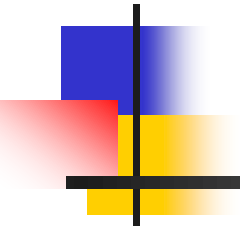


计算机网络（第 6 版）

第 3 章 数据链路层

第 3 章 数据链路层

第 3 章 数据链路层（续）

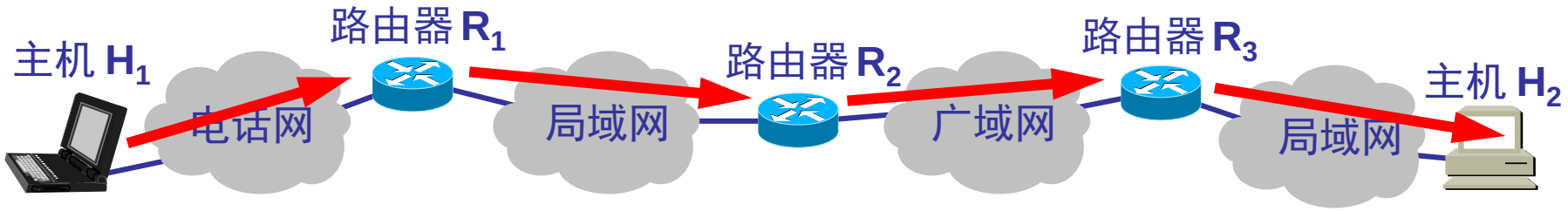
数据链路层

数据链路层使用的信道主要有以下两种类型：

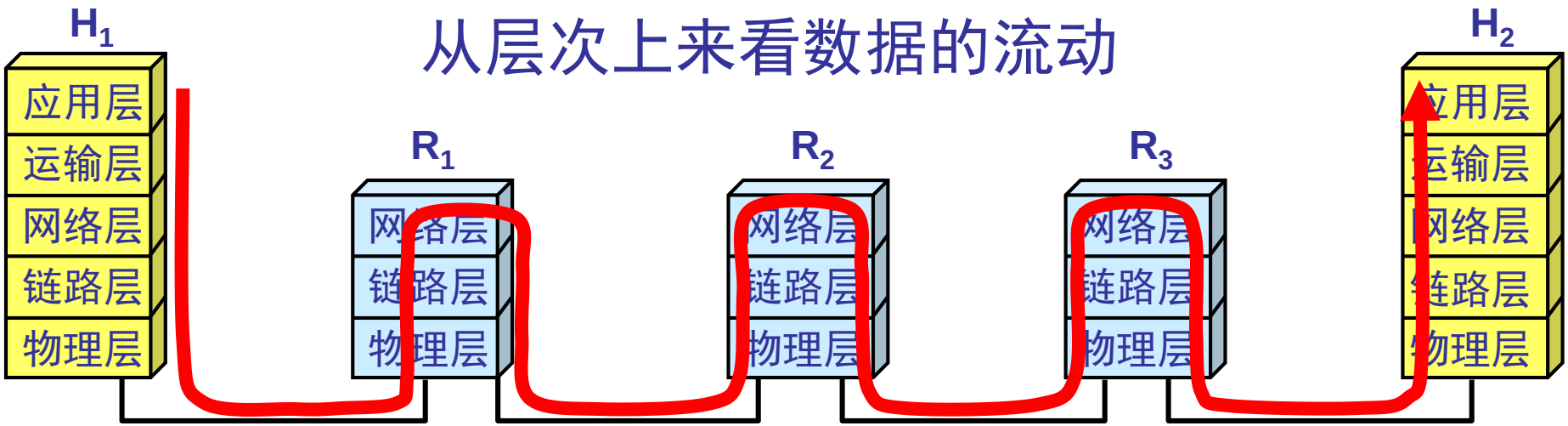
- 点对点信道。这种信道使用一对一的点对点通信方式。
- 广播信道。这种信道使用一对多的广播通信方式，因此过程比较复杂。广播信道上连接的主机很多，因此必须使用专用的共享信道协议来协调这些主机的数据发

数据链路层的简单模型

主机 H_1 向 H_2 发送数据

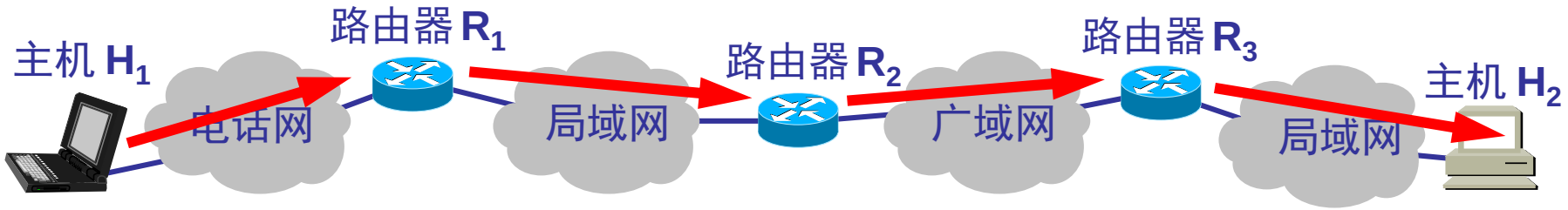


从层次上来看数据的流动

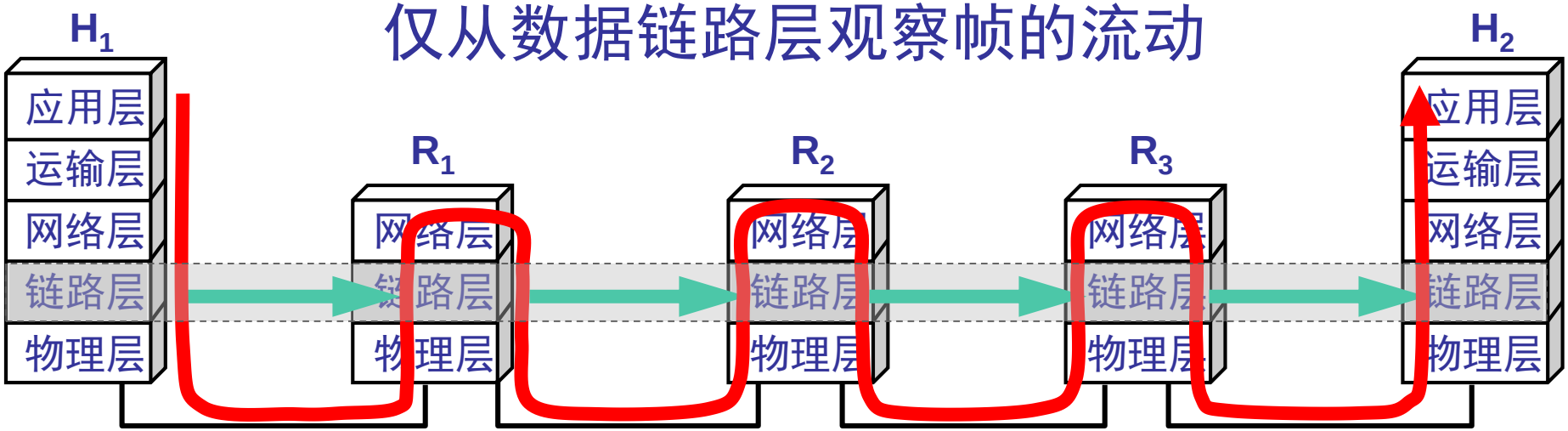


数据链路层的简单模型(续)

主机 H_1 向 H_2 发送数据



仅从数据链路层观察帧的流动

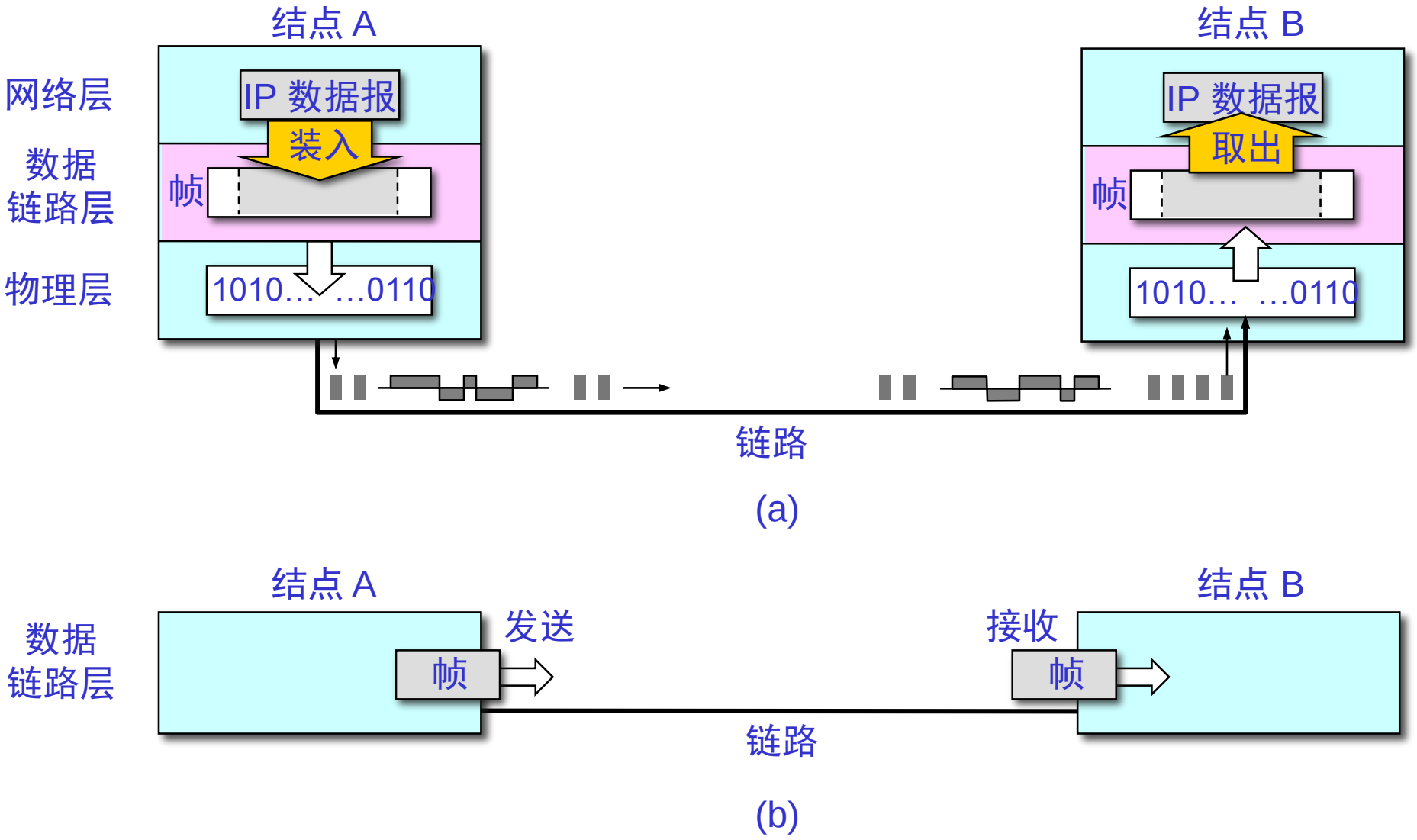


3.1 使用点对点信道的数据链路层

3.1.1 数据链路和帧

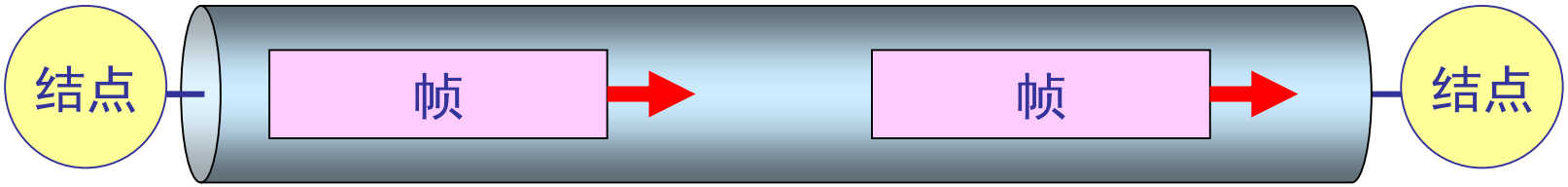
- 链路(link)是一条无源的点到点的物理线路段，中间没有任何其他的交换结点。
  - 一条链路只是一条通路的一个组成部分。
- 数据链路(data link) 除了物理线路外，还必须有通信协议来控制这些数据的传输。若把实现这些协议的硬件和软件加到链路上，就构成了数据链路。
  - 现在最常用的方法是使用适配器（即网卡）来实现这些协议的硬件和软件。
  - 一般的适配器都包括了数据链路层和物理层这两层的功能。

数据链路层传送的是帧


结点 A
结点 B
网络层
数据
链路层
物理层
IP 数据报
装入
帧
1010… …0110
IP 数据报
取出
帧
1010… …0110
链路
(a)
结点 A
结点 B
数据
链路层
发送
帧
接收
帧
链路
(b)

数据链路层像个数字管道

- 常常在两个对等的数据链路层之间画出一个数字管道，而在这条数字管道上传输的数据单位是帧。



- 早期的数据通信协议曾叫作通信规程(procedure)。因此在数据链路层，规程和协议是同义语。

3.1.2 三个基本问题

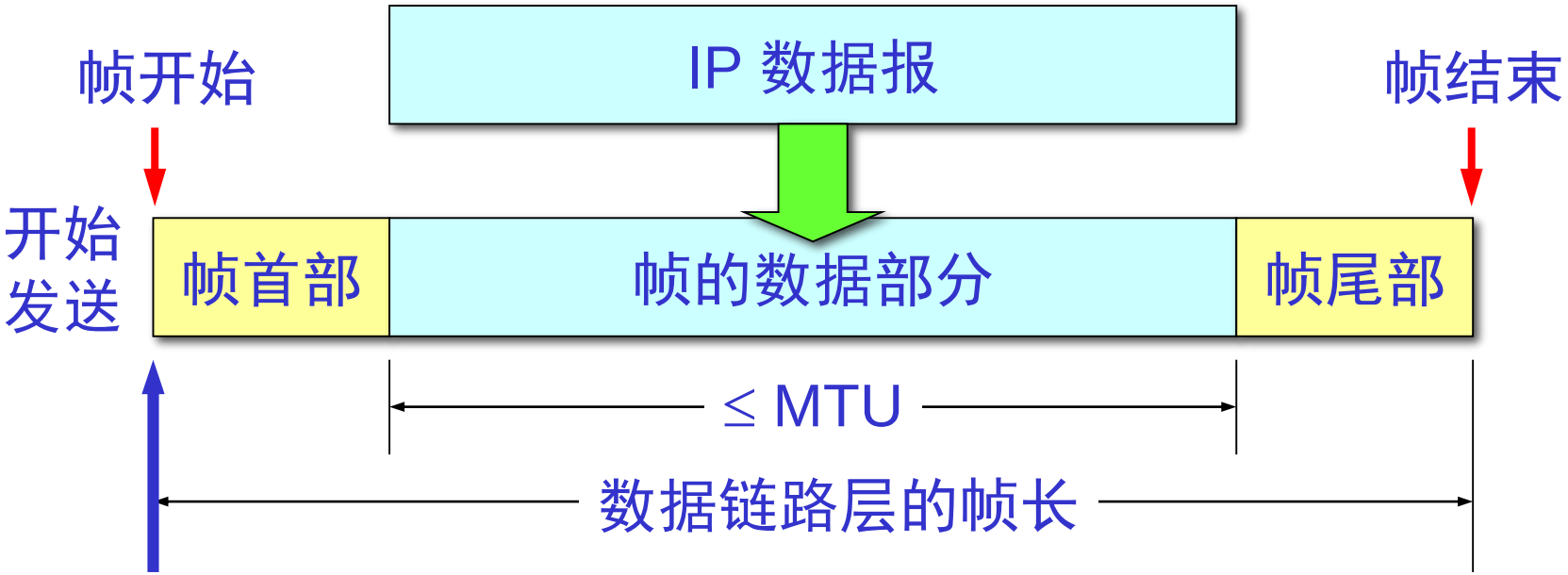
(1) 封装成帧

(2) 透明传输

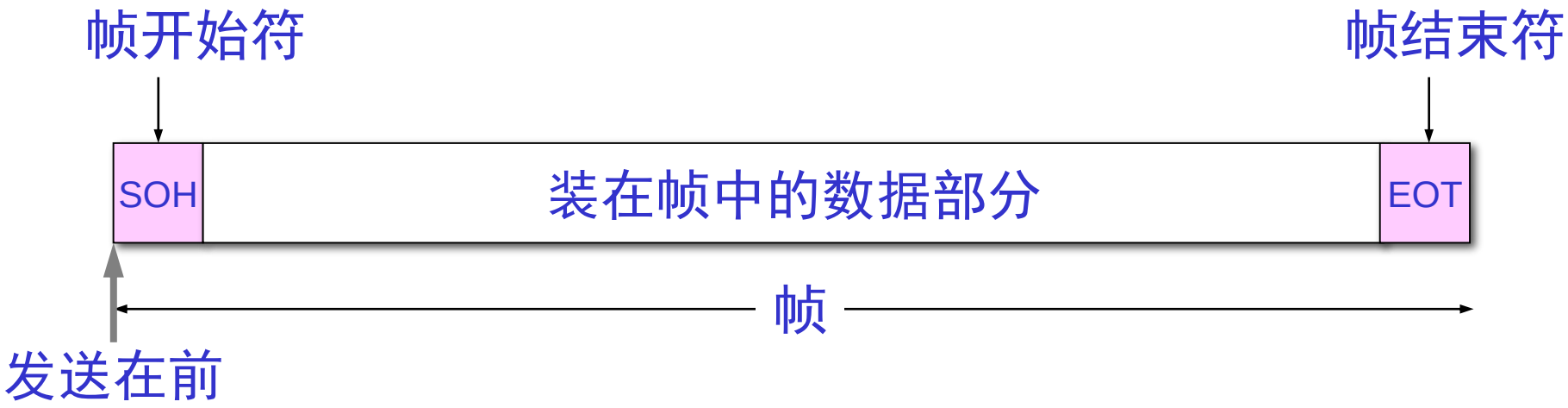
(3) 差错控制

1. 封装成帧

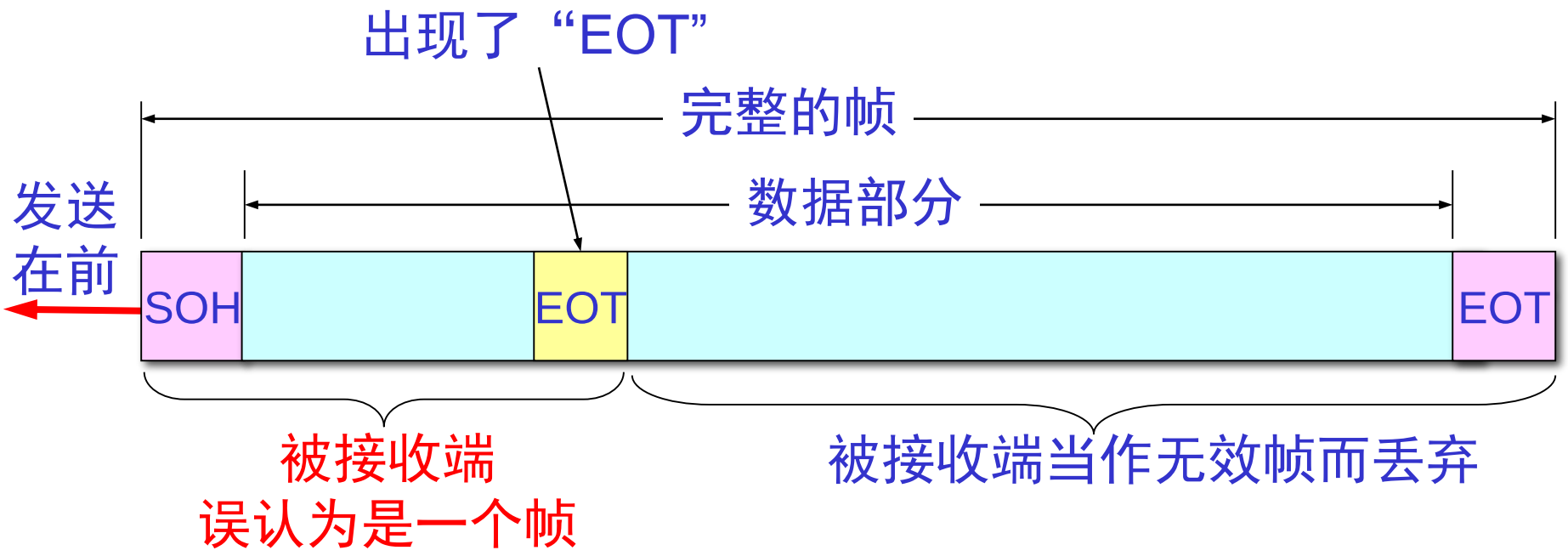
- 封装成帧(framing)就是在一段数据的前后分别添加首部和尾部，然后就构成了一个帧。确定帧的界限。
- 首部和尾部的一个重要作用就是进行帧定界。



用控制字符进行帧定界的方法举例



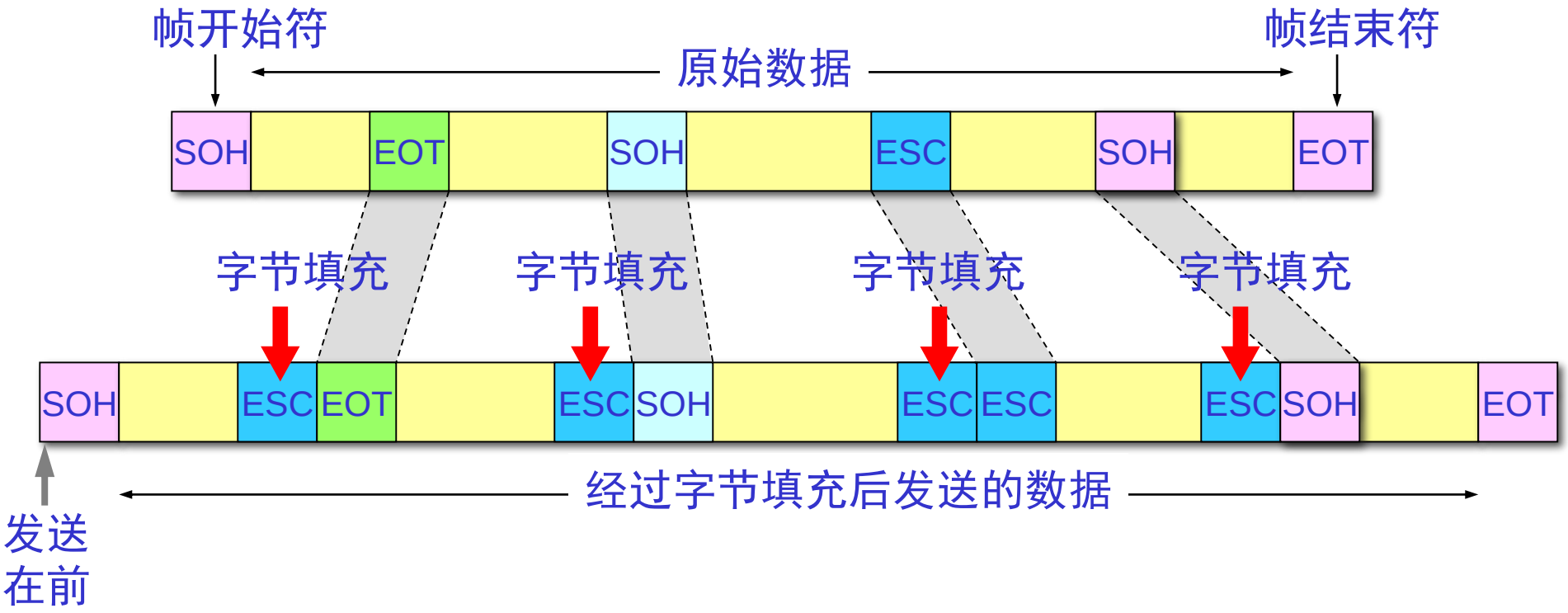
2. 透明传输


出现了“EOT”
完整的帧
数据部分
发送
在前
SOH
EOT
EOT
被接收端
误认为是一个帧
被接收端当作无效帧而丢弃

解决透明传输问题

- 发送端的数据链路层在数据中出现控制字符“SOH”或“EOT”的前面插入一个转义字符“ESC”(其十六进制编码是 1B)。
- 字节填充(byte stuffing)或字符填充(character stuffing)——接收端的数据链路层在将数据送往网络层之前删除插入的转义字符。
- 如果转义字符也出现数据当中，那么应在转义字符前面插入一个转义字符。当接收端收到连续的两个转义字符时，就删除其中前面的一个。

