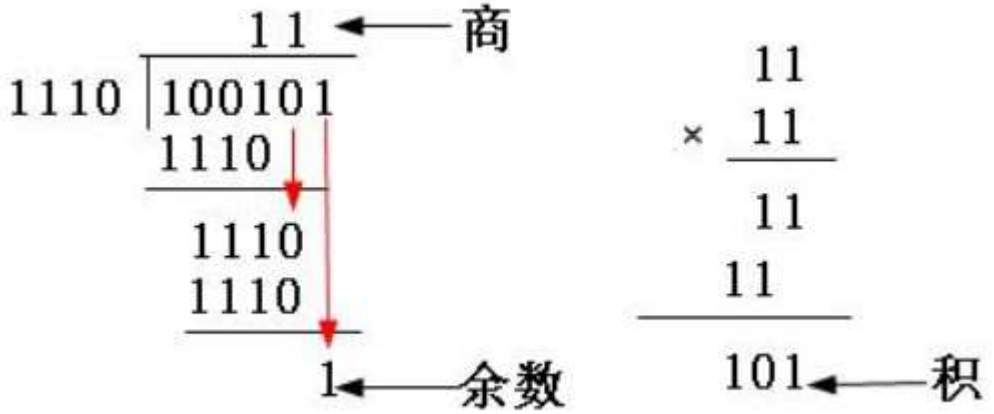


循环冗余校验 – CRC (Cyclic Redundancy Check)

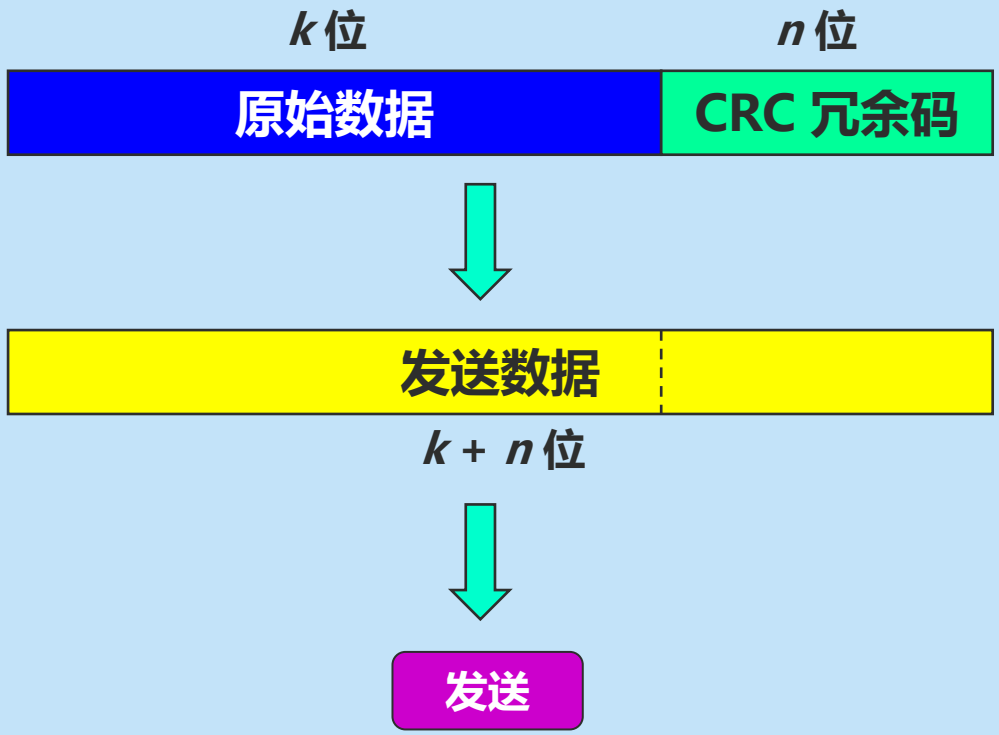
- CRC是比奇偶校验复杂的、强大的一种校验方式。
- 模2运算（加法、减法、乘法、除法）：
  - 1+1=0, 0+1=1, 1+0=1, 0+0=0
  - 1-1=0, 0-1=1, 1-0=1, 0-0=0;

无进位、无借位

  - 乘除法例子 →
  - 加减法效果相同！



循环冗余检验的原理

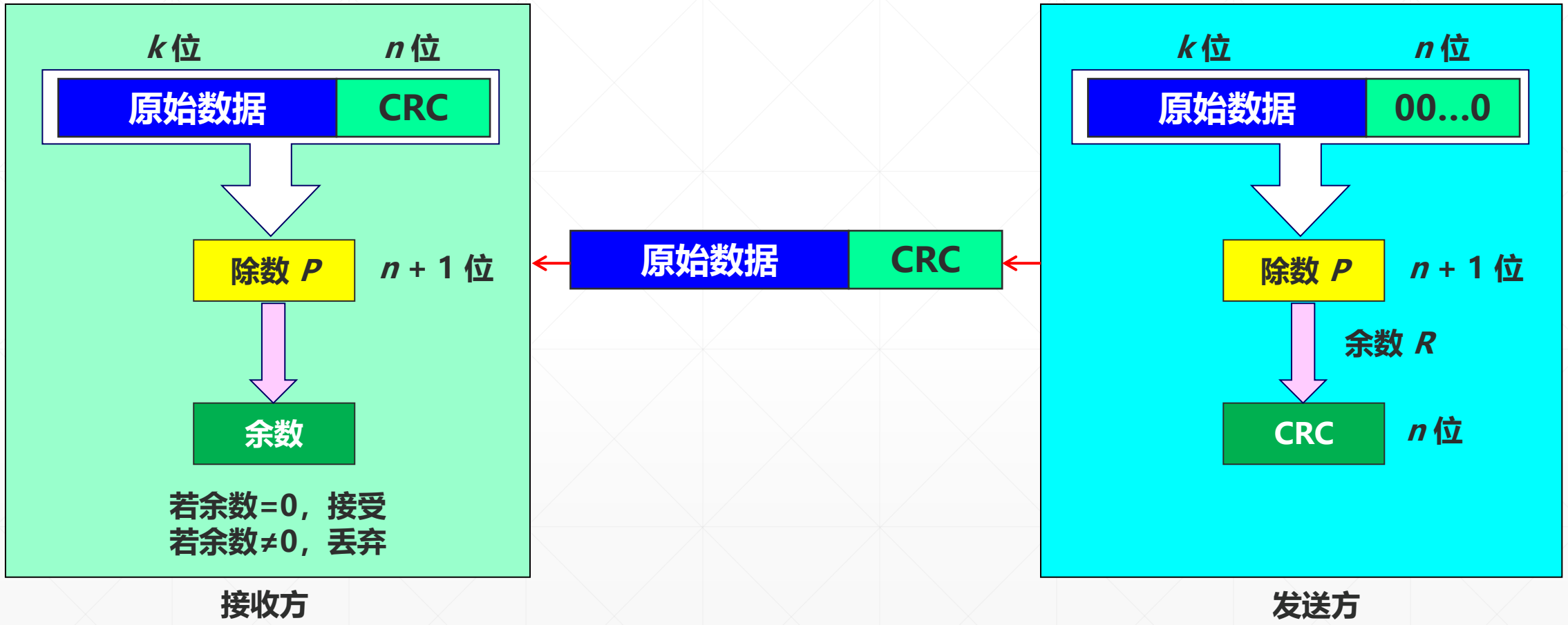


- 在发送端，先把数据划分为组。假定每组 k 个比特。
- 在每组 M 后面再添加供差错检测用的 n 位冗余码，然后一起发送出去。

冗余码的计算

- 用二进制的模 2 运算进行 2^n 乘 M 的运算，这相当于在 M 后面添加 n 个 0。
- 得到的 $(k + n)$ 位的数除以事先选定好的长度为 $(n + 1)$ 位的除数 P，得出商是 Q 而余数是 R，余数 R 比除数 P 少 1 位，即 R 是 n 位。
- 将余数 R 作为冗余码拼接在数据 M 后面，一起发送出去。

冗余码的计算


k 位
n 位
原始数据
CRC
除数 P
n + 1 位
余数
若余数=0，接受
若余数≠0，丢弃
接收方
原始数据
CRC
k 位
n 位
原始数据
00…0
除数 P
n + 1 位
余数 R
CRC
n 位
发送方

接收端对收到的每一帧进行 CRC 检验

- (1) 若得出的余数 $R = 0$，则判定这个帧没有差错，就**接受** (accept)。
- (2) 若余数 $R \neq 0$，则判定这个帧有差错，就**丢弃**。
- 但这种检测方法并不能确定究竟是哪一个或哪几个比特出现了差错。
- 只要经过严格的挑选，并使用位数足够多的除数 P，那么出现检测不到的差错的概率就很小很小。

冗余码的计算举例

- 现在 $k = 6$, $M = 101001$。
- 设 $n = 3$, **除数** $P = 1101$,
- 被除数是 $2^nM = 101001000$。
- 模 2 运算的结果是：**商** $Q = 110101$，**余数** $R = 001$。
- 把余数 R 作为**冗余码**添加在数据 M 的后面发送出去。发送的数据是：$2^nM + R$，即：101001001，共 $(k + n)$ 位。

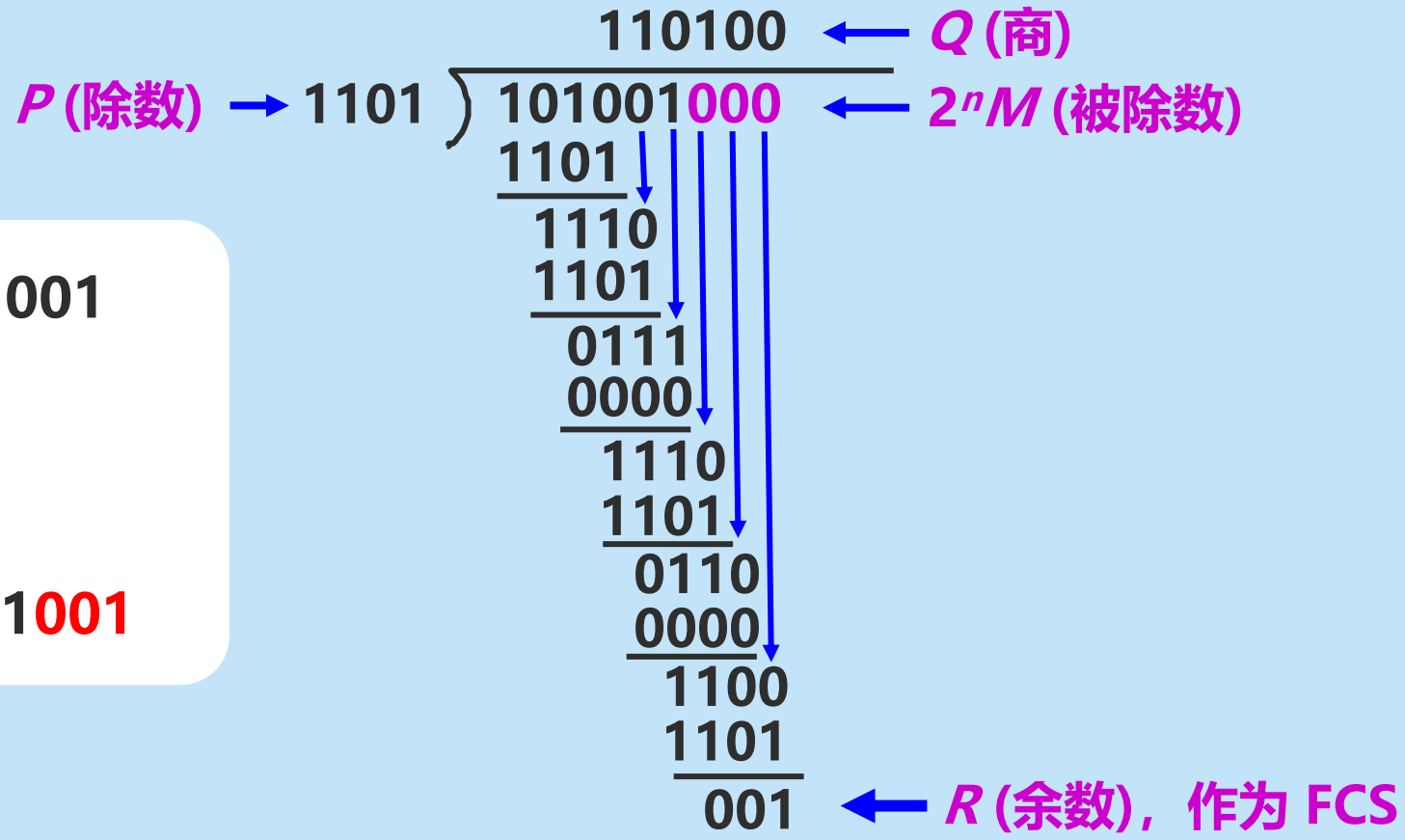
循环冗余检验的原理说明

原始数据 M = 101001

除数 P = 1101

得到：

发送数据 = 101001001



循环冗余校验 – 注意事项

- 选择除数，可以随机选择，也可以按照既定标准选择。除数往往也用多项式表示，所以CRC又称多项式编码方法，这个多项式也称为生成多项式
  - 例：生成多项式$G(x) = x^4 + x^3 + 1$，意味着除数是11001
  - 按照国际通行标准选择，最高位和最低为必须为1。
- 余数的位数一定是比除数位数小一位，即使前面全是0，也不要省略。
- 帧检验序列 FCS – Frame Check Sequence
  - 这里的冗余码就称为FCS
  - 由其他的校验方法，也能得到对应的FCS。

循环冗余校验 – 注意事项

- CRC可以做到“无差错接受”
  - “无差错接受”是指：“凡是接受的帧（即不包括丢弃的帧），我们都能以非常接近于 1 的概率认为这些帧在传输过程中没有产生差错”。
  - 也就是说：“凡是接收端数据链路层接受的帧都没有传输差错”（有差错的帧就丢弃而不接受）。
  - 或称之为“无比特差错”
- 但是，并不能做到可靠传输 –（发送什么就收到什么）
  - 例如，顺序
  - 必须加上确认和重传机制（其他层协议负责）

广播信道、信道复用

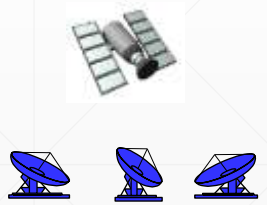
Broadcast channel and multiple access protocols

广播信道

- 链路层有两种不同的“链路”形式
  - 点对点（point-to-point）形式
    - 采用PPP协议进行通信，专享信道。
    - 现在已经用的不多，此处不再讲述
  - 广播形式（共享线路或者传输媒介）
    - 局域网
    - 802.11 无线局域网
    - ……

shared wire (e.g., cabled Ethernet)

shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

多接入协议

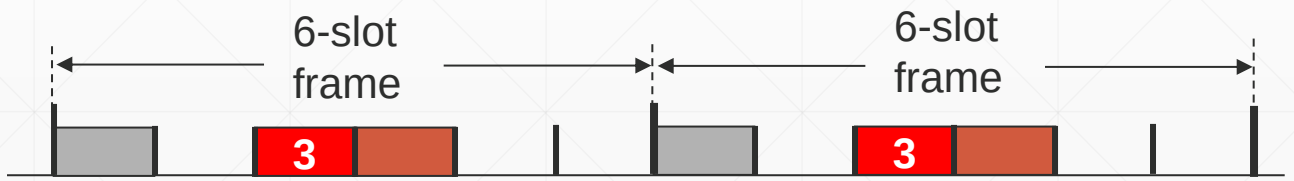
- 广播信道中，往往只有1条共享的广播信道 (single shared broadcast channel)
- 却有两个或多个节点，可能发生同时的数据传输
  - 这会导致干涉的发生（或“碰撞collision”）：如果一个node同时需要发送两个信号，或者多个node需要同时发送多个信号。
- 多接入协议 multiple access protocol
  - 分布式算法 – 每个节点上分别决定是否应该发送信号
  - “how nodes share channel” “何时可以发送信号”
  - 注意，所有的协议通信用到的也是该广播信道自身！即，没有额外的信道用于协调这些节点。

三类多接入协议

- **信道划分 （channel partitioning）**
  - 将信道分为小块（例如分成time slots、根据频率划分、采用不同的编码调制信号等）
  - 每一小块被一个节点独享
- **随机接入 （random access）**
  - 不划分信道，允许碰撞
  - 尝试从“碰撞”中“恢复”
- **受控接入（’taking turns’）**
  - 节点轮流发送信息