用字节填充法解决透明传输的问题



3. 错误检测和纠正（1）

- 差错出现的特点：随机，连续突发（burst）
- 处理差错的两种基本策略
  - 使用纠错码：发送方在每个数据块中加入足够的冗余信息，使得接收方能够判断接收到的数据是否有错，并能纠正错误。
  - 使用检错码：发送方在每个数据块中加入足够的冗余信息，使得接收方能够判断接收到的数据是否有错，但不能判断哪里有错。

3. 错误检测和纠正（2）

1. 纠错码

- 码字（codeword）：一个帧包括m个数据位，r个校验位，n = m + r，则此n比特单元称为n位码字。
- 海明距离（Hamming distance）：两个码字之间不同的比特(对应)位数目。
  - 例：0000000000 与0000011111的海明距离为5
  - 如果两个码字的海明距离为d，则需要d个单比特错就可以把一个码字转换成另一个码字；
- 对于n位码字的集合，只有2^m个码字是有效的。也就是说，通常并未使用所有2^n个码字。

检错和纠错

- 一种编码的检错和纠错能力取决于编码后码字海明距离的大小。
- 为了检测出d个比特的错，需要使用海明距离为d+1的编码
  例如：数据后加奇偶校验位，编码后的海明距离为2，能检测1比特错。
- 例如：
  000 0(前三个数据位，后一个是冗余的奇偶校验位)
  001 1
  010 1
  100 1
  011 0
  100 1
  101 0
  110 0
  111 1

  共有8个有效的码字，你会发现两位不同，即海明距离为2，也就是需要改变两位才能变成另一个有效的码字，而奇偶校验只能检测一位错。(此例为偶校验，码字为奇校验的为无效码字)

检错和纠错

- 在任意两个有效码字间找出具有最小海明距离的两个码字，该海明距离便定义为全部码字的海明距离。
- 为了纠正d个比特的错，必须用距离为2d+1的编码。

  例如有4个有效码字：它们是0000000000，0000011111，1111100000，1111111111，海明距离为5，能纠正2比特错。

3. 错误检测和纠正（3）

- 最简单的例子是奇偶校验，在数据后填加一个奇偶位
  - 例：使用偶校验（“1”的个数为偶数）

    10110101——>101101011

    10110001——>101100010
  - 奇偶校验可以用来检查奇数个错误。
- 要求设计仅纠正单比特错的纠错码

设计纠错码

- 要求：m个信息位，r个校验位，当r满足什么条件时，能纠正所有单比特错；
- 对2^m个合法报文的任何一个而言，有n个与该报文距离为1的无效码字，所以2^m个合法报文的每一个都对应有n+1个各不相同的位图，n位码字的总的位图是2^n个：

$(n+1)\ 2^m <= 2^n$，$n=m+r$代入

$(m+r+1)\ 2^m <= 2^{m+r}$

$2^r >= m+r+1$ ⇨ 纠正单比特误码的校验位下界r

3. 错误检测和纠正（4）

- 海明码
  - 码位从左边开始编号，从“1”开始；
  - 位号为2的幂的位是校验位，其余是信息位；
  - 每个校验位使得包括自己在内的一些位的奇偶值为偶数（或奇数）。
  - 为看清数据位k对哪些校验位有影响，将k写成2的幂的和。例：11 = 1 + 2 + 8

3. 错误检测和纠正（5）

- 海明码工作过程
  - 每个码字到来前，接收方计数器清零；
  - 接收方检查每个校验位k (k = 1, 2, 4 ...)的奇偶值是否正确；
  - 若第 k 位奇偶值不对，计数器加 k；
  - 所有校验位检查完后，若计数器值为0，则码字有效；若计数器值为m，则第m位出错。例：若校验位1、2、8出错，则第11位变反。

3. 错误检测和纠正（6）

- 使用海明码纠正突发错误
  - 可采用k个码字（n = m + r）组成 k × n 矩阵，按列发送，接收方恢复成 k × n 矩阵
  - kr个校验位，km个数据位，可纠正最多为k个的突发性连续比特错。

海明编码举例

字符	ASCII	校验位
H	1001000	00110010000
a	1100001	10111001001
m	1101101	11101010101
m	1101101	11101010101
i	1101001	01101011001
n	1101110	01101010110
g	1100111	11111001111
	0100000	10011000000
c	1100011	11111000011
o	1101111	00101011111
d	1100100	11111001100
e	1100101	00111000101

位传输的顺序

1	***2***	3	***4***	5	6	7	***8***	9	10	11
		1		1		1		1		1
		2			2	2			2	2
				4	4	4				
								8	8	8

K取不同值时，所对应的校验位。如，位置3由1、2位置的校验位校验，5由1、4校验…

上例中，m=7，r=4，n=11，显然2^4>=11+1，采用偶校验

3=1+2，5=1+4

6=2+4，7=1+2+4

9=1+8，10=2+8

11=1+2+8

校验位：1∈（3，5，7，9，11）

2 ∈（3，6，7，10，11）

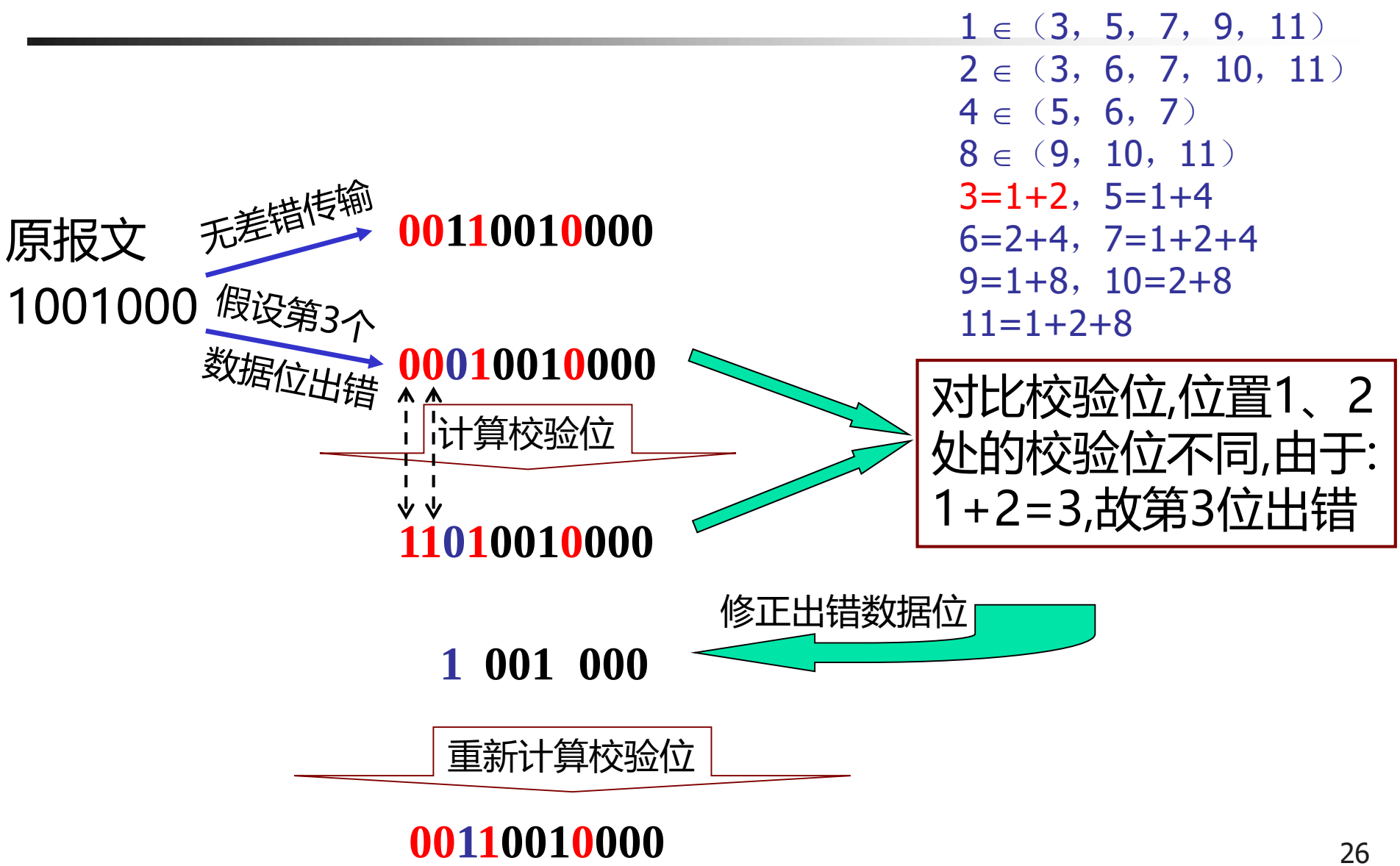
4 ∈（5，6，7）

8 ∈（9，10，11）

能纠正单比特错!

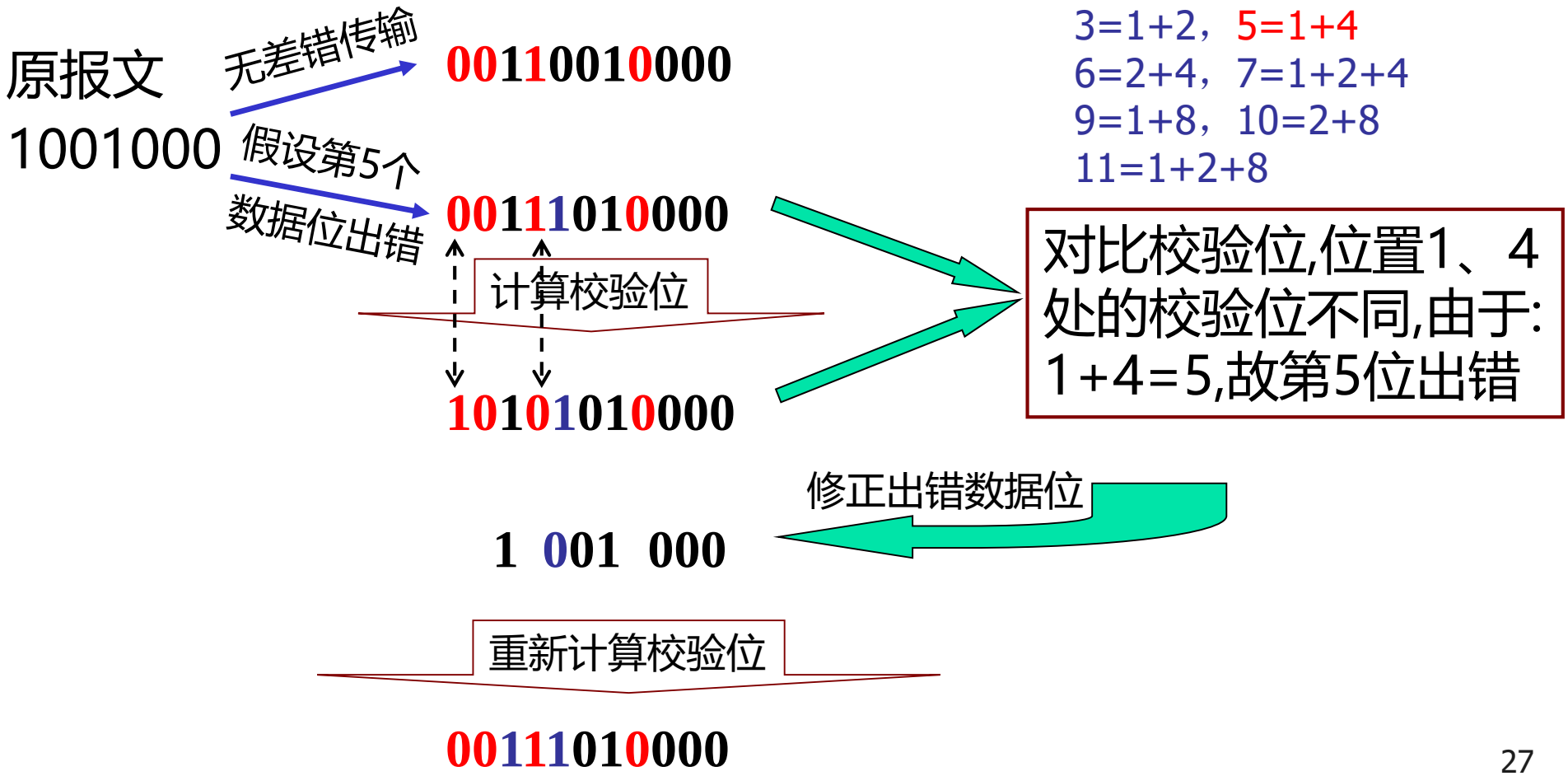
例如，在接收方，如果校验位1不满足偶校验，而其他校验位都满足，则第1位出错，…

纠正码字中的单个数据位错误

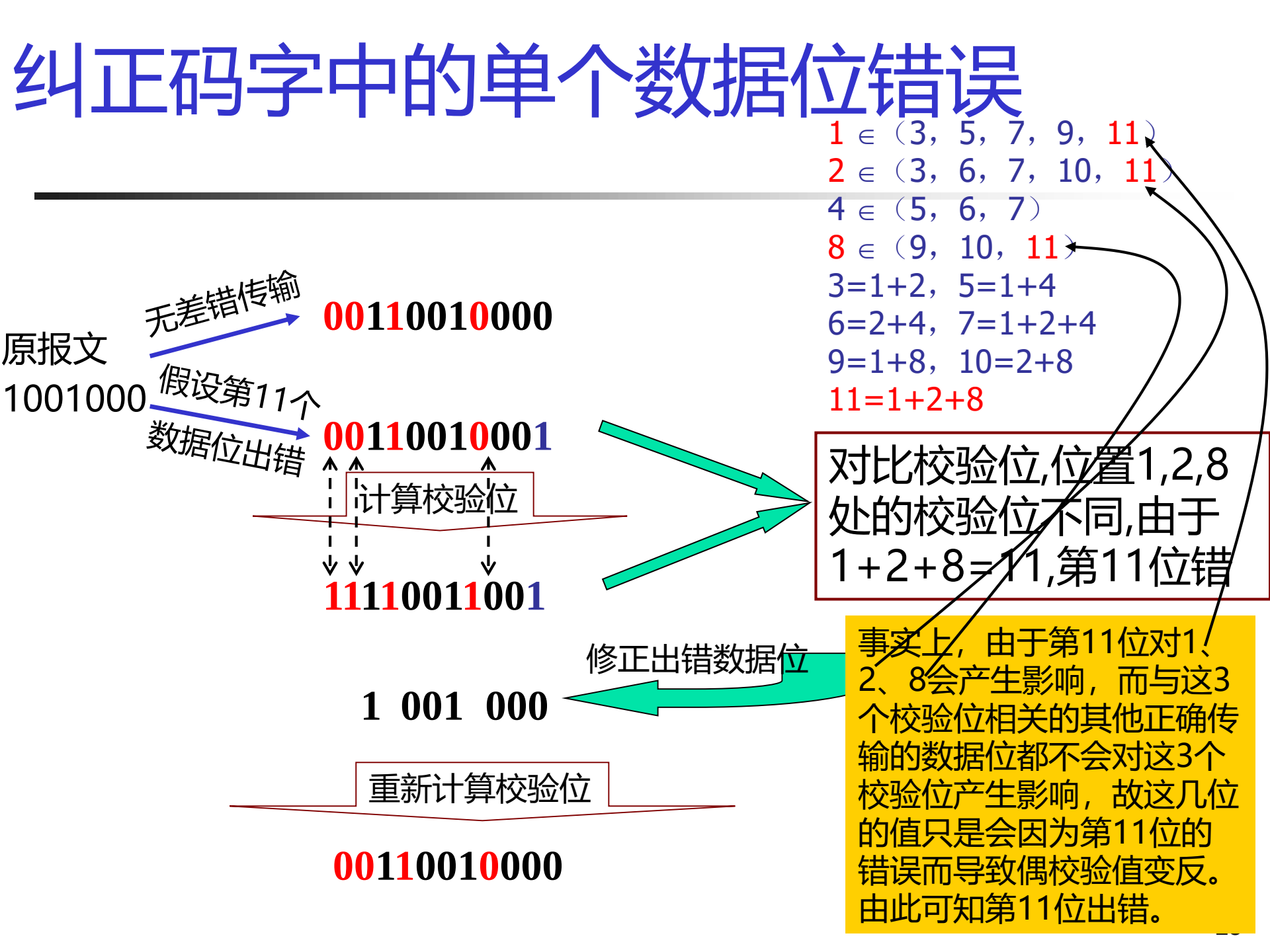


纠正码字中的单个数据位错误

1 ∈（3，5，7，9，11）
2 ∈（3，6，7，10，11）
4 ∈（5，6，7）
8 ∈（9，10，11）
3=1+2，5=1+4
6=2+4，7=1+2+4
9=1+8，10=2+8
11=1+2+8

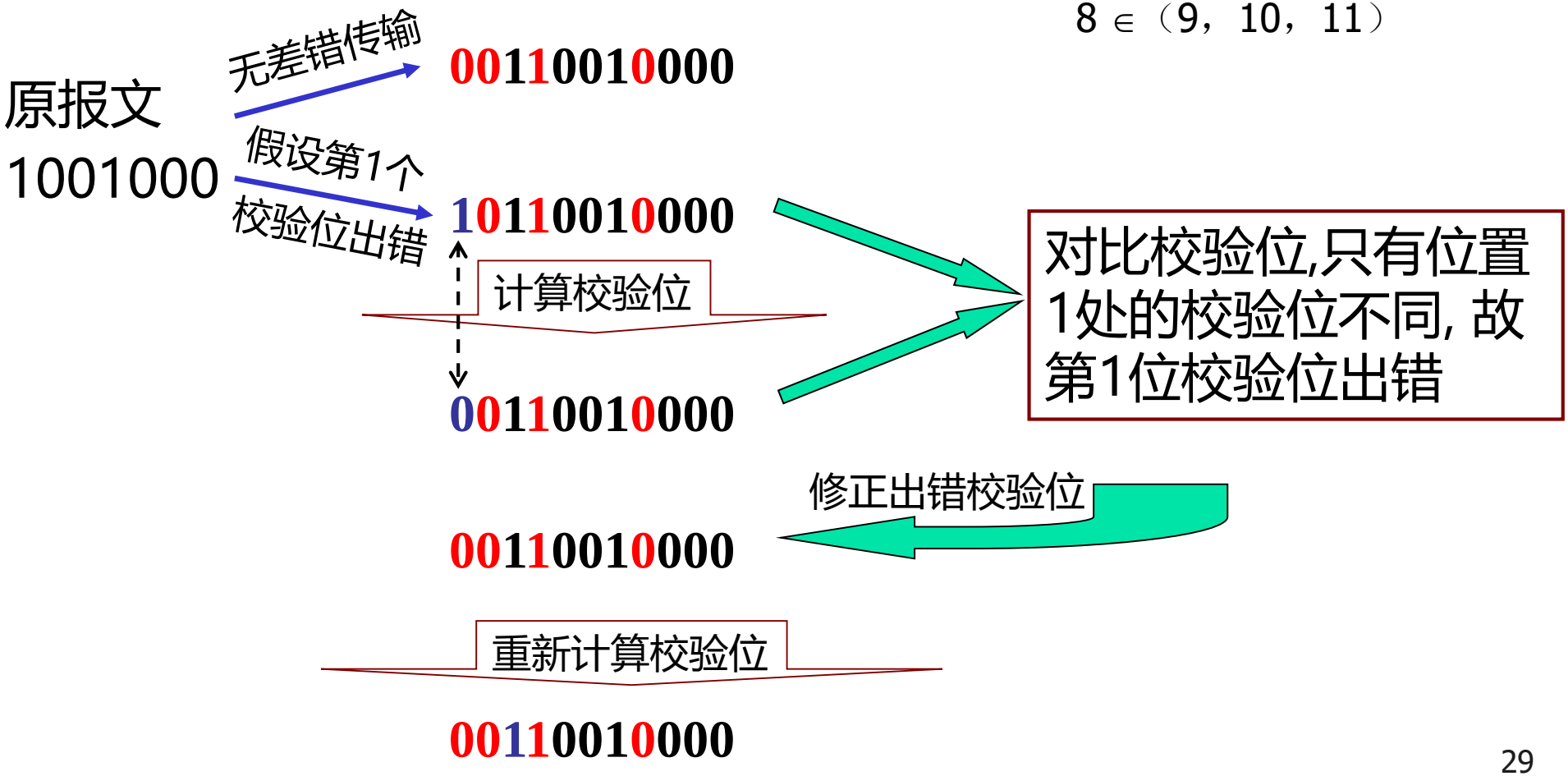


纠正码字中的单个数据位错误



纠正码字中的单个校验位错误

1 ∈（3，5，7，9，11）
2 ∈（3，6，7，10，11）
4 ∈（5，6，7）
8 ∈（9，10，11）

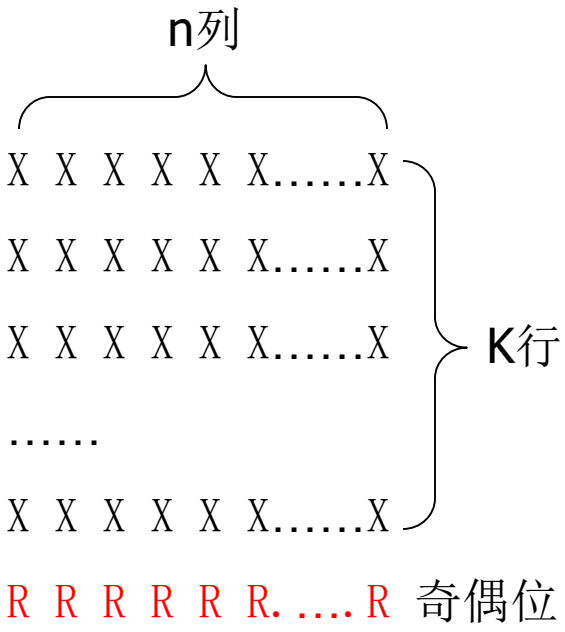


纠错码与检错码

- 在实际通信中使用纠错码好还是检错码好呢?
- 例如：假设一个信道误码率是10^{-6}，且出错是孤立产生的（即只有单比特错），数据块长度为1000比特，如果采用纠错编码，需要10个校验位（$2^{10}>1011$），传送1M数据需要10000个校验位；如果采用检错编码，每个数据块只需一个奇偶校验位，传送1M数据只需1000个校验位和一个重传的数据1001位，共需要2001比特的额外开销。
- 在多数通信中采用检错编码，但在单工信道中需要纠错编码。

改进的奇偶校验

- 将数据位组成一个n位宽，K位高的长方形距阵来发送，然后对每一列单独计算奇偶位，并附在最后一行作为冗余位。



检错率：

1.该方法可以检测长度为n的突发性错误，但不能检测长度为n+1的突发性错误。

2.假设n列中任意一列检测出错的概率为1/2，那么，整个数据块的错判率为（1/2）^{n}。

该方法用在ICMP报头检验中。

3. 错误检测和纠正（7）

2. 检错码

- 使用纠错码传数据，效率低，适用于不可能重传的场合；大多数情况采用检错码加重传。
- 循环冗余码（CRC码，多项式编码）
  - 110001，表示成多项式 $x^5 + x^4 + 1$
- 生成多项式G(x)
  - 发方、收方事前商定；
  - 生成多项式的高位和低位必须为1
  - 生成多项式必须比传输信息对应的多项式短。

3. 错误检测和纠正（8）

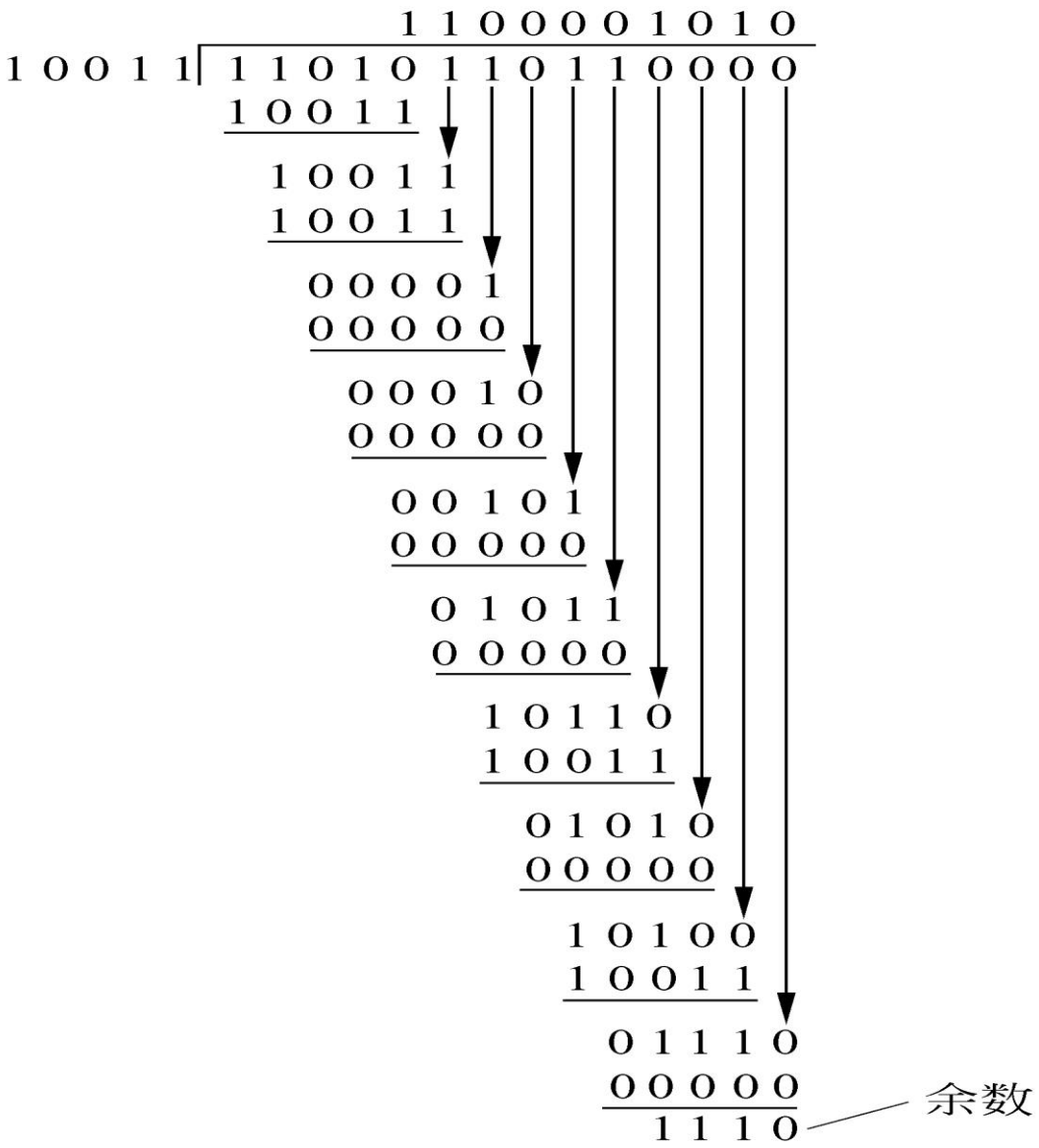
- CRC码基本思想
  - 校验和（checksum）加在帧尾，使带校验和的帧的多项式能被G(x)除尽；收方接收时，用G(x)去除它，若有余数，则传输出错。
- 校验和计算算法
  - 设G(x)为 r 阶，在帧的末尾加 r 个0，使帧为m+r位，相应多项式为$x^rM(x)$；
  - 按模2除法用对应于G(x)的位串去除对应于$x^rM(x)$的位串；
  - 按模2减法从对应于$x^rM(x)$的位串中减去余数(等于或小于r位)，结果就是要传送的带校验和的多项式T(x)。

帧 ：1 1 0 1 0 1 1 0 1 1 (**M(X)**)

除数 ：1 0 0 1 1 (**G(X)**)

附加 4 个零后形成的串 ：1 1 0 1 0 1 1 0 1 1 0 0 0 0 (**M(X)*X^r**)

多项式编码校验和计算示例



传输的帧 ：1 1 0 1 0 1 1 0 1 1 1 1 1 0 (**T(X)**)

3. 错误检测和纠正（9）

- CRC的检错能力
  - 发送：T(x)；接收：T(x) + E(x), $E(x) \neq 0$；
  - 余数((T(x) + E(x)) / G(x)) = 0 + 余数(E(x) / G(x))
  - 若余数(E(x) / G(x)) = 0，则差错不能发现；否则，可以发现。

3. 错误检测和纠正（10）

- CRC检错能力的几种情况分析
  - 如果只有单比特错，即$E(x) = x^i$，而G(x)中至少有两项，余数$(E(x) / G(x)) \neq 0$，所以可以查出单比特错；
  - 如果发生两个孤立单比特错，即$E(x) = x^i + x^j = x^j(x^{i-j} + 1)$，假定G(x)不能被x整除，那么能够发现两个比特错的充分条件是：$x^k + 1$不能被G(x)整除$(k \leq i - j)$；
  - 如果有奇数个比特错，即E(x)包括奇数个项，G(x)选$(x + 1)$的倍数就能查出奇数个比特错；

3. 错误检测和纠正（11）

- 具有r个校验位的多项式能检查出所有长度 ≤ r 的突发性差错。长度为k的突发性连续差错可表示为 $x^i (x^{k-1} + \ldots + 1)$，若G(x)包括$x^0$项，且 k - 1小于G(x)的阶，则 余数(E(x) / G(x)) ≠ 0;
- 如果突发差错长度为 r + 1，当且仅当突发差错和G(x)一样时，余数(E(x) / G(x)) = 0，概率为$1/2^{r-1}$;
- 长度大于 r + 1的突发差错或几个较短的突发差错发生后，坏帧被接收的概率为 $1/2^r$。

帧检验序列 FCS

- 在数据后面添加上的冗余码称为帧检验序列 FCS (Frame Check Sequence)。
- 循环冗余检验 CRC 和帧检验序列 FCS并不等同。
  - CRC 是一种常用的检错方法，而 FCS 是添加在数据后面的冗余码。
  - FCS 可以用 CRC 这种方法得出，但 CRC 并非用来获得 FCS 的惟一方法。

检测出差错

- 只要得出的余数R不为0，就表示检测到了差错。
- 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。
- 一旦检测出差错，就丢弃这个出现差错的帧。
- 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。

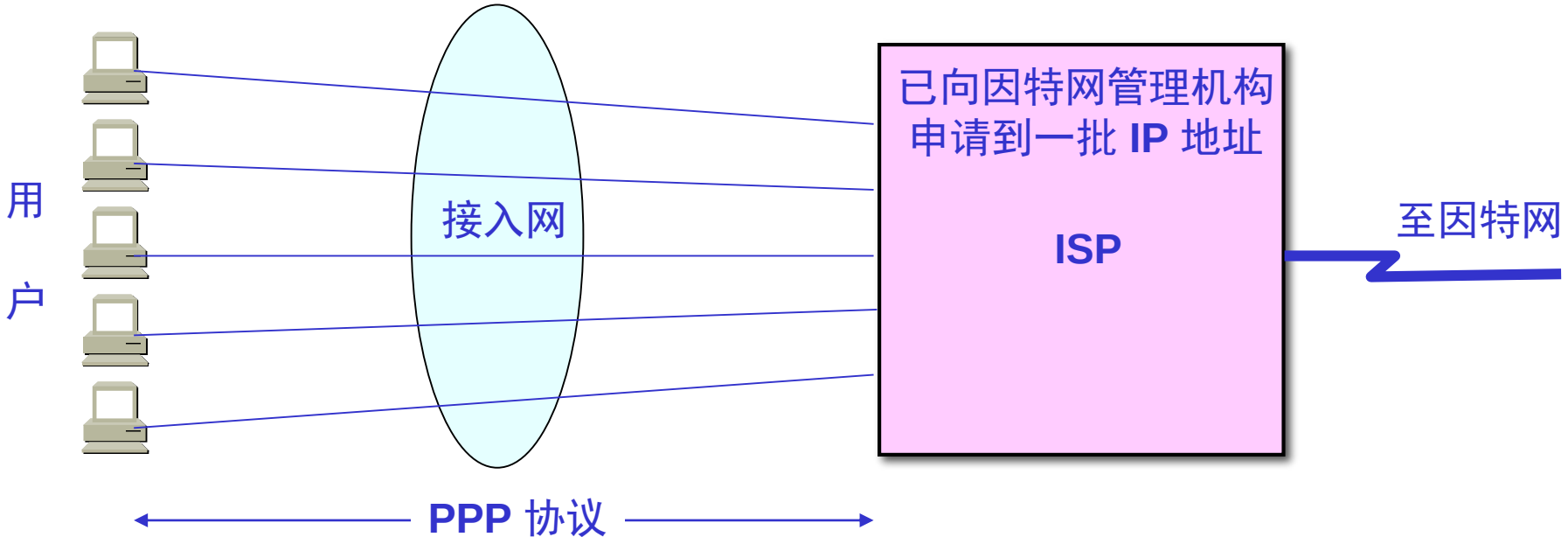
应当注意

- 仅用循环冗余检验 CRC 差错检测技术只能做到无差错接受(accept)。
- “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：“凡是接受的帧都没有传输差错”（有差错的帧就丢弃而不接受）。
- 要做到“可靠传输”（即发送什么就收到什么）就必须再加上确认和重传机制。

3.2 点对点协议 PPP
3.2.1 PPP 协议的特点

- 现在全世界使用得最多的数据链路层协议是点对点协议 PPP (Point-to-Point Protocol)。
- 用户使用拨号电话线接入因特网时，一般都是使用 PPP 协议。

用户到 ISP 的链路使用 PPP 协议



1. PPP 协议应满足的需求

- 简单——这是首要的要求
- 封装成帧
- 透明性
- 多种网络层协议
- 多种类型链路
- 差错检测
- 检测连接状态
- 最大传送单元
- 网络层地址协商
- 数据压缩协商

2. PPP 协议不需要的功能

- 纠错
- 流量控制
- 序号
- 多点线路
- 半双工或单工链路

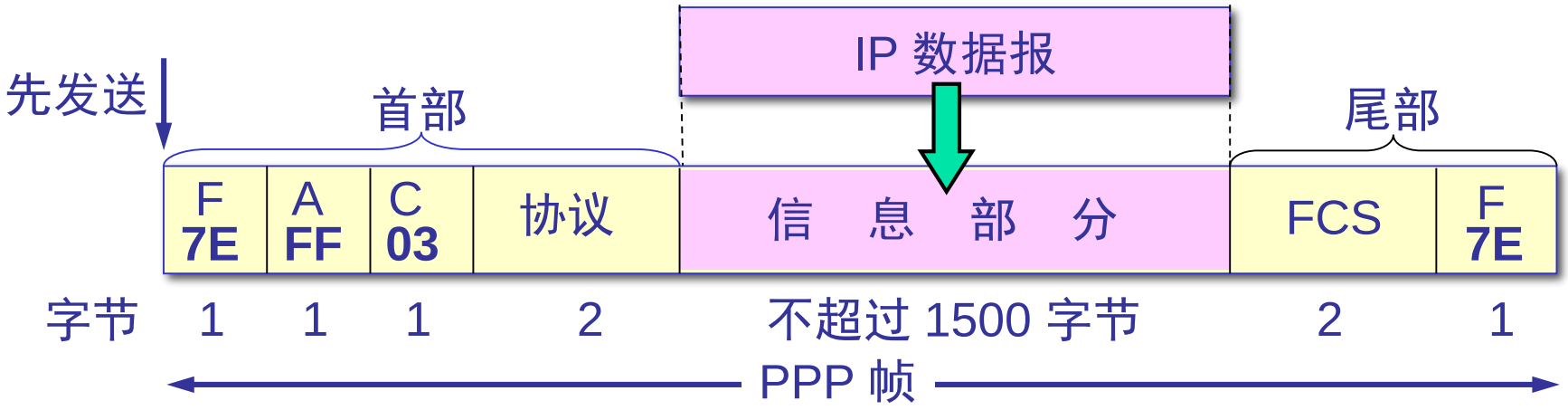
3. PPP 协议的组成

- 1992 年制订了 PPP 协议。经过 1993 年和 1994 年的修订，现在的 PPP 协议已成为因特网的正式标准[RFC 1661]。
- PPP 协议有三个组成部分
  - 一个将 IP 数据报封装到串行链路的方法。
  - 链路控制协议 LCP (Link Control Protocol)。
  - 网络控制协议 NCP (Network Control Protocol)。

3.2.2 PPP 协议的帧格式

- 标志字段 F = 0x7E （符号“0x”表示后面的字符是用十六进制表示。十六进制的 7E 的二进制表示是 01111110）。
- 地址字段 A 只置为 0xFF。地址字段实际上并不起作用。
- 控制字段 C 通常置为 0x03。
- PPP 是面向字节的，所有的 PPP 帧的长度都是整数字节。

PPP 协议的帧格式



- PPP 有一个 2 个字节的协议字段。
  - 当协议字段为 0x0021 时，PPP 帧的信息字段就是IP 数据报。
  - 若为 0xC021, 则信息字段是 PPP 链路控制数据。
  - 若为 0x8021，则表示这是网络控制数据。

透明传输问题

- 当 PPP 用在同步传输链路时，协议规定采用硬件来完成比特填充（和 HDLC 的做法一样）。
- 当 PPP 用在异步传输时，就使用一种特殊的字符填充法。

字符填充

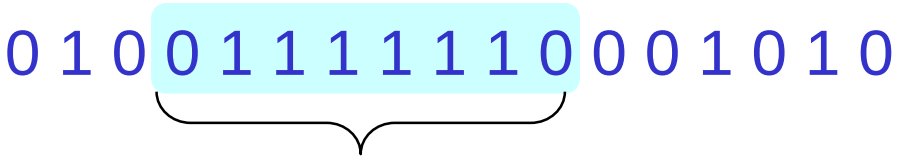
- 将信息字段中出现的每一个 0x7E 字节转变成为 2 字节序列(0x7D, 0x5E)。
- 若信息字段中出现一个 0x7D 的字节, 则将其转变成为 2 字节序列(0x7D, 0x5D)。
- 若信息字段中出现 ASCII 码的控制字符（即数值小于 0x20 的字符），则在该字符前面要加入一个 0x7D 字节，同时将该字符的编码加以改变。

零比特填充

- PPP 协议用在 SONET/SDH 链路时，是使用同步传输（一连串的比特连续传送）。这时 PPP 协议采用零比特填充方法来实现透明传输。
- 在发送端，只要发现有 5 个连续 1，则立即填入一个 0。接收端对帧中的比特流进行扫描。每当发现 5 个连续1时，就把这 5 个连续 1 后的一个 0 删除，

零比特填充

信息字段中出现了和标志字段 F 完全一样的 8 比特组合



发送端在 5 个连 1 之后填入 0 比特再发送出去

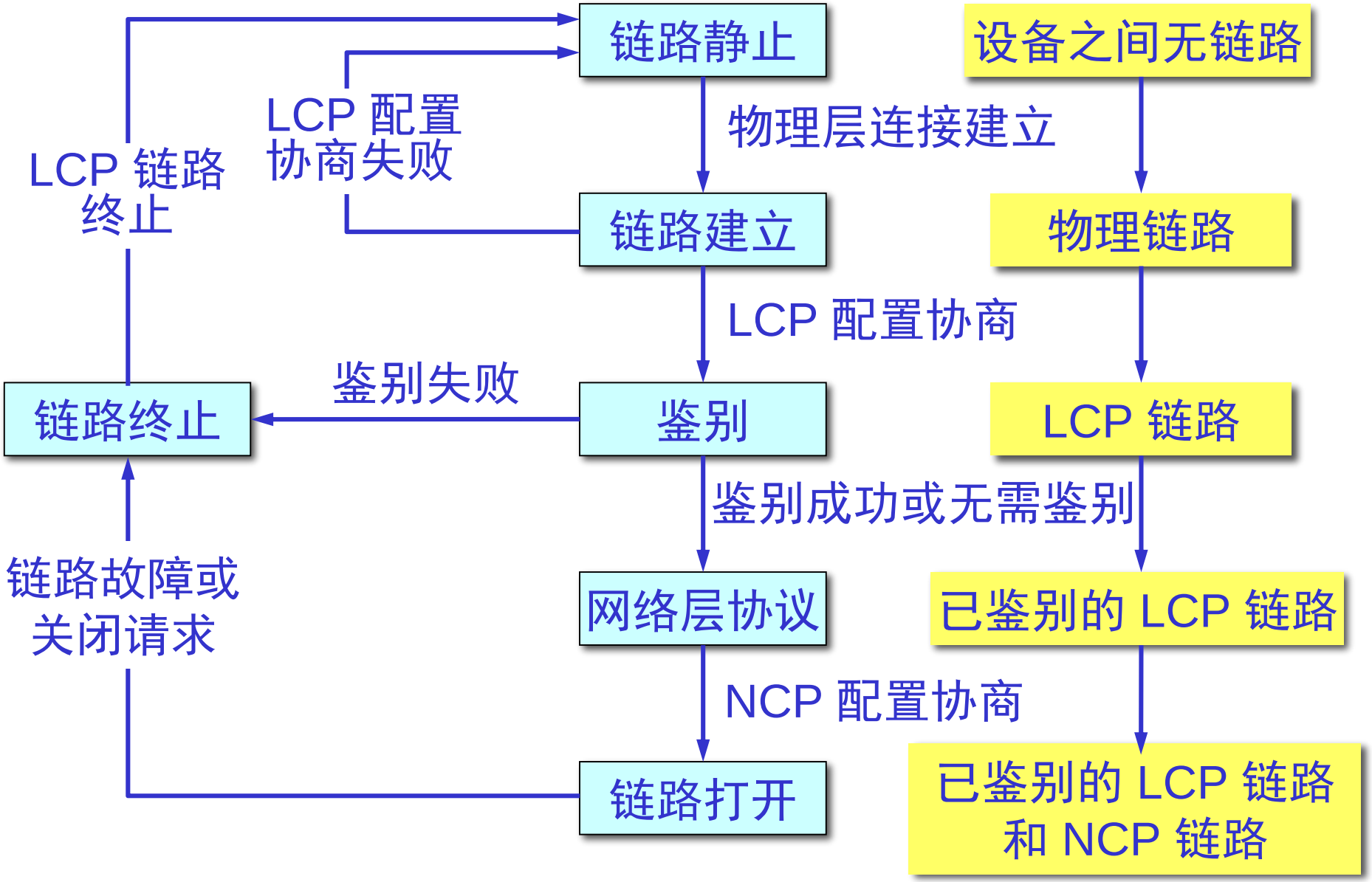
在接收端把 5 个连 1 之后的 0 比特删除

不提供使用序号和确认的可靠传输

- PPP 协议之所以不使用序号和确认机制是出于以下的考虑：
  - 在数据链路层出现差错的概率不大时，使用比较简单的 PPP 协议较为合理。
  - 在因特网环境下，PPP 的信息字段放入的数据是 IP 数据报。数据链路层的可靠传输并不能够保证网络层的传输也是可靠的。
  - 帧检验序列 FCS 字段可保证无差错接受。

3.2.3 PPP 协议的工作状态

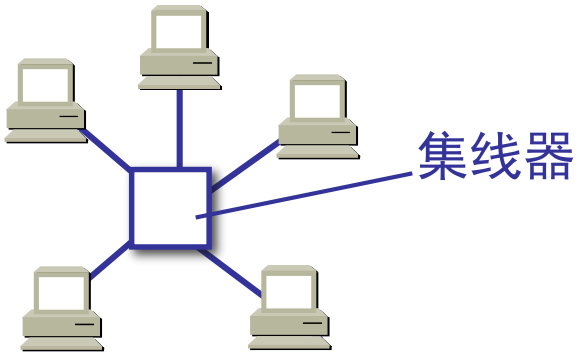
- 当用户拨号接入 ISP 时，路由器的调制解调器对拨号做出确认，并建立一条物理连接。
- PC 机向路由器发送一系列的 LCP 分组（封装成多个 PPP 帧）。
- 这些分组及其响应选择一些 PPP 参数，和进行网络层配置，NCP 给新接入的 PC机分配一个临时的 IP 地址，使 PC 机成为因特网上的一个主机。
- 通信完毕时，NCP 释放网络层连接，收回原来分配出去的 IP 地址。接着，LCP 释放数据链路层连接。最后释放的是物理层的连接。


链路静止
设备之间无链路
LCP 配置
协商失败
物理层连接建立
LCP 链路
终止
链路建立
物理链路
LCP 配置协商
鉴别失败
链路终止
鉴别
LCP 链路
鉴别成功或无需鉴别
链路故障或
关闭请求
网络层协议
已鉴别的 LCP 链路
NCP 配置协商
链路打开
已鉴别的 LCP 链路
和 NCP 链路

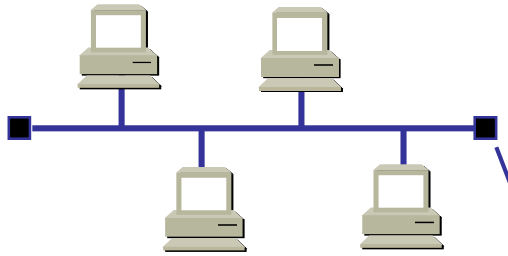
3.3 使用广播信道的数据链路层
3.3.1 局域网的数据链路层

- 局域网最主要的特点是：网络为一个单位所拥有，且地理范围和站点数目均有限。
- 局域网具有如下的一些主要优点：
  - 具有广播功能，从一个站点可很方便地访问全网。局域网上的主机可共享连接在局域网上的各种硬件和软件资源。
  - 便于系统的扩展和逐渐地演变，各设备的位置可灵活调整和改变。
  - 提高了系统的可靠性、可用性和残存性。

局域网的拓扑

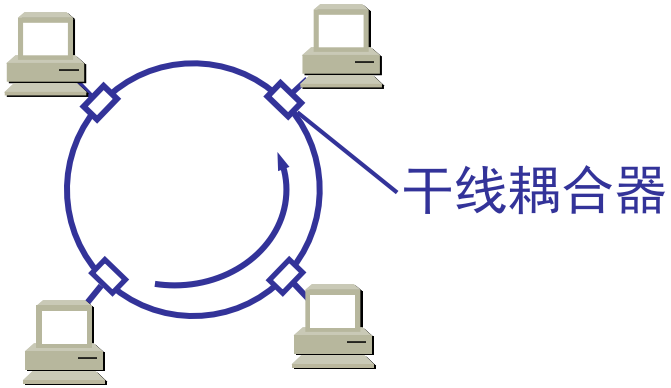


星形网



总线网

匹配电阻



环形网

媒体共享技术

- 静态划分信道
  - 频分复用
  - 时分复用
  - 波分复用
  - 码分复用
- 动态媒体接入控制（多点接入）
  - 随机接入
  - 受控接入，如多点线路探询(polling)，或轮询。