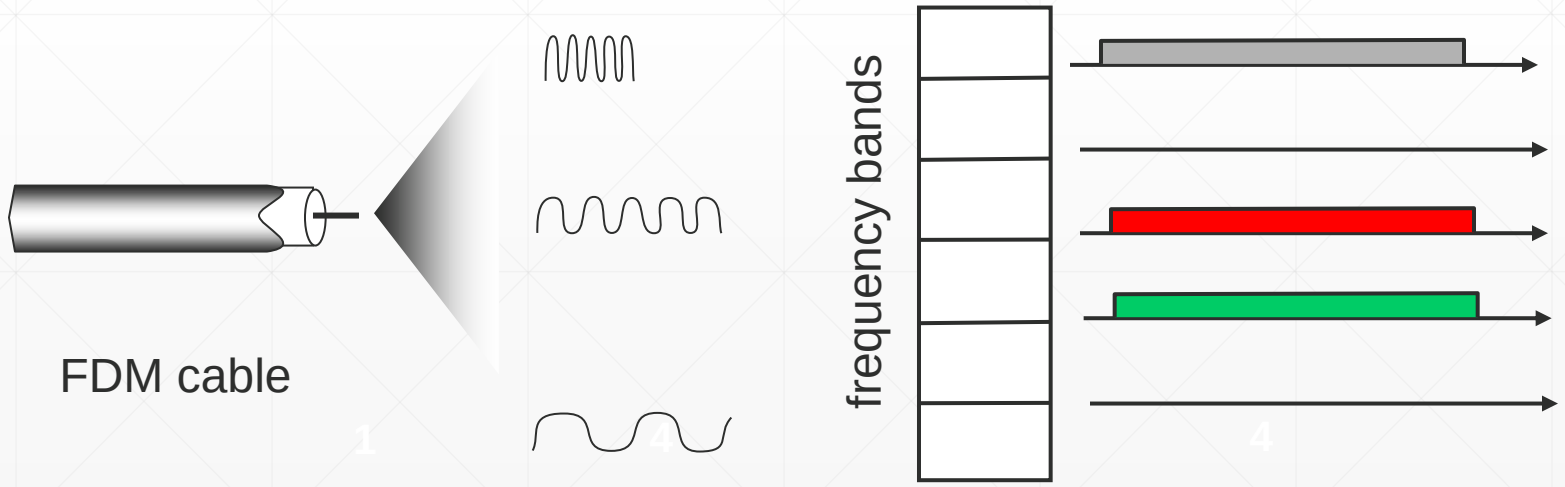
信道划分 – TDMA

- **TDMA：time division multiple access 时分复用**
- 节点分轮（in “rounds”）使用信道
- 每个节点在每一轮都会获得定长的slot（通常 slot的长度等于传输packet的时间）
- 分配但未用的slot则保持闲置
- 例：6个节点，节点1、3、4发送数据，而节点2、5、6不发送数据



信道划分 – FDMA

- **FDMA：frequency division multiple access 频分复用**
- 信道的频谱划分为不同的频段
- 每个节点分配使用固定的频段发送数据
- 对于不发送数据的节点，对应的频段即为闲置状态
- 例：和上页类似的
  6个节点的网络。



信道划分 – CDMA

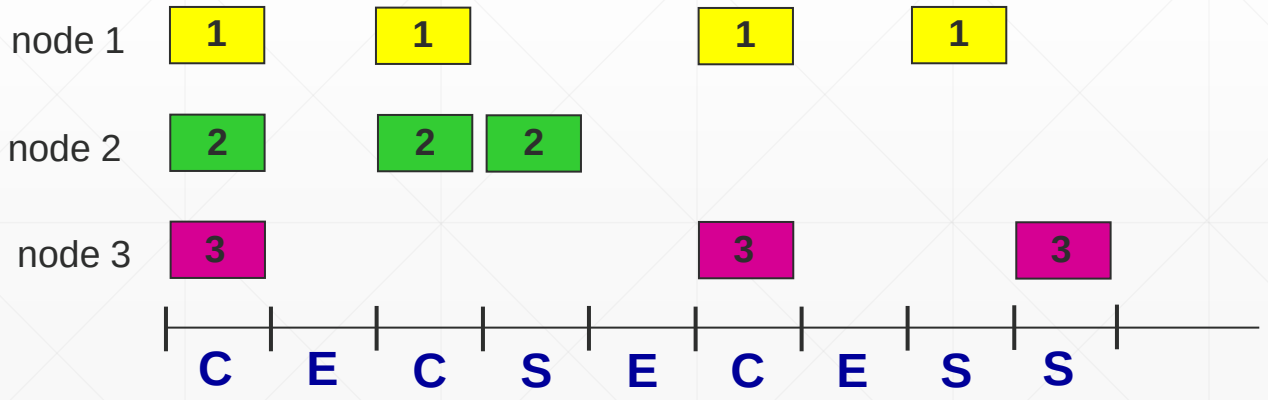
- **CDMA：code division multiple access 码分复用**
- 这个比较复杂，将需要发送的信号通过不同的编码进行调制，然后复用广播信道。
- TDMA、FDMA、CDMA应在数字通信、信号与系统等类似课程中学习。尤其是CDMA。此处不再详述。

随机接入

- **“随机”**：当一个节点有数据需要发送时：
  - 以信道带宽R全速发送；且不需要经过节点之间的事先协调。
- **“碰撞”**：当多个节点同时发送数据，则会导致碰撞
- “随机接入”的信道复用协议则规定：
  - 如何侦测“碰撞”的出现？
  - 如何从“碰撞”中恢复？（例如，延迟一定时间，再重新传输数据）
- 典型的随机接入信道复用协议：
  - Slotted ALOHA; ALOHA
  - CSMA, CSMA/CD, CSMA/CA

Slotted ALOHA

- **基本假设：**
  - 所有的帧大小一致
  - 时间轴划分为等长的时隙（slots），通常是传输一帧的时间
  - 节点均只在时隙开始的时候传输
  - 节点的行为是同步的
  - 如果超过两个节点在一个时隙同时传输，则所有节点能侦测到碰撞
- **操作：**
  - 当一个节点获得一个新的帧，则在下一个时隙开始的时候传输：
    - 如果没有碰撞，则传输成功，该节点成功传输了这个帧。
    - 如果发生碰撞，则传输失败。该节点在以后的每一个时隙均以概率p进行重传，直至传输成功。



Slotted ALOHA

- **优点：**
  - 每个时隙中，都有一个节点可以一直以全速进行传输。
  - 高度分布式算法：只需把所有节点的时隙进行同步即可，其余全部是通过节点自行探测解决。
  - 算法十分简单
- **缺点：**
  - 碰撞，且浪费了很多时隙，也会有很多时隙是闲置的
  - 需要一个时隙去侦测碰撞的发生。但事实上节点可能只需要更少的时间就可以侦测碰撞
  - 需要时钟同步

Slotted ALOHA

- **效率：**
  - 假设在长期运行中，有许多节点，每个节点都有很多帧需要发送。节点数N，每个节点在每个时隙发送的概率都是p。
  - 则“某个给点节点在一个时隙成功发送的概率”是$p(1-p)^{N-1}$。
  - “任意节点在一个时隙成功发送的概率”是$\mathrm{p}_0 = Np(1-p)^{N-1}$。
  - 求出$p = p_*$，使得其最大化p_0。可以得到$p_* = \frac{1}{N}, \mathrm{p}_0(\max) = \left(1-\frac{1}{N}\right)^{N-1}$
  - 对于N很大的情况，可以求得**slotted ALOHA的最大效率**是$\lim_{N\to\infty}\left(1-\frac{1}{N}\right)^{N-1} = \frac{1}{e} = 36.8\%$

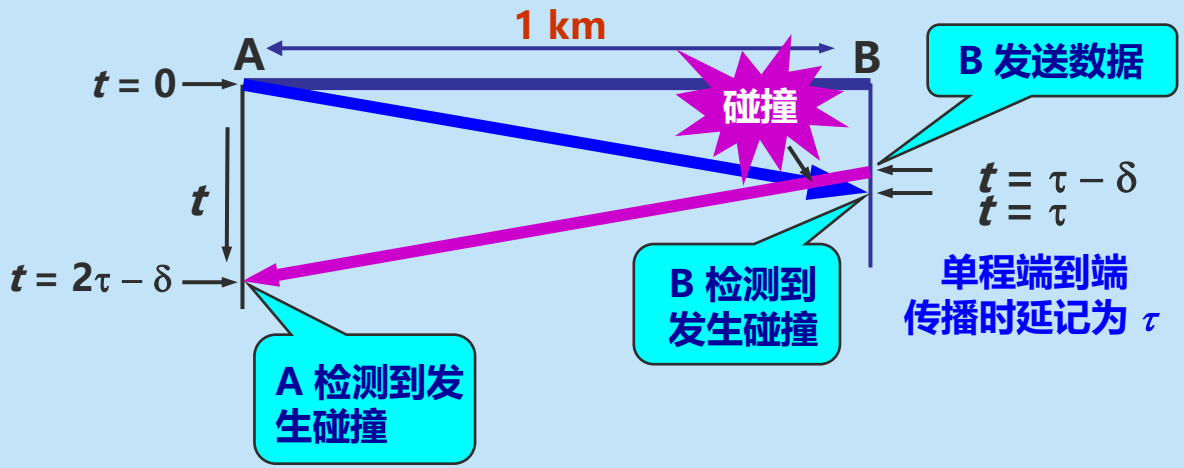
CSMA协议家族

- **CSMA – Carrier Sense Multiple Access 载波监听多点接入**
- 发送数据前先监听广播信道，如果信道空闲，则发送整个帧；如果信道正忙，则延迟发送
- 类比：“不打扰他人说话”
- 此处主要讲述**CSMA/CD**（Collision Detection 碰撞检测）
  - 短时间内先侦测是否发生碰撞，如碰撞发生，则中止传输。
  - IEEE 802.3 有线局域网中使用
  - 无线局域网中不适用：因为无线网络中很难通过信号强度判定碰撞是否发生

CSMA/CD 碰撞检测

- **“碰撞检测”**就是计算机边发送数据边检测信道上的信号电压大小。
- 当几个站同时在总线上发送数据时，总线上的信号电压摆动值将会增大（互相叠加）。
- 当一个站检测到的信号电压摆动值超过一定的门限值时，就认为总线上至少有两个站同时在发送数据，表明产生了碰撞。
- 所谓“碰撞”就是发生了冲突。因此“碰撞检测”也称为“冲突检测”。
- 在发生碰撞时，总线上传输的信号产生了严重的失真，无法从中恢复出有用的信息来。
- 每一个正在发送数据的站，一旦发现总线上出现了碰撞，就要立即停止发送，免得继续浪费网络资源，然后等待一段随机时间后再次发送。

碰撞检测



A 需要单程传播时延的 2 倍的时间，才能检测到与 B 的发送产生了冲突

碰撞检测

- 最先发送数据帧的站，在发送数据帧后至多经过时间2τ（两倍的端到端往返时延）就可知道发送的数据帧是否遭受了碰撞。
- 以太网的端到端往返时延2τ称为**争用期**，或碰撞窗口。
- 经过争用期这段时间还没有检测到碰撞，才能肯定这次发送不会发生碰撞。
- 在10M bit/s的以太网中，传统取$51.2\mu s$为争用期的长度。对应在争用期内，可以发送512bit（64 byte）的数据
  - 即：若前64字节没有发生冲突，则该帧的后续数据不会发生冲突
  - 一半时间（$25.6\mu s$）对应着以太网的最大端到端的长度，约为5km（玻璃纤维中的信号传播速度约为200km/ms）

CSMA/CD 算法

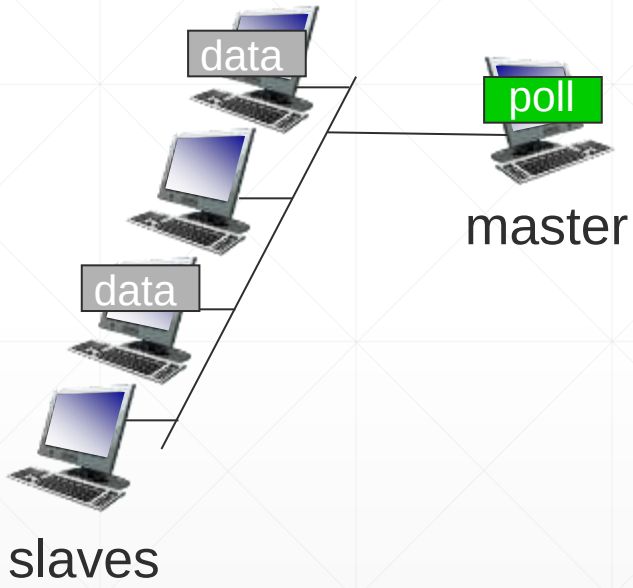
- 1. 网卡（NIC）从网络层收到数据报文，包装成帧
- 2. 如果NIC检测到信道空闲，则开始发送数据；反之，则等待直至信道空闲，然后发送数据
- 3. 如果网卡成功传输了整个帧，并没有遇到其他同时的传输，则该帧传输完成！
- 4. 如果NIC在发送帧的过程中检测到发生碰撞，则中止发送，并发送拥塞信号
- 5. 中止发送后，NIC进入**二进制指数退避**（binary exponential backoff）：
  - 基本退避时间为争用期2τ。
  - 若已经经过m次重传，则从整数集合$\{0,1,\dots,2^m-1\}$中随机取出一个数，定为k。重传所需的时延即为k倍的争用期。
  - 重传16次仍不能成功，则丢弃该帧，并向更高层次的协议报告。
  - Truncated binary exponential backoff：经过m次重传后，整数集合选定为$\{0,1,\dots,2^{m'}-1\}, m'=\min\{m,10\}.$

CSMA/CD的效率

- 用t_{prop}表示在局域网中任意两个节点之间的最大传播时延，t_{trans}表示需要传输最大的帧所用的时间，则CSMA/CD的效率可以用以下公式近似表示：
  - $efficiency = \frac{1}{1+5\frac{t_{prop}}{t_{trans}}}$
  - 该公式的推导不属于本课程范围，可参见参考书“自顶向下方法”第300页给出的参考文献。
- CSMA/CD的性能比ALOHA好，而且简单、便宜、完全去中心化。
- 使用 CSMA/CD 协议的以太网不能进行全双工通信而只能进行双向交替通信（半双工通信）。
- 每个站在发送数据之后的一小段时间内，存在着遭遇碰撞的可能性。

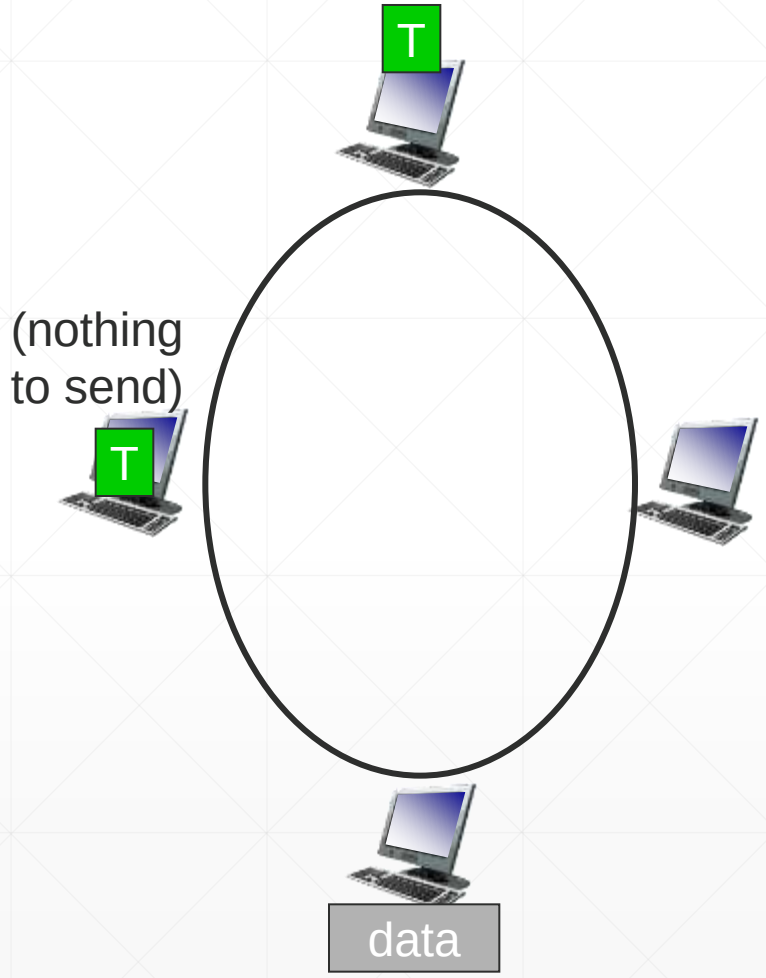
Taking turns 轮流接入

- **轮询 polling：**
  - 主节点（master）邀请次节点（slave nodes）轮流发送数据
  - 典型场景：次节点不具备智能，是dumb devices
  - 存在的问题：
    - 轮询的开销
    - 延迟
    - 单点故障（即master节点）



Taking turns 轮流接入

- **令牌传递 token passing：**
  - 受控的令牌从一个节点依次序传递到下一个节点
  - 有令牌的节点即可发送帧
  - 存在问题：
    - 传递令牌的开销
    - 延迟
    - 单点故障（令牌本身）



链路层寻址

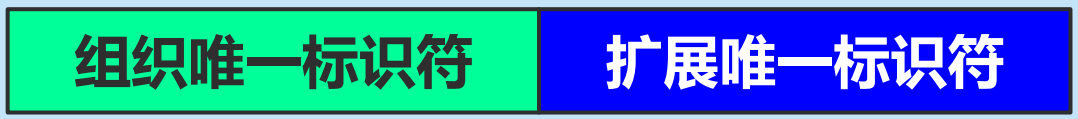
MAC addresses and ARP

MAC地址

- Media Access Control – MAC地址（也称为局域网地址、硬件地址、物理地址、以太网地址等）
- 作用：*used 'locally" to get frame from one interface to another physically-connected interface (same network, in IP-addressing sense)*
- 48位，通常都烧录进NIC的ROM中，有时候也可以通过软件设定
- 注意：并不是一台电脑或者一台路由器/交换机只有一个MAC地址，MAC地址是和端口绑定的！

48 位的 MAC 地址

- IEEE 802 标准规定 MAC 地址字段可采用 6 字节 (48位) 或 2 字节 (16 位) 这两种中的一种。
- IEEE 的注册管理机构 RA 负责向厂家分配地址字段 6 个字节中的前三个字节 (即高位 24 位)，称为组织唯一标识符。
- 地址字段 6 个字节中的后三个字节 (即低位 24 位) 由厂家自行指派，称为扩展唯一标识符，必须保证生产出的适配器没有重复地址。