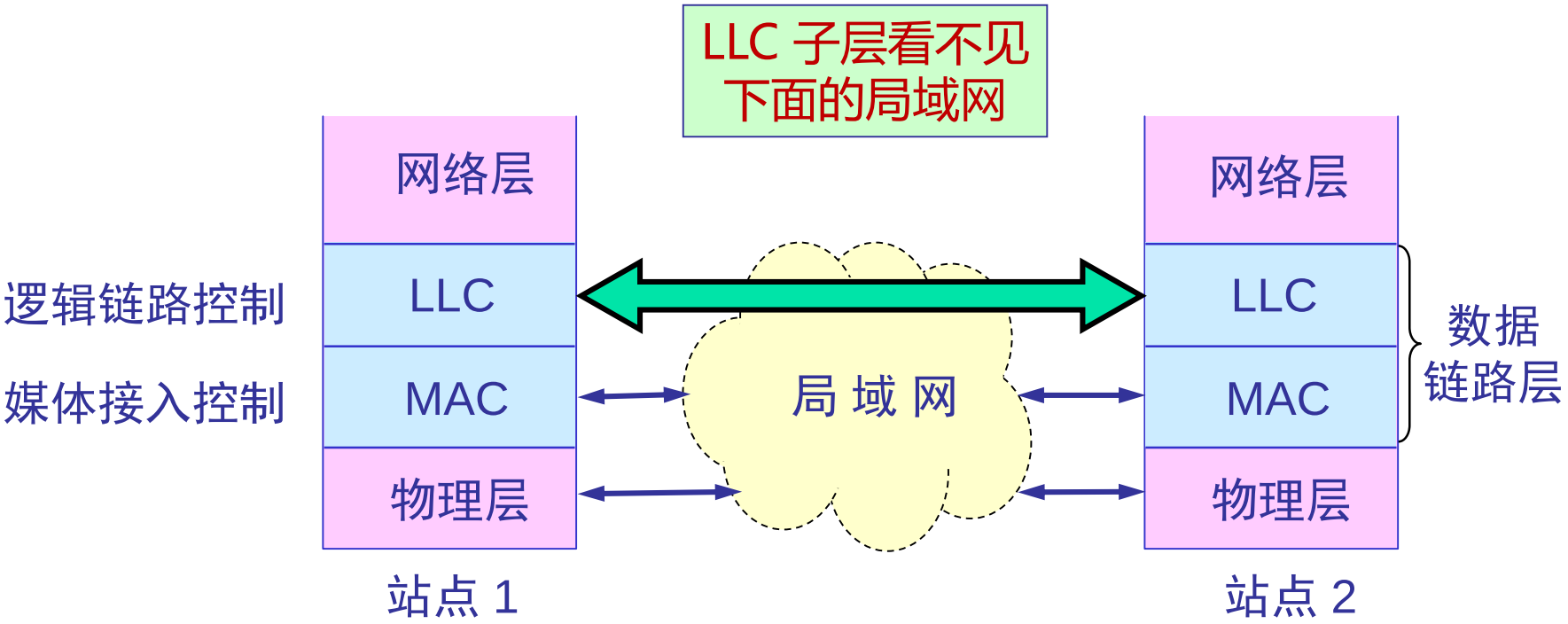
以太网的两个标准

- DIX Ethernet V2 是世界上第一个局域网产品（以太网）的规约。
- IEEE 的 802.3 标准。
- DIX Ethernet V2 标准与 IEEE 的 802.3 标准只有很小的差别，因此可以将 802.3 局域网简称为“以太网”。
- 严格说来，“以太网”应当是指符合 DIX Ethernet V2 标准的局域网

数据链路层的两个子层

- 为了使数据链路层能更好地适应多种局域网标准，802 委员会就将局域网的数据链路层拆成两个子层：
  - 逻辑链路控制 LLC (Logical Link Control)子层
  - 媒体接入控制 MAC (Medium Access Control)子层。
- 与接入到传输媒体有关的内容都放在 MAC子层，而 LLC 子层则与传输媒体无关，不管采用何种协议的局域网对 LLC 子层来说都是透明的

局域网对 LLC 子层是透明的



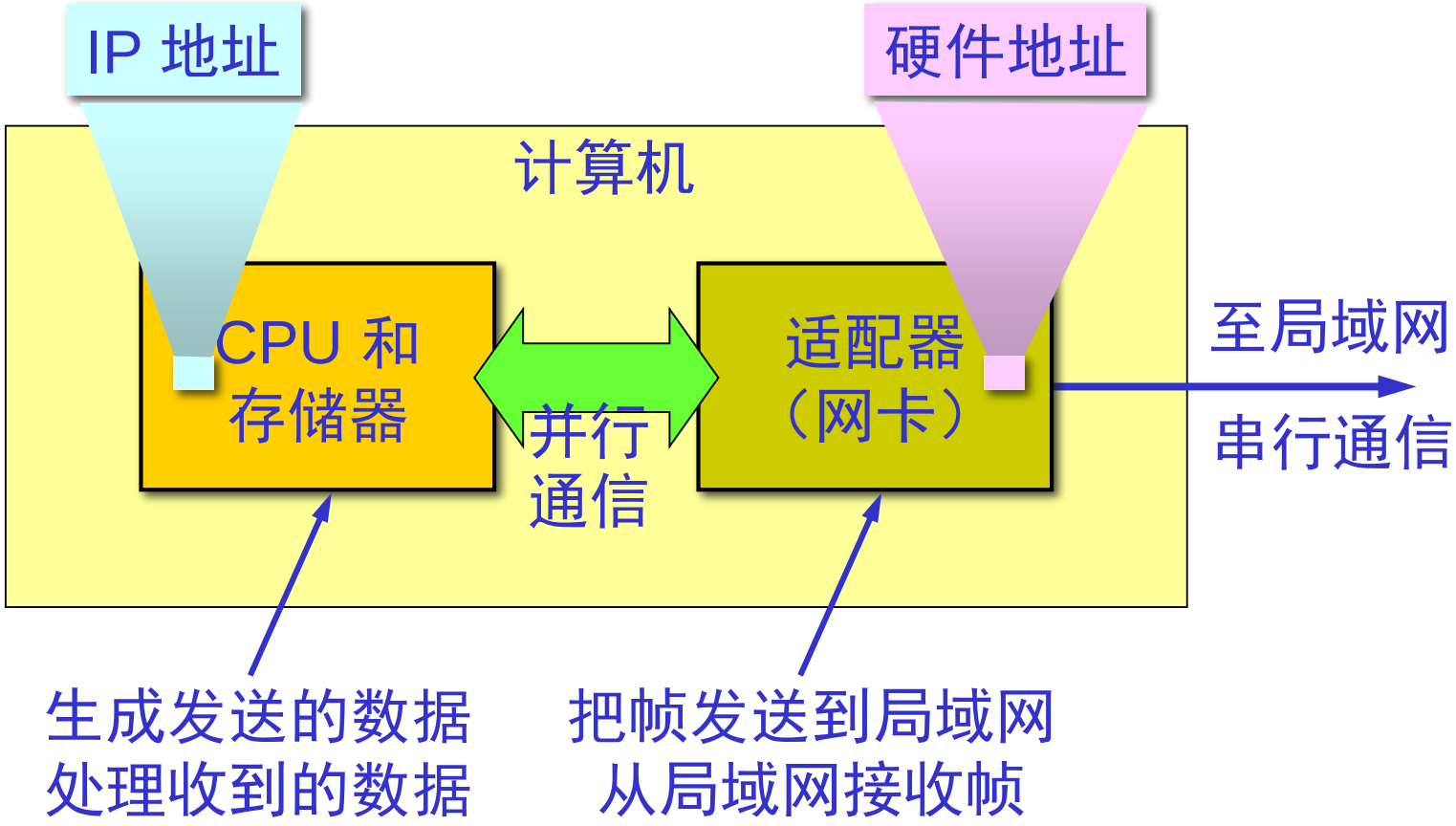
以后一般不考虑 LLC 子层

- 由于 TCP/IP 体系经常使用的局域网是 DIX Ethernet V2 而不是 802.3 标准中的几种局域网，因此现在 802 委员会制定的逻辑链路控制子层 LLC（即 802.2 标准）的作用已经不大了。
- 很多厂商生产的适配器上就仅装有 MAC 协议而没有 LLC 协议。

2．适配器的作用

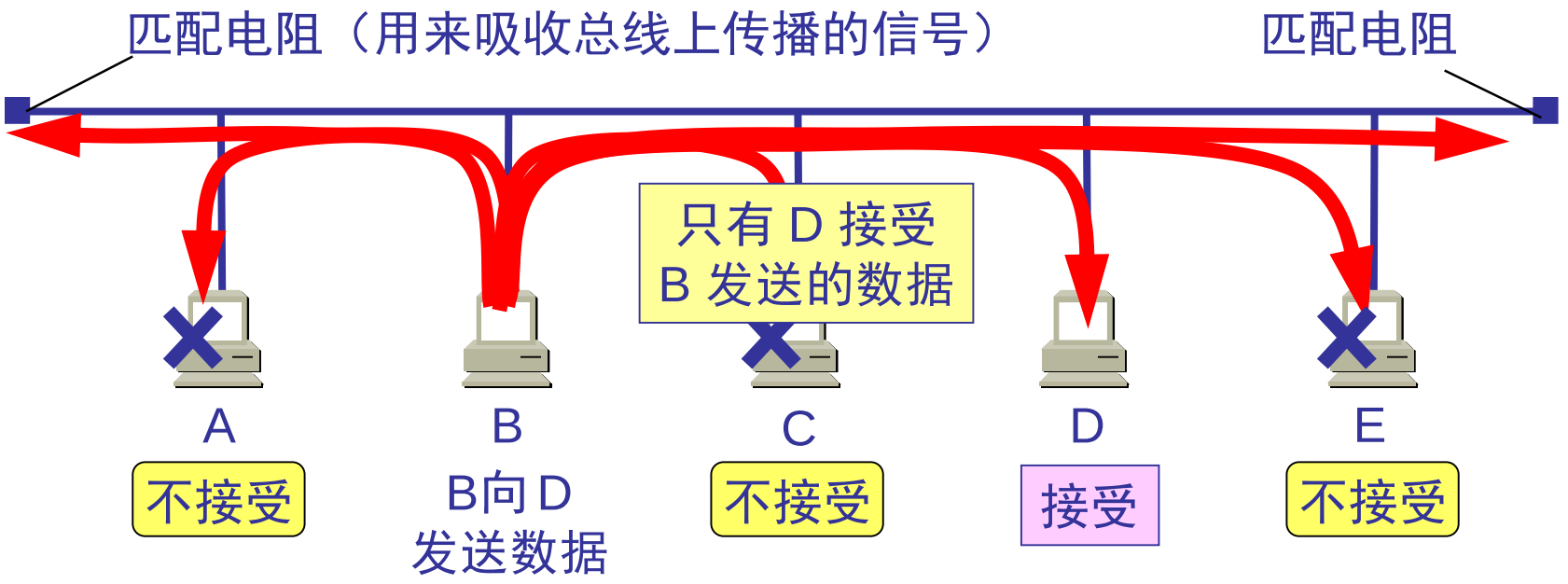
- 网络接口板又称为通信适配器(adapter)或网络接口卡 NIC (Network Interface Card)，或“网卡”。
- 适配器的重要功能：
  - 进行串行/并行转换。
  - 对数据进行缓存。
  - 在计算机的操作系统安装设备驱动程序。
  - 实现以太网协议。

计算机通过适配器和局域网进行通信



3.3.2 CSMA/CD 协议

- 最初的以太网是将许多计算机都连接到一根总线上。当初认为这样的连接方法既简单又可靠，因为总线上没有有源器件。



以太网的广播方式发送

- 总线上的每一个工作的计算机都能检测到 B 发送的数据信号。
- 由于只有计算机 D 的地址与数据帧首部写入的地址一致，因此只有 D 才接收这个数据帧。
- 其他所有的计算机（A, C 和 E）都检测到不是发送给它们的数据帧，因此就丢弃这个数据帧而不能够收下来。
- 具有广播特性的总线上实现了一对一的通信。

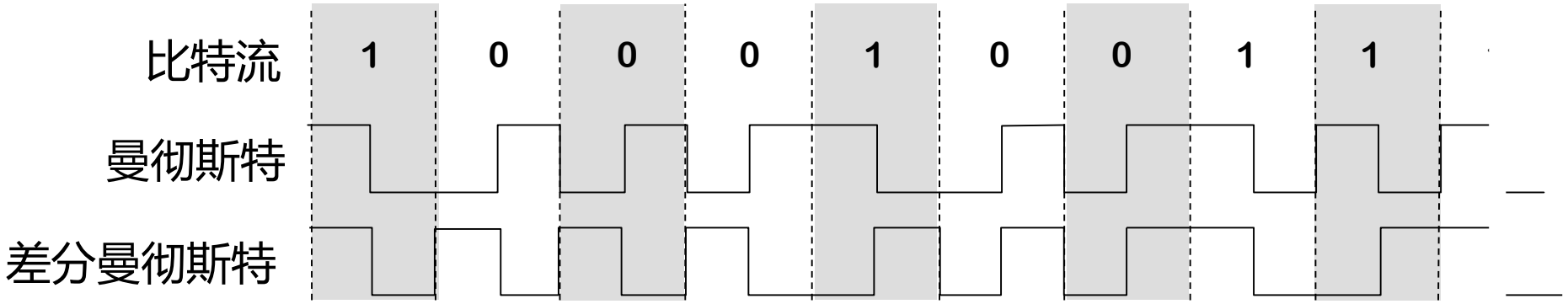
为了通信的简便
以太网采取了两种重要的措施

- 采用较为灵活的无连接的工作方式，即不必先建立连接就可以直接发送数据。
- 以太网对发送的数据帧不进行编号，也不要求对方发回确认。
  - 这样做的理由是局域网信道的质量很好，因信道质量产生差错的概率是很小的。

以太网提供的服务

- 以太网提供的服务是不可靠的交付，即尽最大努力的交付。
- 当目的站收到有差错的数据帧时就丢弃此帧，其他什么也不做。差错的纠正由高层来决定。
- 如果高层发现丢失了一些数据而进行重传，但以太网并不知道这是一个重传的帧，而是当作一个新的数据帧来发送。

以太网发送的数据都使用曼彻斯特(Manchester)编码



- 曼彻斯特编码中，每一位的中间有一跳变，位中间的跳变既作时钟信号，又作数据信号；从高到低跳变表示" 1"，从低到高跳变表示" 0"。
- 差分曼彻斯特编码，每位中间的跳变仅提供时钟定时，而用每位开始时有无跳变表示"0"或"1"，有跳变为"0"，无跳变为"1"。
- 两种曼彻斯特编码是将时钟和数据包含在数据流中，在传输代码信息的同时，也将时钟同步信号一起传输到对方，每位编码中有一跳变，不存在直流分量，因此具有自同步能力和良好的抗干扰性能。但每一个码元都被调成两个电平，所以数据传输速率只有调制速率的1/2。

载波监听多点接入/碰撞检测 CSMA/CD

- CSMA/CD 表示 Carrier Sense Multiple Access with Collision Detection。
- “多点接入”表示许多计算机以多点接入的方式连接在一根总线上。
- “载波监听”是指每一个站在发送数据之前先要检测一下总线上是否有其他计算机在发送数据，如果有，则暂时不要发送数据，以免发生碰撞。
- 总线上并没有什么“载波”。因此，“载波监听”就是用电子技术检测总线上有没有其他计算机发送的数据信号。

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”。

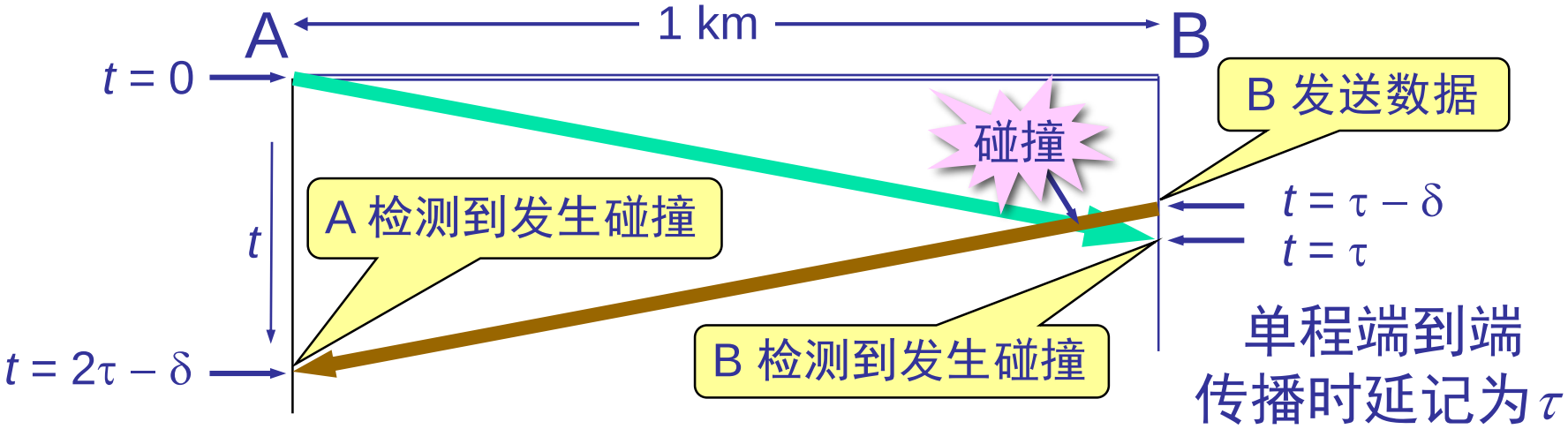
检测到碰撞后

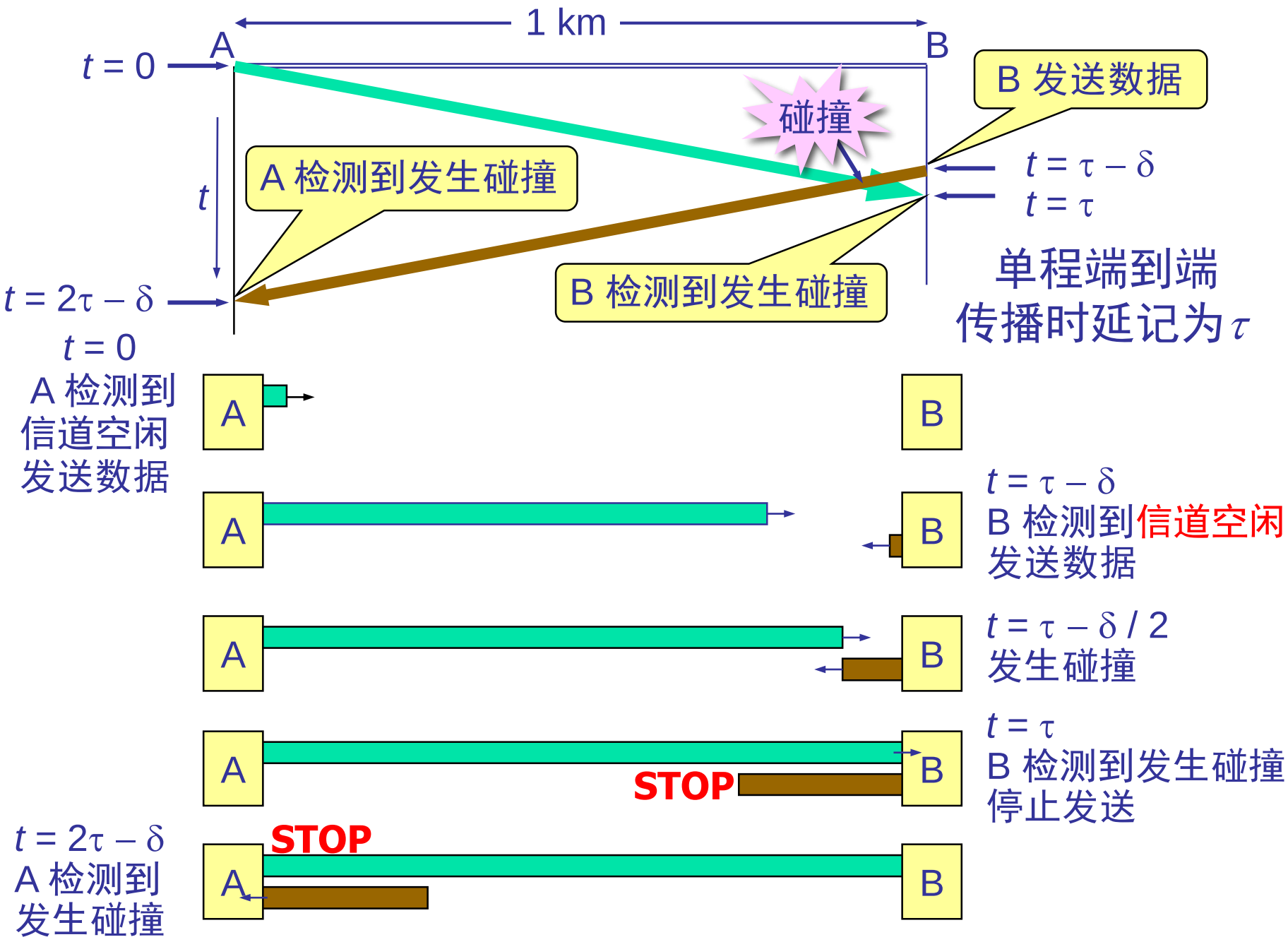
- 在发生碰撞时，总线上传输的信号产生了严重的失真，无法从中恢复出有用的信息来。
- 每一个正在发送数据的站，一旦发现总线上出现了碰撞，就要立即停止发送，免得继续浪费网络资源，然后等待一段随机时间后再次发送。

电磁波在总线上的有限传播速率的影响

- 当某个站监听到总线是空闲时，也可能总线并非真正是空闲的。
- A 向 B 发出的信息，要经过一定的时间后才能传送到 B。
- B 若在 A 发送的信息到达 B 之前发送自己的帧(因为这时 B 的载波监听检测不到 A 所发送的信息)，则必然要在某个时间和 A 发送的帧发生碰撞。
- 碰撞的结果是两个帧都变得无用。

传播时延对载波监听的影响




1 km
A
B
$t = 0$
t
$t = 2\tau - \delta$
碰撞
B 发送数据
A 检测到发生碰撞
B 检测到发生碰撞
$t = \tau - \delta$
$t = \tau$
单程端到端
传播时延记为τ
$t = 0$
A 检测到
信道空闲
发送数据
$t = \tau - \delta$
B 检测到信道空闲
发送数据
$t = \tau - \delta / 2$
发生碰撞
$t = \tau$
B 检测到发生碰撞
停止发送
STOP
STOP
$t = 2\tau - \delta$
A 检测到
发生碰撞

重要特性

- 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。
- 这种发送的不确定性使整个以太网的平均通信量远小于以太网的最高数据率。

争用期

- 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。

二进制指数类型退避算法 (truncated binary exponential type)

- 发生碰撞的站在停止发送数据后，要推迟（退避）一个随机时间才能再发送数据。
  - 基本退避时间取为争用期 2τ。
  - 从整数集合[0,1,…, $(2^k - 1)$]中随机地取出一个数，记为 r。重传所需的时延就是 r 倍的基本退避时间。
  - 参数 k 按下面的公式计算：

    k = Min[重传次数, 10]
  - 当 $k \le 10$ 时，参数 k 等于重传次数。
  - 当重传达 16 次仍不能成功时即丢弃该帧，并向高层报告。

争用期的长度

- 以太网取 51.2 μs 为争用期的长度。
- 对于 10 Mb/s 以太网，在争用期内可发送512 bit，即 64 字节。
- 以太网在发送数据时，若前 64 字节没有发生冲突，则后续的数据就不会发生冲突。

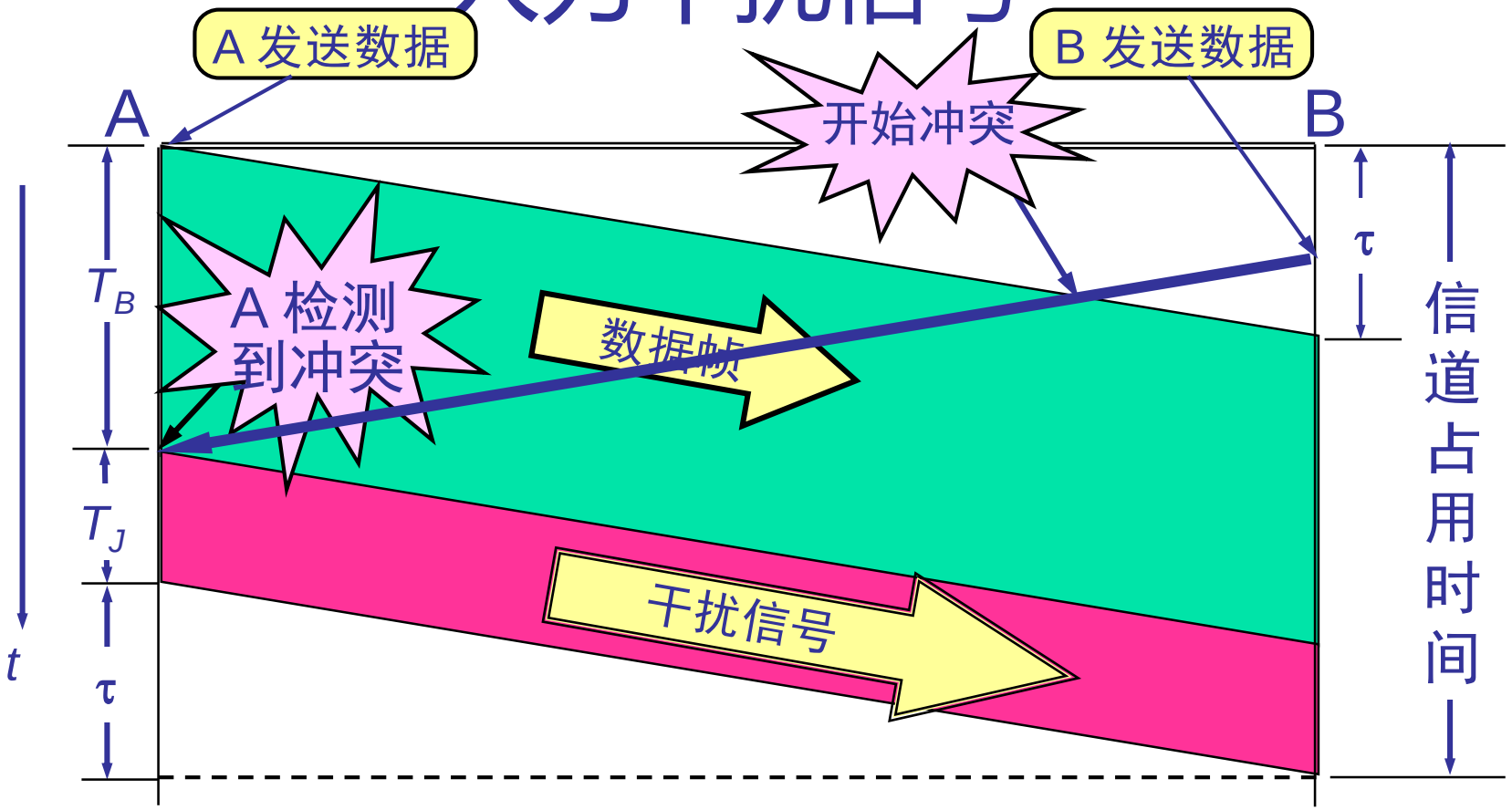
最短有效帧长

- 如果发生冲突，就一定是在发送的前 64 字节之内。
- 由于一检测到冲突就立即中止发送，这时已经发送出去的数据一定小于 64 字节。
- 以太网规定了最短有效帧长为 64 字节，凡长度小于 64 字节的帧都是由于冲突而异常中止的无效帧。

强化碰撞

- 当发送数据的站一旦发现发生了碰撞时：
  - 立即停止发送数据；
  - 再继续发送若干比特的人为干扰信号(jamming signal)，以便让所有用户都知道现在已经发生了碰撞。

人为干扰信号

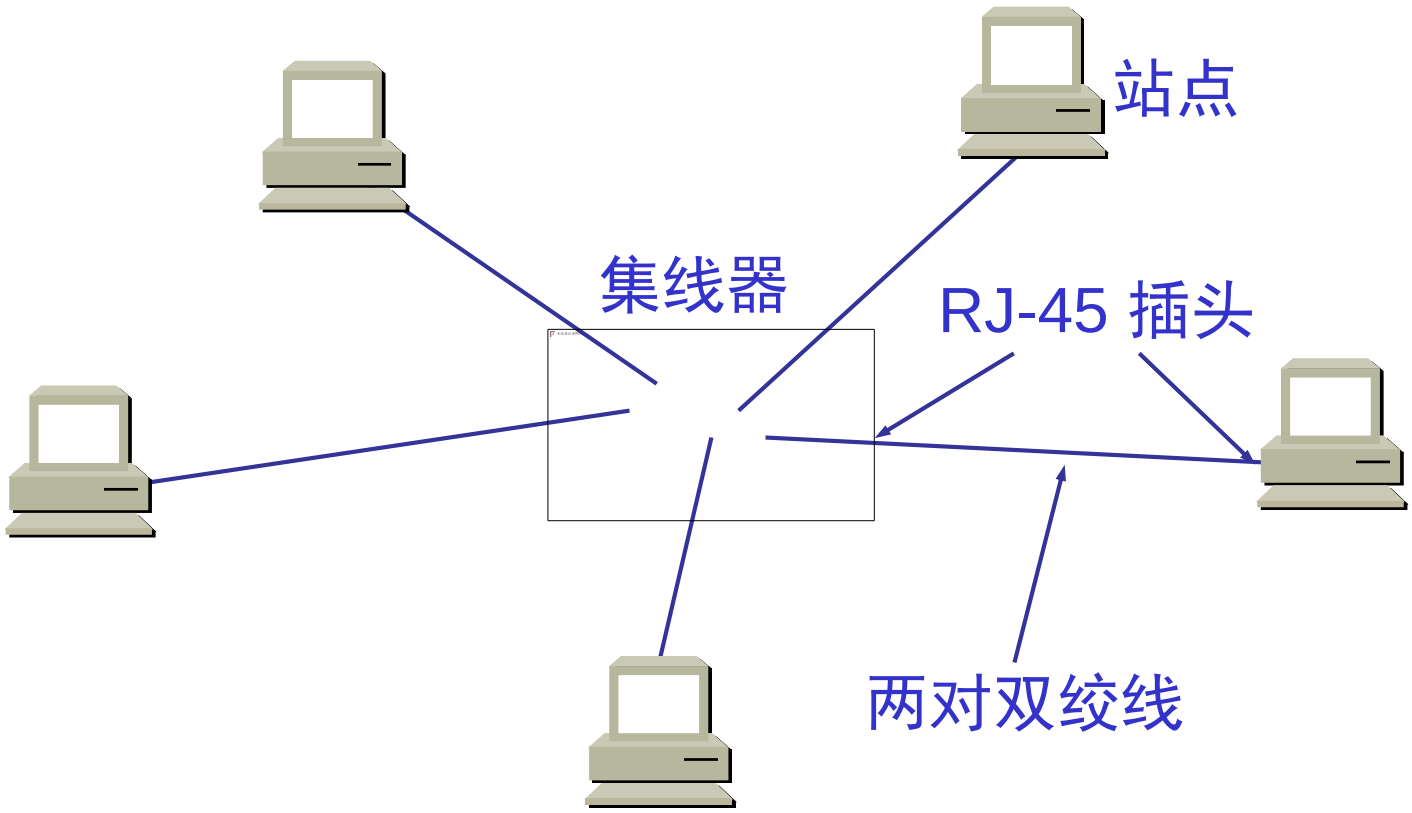


B 也能够检测到冲突，并立即停止发送数据帧，接着就发送干扰信号。这里为了简单起见，只画出 A 发送干扰信号的情况。

3.4 使用广播信道的以太网
3.4.1 使用集线器的星形拓扑

- 传统以太网最初是使用粗同轴电缆，后来演进到使用比较便宜的细同轴电缆，最后发展为使用更便宜和更灵活的双绞线。
- 这种以太网采用星形拓扑，在星形的中心则增加了一种可靠性非常高的设备，叫做集线器(hub)

使用集线器的双绞线以太网



星形网 10BASE-T

- 不用电缆而使用无屏蔽双绞线。每个站需要用两对双绞线，分别用于发送和接收。
- 集线器使用了大规模集成电路芯片，因此这样的硬件设备的可靠性已大大提高了。

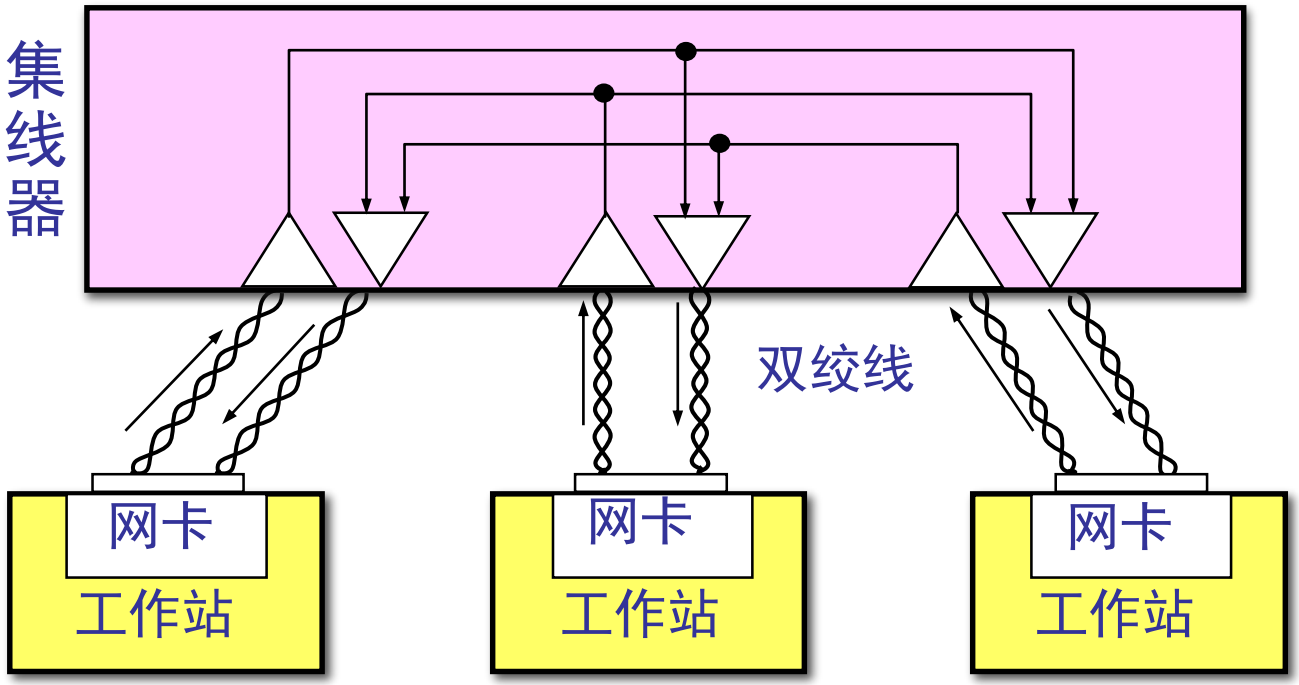
以太网在局域网中的统治地位

- 10BASE-T 的通信距离稍短，每个站到集线器的距离不超过 100 m。
- 这种 10 Mb/s 速率的无屏蔽双绞线星形网的出现，既降低了成本，又提高了可靠性。
- 10BASE-T 双绞线以太网的出现，是局域网发展史上的一个非常重要的里程碑，它为以太网在局域网中的统治地位奠定了牢固的基础。

集线器的一些特点

- 集线器是使用电子器件来模拟实际电缆线的工作，因此整个系统仍然像一个传统的以太网那样运行。
- 使用集线器的以太网在逻辑上仍是一个总线网，各工作站使用的还是 CSMA/CD 协议，并共享逻辑上的总线。
- 集线器很像一个多接口的转发器，工作在物理层。

具有三个接口的集线器

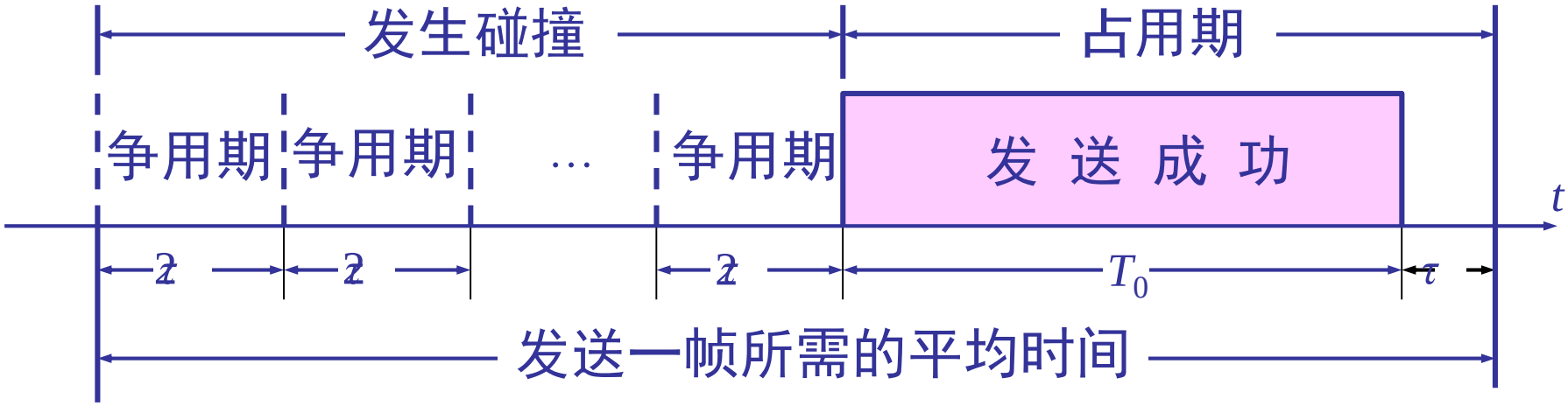


3.4.2 以太网的信道利用率

- 以太网的信道被占用的情况：
- 争用期长度为 2τ，即端到端传播时延的两倍。检测到碰撞后不发送干扰信号。
- 帧长为 L (bit)，数据发送速率为 C (b/s)，因而帧的发送时间为 $L/C = T_0$ (s)。

以太网的信道利用率

- 一个帧从开始发送，经可能发生的碰撞后，将再重传数次，到发送成功且信道转为空闲(即再经过时间 τ 使得信道上无信号在传播)时为止，是发送一帧所需的平均时间。



$a = \frac{\tau}{T_0}$

参数 a

- 要提高以太网的信道利用率，就必须减小 τ 与 T_0 之比。在以太网中定义了参数 a，它是以太网单程端到端时延 τ 与帧的发送时间 T_0 之比：

$$a = \frac{\tau}{T_0} \qquad (3\text{-}2)$$

  - $a \to 0$ 表示一发生碰撞就立即可以检测出来，并立即停止发送，因而信道利用率很高。
  - a 越大，表明争用期所占的比例增大，每发生一次碰撞就浪费许多信道资源，使得信道利用率明显降低。

对以太网参数的要求

- 当数据率一定时，以太网的连线的长度受到限制，否则 τ 的数值会太大。
- 以太网的帧长不能太短，否则 T_0 的值会太小，使 a 值太大。

信道利用率的最大值 S_{max}

- 在理想化的情况下，以太网上的各站发送数据都不会产生碰撞（这显然已经不是 CSMA/CD，而是需要使用一种特殊的调度方法），即总线一旦空闲就有某一个站立即发送数据。
- 发送一帧占用线路的时间是 $T_0 + \tau$，而帧本身的发送时间是 T_0。于是我们可计算出理想情况下的极限信道利用率 S_{max}为：

$$S_{max} = \frac{T_0}{T_0 + \tau} = \frac{1}{1 + a} \qquad (3\text{-}3)$$

3.4.3 以太网的 MAC 层

1. MAC 层的硬件地址

- 在局域网中，硬件地址又称为物理地址，或 MAC 地址。
- 802 标准所说的“地址”严格地讲应当是每一个站的“名字”或标识符。
- 但鉴于大家都早已习惯了将这种 48 位的“名字”称为“地址”，所以本书也采用这种习惯用法，尽管这种说法并不太严格。

48 位的 MAC 地址

- IEEE 的注册管理机构 RA 负责向厂家分配地址字段的前三个字节(即高位 24 位)。
- 地址字段中的后三个字节(即低位 24 位)由厂家自行指派，称为扩展标识符，必须保证生产出的适配器没有重复地址。
- 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。
- “MAC地址”实际上就是适配器地址或适配器标识符EUI-48。

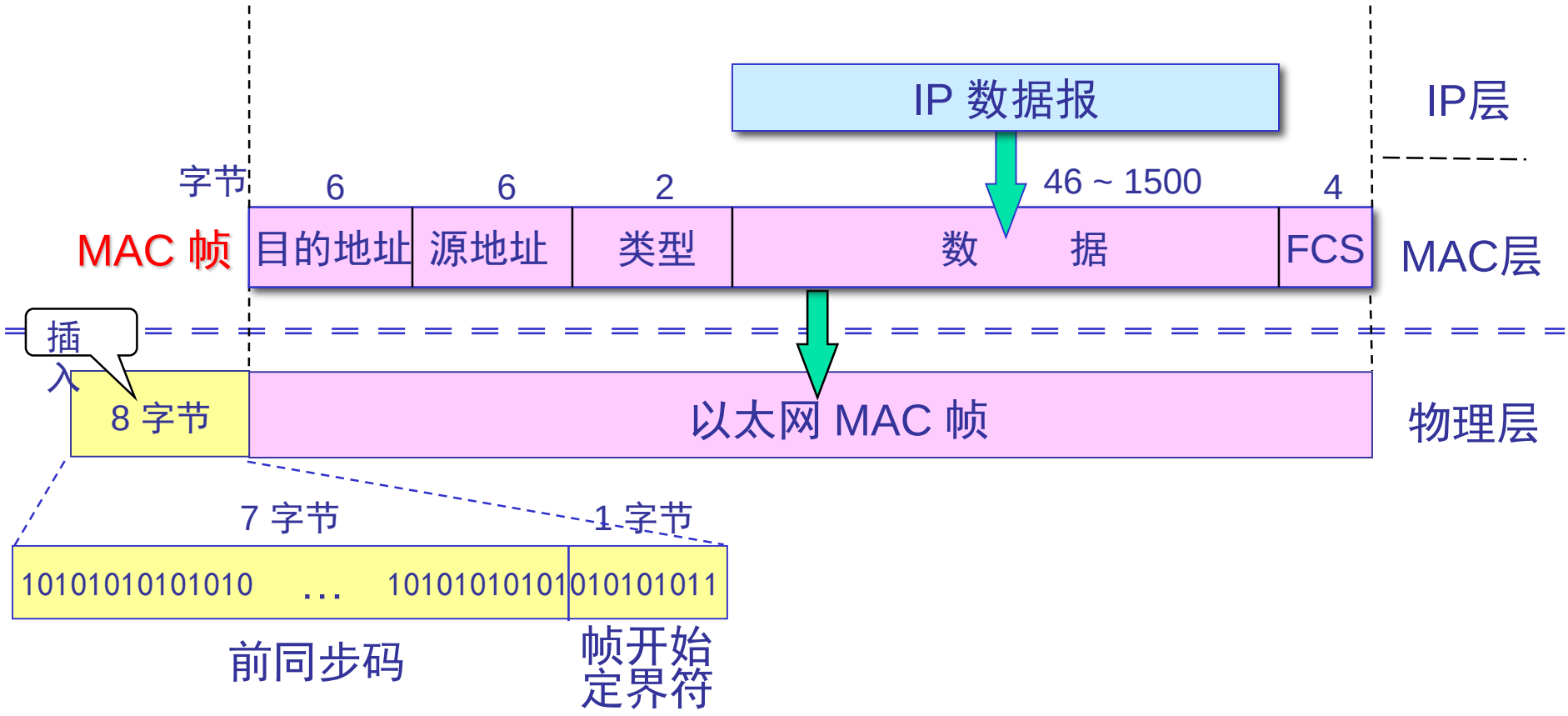
适配器检查 MAC 地址

- 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）

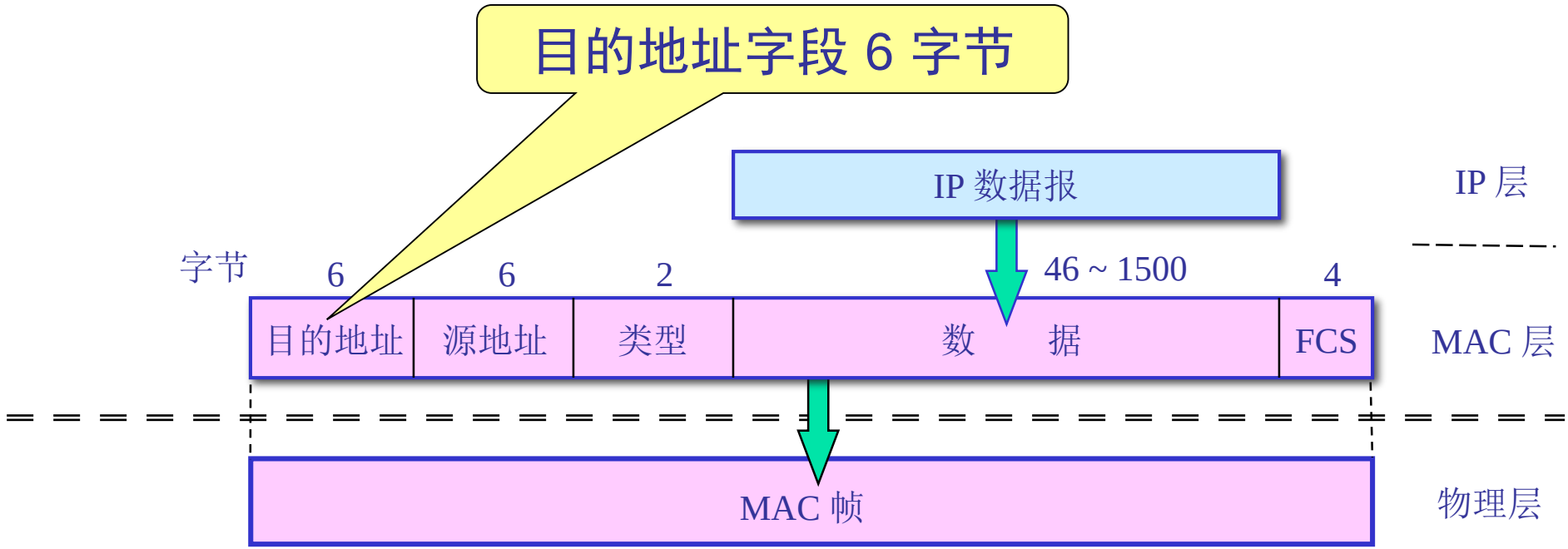
2. MAC 帧的格式

- 常用的以太网MAC帧格式有两种标准：
  - DIX Ethernet V2 标准
  - IEEE 的 802.3 标准
- 最常用的 MAC 帧是以太网 V2 的格式。

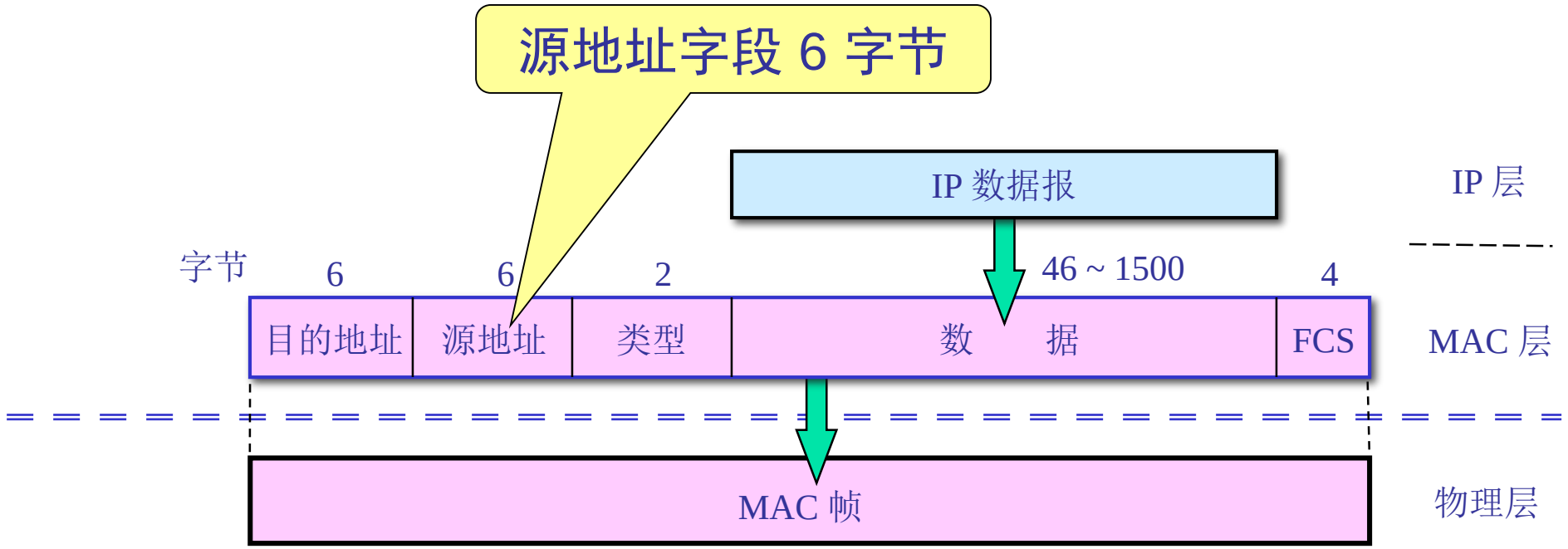
以太网的 MAC 帧格式


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
7 字节
1 字节
10101010101010 ... 10101010101010101011
前同步码
帧开始定界符