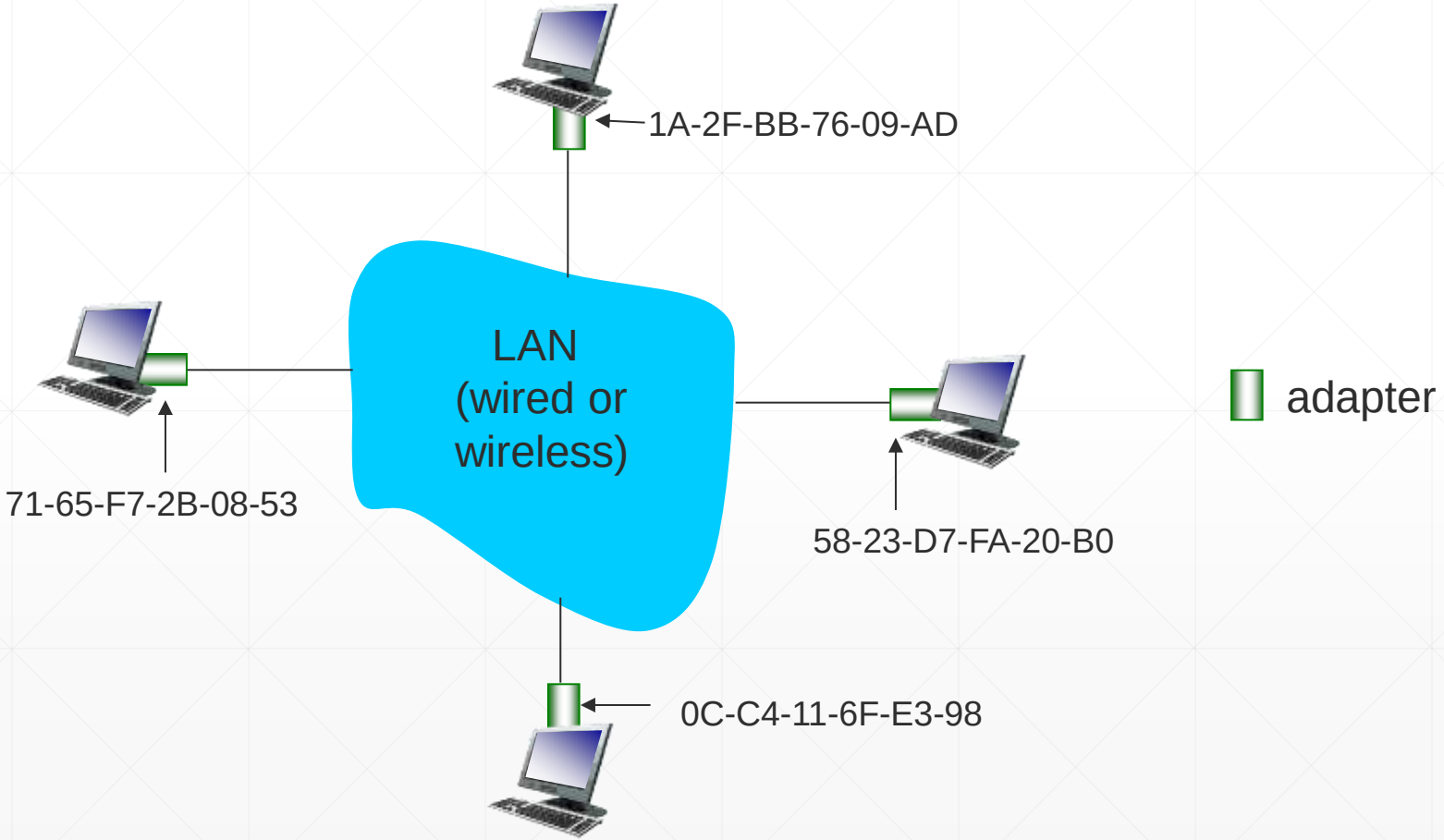
48 位的 MAC 地址

48 位的 MAC 地址

- 一个地址块可以生成 2^{24} 个不同的地址。这种 48 位地址称为 MAC-48，它的通用名称是 EUI-48。
- 生产适配器时，6 字节的 MAC 地址已被固化在适配器的 ROM，因此，MAC 地址也叫做**硬件地址** (hardware address) 或**物理地址**。
- “MAC 地址”实际上就是适配器地址或适配器标识符 EUI-48。

MAC地址

- 每个适配器、每个接口/端口，有唯一的MAC地址
- 用12个十六进制字符表示。
- MAC地址不是等级化的，保证了可移植性（可以把网卡从这个局域网移动到另一个局域网使用）



NIC检查MAC地址

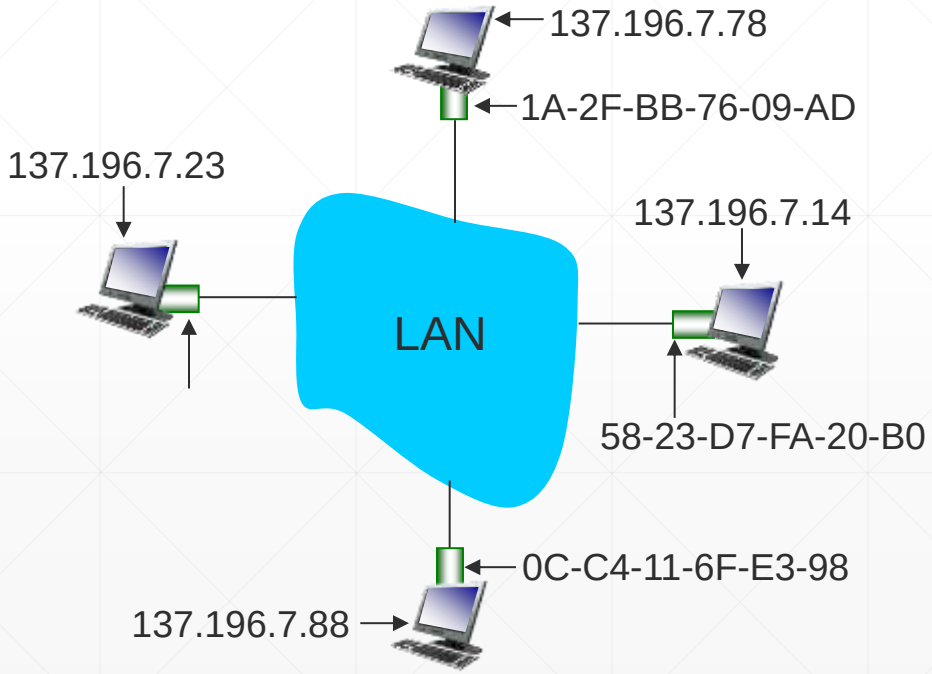
- 适配器从网络上每收到一个 MAC 帧就首先用硬件检查 MAC 帧中的 MAC 地址。
  - 如果是发往本站的帧则收下，然后再进行其他的处理。
  - 否则就将此帧丢弃，不再进行其他的处理。
- “发往本站的帧”包括以下三种帧：
  - **单播 (unicast)** 帧（一对一）
  - **广播 (broadcast)** 帧（一对全体）
  - **多播 (multicast)** 帧（一对多）

NIC检查MAC地址

- 所有的适配器都至少能够识别前两种帧，即能够识别单播地址和广播地址。
- 有的适配器可用编程方法识别多播地址。
- 只有目的地址才能使用广播地址和多播地址。
- 以混杂方式 (promiscuous mode) 工作的以太网适配器只要“听到”有帧在以太网上传输就都接收下来。

ARP协议 – Address Resolution Protocol

问题：已知一个端口的IP地址，如何获知它的MAC地址?



- **ARP表**：局域网中的每个节点（主机、路由器等）都有一张表，称之为ARP表。
  - 存储了局域网中某些节点的IP地址/MAC地址的映射关系。
  - <IP address; MAC address; TTL>
  - TTL (time to live): 多久之后忘记该映射（典型值：20分钟）。
  - 如何学习ARP表?

ARP协议 – 同一局域网内

- A需要发送数据报给B
  - 假设A的ARP表中并没有B的MAC地址
  - 虽然A知道B的IP地址，在网络层A知道B的位置，但是链路层A并不知道
- A发送**广播**ARP查询packet，包含了B的IP地址
  - 该packet中的目的MAC地址是FF-FF-FF-FF-FF-FF（广播）
  - 局域网中所有节点都收到了该查询
- B收到该packet，将自身的MAC地址回复给A
  - 这个回复帧，发送给A的MAC地址（**单播**）

- A将学习到的IP-MAC地址映射存放在其ARP表中，直到该信息过时：
  - 除非更新信息，否则在TTL后该信息即为过时
- ARP协议是“即插即用”的：
  - 节点创建自己的ARP表格，并不需要网络管理员的人工干涉。

ARP协议 – 不在同一局域网内

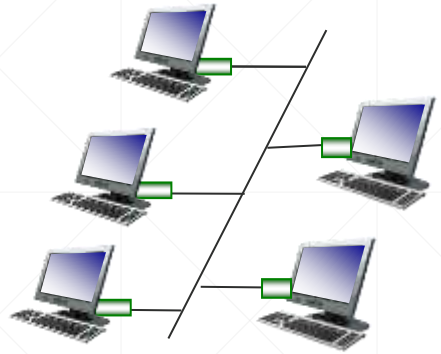
- 过程基本相同，但需要分开先从边缘路由器跳转，获得各自的MAC地址
- 具体过程，需要先学习IP协议，此处不展开
- 待学习IP协议后再讲述

以太网

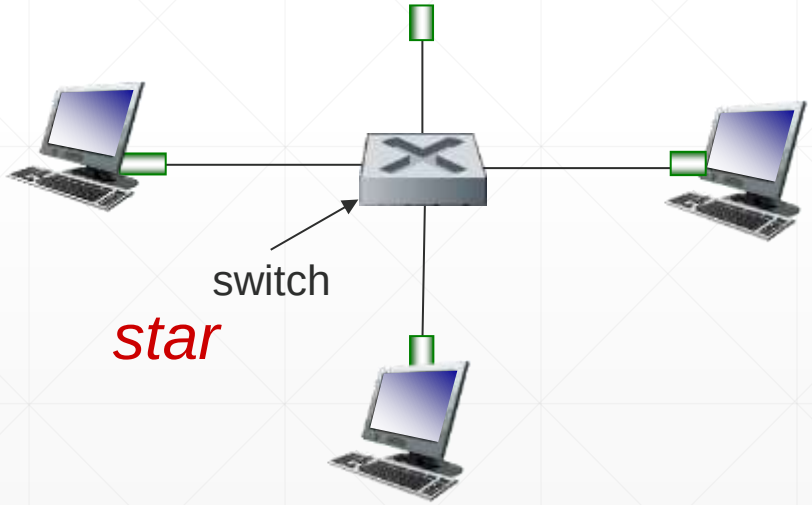
Ethernet

Ethernet

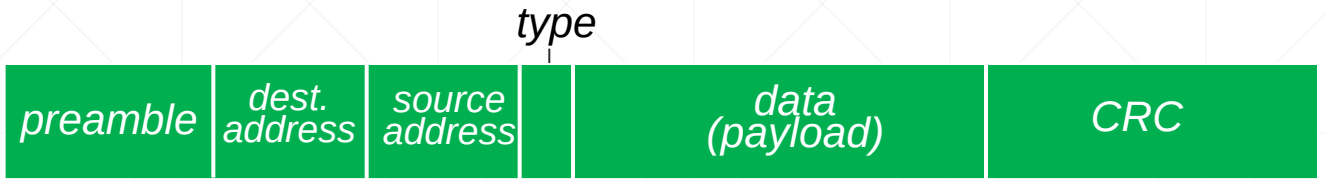
- 以太网，是现在的事实上的主流有线局域网技术
  - 单一芯片，可以实现多种带宽
  - 第一个广泛采用的局域网技术
  - 简单、便宜、速度快
- 以太网的物理拓扑（physical topology）
  - **总线拓扑**（直至90年代中期）
    - 所有节点处于同一碰撞域（可能互相发生碰撞）
  - **星形拓扑**（现今）
    - 采用交换机（switch）
    - 每个“辐条”运行各自的以太网协议，各自组成碰撞域



bus: coaxial cable

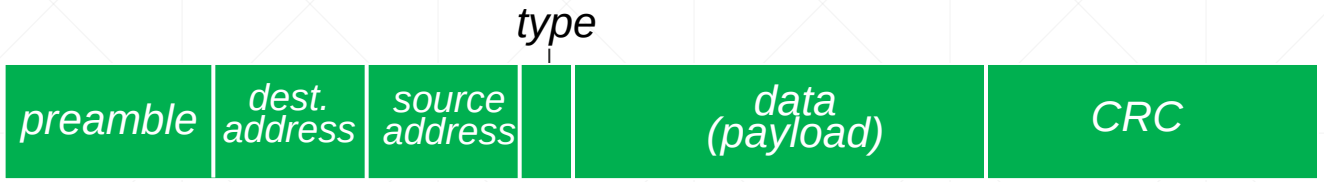


以太网的帧结构

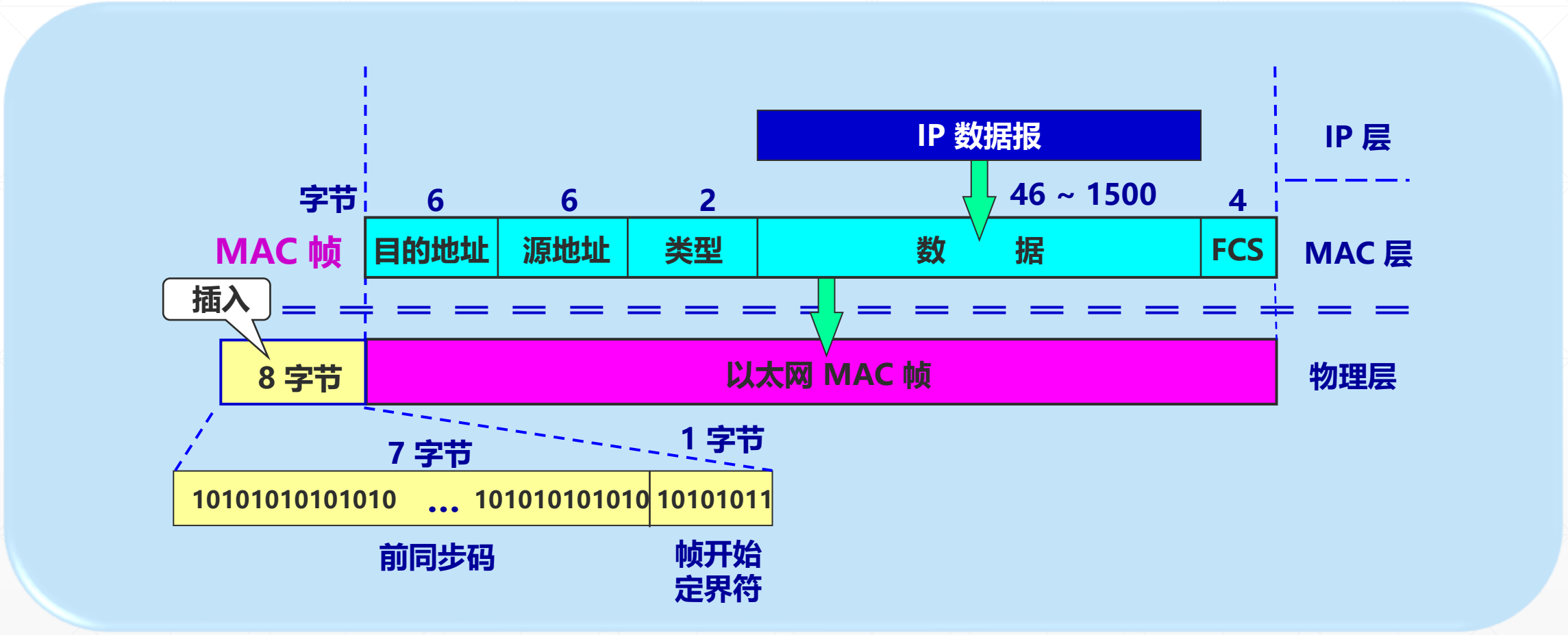


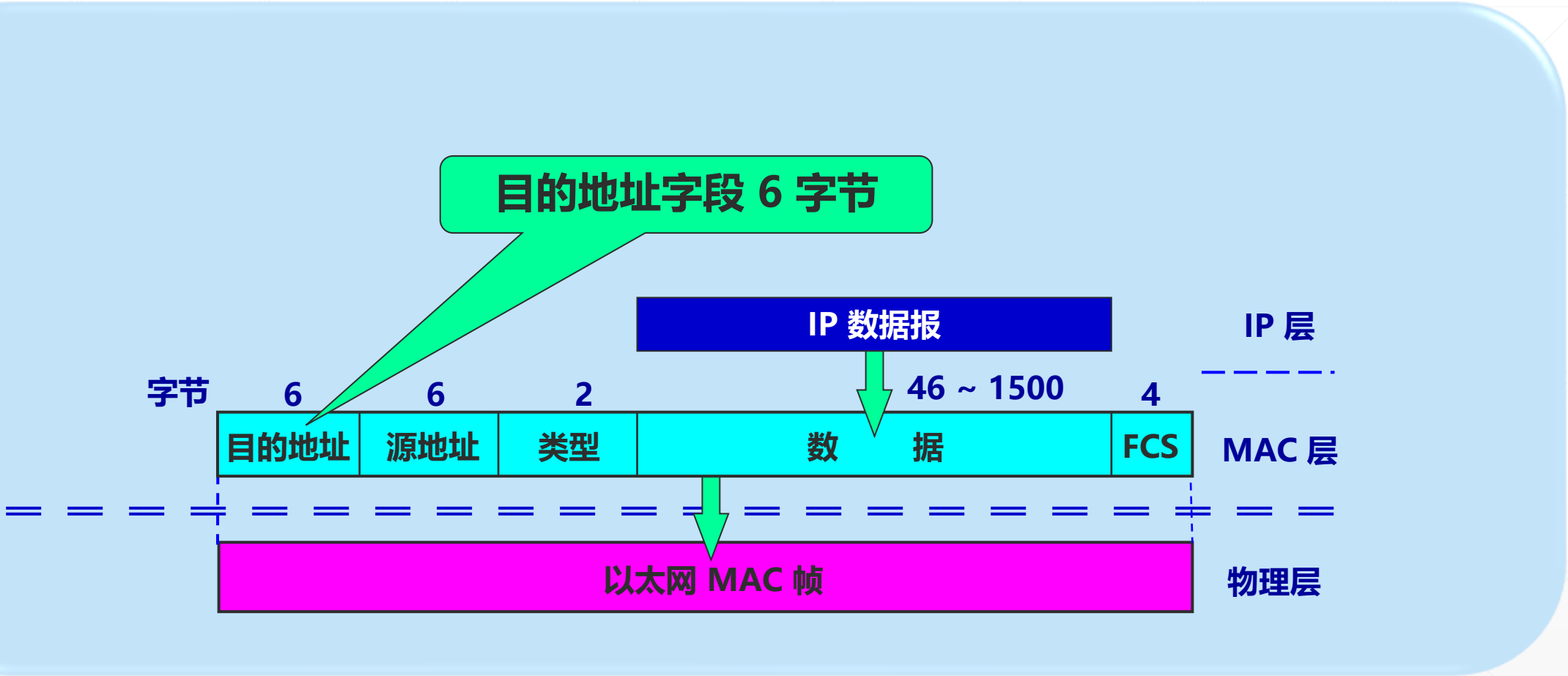
- 发送帧的NIC，负责将IP数据报（从网络层来的，也可能是其他网络层协议的报文）包装成以太网的帧。
- **preamble：**
  - 一共8个字节，前面7个字节是重复的10101010；最后一个字节是10101011.
  - 用于实现MAC帧的比特同步，及界定MAC帧的开始。
- **dest address/source address：**
  - 各自6字节的MAC地址。
  - 如果一个NIC收到的帧，其目的地址匹配自己的MAC地址，或是收到广播地址，则传输该帧至上一层网络层协议。否则丢弃该帧