以太网 V2 的 MAC 帧格式

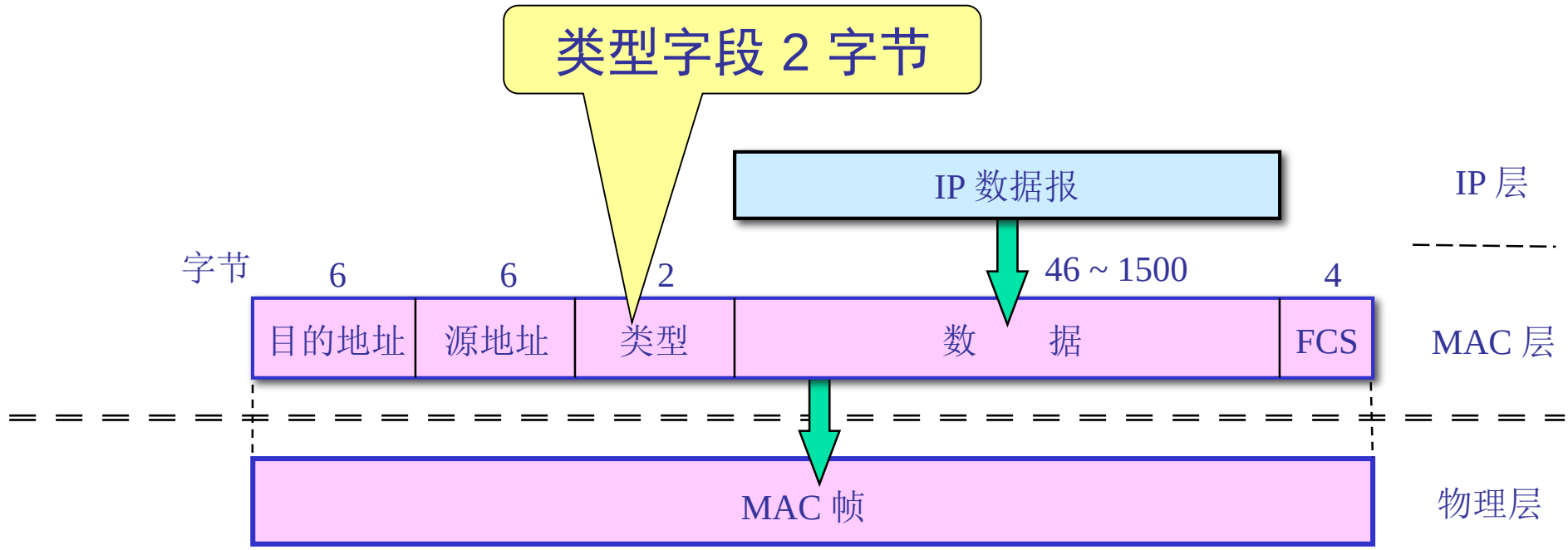


以太网 V2 的 MAC 帧格式



以太网 V2 的 MAC 帧格式

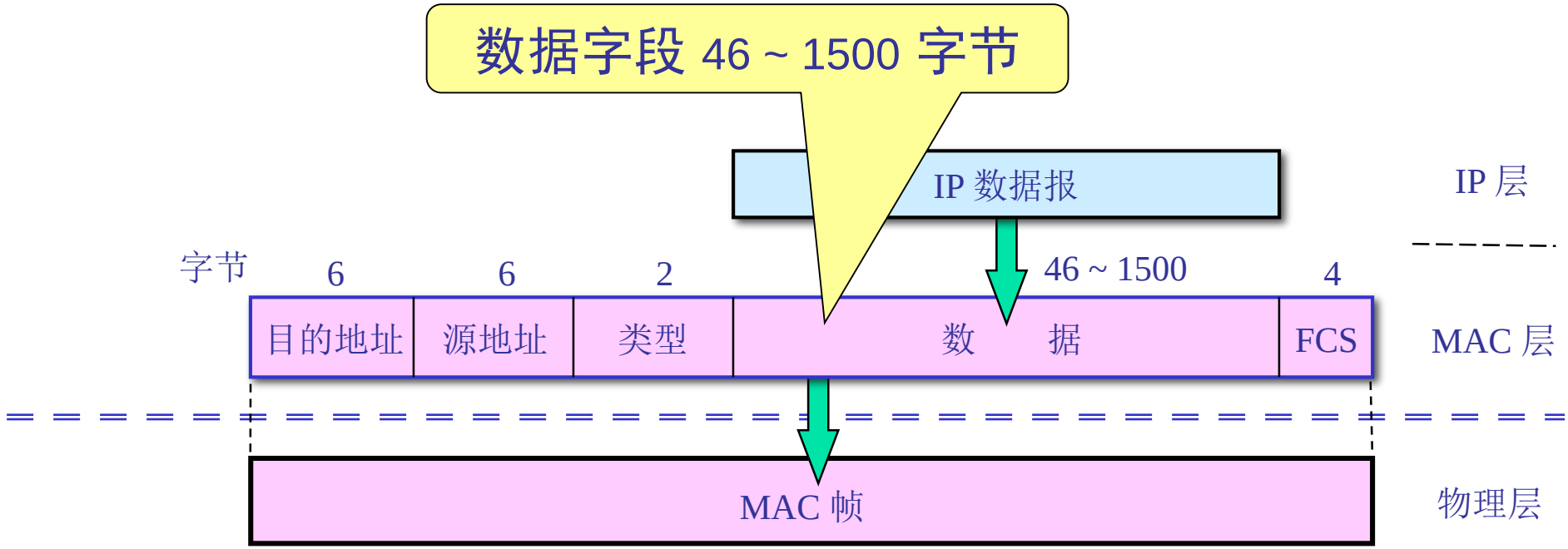
类型字段用来标志上一层使用的是什么协议，
以便把收到的 MAC 帧的数据上交给上一层的这个协议。



以太网 V2 的 MAC 帧格式

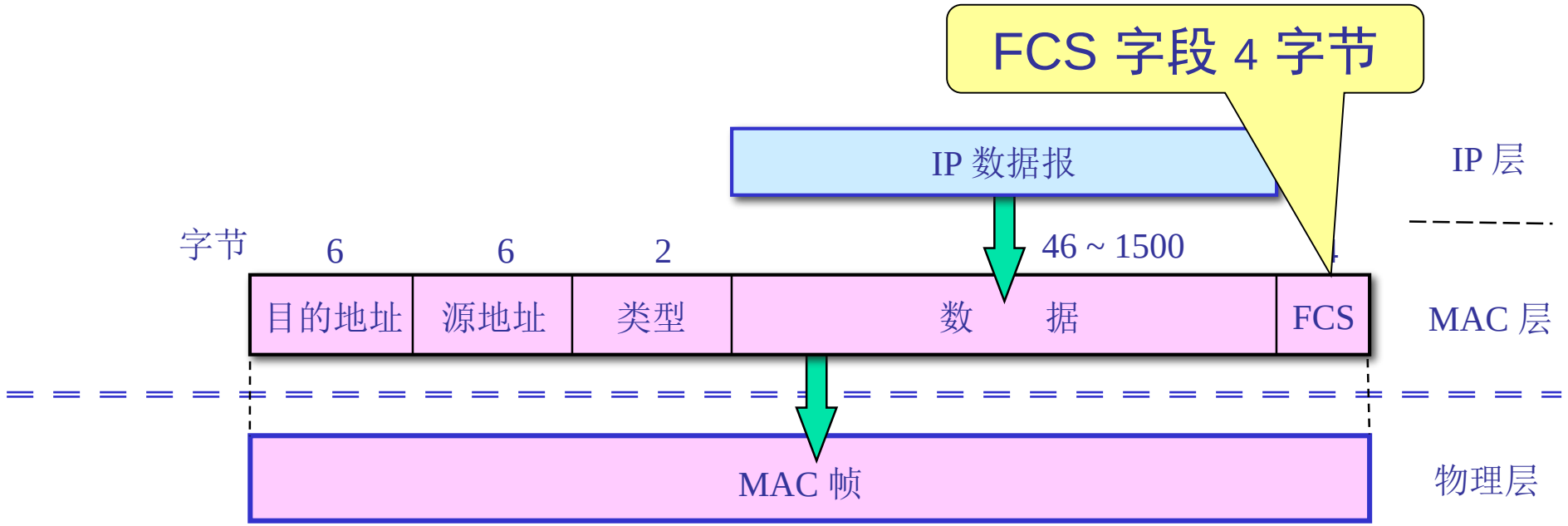
数据字段的正式名称是 MAC 客户数据字段

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



以太网 V2 的 MAC 帧格式

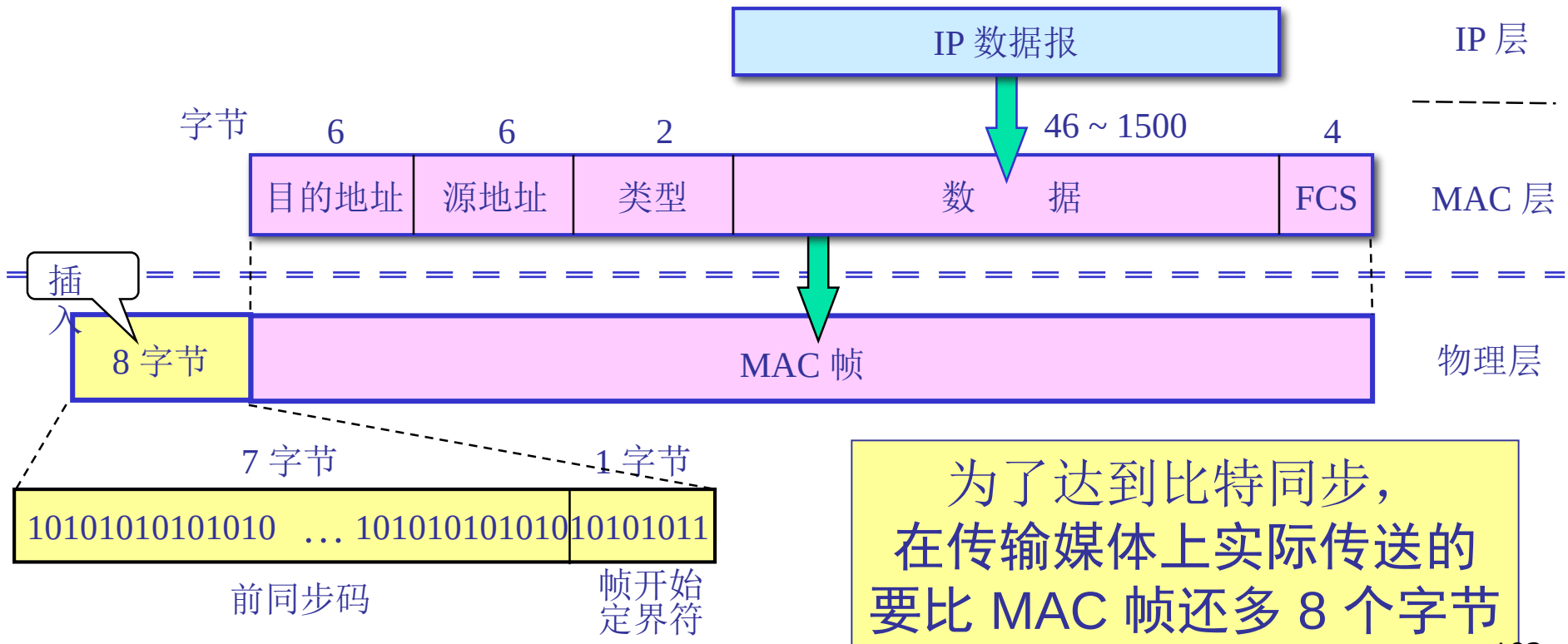
当传输媒体的误码率为 1×10^{-8} 时，
MAC 子层可使未检测到的差错小于 1×10^{-14}。



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。

以太网 V2 的 MAC 帧格式

在帧的前面插入的 8 字节中的第一个字段共 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

无效的 MAC 帧

- 数据字段的长度与长度字段的值不一致;
- 帧的长度不是整数个字节;
- 用收到的帧检验序列 FCS 查出有差错;
- 数据字段的长度不在 46 ~ 1500 字节之间。
- 有效的 MAC 帧长度为 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。

帧间最小间隔

- 帧间最小间隔为 9.6 μs，相当于 96 bit 的发送时间。
- 一个站在检测到总线开始空闲后，还要等待 9.6 μs 才能再次发送数据。
- 这样做是为了使刚刚收到数据帧的站的接收缓存来得及清理，做好接收下一帧的准备。

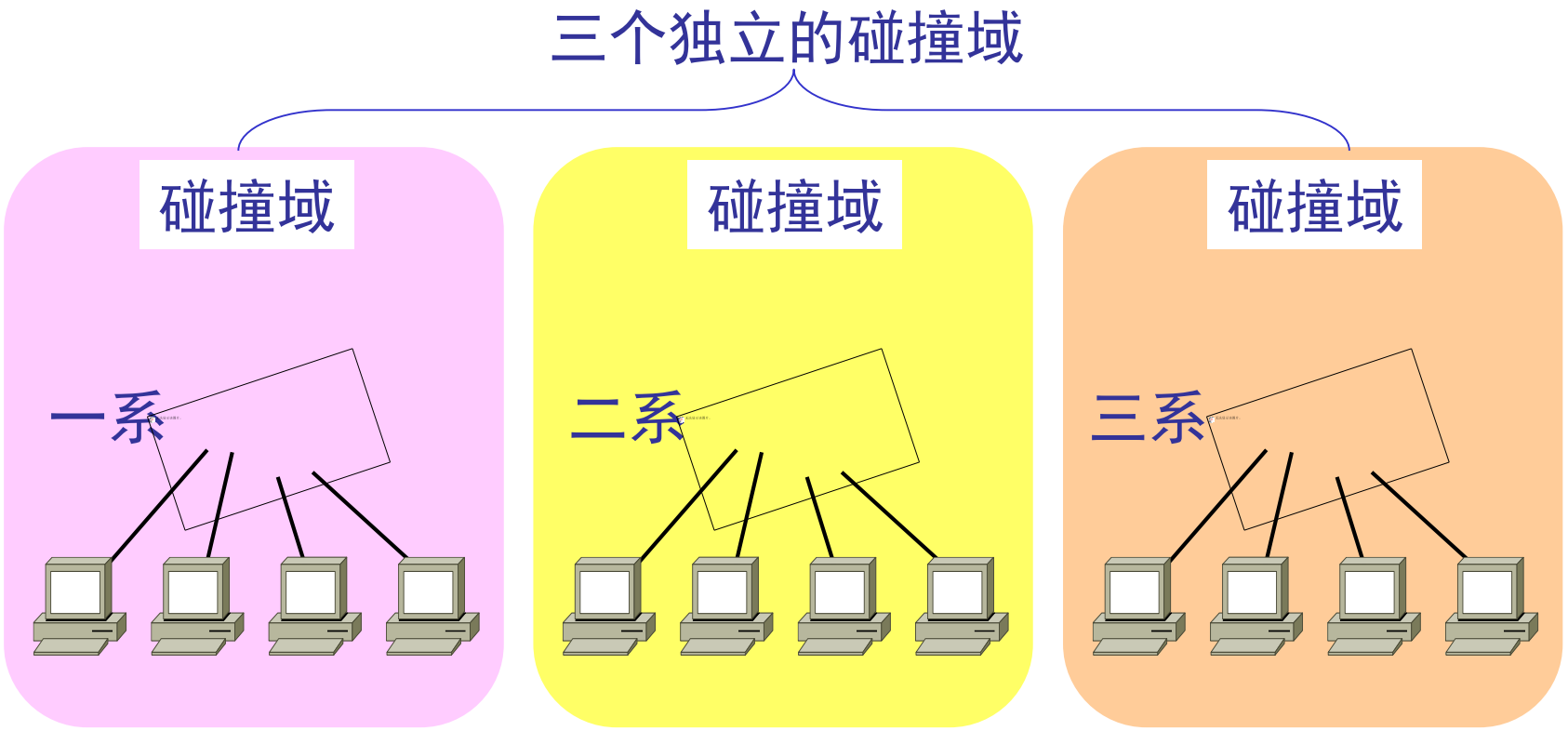
3.5 扩展的局域网

3.5.1 在物理层扩展局域网

- 主机使用光纤和一对光纤调制解调器连接到集线器

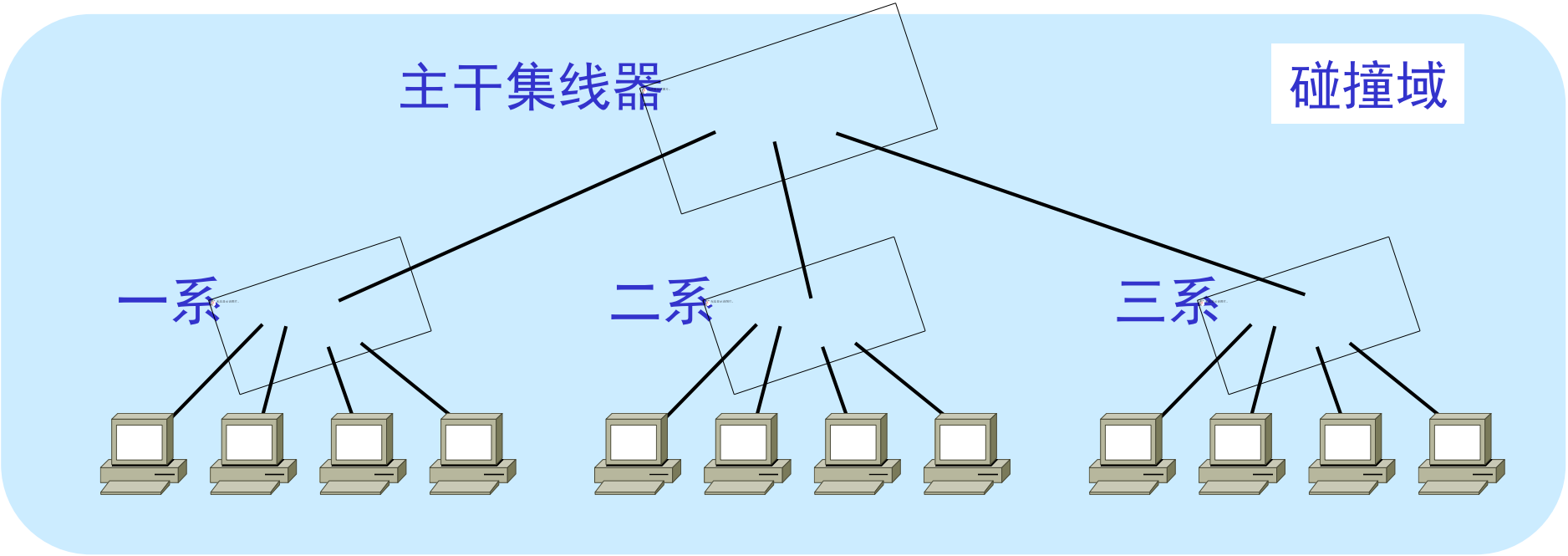
用多个集线器可连成更大的局域网

- 某大学有三个系，各自有一个局域网



用集线器组成更大的局域网
都在一个碰撞域中

一个更大的碰撞域



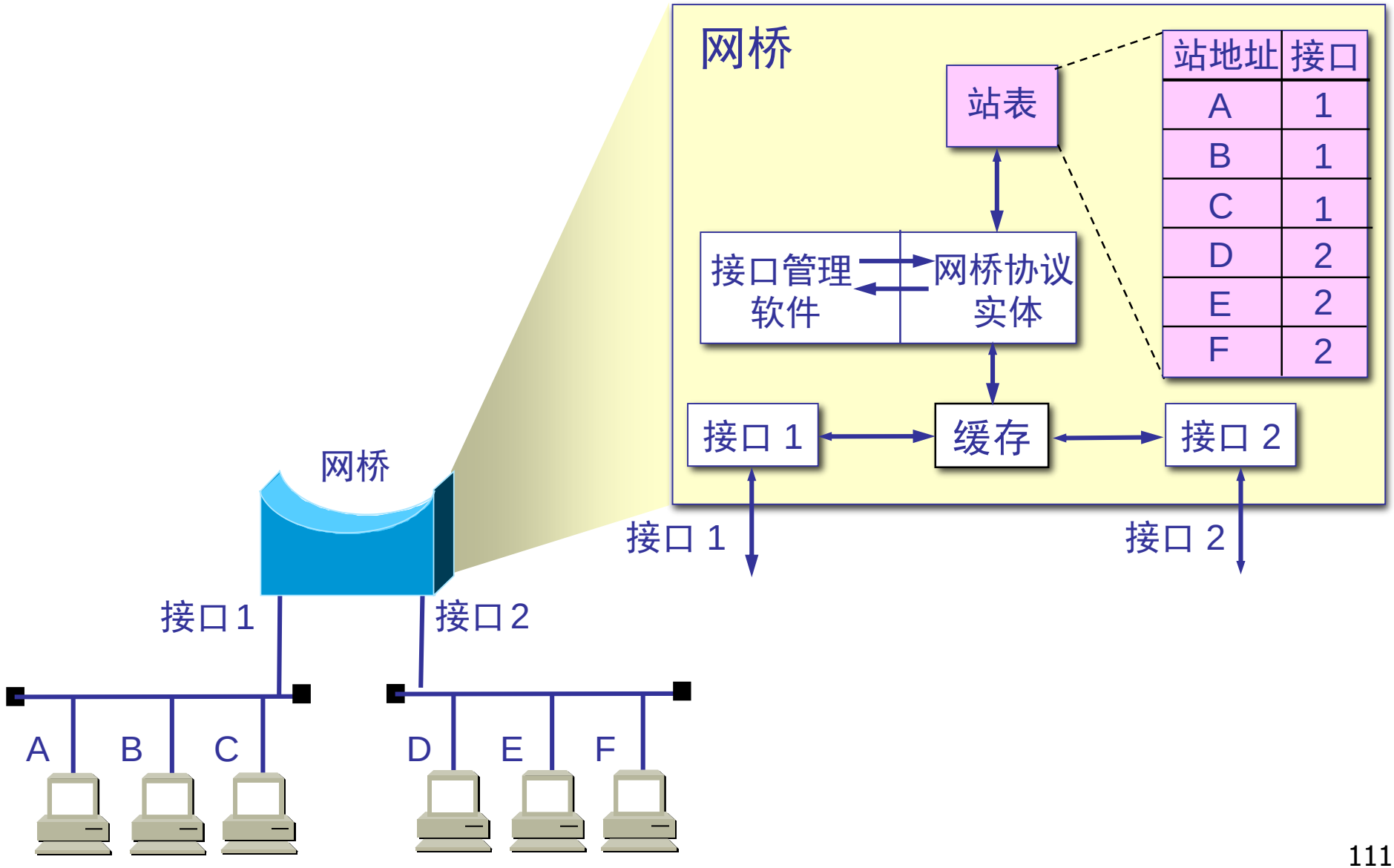
用集线器扩展局域网

- 优点
  - 使原来属于不同碰撞域的局域网上的计算机能够进行跨碰撞域的通信。
  - 扩大了局域网覆盖的地理范围。
- 缺点
  - 碰撞域增大了，但总的吞吐量并未提高。
  - 如果不同的碰撞域使用不同的数据率，那么就不能用集线器将它们互连起来。

3.5.2 在数据链路层扩展局域网

- 在数据链路层扩展局域网是使用网桥。
- 网桥工作在数据链路层，它根据 MAC 帧的目的地址对收到的帧进行转发。
- 网桥具有过滤帧的功能。当网桥收到一个帧时，并不是向所有的接口转发此帧，而是先检查此帧的目的 MAC 地址，然后再确定将该帧转发到哪一个接口

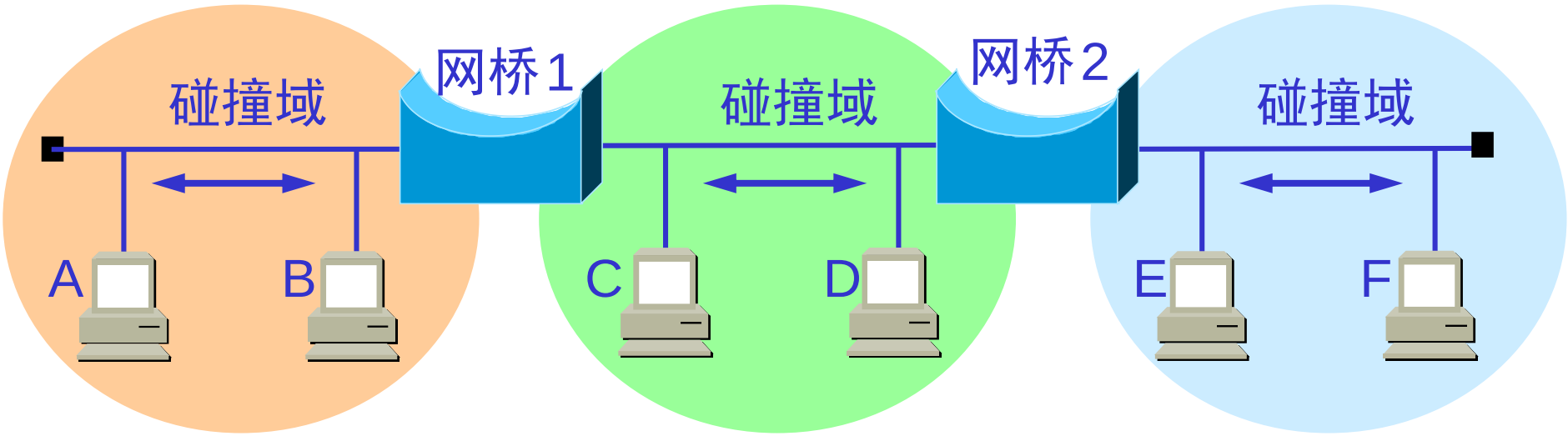
1. 网桥的内部结构


网桥
站表
站地址
接口
A
1
B
1
C
1
D
2
E
2
F
2
接口管理
软件
网桥协议
实体
接口 1
缓存
接口 2
接口 1
接口 2
网桥
接口1
接口2
A
B
C
D
E
F

使用网桥带来的好处

- 过滤通信量。
- 扩大了物理范围。
- 提高了可靠性。
- 可互连不同物理层、不同 MAC 子层和不同速率（如10 Mb/s 和 100 Mb/s 以太网）的局域网。

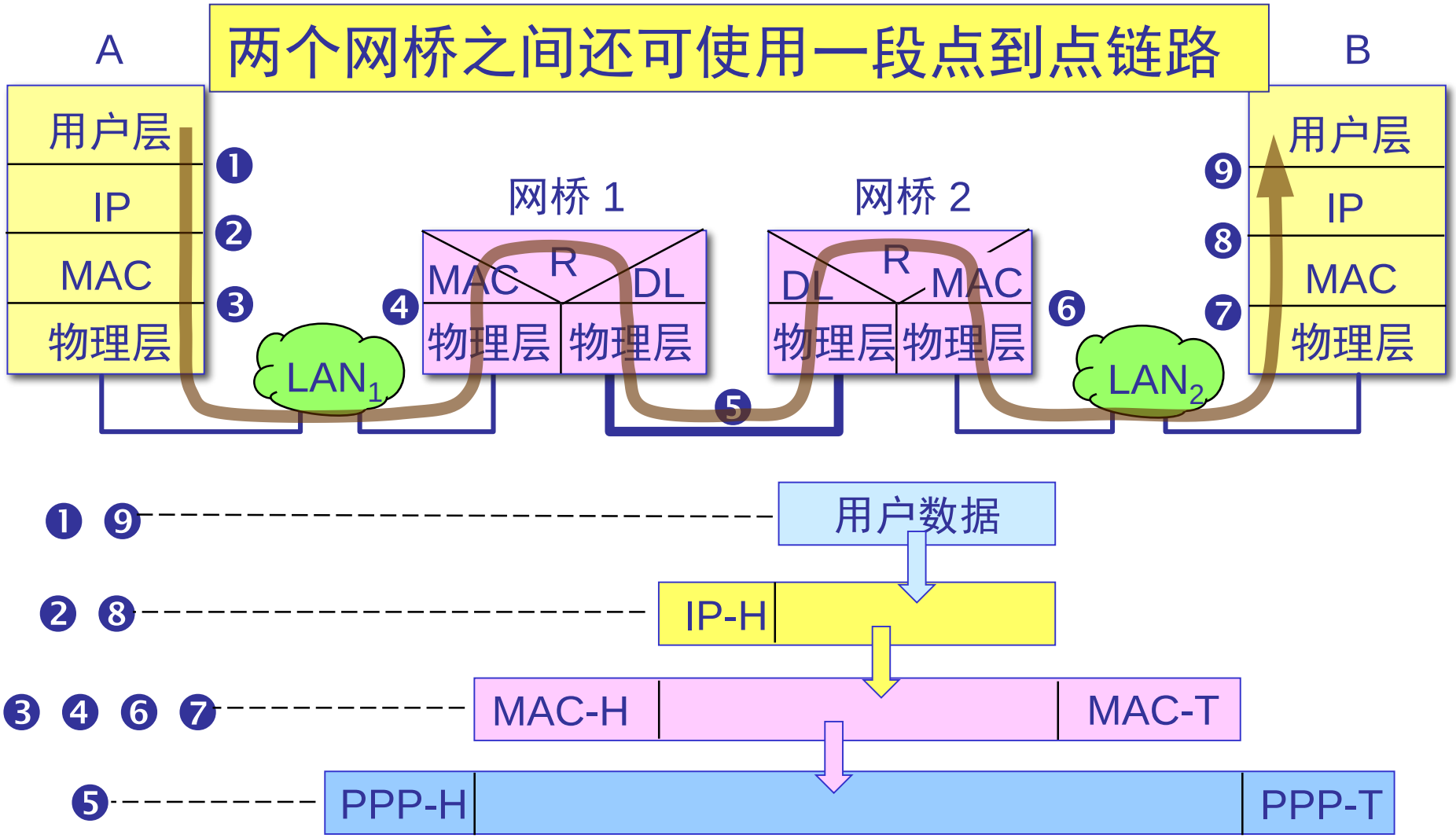
网桥使各网段成为隔离开的碰撞域



使用网桥带来的缺点

- 存储转发增加了时延。
- 在MAC 子层并没有流量控制功能。
- 具有不同 MAC 子层的网段桥接在一起时时延更大。
- 网桥只适合于用户数不太多(不超过几百个)和通信量不太大的局域网，否则有时还会因传播过多的广播信息而产生网络拥塞。这就是所谓的广播风暴。

两个网桥之间还可使用一段点到点链路



网桥不改变它转发的帧的源地址

网桥和集线器（或转发器）不同

- 集线器在转发帧时，不对传输媒体进行检测。
- 网桥在转发帧之前必须执行 CSMA/CD 算法。
  - 若在发送过程中出现碰撞，就必须停止发送和进行退避。

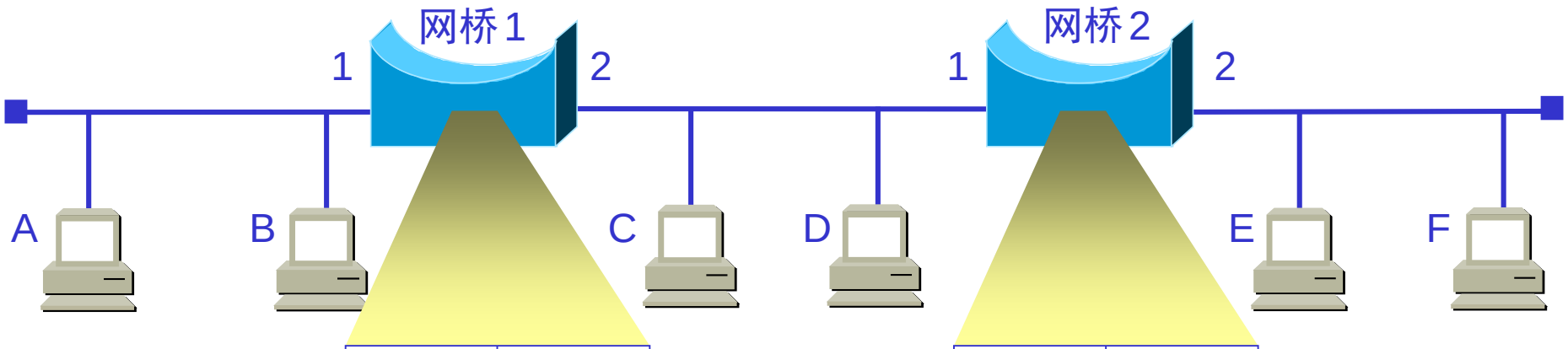
2. 透明网桥

- 目前使用得最多的网桥是透明网桥(transparent bridge)。
- “透明”是指局域网上的站点并不知道所发送的帧将经过哪几个网桥，因为网桥对各站来说是看不见的。
- 透明网桥是一种即插即用设备，其标准是 IEEE 802.1D。

网桥应当按照以下自学习算法处理收到的帧和建立转发表

- 若从 A 发出的帧从接口 x 进入了某网桥，那么从这个接口出发沿相反方向一定可把一个帧传送到 A。
- 网桥每收到一个帧，就记下其源地址和进入网桥的接口，作为转发表中的一个项目。
- 在建立转发表时是把帧首部中的源地址写在“地址”这一栏的下面。
- 在转发帧时，则是根据收到的帧首部中的目的地址来转发的。这时就把在“地址”栏下面已经记下的源地址当作目的地址，而把记下的进入接口当作转发接口。

转发表的建立过程举例



	地址	接口
A → B	A	1
F → C	F	2
B → A	B	1
	…	…

	地址	接口
A → B	A	1
F → C	F	2
	…	…

网桥在转发表中登记以下三个信息

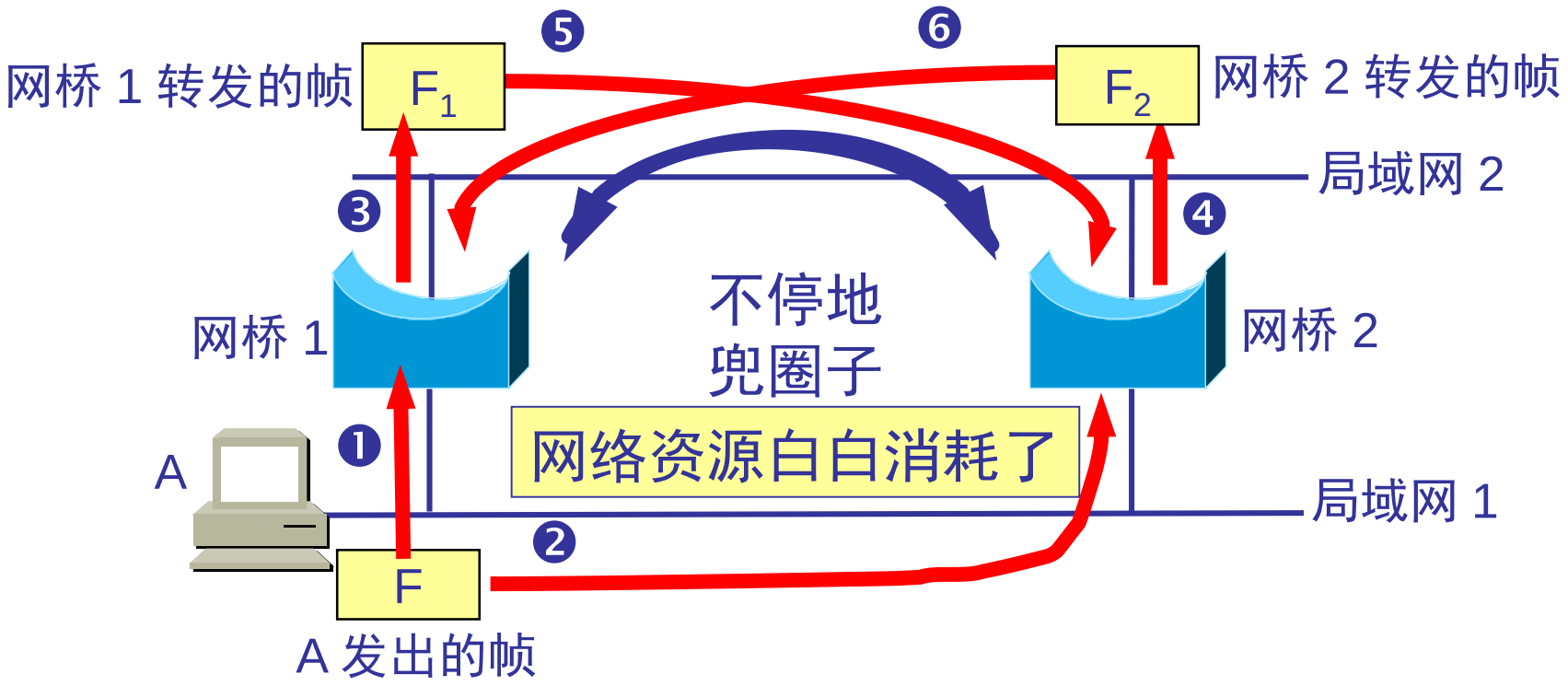
- 在网桥的转发表中写入的信息除了地址和接口外，还有帧进入该网桥的时间。
- 这是因为以太网的拓扑可能经常会发生变化，站点也可能会更换适配器（这就改变了站点的地址）。另外，以太网上的工作站并非总是接通电源的。
- 把每个帧到达网桥的时间登记下来，就可以在转发表中只保留网络拓扑的最新状态信息。这样就使得网桥中的转发表能反映当前网络的最新拓扑状态。

网桥的自学习和转发帧的步骤归纳

- 网桥收到一帧后先进行自学习。查找转发表中与收到帧的源地址有无相匹配的项目。如没有，就在转发表中增加一个项目（源地址、进入的接口和时间）。如有，则把原有的项目进行更新。
- 转发帧。查找转发表中与收到帧的目的地址有无相匹配的项目。
  - 如没有，则通过所有其他接口（但进入网桥的接口除外）按进行转发。
  - 如有，则按转发表中给出的接口进行转发。
  - 若转发表中给出的接口就是该帧进入网桥的接口，则应丢弃这个帧（因为这时不需要经过网桥进行转发）。

透明网桥使用了生成树算法

- 这是为了避免产生转发的帧在网络中不断地兜圈子。



生成树的得出

- 互连在一起的网桥在进行彼此通信后，就能找出原来的网络拓扑的一个子集。在这个子集里，整个连通的网络中不存在回路，即在任何两个站之间只有一条路径。
- 为了避免产生转发的帧在网络中不断地兜圈子。
- 为了得出能够反映网络拓扑发生变化时的生成树，在生成树上的根网桥每隔一段时间还要对生成树的拓扑进行更新。

生成树算法思想

- 思想
  - 让网桥之间互相通信，用一棵连接每个LAN的生成树（Spanning Tree）覆盖实际的拓扑结构
- 构造生成树
  - 每个桥广播自己的桥编号，号最小的桥称为生成树的根；
  - 每个网桥计算自己到根的最短路径，构造出生成树，使得每个LAN和桥到根的路径最短；
  - 当某个LAN或网桥发生故障时，要重新计算生成树；
  - 生成树构造完后，算法继续执行以便自动发现拓扑结构变化，更新生成树。

举例

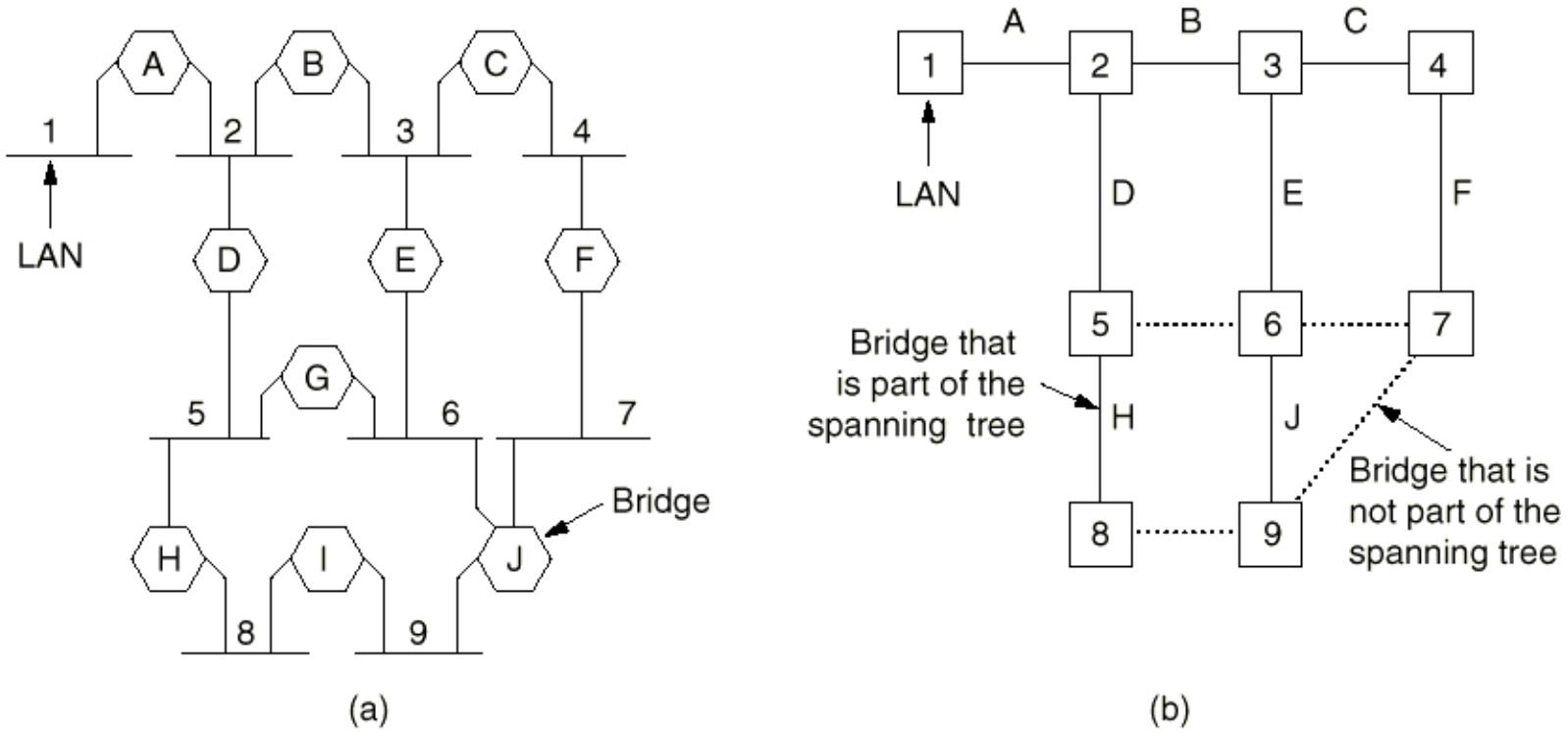


Fig. 4-40. (a) Interconnected LANs. (b) A spanning tree covering the LANs. The dotted lines are not part of the spanning tree.

3. 源路由网桥

- 透明网桥容易安装，但网络资源的利用不充分。
- 源路由(source route)网桥在发送帧时将详细的路由信息放在帧的首部中。
- 源站以广播方式向欲通信的目的站发送一个发现帧，每个发现帧都记录所经过的路由。
- 发现帧到达目的站时就沿各自的路由返回源站。源站在得知这些路由后，从所有可能的路由中选择出一个最佳路由。凡从该源站向该目的站发送的帧的首部，都必须携带源站所确定的这一路由信息。

4. 多接口网桥——以太网交换机

- 1990 年问世的交换式集线器(switching hub)，可明显地提高局域网的性能。
- 交换式集线器常称为以太网交换机(switch)或第二层交换机（表明此交换机工作在数据链路层）。
- 以太网交换机通常都有十几个接口。因此，以太网交换机实质上就是一个多接口的网桥，可见交换机工作在数据链路层。

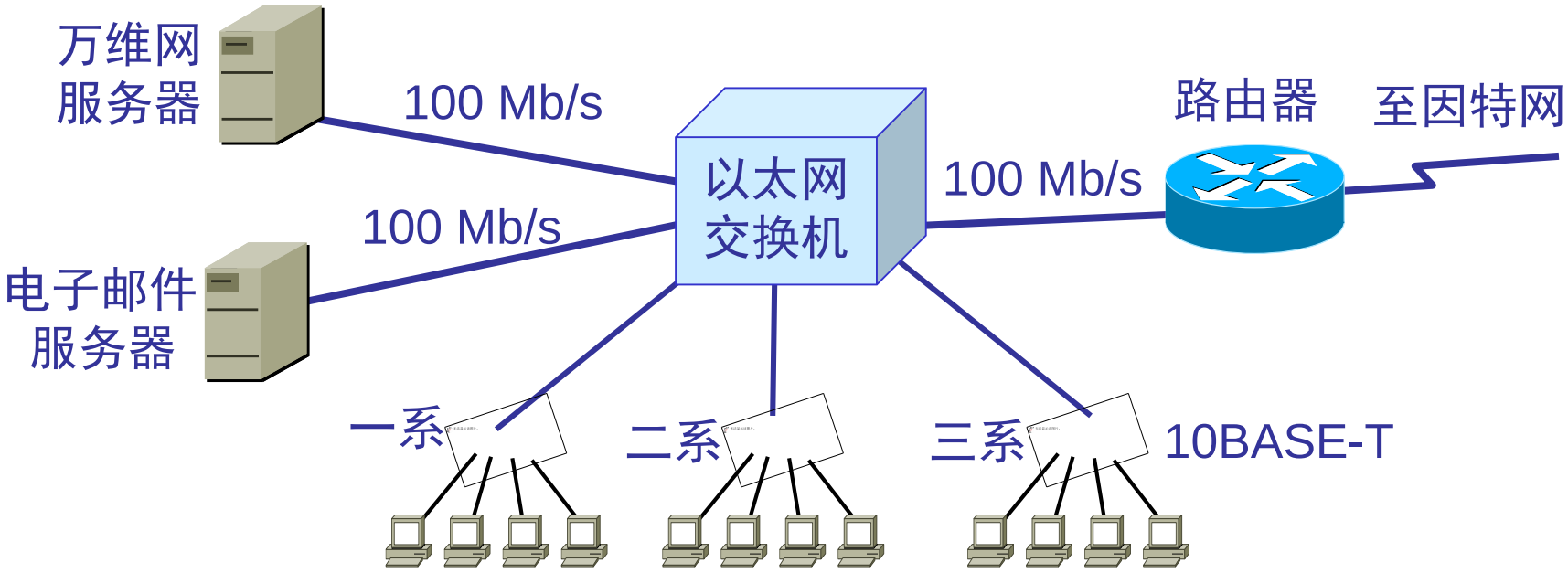
以太网交换机的特点

- 以太网交换机的每个接口都直接与主机相连，并且一般都工作在全双工方式。
- 交换机能同时连通许多对的接口，使每一对相互通信的主机都能像独占通信媒体那样，进行无碰撞地传输数据。
- 以太网交换机由于使用了专用的交换结构芯片，其交换速率就较高。