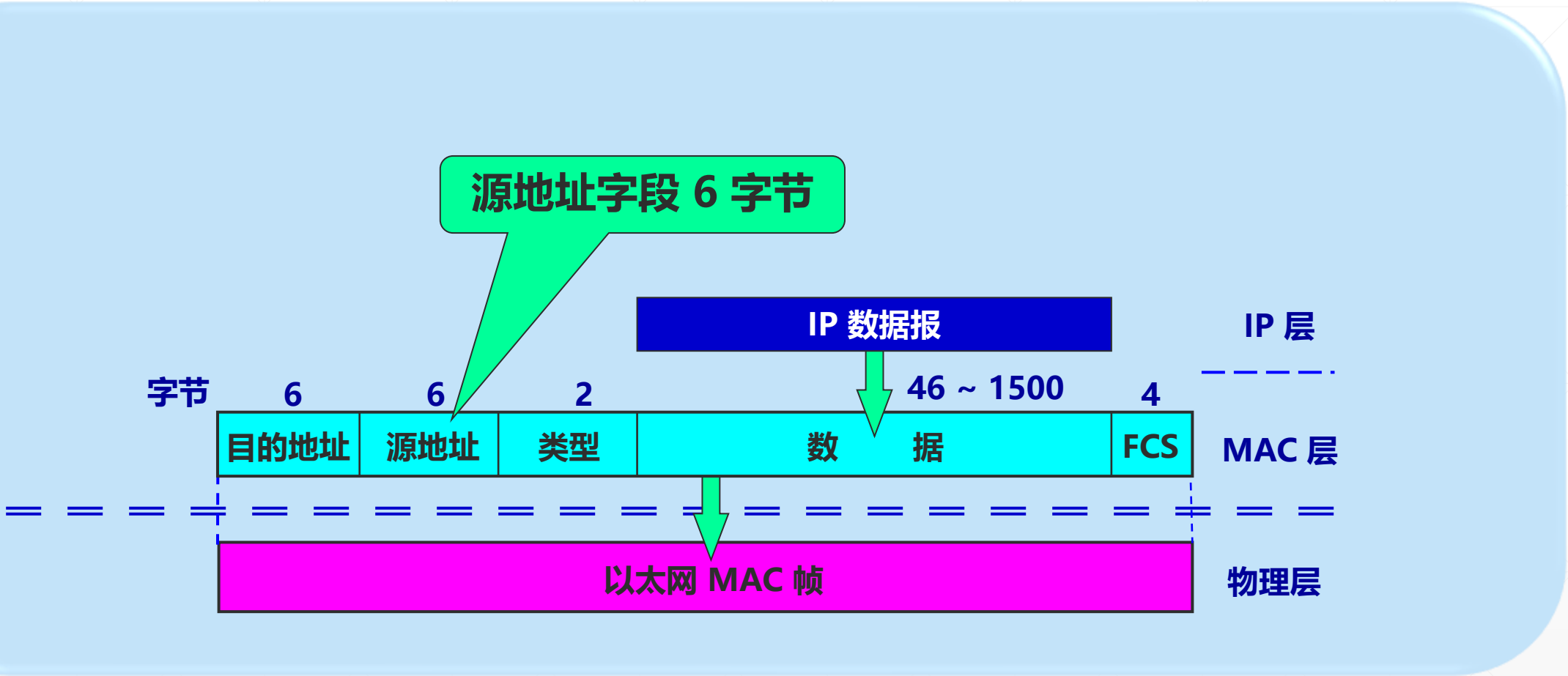
以太网的帧结构



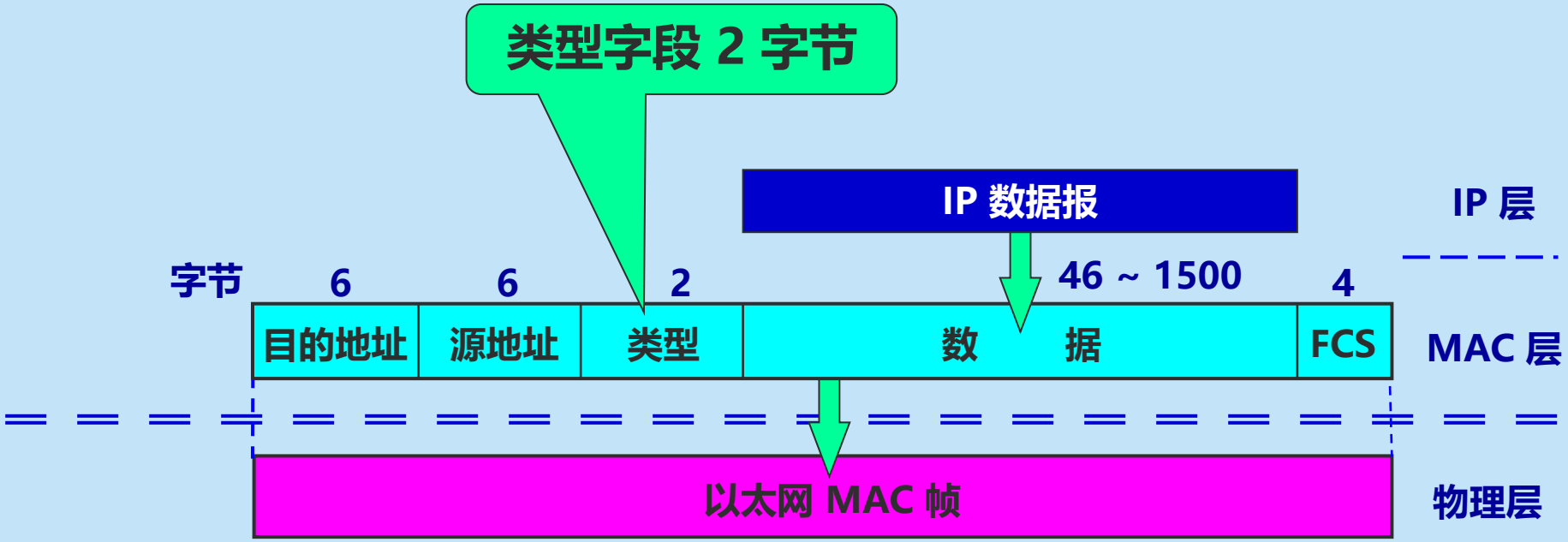
- **type：**
  - 表示“上一层协议”，即收到该帧之后，应传送给网络层的何种协议。
  - 多数情况下是IP协议，也有可能有其他的IP层协议
- **CRC：**
  - 4字节的CRC校验码。
  - 若检查出错误，则丢弃该帧；没有重传！


IP 数据报
IP 层
字节
6
6
2
46 ~ 1500
4
MAC 帧
目的地址
源地址
类型
数
据
FCS
MAC 层
插入
8 字节
以太网 MAC 帧
物理层
1 字节
7 字节
10101010101010 ... 10101010101010
10101011
前同步码
帧开始
定界符

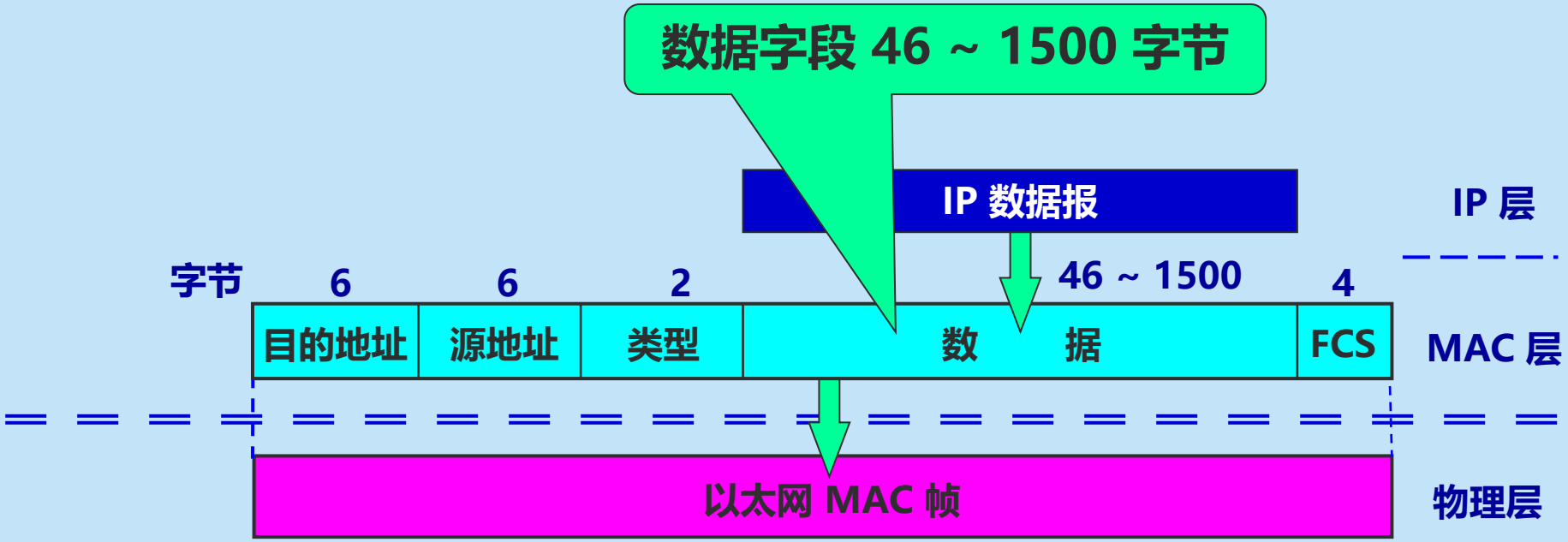

目的地址字段 6 字节
IP 数据报
IP 层
字节
6
6
2
46 ~ 1500
4
目的地址
源地址
类型
数
据
FCS
MAC 层
以太网 MAC 帧
物理层


源地址字段 6 字节
IP 数据报
IP 层
字节
6
6
2
46 ~ 1500
4
目的地址
源地址
类型
数
据
FCS
MAC 层
以太网 MAC 帧
物理层

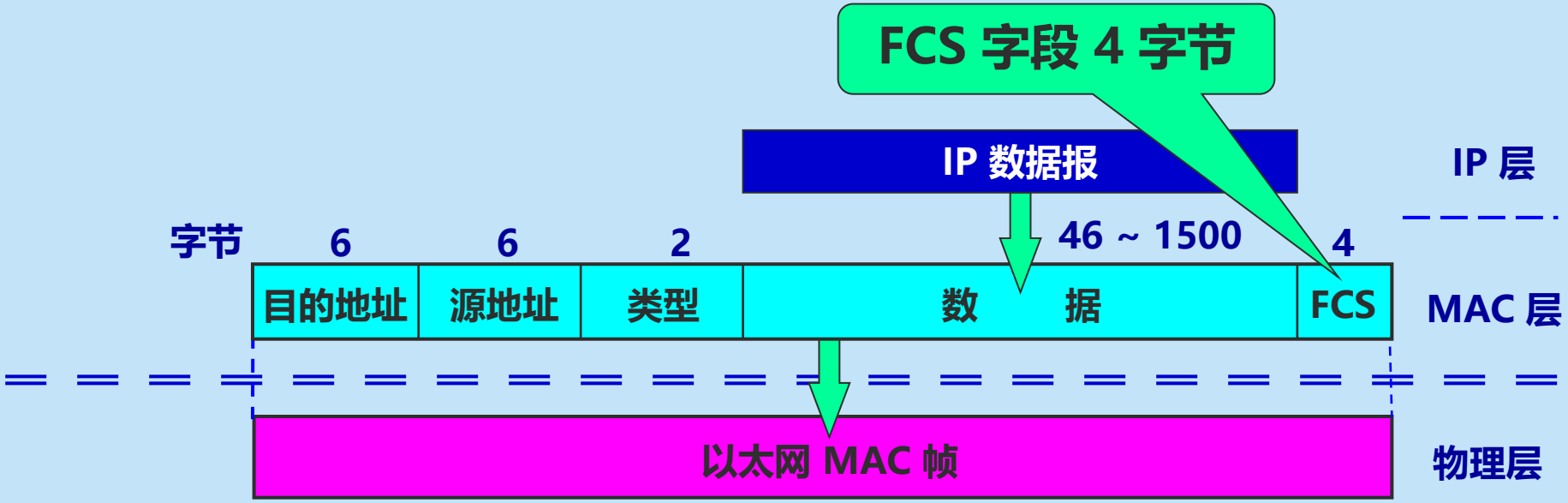
类型字段用来标志上一层使用的是什么协议，
以便把收到的 MAC 帧的数据上交给上一层的这个协议。



数据字段的正式名称是 MAC 客户数据字段。
最小长度 64 字节 - 18 字节的首部和尾部 = 数据字段的最小长度（46字节）

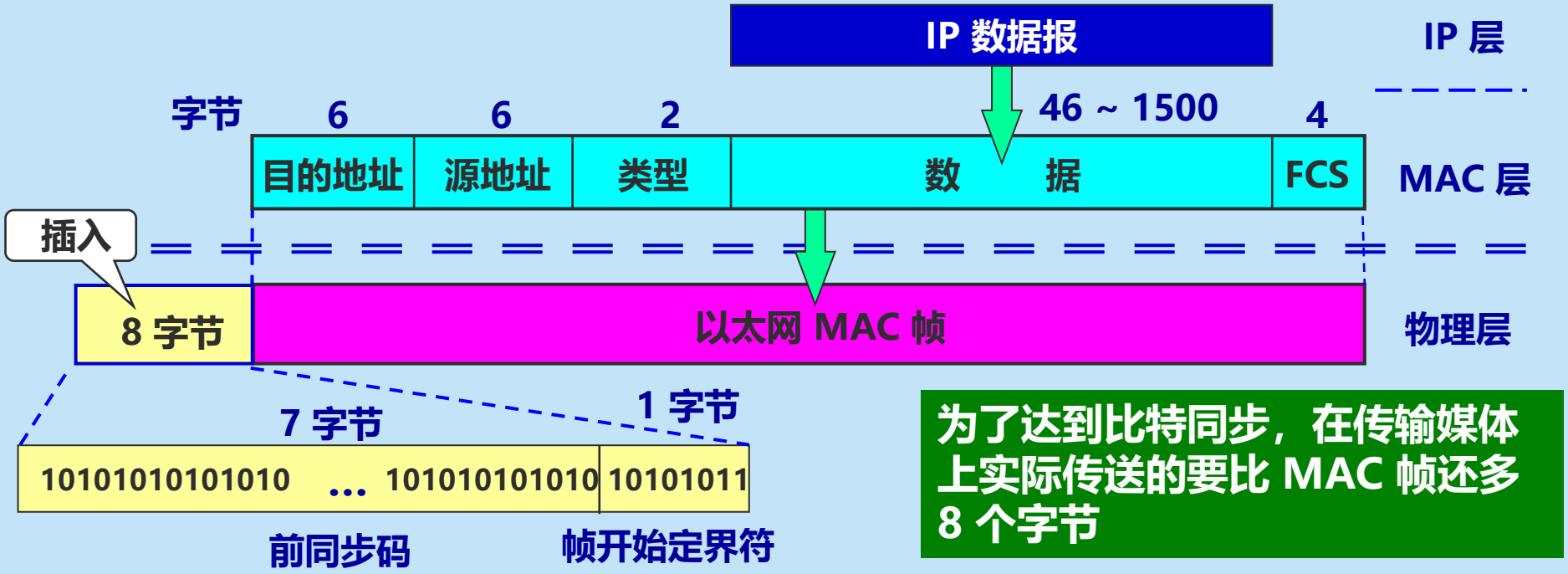


当传输媒体的误码率为 $1x10^{-8}$ 时，MAC 子层可使未检测到的差错小于 $1x10^{-14}$ 。



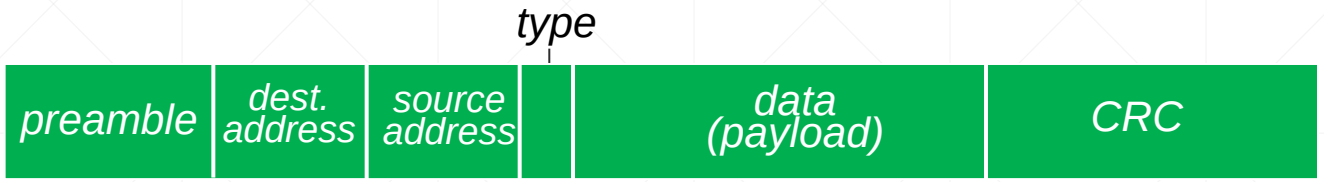
当数据字段的长度小于 46 字节时，应在数据字段的后面加入整数字节的**填充字段**，以保证以太网的 MAC 帧长不小于 64 字节。

在帧的前面插入（硬件生成）的 8 字节中，第一个字段共 7 个字节，是前同步码，用来迅速实现 MAC 帧的比特同步。第二个字段 1 个字节是帧开始定界符，表示后面的信息就是 MAC 帧。



为了达到比特同步，在传输媒体上实际传送的要比 MAC 帧还多 8 个字节

以太网的帧结构



- 什么是无效的帧?
  - 帧长度不是整数字节
  - CRC校验失败
  - 数据字段的长度不在46~1500字节之间
  - MAC帧长度不在64~1518字节之间

对于检查出的无效 MAC 帧就简单地丢弃。以太网不负责重传丢弃的帧。

不可靠、无连接的以太网

- **无连接 connectionless**：发送的NIC和接收的NIC并不需要用握手来建立连接
- **不可靠**：接收的NIC不需要向发送的NIC发送ACK或者NACK等packet来表达“是否收到帧”
  - 若帧被丢弃，则只能通过上层协议来恢复（例如，提供了可靠数据传输的协议，如TCP，之后会学到）；如没有通过上层协议恢复，则数据就丢失了
- Ethernet中用到的信道复用协议是unslotted **CSMA/CD with binary backoff**。

802.3 Ethernet标准家族

- 有很多种不同的Ethernet标准
  - 主要规定物理层、链路层行为
  - 通常都共用了信道复用协议和帧格式
  - 但是提供不同的带宽/速度：2Mbps, 10Mbps, 100Mbps, 1Gbps, 10Gbps, 40Gbps…
  - 采用不同的物理层媒介：光纤、光缆等
  - http://www.ieee802.org/3/

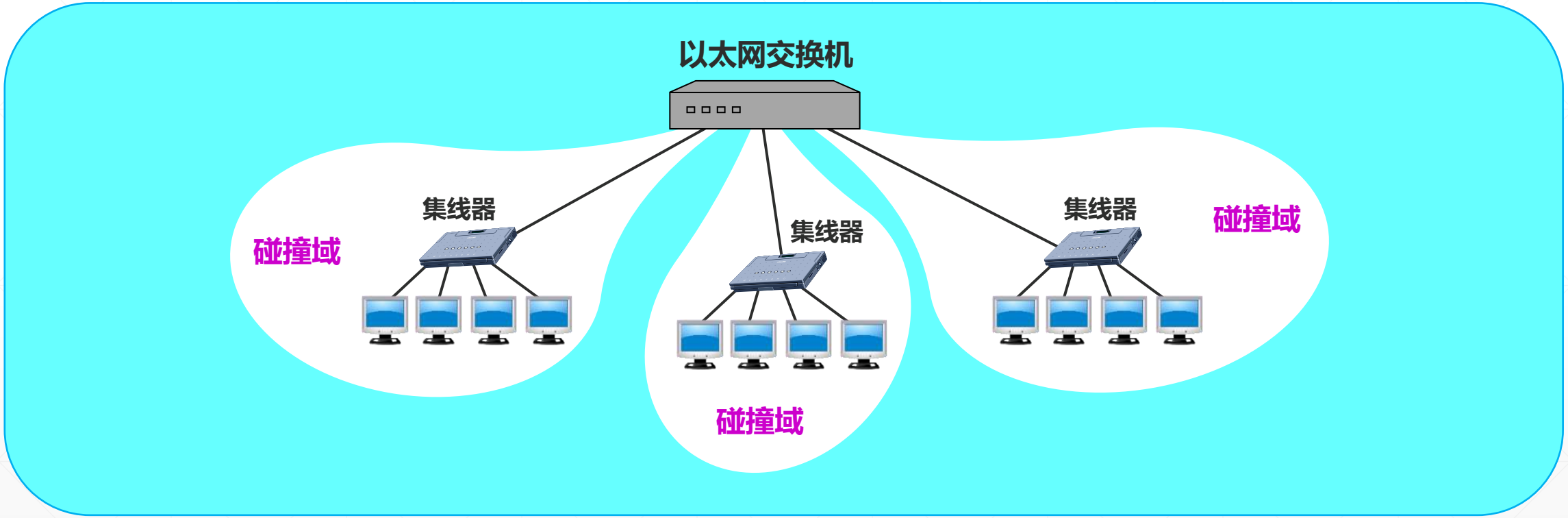
交换机

Switches

为什么要使用交换机?

- 在链路层扩展以太网，常用的方法是使用以太网交换机
  - 扩大以太网规模，将不同的网络连接在一起
- 交换机是链路层设备，通常也称为“第二层交换机”（L2 switch），强调这种交换机工作在链路层
  - 存储并转发以太网帧
  - 检查收到的帧的MAC地址，选择性地转发该帧到一个或多个链路
- 交换机是透明的：主机并不知道交换机的存在
- 即插即用、自学习：交换机不需要手动配置，它自动学习网络拓扑

- 相互通信的主机都是独占传输媒体，**无碰撞**地传输数据。

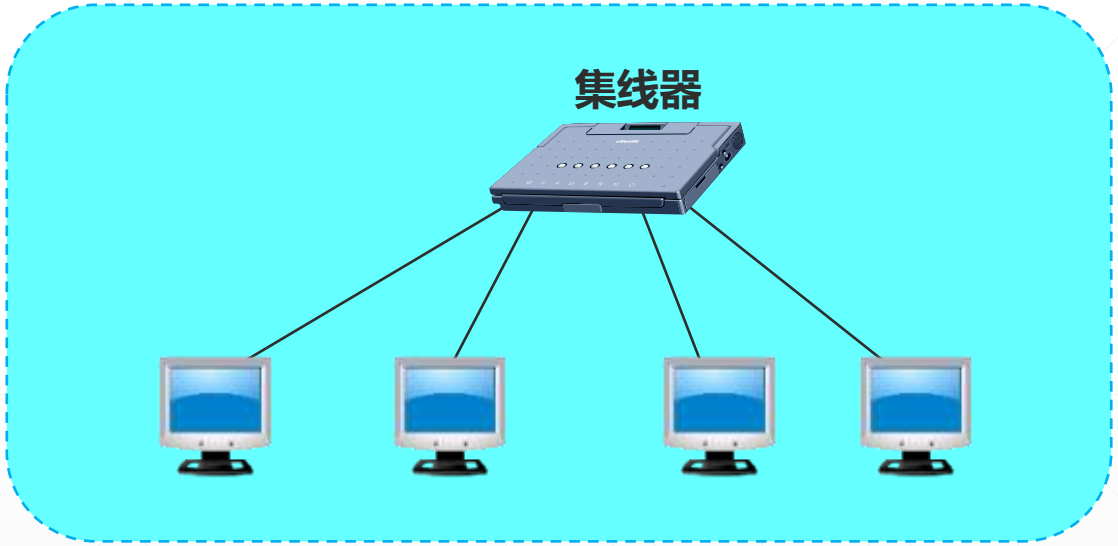


以太网交换机的每个接口是一个碰撞域

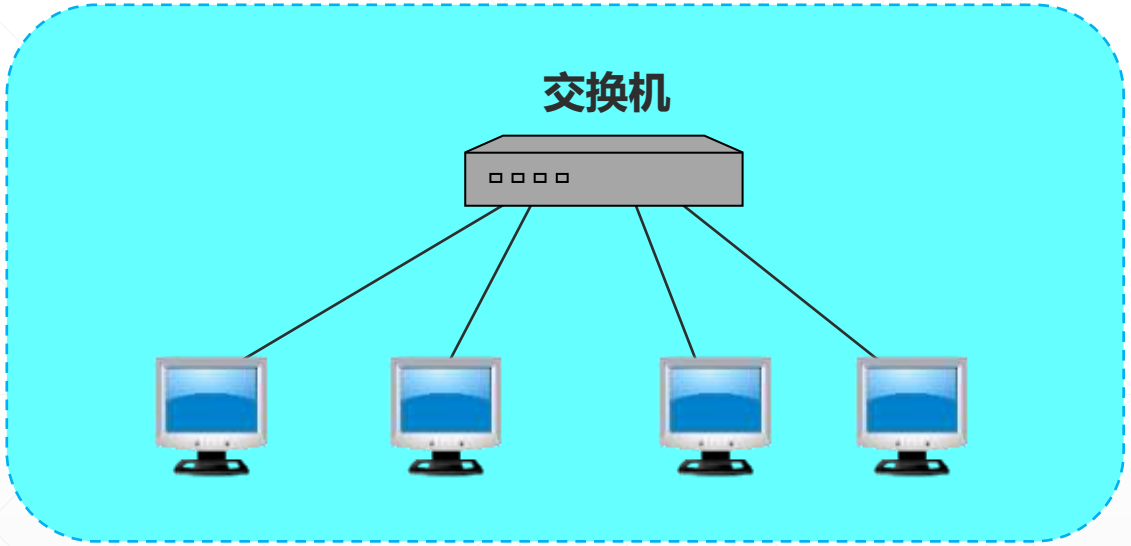
交换机的特点

- 交换机通常都有十几个或更多的接口。
  - 每个接口都直接与一个单台主机或另一个交换机相连，并且一般都工作在全双工方式。
- 交换机具有并行性。
  - 能同时连通多对接口，使多对主机能同时通信。
- 交换机的接口有存储器，能在输出端口繁忙时把到来的帧进行缓存。
- 交换机使用了专用的交换结构芯片，用硬件转发，其转发速率非常高。
- 交换机的性能远远超过普通的集线器，而且价格并不贵。

- 用户独享带宽，增加了总容量。



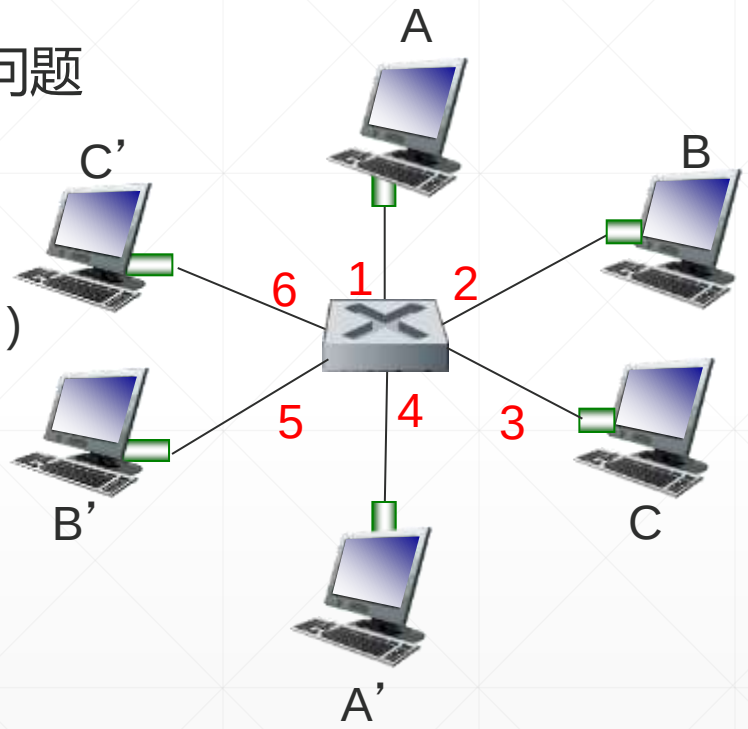
- N 个用户共享集线器提供的带宽 B。
- 平均每个用户仅占有 B/N 的带宽。



- 交换机为每个端口提供带宽 B。
- N个用户，每个用户独占带宽 B。
- 交换机总带宽达 B × N 。

交换表/转发表 forwarding table

- 交换机如何知道“主机通过哪个接口可以访问？”这样的问题
  - A’ 可以通过接口4访问；B’ 可以通过接口5访问。。。
  - 每个交换机有一个交换表
  - 表的内容是（主机的MAC地址，通过哪个接口访问，有效时间）
  - 就像路由表一样!
- 上面的交换表中的条目如何创建和维护?
  - 交换机的自学习功能（self-learning）



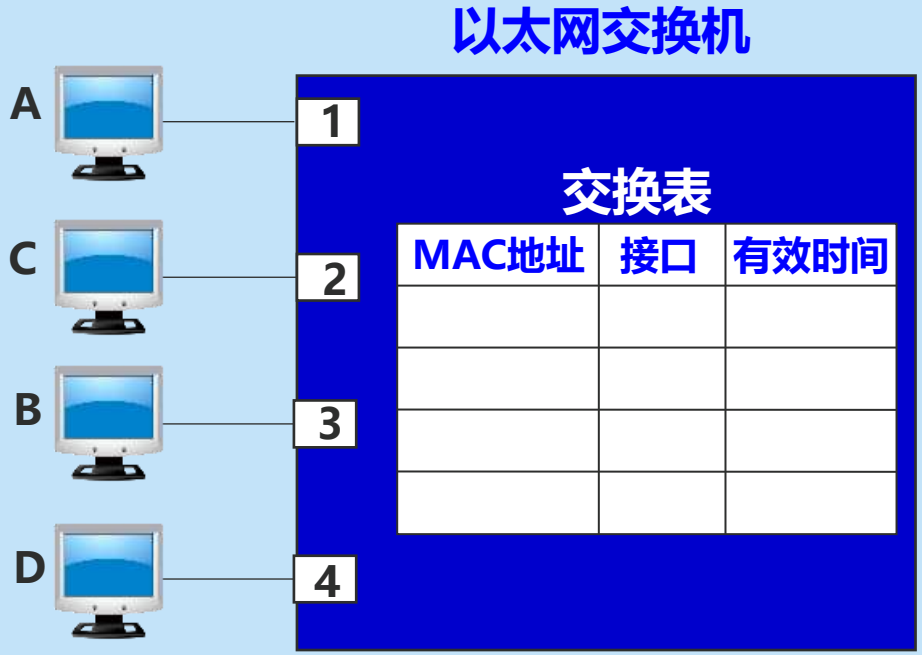
六个接口的交换机
(1,2,3,4,5,6)

交换机功能：过滤帧、转发帧

- 当交换机收到一个帧
- 1. 记录进入交换机的链路，以及发送主机的MAC地址
- 2. 根据目的地MAC地址，从交换表里面寻找对应条目
  - 2.1 如果找到了对应条目
    - 2.1.1 如果目的地和帧的来源一样，则丢弃该帧
    - 2.1.2 否则，根据找到的对应条目，从正确的接口将帧转发出去
  - 2.2 如果没有找到对应条目
    - 2.2.1 则泛洪该帧（通过除了进来的接口以外的其他所有接口转发该帧）

以太网交换机的自学习功能

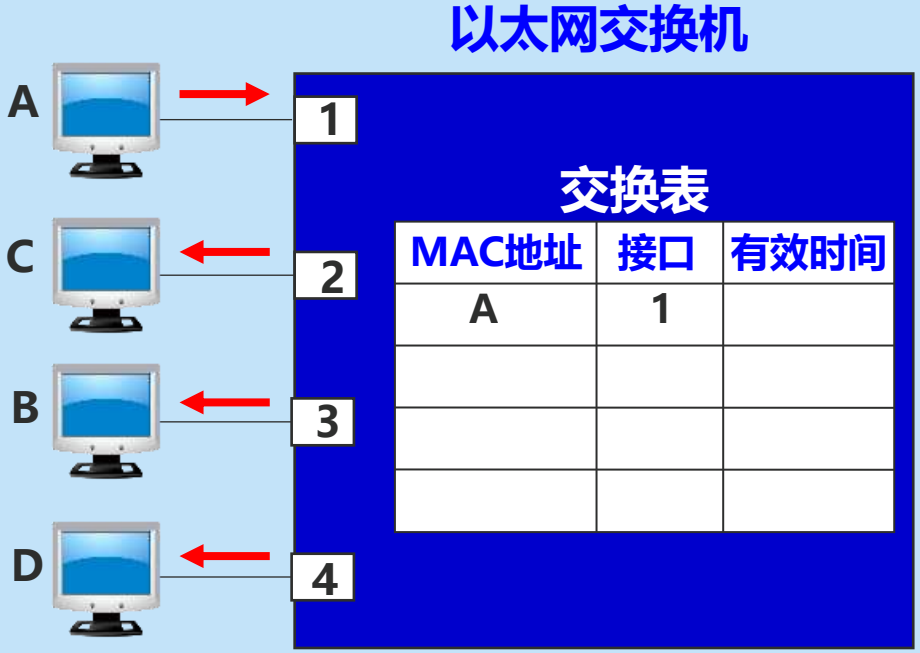
- 以太网交换机运行自学习算法自动维护交换表。



开始时，交换表是空的

以太网交换机的自学习功能

- 以太网交换机运行自学习算法自动维护交换表。



以太网帧

目的地址	源地址	类型	数据	FCS
B	A			

A 先向 B 发送一帧。该帧从接口 1 进入到交换机。

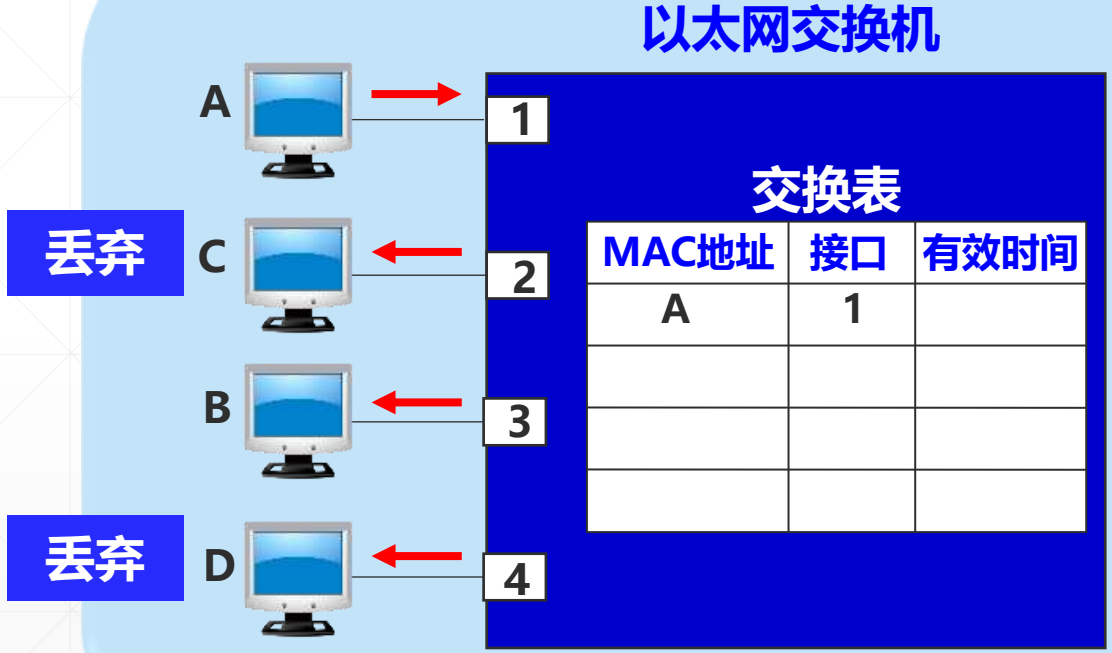
交换机收到帧后，先查找交换表。没有查到应从哪个接口转发这个帧给 B。

交换机把这个帧的源地址 A 和接口 1 写入交换表中。

交换机向除接口 1 以外的所有的接口广播这个帧。

以太网交换机的自学习功能

- 以太网交换机运行自学习算法自动维护交换表。



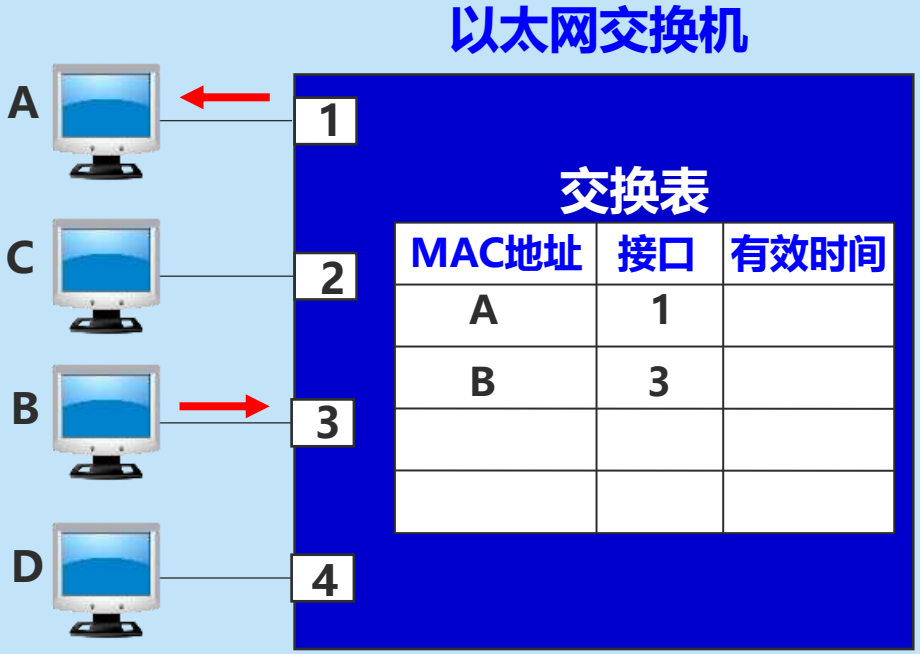
以太网帧

目的地址	源地址	类型	数据	FCS
B	A			

由于与该帧的目的地址不相符，C 和 D 将丢弃该帧。

以太网交换机的自学习功能

- 以太网交换机运行自学习算法自动维护交换表。



以太网帧

目的地址	源地址	类型	数据	FCS
A	B			

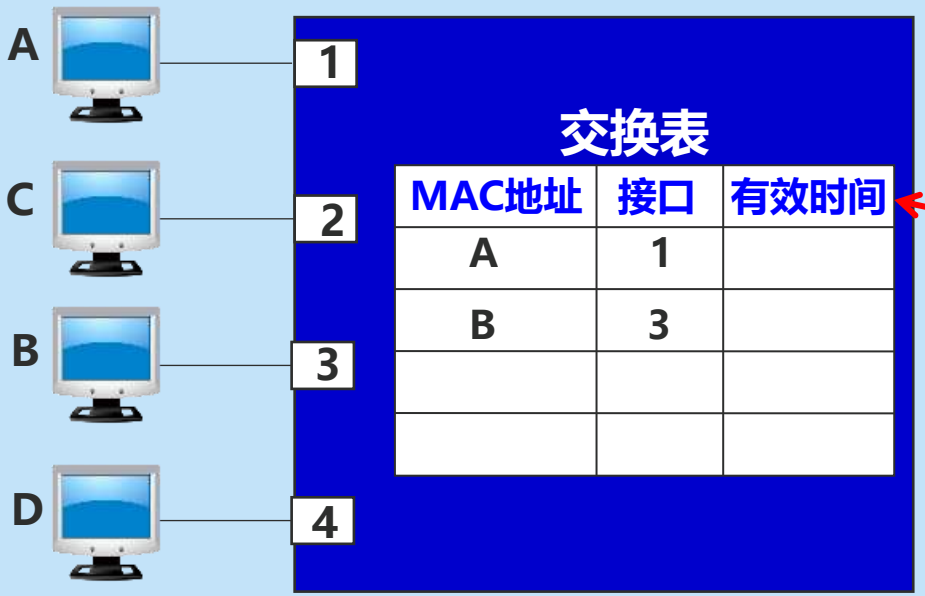
B 向 A 发送一帧。该帧从接口 3 进入到交换机。

交换机收到帧后，先查找交换表。发现交换表中的 MAC 地址有 A，表明要发送给A的帧应从接口1转发出去。于是就把这个帧传送到接口 1 转发给 A。

交换机把这个帧的源地址 B 和接口 3 写入交换表中。

以太网交换机的自学习功能

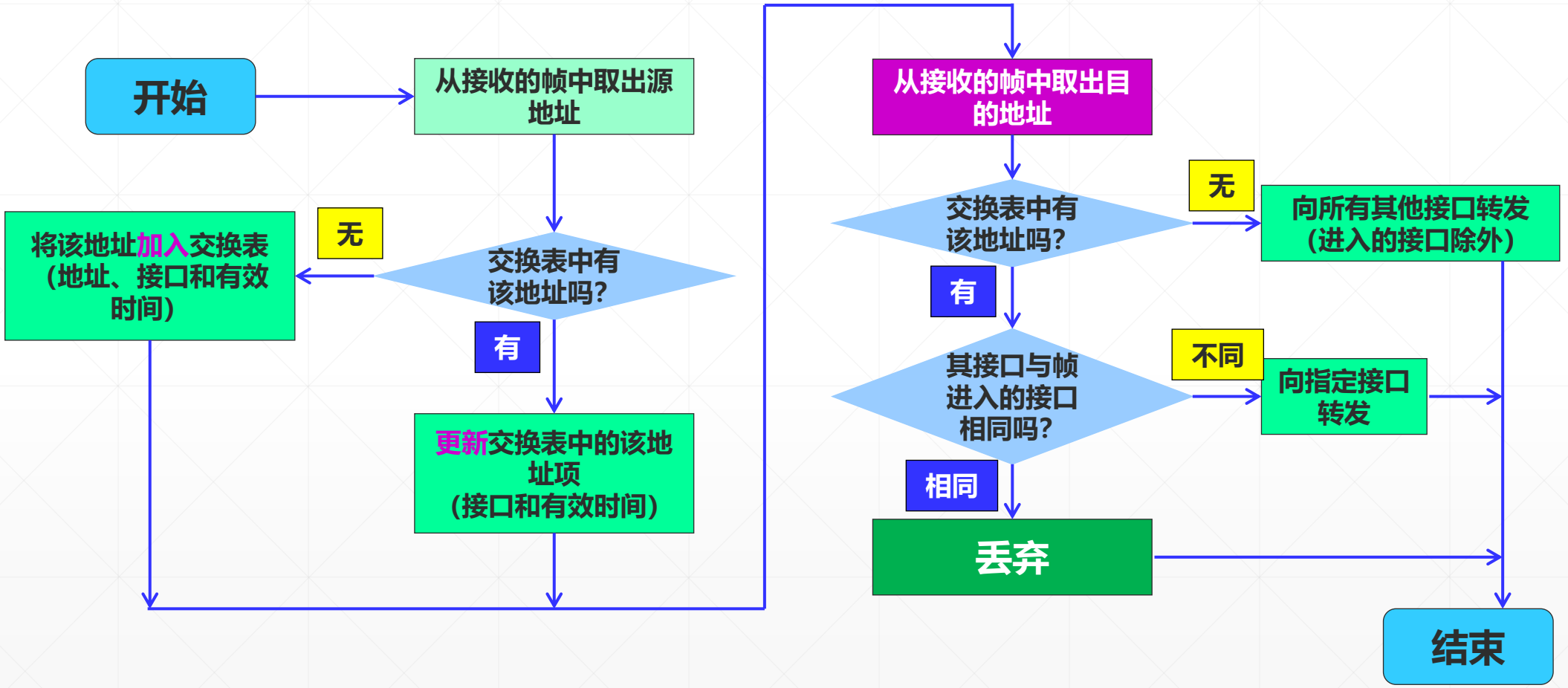
- 以太网交换机运行自学习算法自动维护交换表。



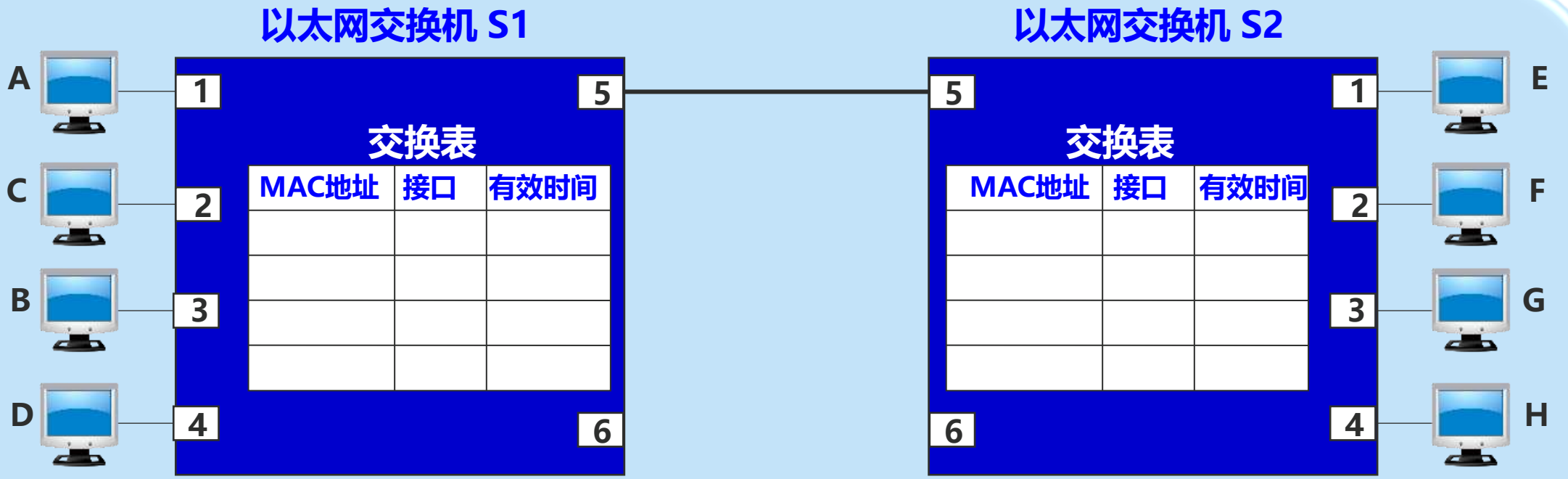
考虑到可能有时要在交换机的接口更换主机，或者主机要更换其网络适配器，这就需要更改交换表中的项目。为此，在交换表中每个项目都设有一定的**有效时间**。**过期的项目就自动被删除。**

以太网交换机的这种自学习方法使得以太网交换机能够即插即用，不必人工进行配置，因此非常方便。

交换机自学习和转发帧的步骤归纳



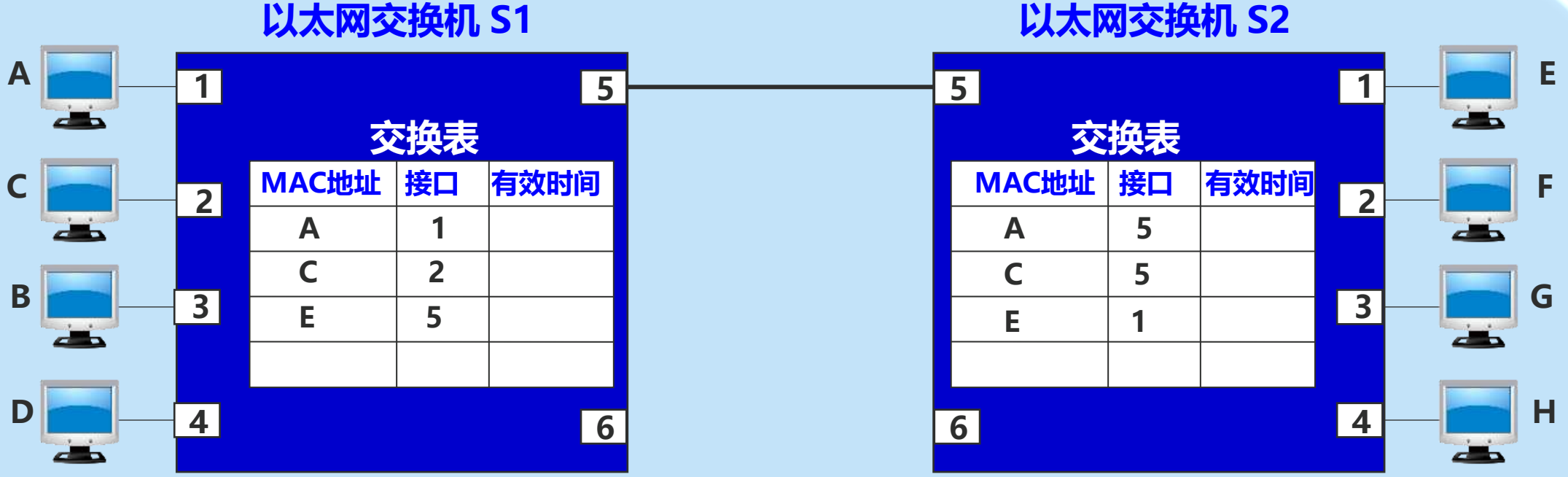
理解以太网交换机的自学习功能



假设：A 向 B 发送了一帧，C 向 E 发送了一帧，E 向 A 发送了一帧。

请分析：此时，S1 和 S2 的交换表内容分别是什么?

理解以太网交换机的自学习功能



假设：A 向 B 发送了一帧，C 向 E 发送了一帧，E 向 A 发送了一帧。

请分析：此时，S1 和 S2 的交换表内容分别是什么?

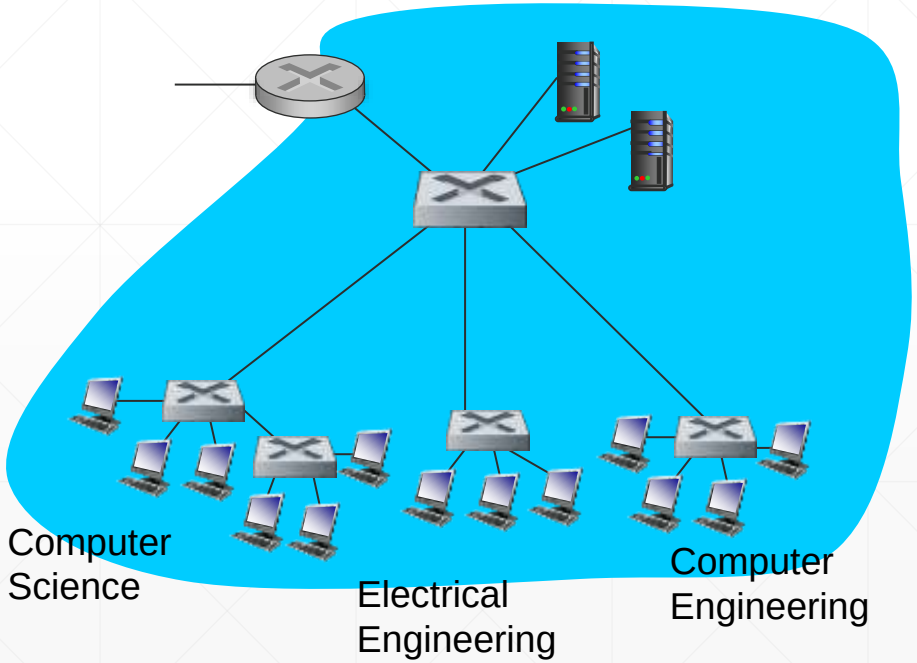
虚拟局域网

VLAN – Virtual LAN

VLAN – 为什么要有VLAN?

- 考虑一个场景：
  - CS用户把办公室移动到EE系的楼，但是仍然希望连接CS系的局域网
  - 如果只采用一个广播域
    - 则所有的链路层流量（例如ARP、DHCP等）都需要穿过整个局域网
    - 带来安全、隐私、效率问题

VLAN：switches supporting VLAN capabilities can be configured to define multiple *virtual* LANs over single physical LAN infrastructure.



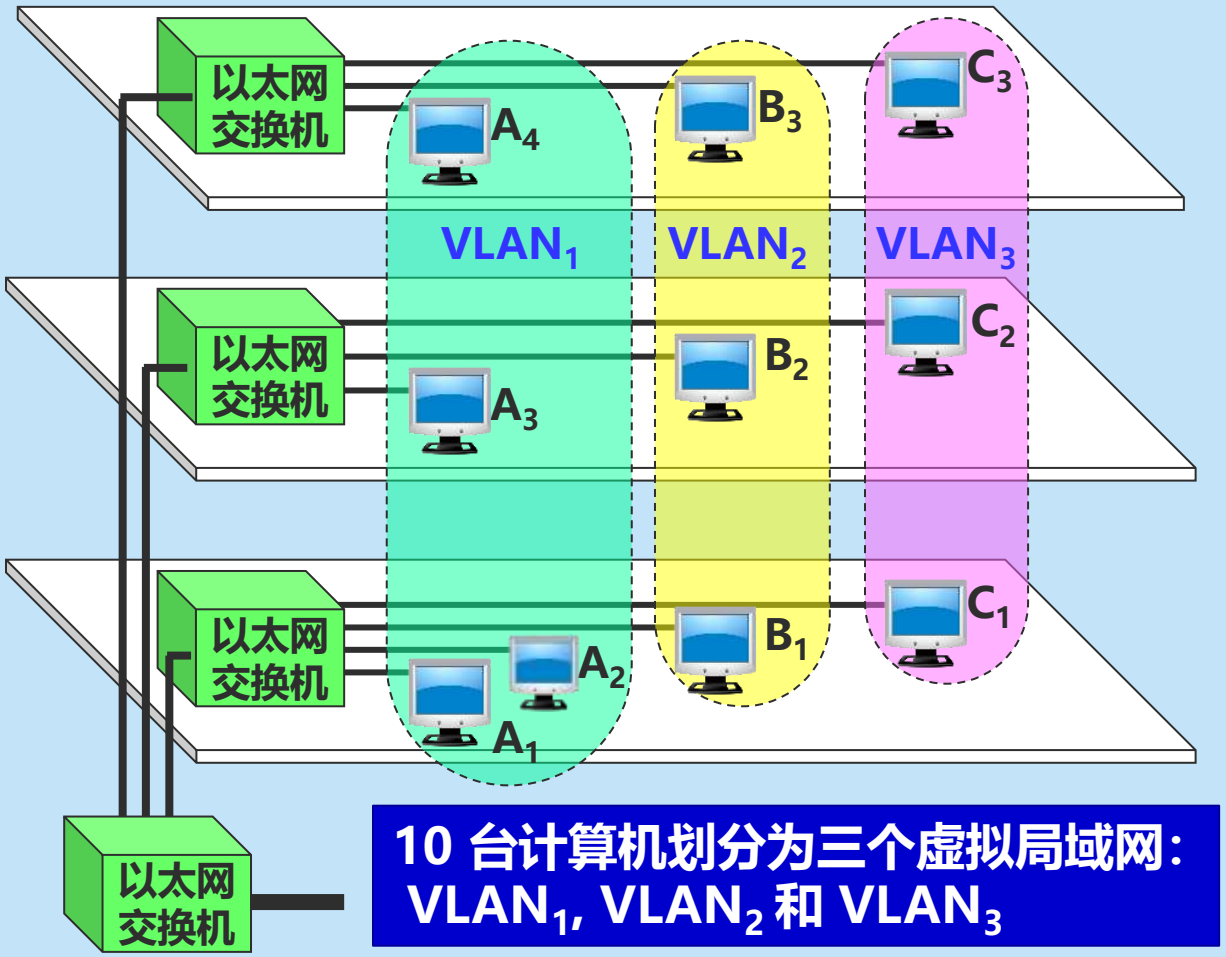
虚拟局域网

- 利用以太网交换机可以很方便地实现虚拟局域网 VLAN (Virtual LAN)。
- IEEE 802.1Q 对虚拟局域网 VLAN 的定义：

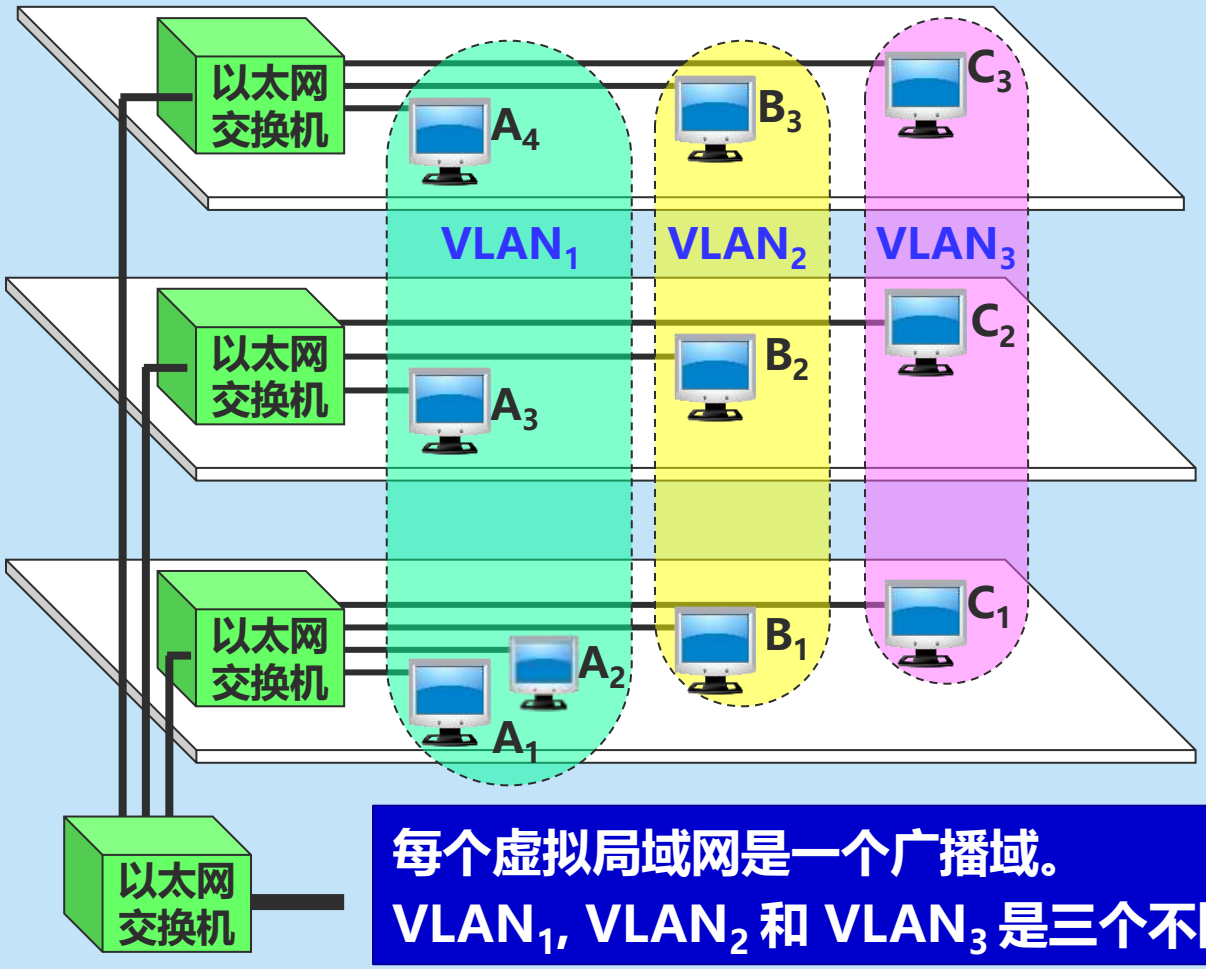
  虚拟局域网 VLAN 是由一些局域网网段构成的与物理位置无关的逻辑组，而这些网段具有某些共同的需求。每一个 VLAN 的帧都有一个明确的标识符，指明发送这个帧的计算机是属于哪一个 VLAN。

虚拟局域网

- **虚拟局域网其实只是局域网给用户提供的一种服务，而并不是一种新型局域网。**
- **由于虚拟局域网是用户和网络资源的逻辑组合，因此可按照需要将有关设备和资源非常方便地重新组合，使用户从不同的服务器或数据库中存取所需的资源。**

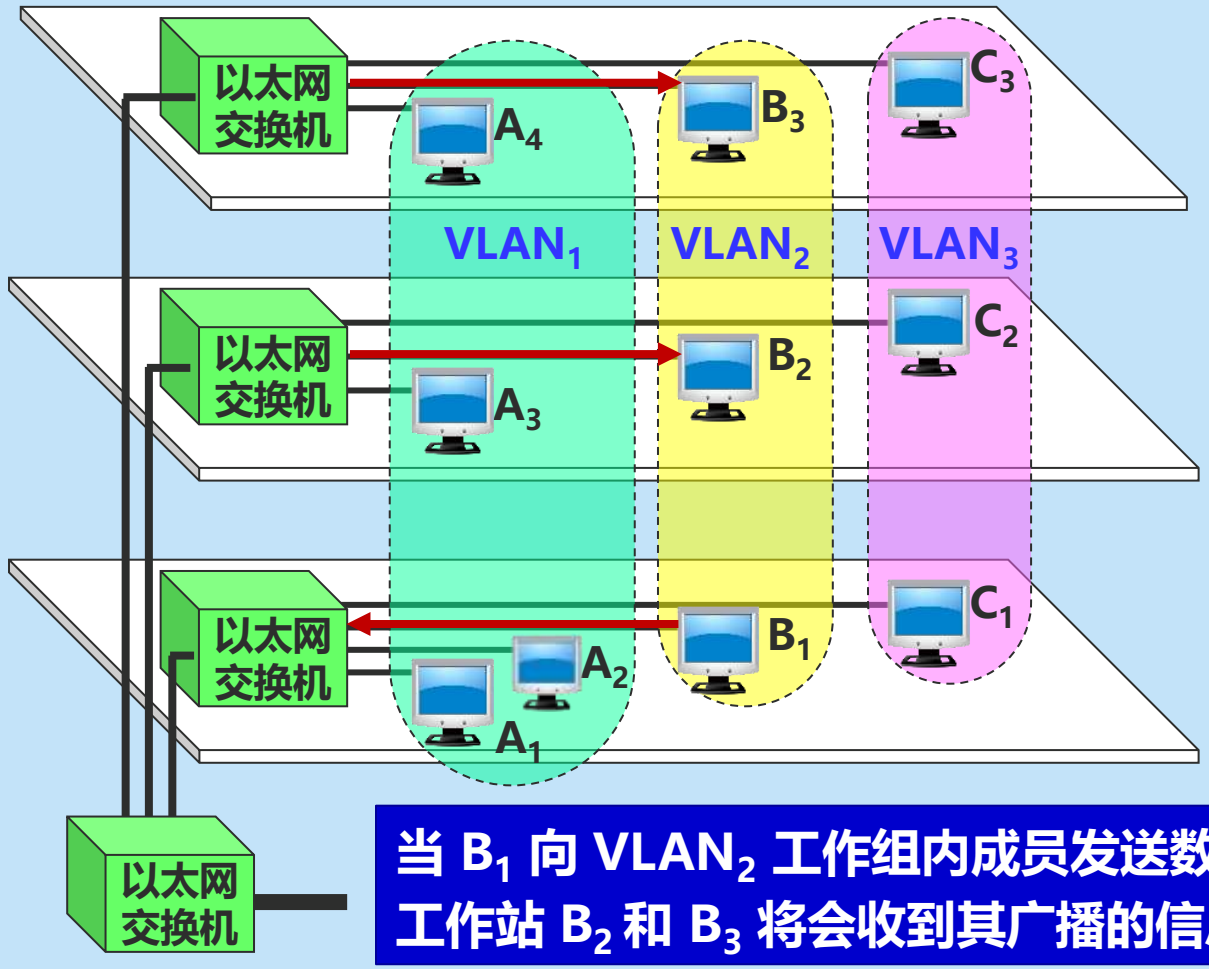


10 台计算机划分为三个虚拟局域网：
$VLAN_1$, $VLAN_2$ 和 $VLAN_3$

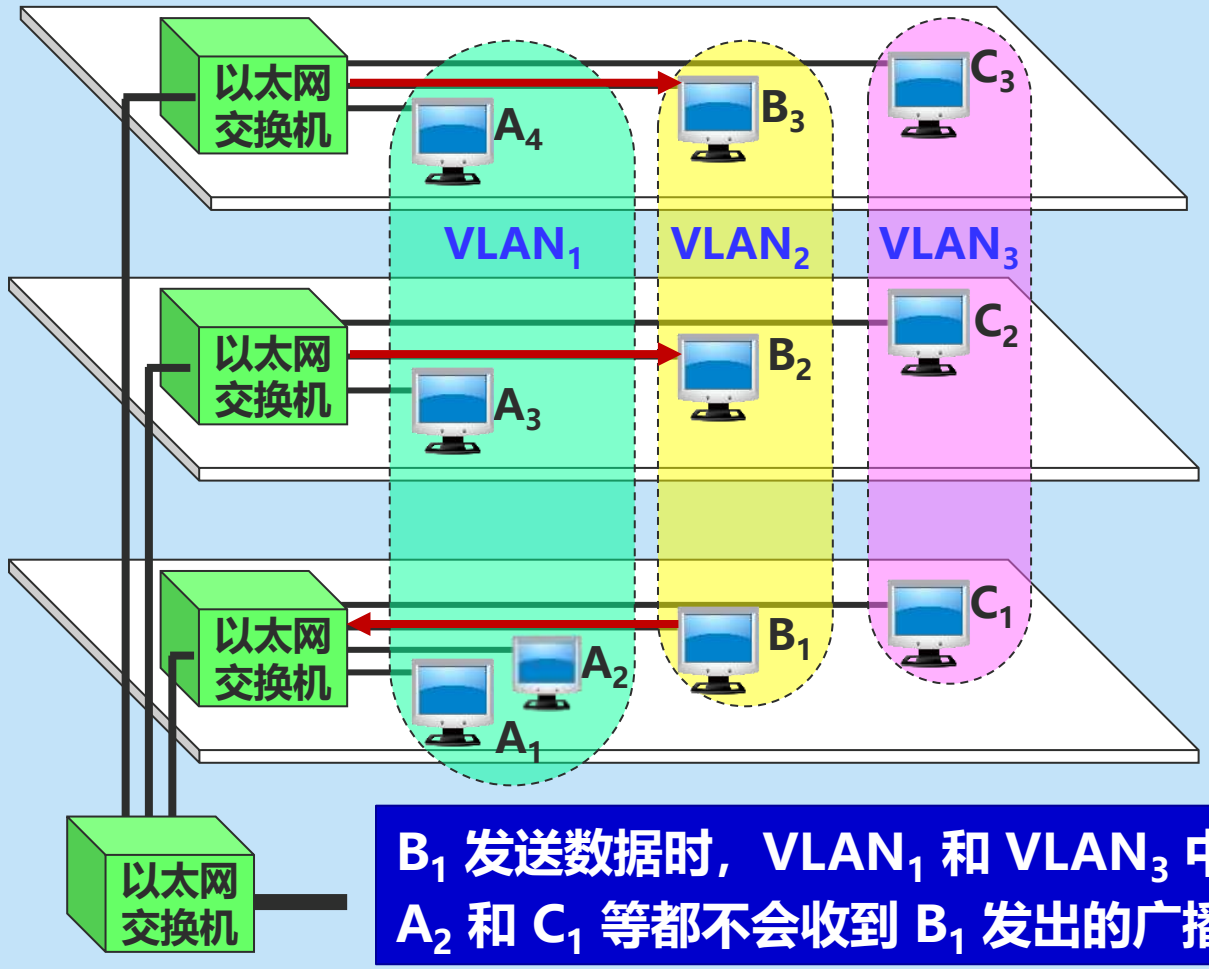


每个虚拟局域网是一个广播域。

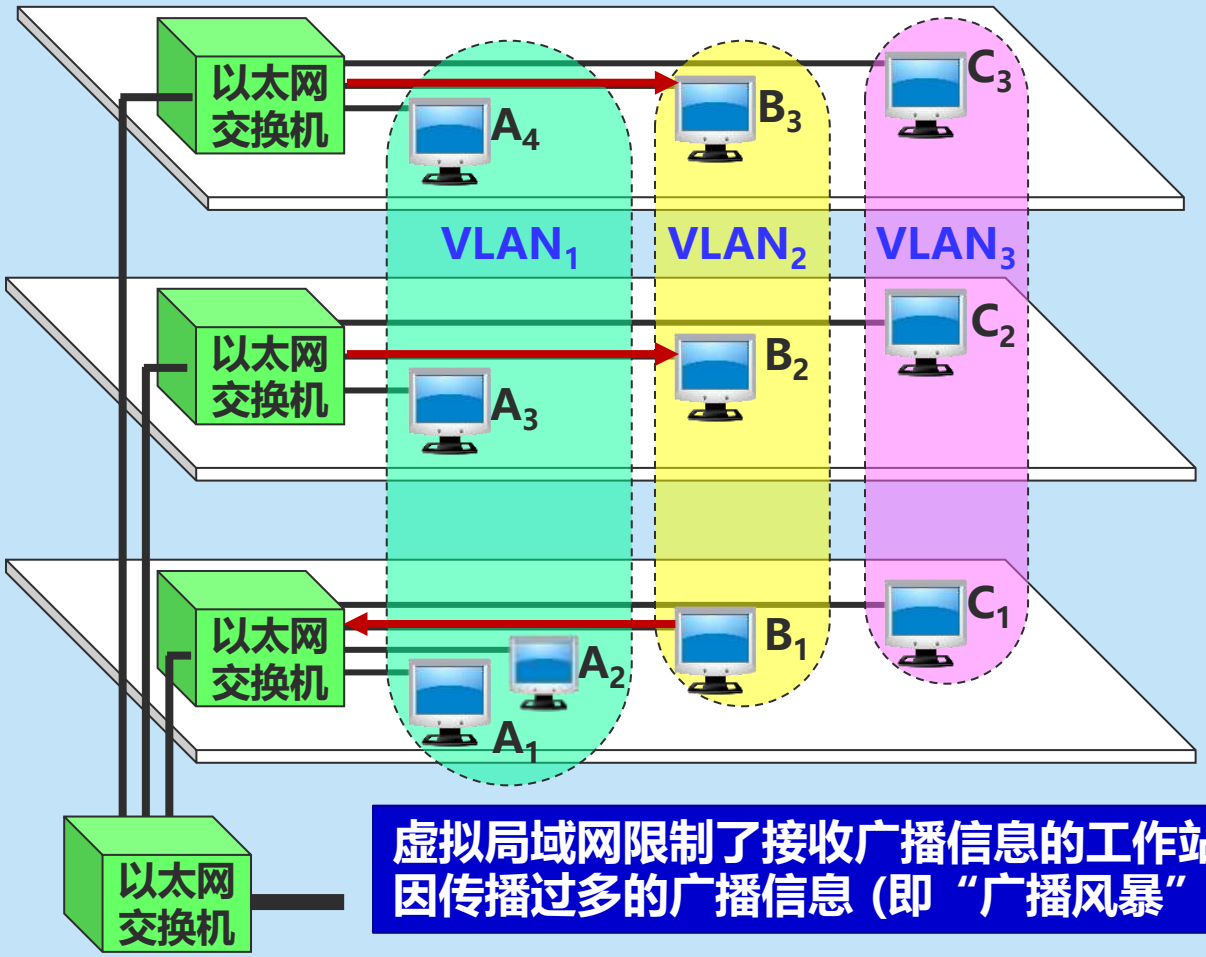
$VLAN_1$, $VLAN_2$ 和 $VLAN_3$ 是三个不同的广播域。



当 B_1 向 $VLAN_2$ 工作组内成员发送数据时，工作站 B_2 和 B_3 将会收到其广播的信息。



B_1 发送数据时，$VLAN_1$ 和 $VLAN_3$ 中的工作站 A_1，A_2 和 C_1 等都不会收到 B_1 发出的广播信息。



虚拟局域网限制了接收广播信息的工作站数，使得网络不会因传播过多的广播信息（即“广播风暴”）而引起性能恶化。

虚拟局域网优点

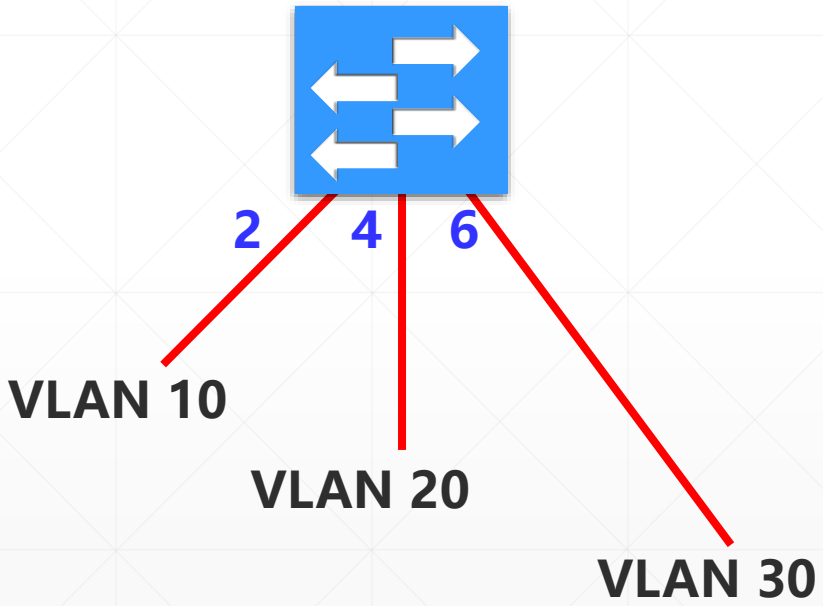
- 虚拟局域网（VLAN）技术具有以下主要优点：
  1. 改善了性能
  2. 简化了管理
  3. 降低了成本
  4. 改善了安全性

划分虚拟局域网的方法

- 基于交换机端口
- 基于计算机网卡的MAC地址
- 基于协议类型
- 基于IP子网地址
- 基于高层应用或服务

基于交换机端口的方法

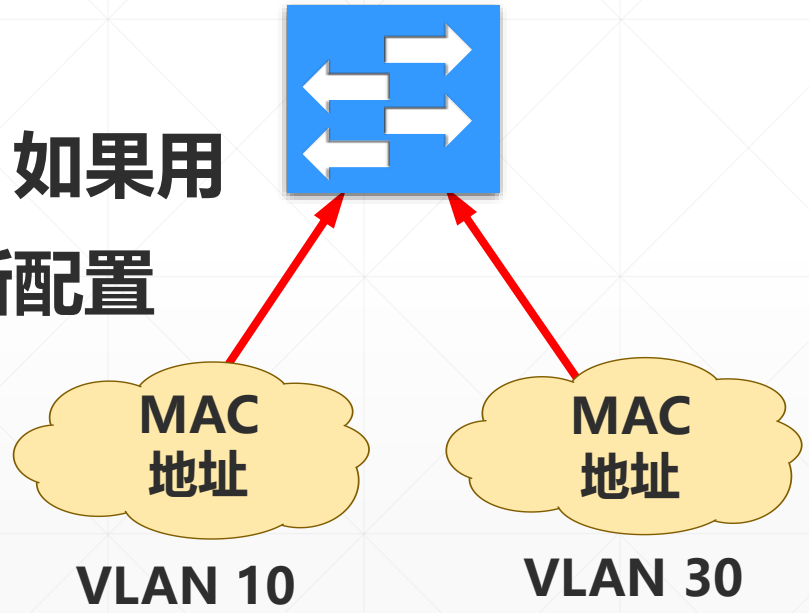
- 最简单、也是最常用的方法。
- 属于在第一层划分虚拟局域网的方法。
- **缺点**：不允许用户移动。



基于计算机网卡的MAC地址的方法

- 根据用户计算机的MAC地址划分虚拟局域网。
- 属于在第二层划分虚拟局域网的方法。
- 允许用户移动。
- **缺点**：需要输入和管理大量的MAC地址。如果用户的MAC地址改变了，则需要管理员重新配置VLAN。

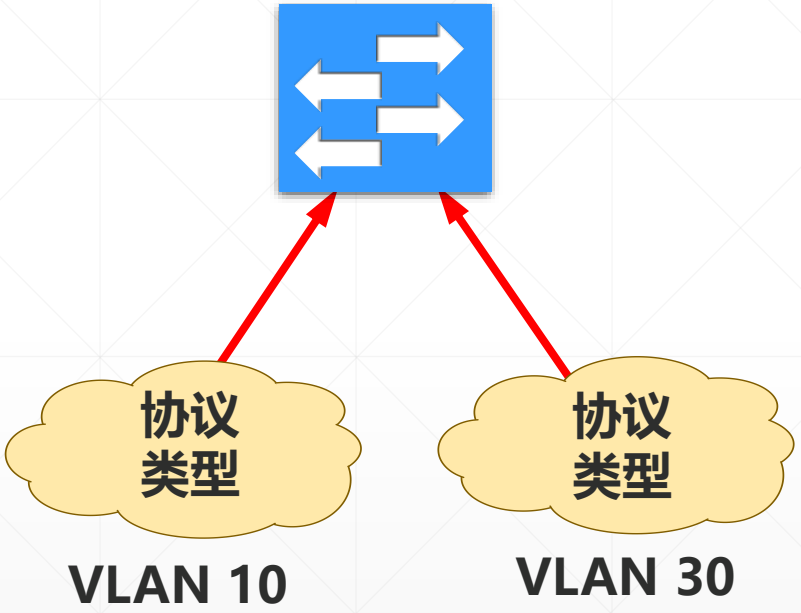
MAC 地址	VLAN
00-15-F5-CC-C8-14	10
C0-AB-D5-00-18-F4	10
C0-C5-18-DE-BC-E6	30



基于协议类型的方法

- 根据以太网帧的第三个字段“类型”字段确定该类型的协议属于哪一个虚拟局域网。
- 属于在第二层划分虚拟局域网的方法。

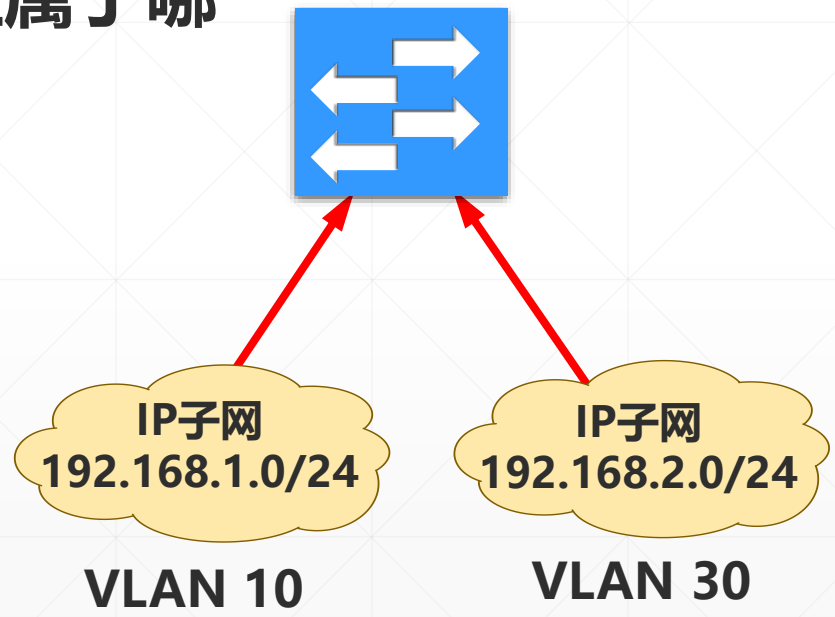
“类型”	VLAN
IP	10
IPX	30
……	…



基于IP子网地址的方法

- 根据以太网帧的第三个字段“类型”字段和IP分组首部中的源 IP 地址字段确定该 IP 分组属于哪一个虚拟局域网。
- 属于在第三层划分虚拟局域网的方法。

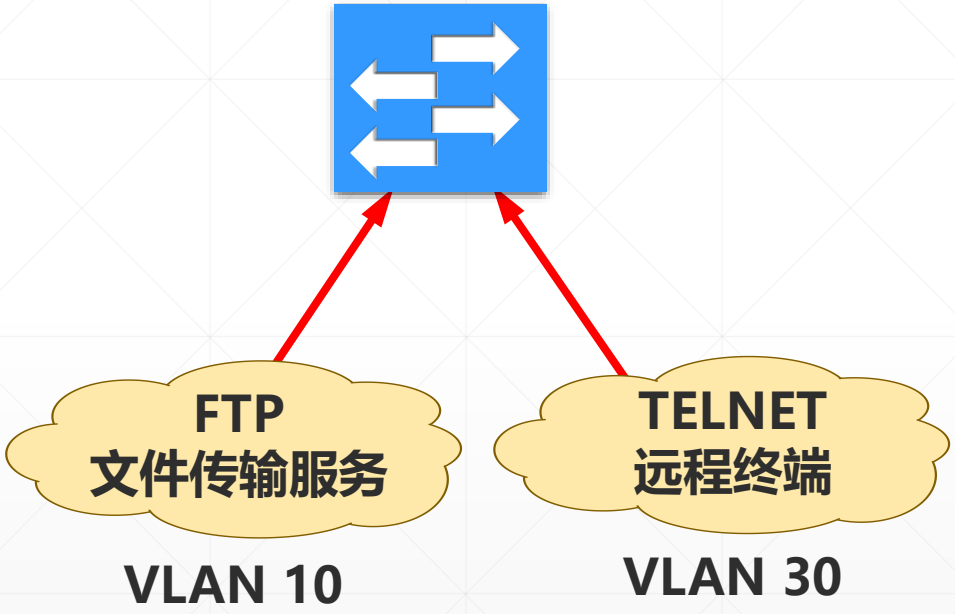
IP 子网	VLAN
192.168.1.0/24	10
192.168.2.0/24	30
……	…



基于高层应用或服务的方法

- 根据高层应用或服务、或者它们的组合划分虚拟局域网。
- 更加灵活，但更加复杂。

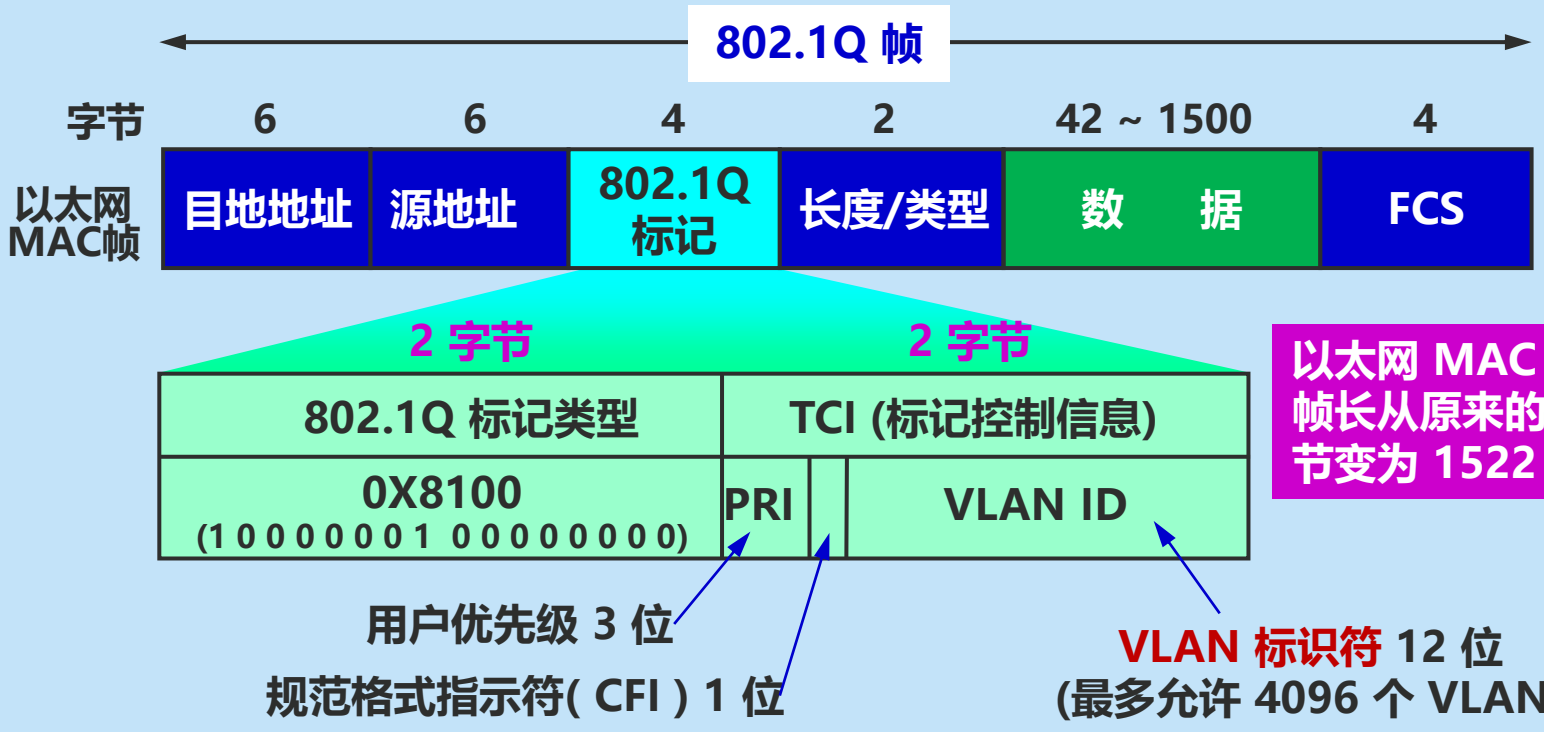
应用	VLAN
FTP	10
TELNET	30
……	…



虚拟局域网使用的以太网帧格式

- IEEE 批准了 802.3ac 标准，该标准定义了以太网的帧格式的扩展，以支持虚拟局域网。
- 虚拟局域网协议允许在以太网的帧格式中插入一个4字节的标识符，称为 **VLAN 标记** (tag)，用来指明该帧属于哪一个虚拟局域网。
- 插入VLAN标记得出的帧称为 **802.1Q** 帧或**带标记的以太网帧**。

虚拟局域网使用的以太网帧格式



以太网 MAC 帧的最大帧长从原来的 1518 字节变为 1522 字节。

插入 VLAN 标记后变成了 802.1Q 帧

虚拟局域网使用的以太网帧格式