独占传输媒体的带宽

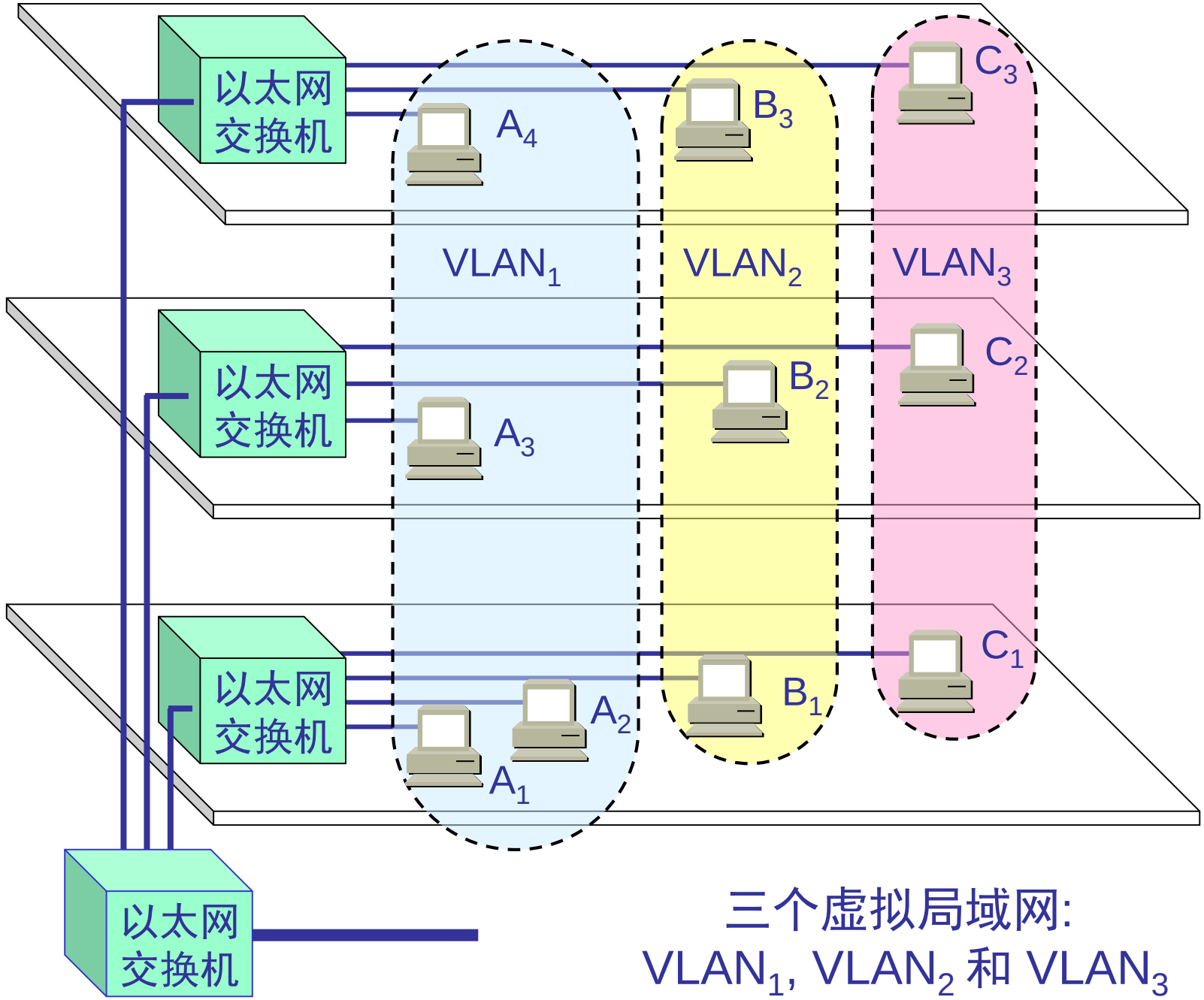
- 对于普通 10 Mb/s 的共享式以太网，若共有 N 个用户，则每个用户占有的平均带宽只有总带宽(10 Mb/s)的 N 分之一。
- 使用以太网交换机时，虽然在每个接口到主机的带宽还是 10 Mb/s，但由于一个用户在通信时是独占而不是和其他网络用户共享传输媒体的带宽，因此对于拥有 N 对接口的交换机的总容量为 $N \times 10$ Mb/s。这正是交换机的最大优点。

用以太网交换机扩展局域网

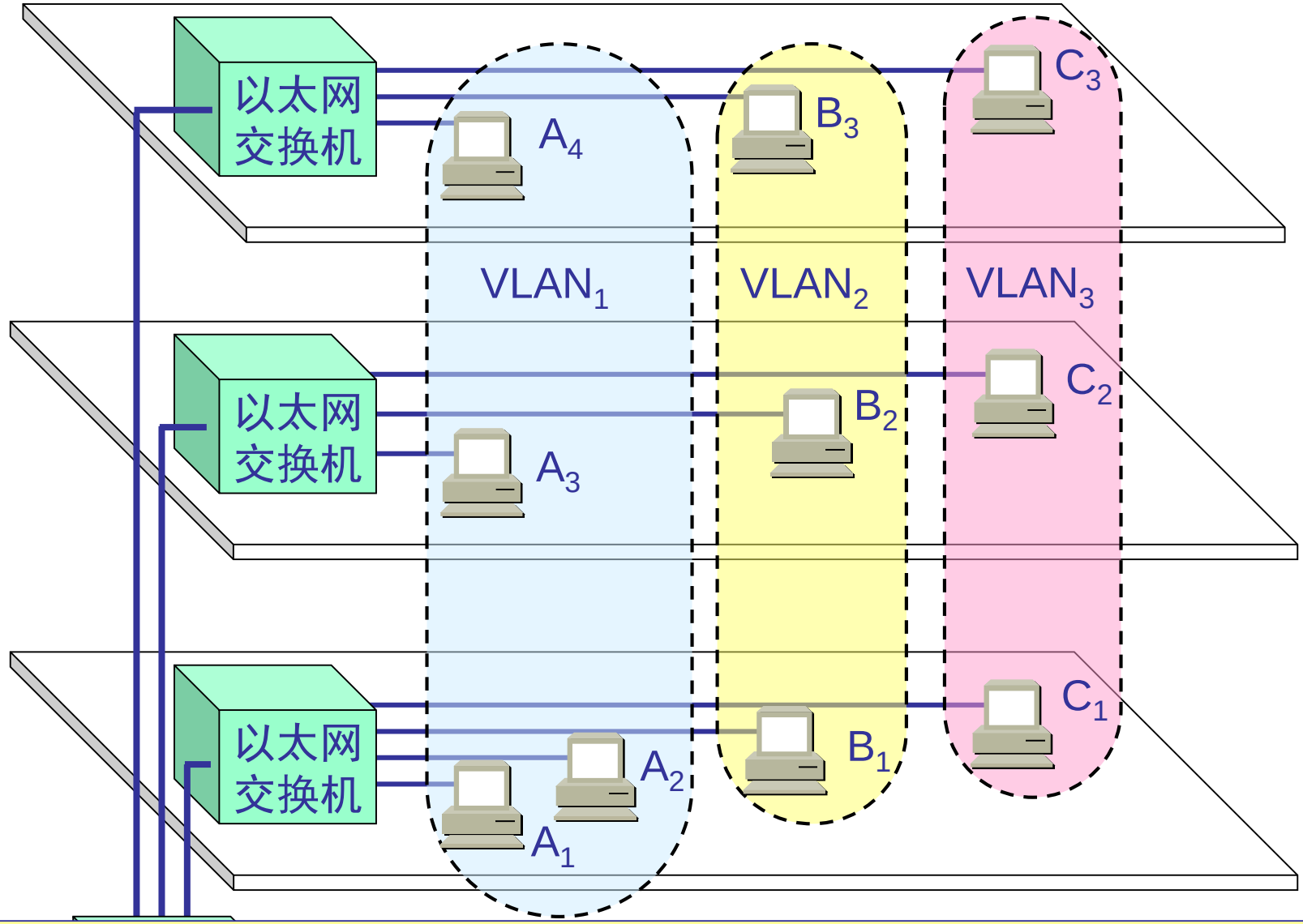


利用以太网交换机可以很方便地实现虚拟局域网

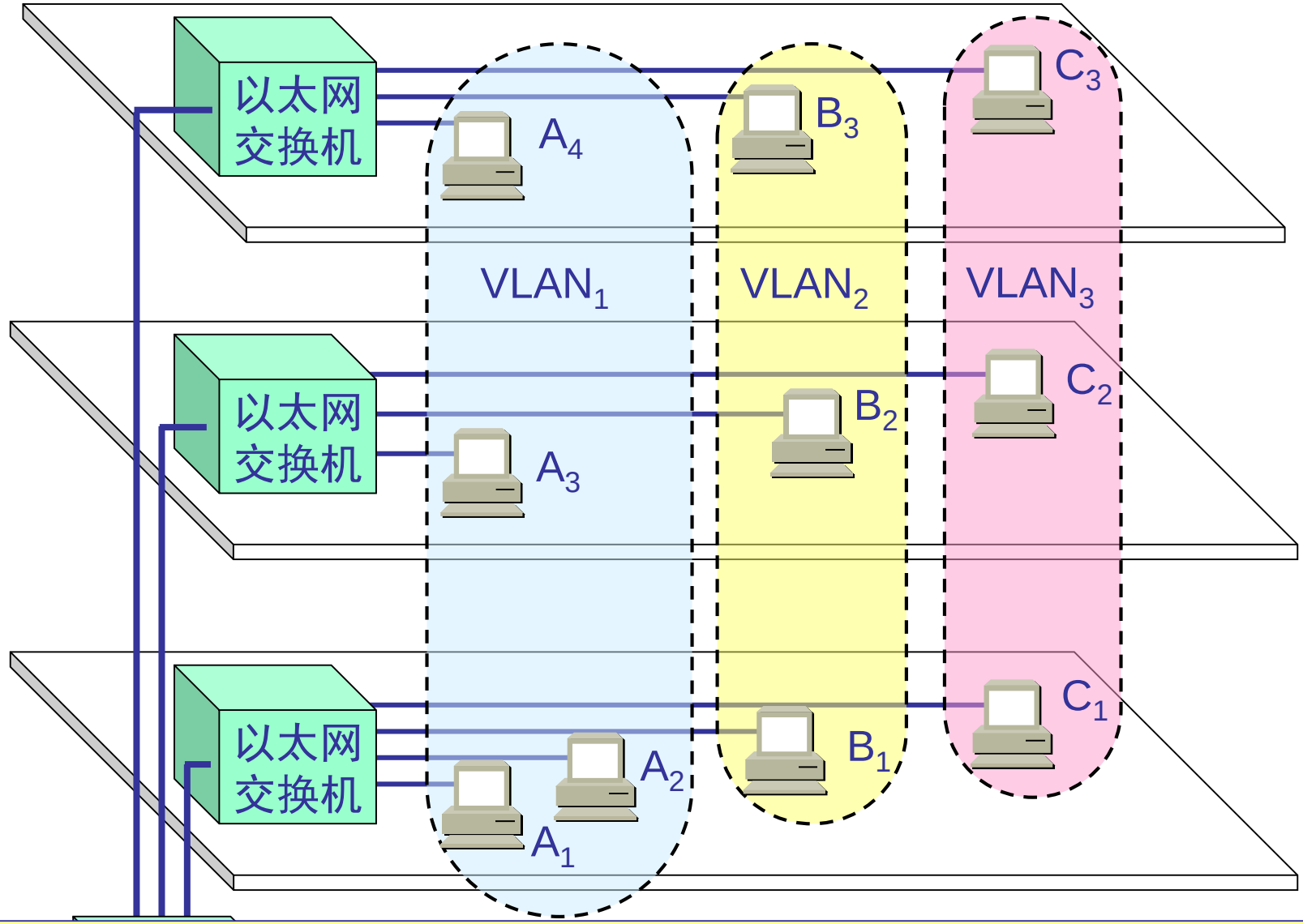
- 虚拟局域网 VLAN 是由一些局域网网段构成的与物理位置无关的逻辑组。
  - 这些网段具有某些共同的需求。
  - 每一个 VLAN 的帧都有一个明确的标识符，指明发送这个帧的工作站是属于哪一个 VLAN。
- 虚拟局域网其实只是局域网给用户提供的一种服务，而并不是一种新型局域网。



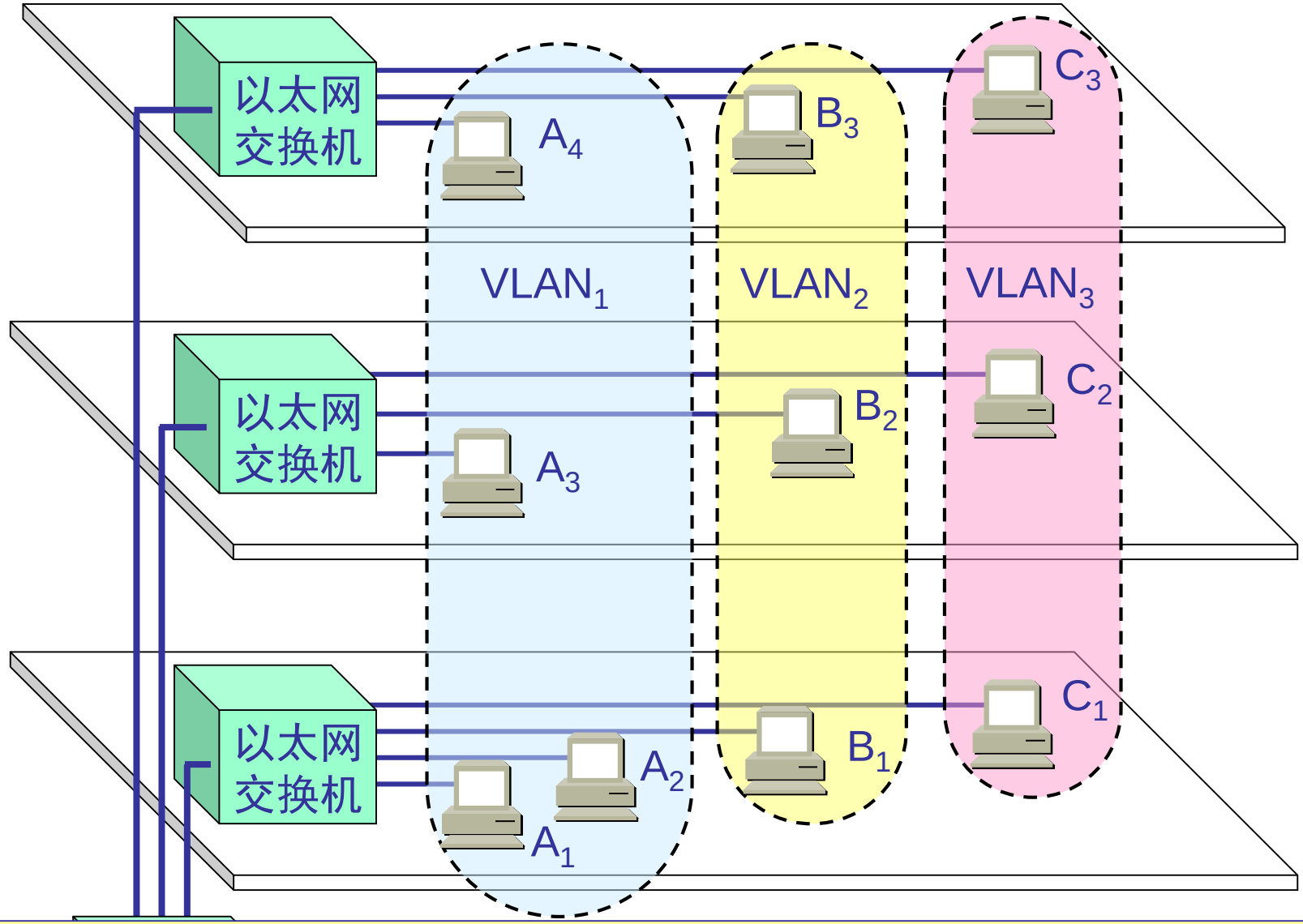
三个虚拟局域网:
$VLAN_1$, $VLAN_2$ 和 $VLAN_3$



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 将会收到广播的信息。



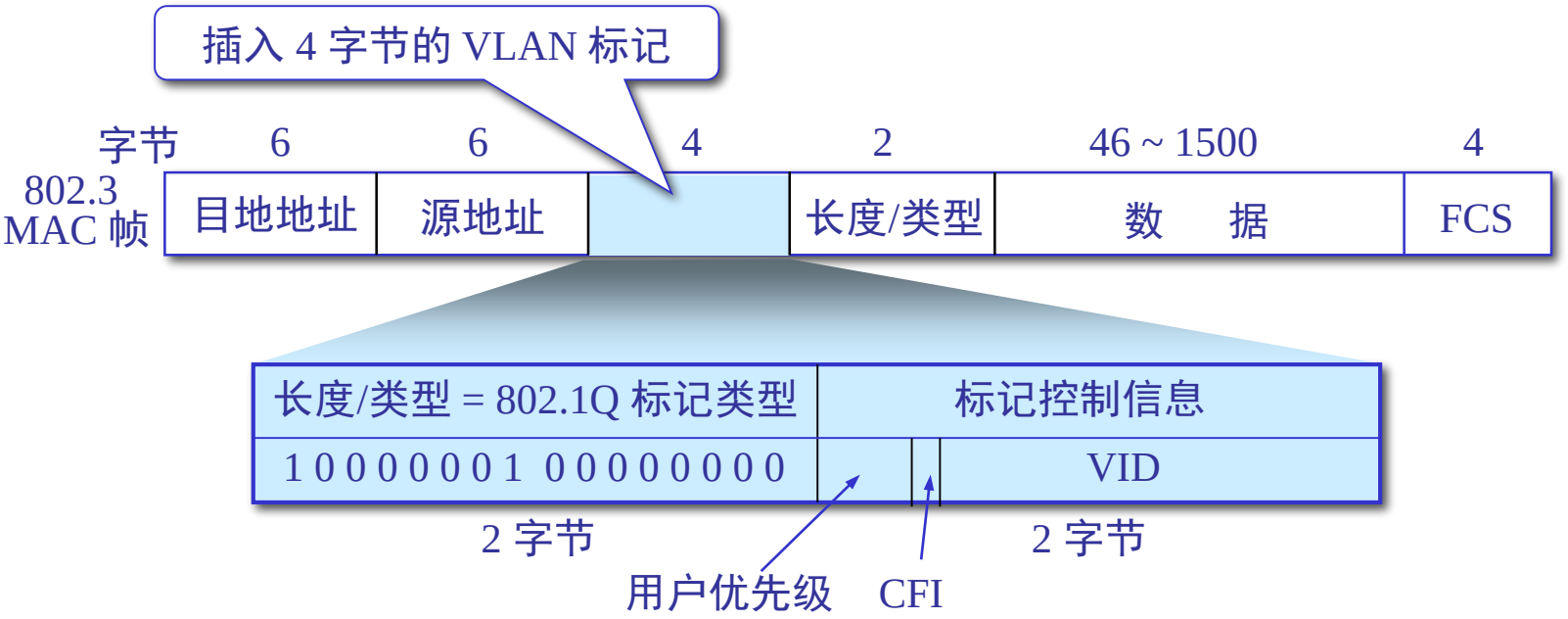
B_1 发送数据时，工作站 A_1, A_2 和 C_1 都不会收到 B_1 发出的广播信息。



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。

虚拟局域网使用的以太网帧格式

- 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)，用来指明发送该帧的工作站属于哪一个虚拟局域网。



3.6 高速以太网

3.6.1 100BASE-T 以太网

- 速率达到或超过 100 Mb/s 的以太网称为高速以太网。
- 在双绞线上传送 100 Mb/s 基带信号的星型拓扑以太网，仍使用 IEEE 802.3 的CSMA/CD 协议。100BASE-T 以太网又称为快速以太网 (Fast Ethernet)。

100BASE-T 以太网的特点

- 可在全双工方式下工作而无冲突发生。因此，不使用 CSMA/CD 协议。
- MAC 帧格式仍然是 802.3 标准规定的。
- 保持最短帧长不变，但将一个网段的最大电缆长度减小到 100 m。
- 帧间时间间隔从原来的 9.6 μs 改为现在的 0.96 μs。

100 Mb/s 以太网的三种不同的物理层标准

- 100BASE-TX
  - 使用 2 对 UTP 5 类线或屏蔽双绞线 STP。
- 100BASE-FX
  - 使用 2 对光纤。
- 100BASE-T4
  - 使用 4 对 UTP 3 类线或 5 类线。

3.6.2 吉比特以太网

- 允许在 1 Gb/s 下全双工和半双工两种方式工作。
- 使用 802.3 协议规定的帧格式。
- 在半双工方式下使用 CSMA/CD 协议（全双工方式不需要使用 CSMA/CD 协议）。
- 与 10BASE-T 和 100BASE-T 技术向后兼容。

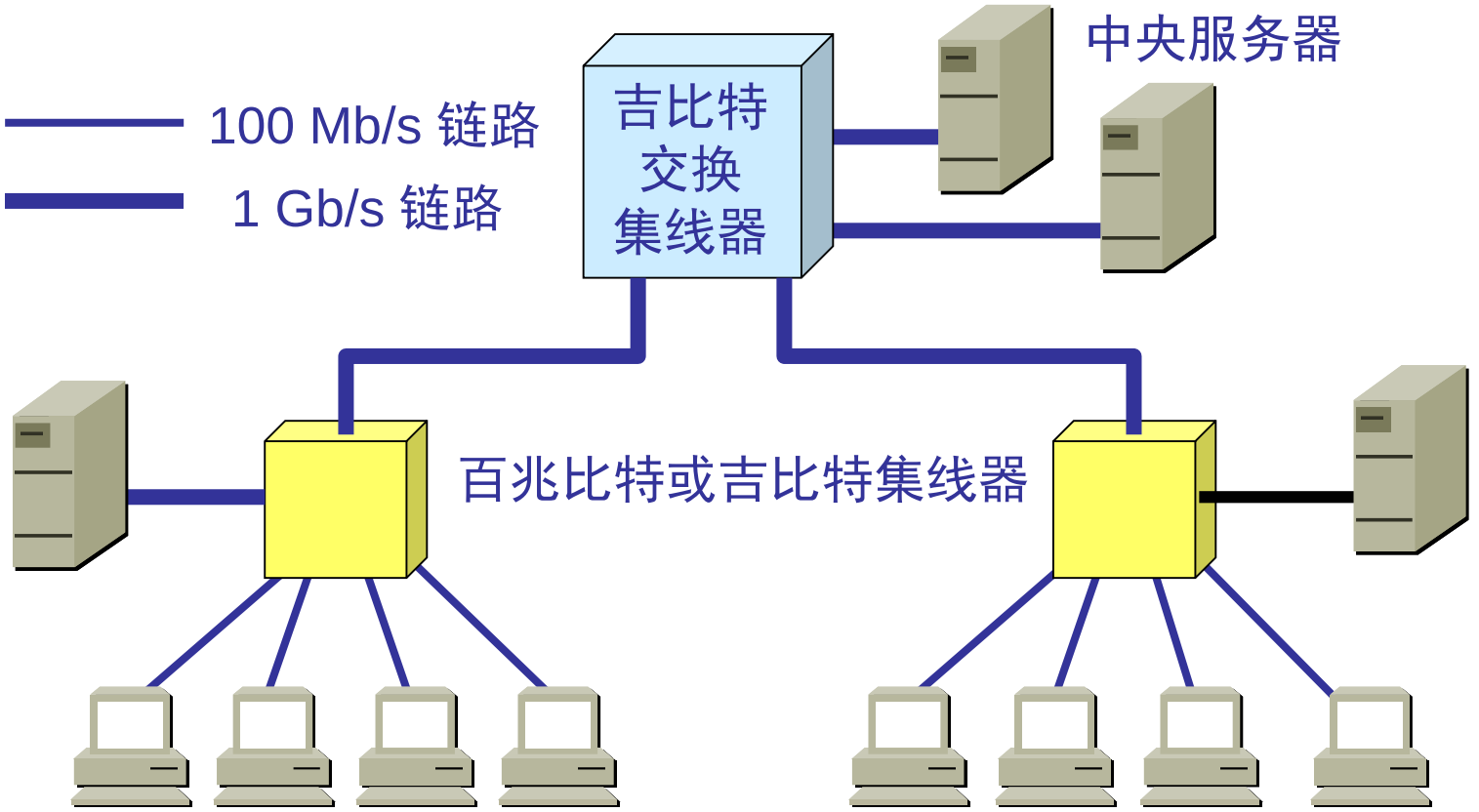
吉比特以太网的物理层

- 1000BASE-X　基于光纤通道的物理层：
  - 1000BASE-SX　SX表示短波长
  - 1000BASE-LX　LX表示长波长
  - 1000BASE-CX　CX表示铜线
- 1000BASE-T
  - 使用 4对 5 类线 UTP

全双工方式

- 当吉比特以太网工作在全双工方式时（即通信双方可同时进行发送和接收数据），不使用载波延伸和分组突发。

吉比特以太网的配置举例



3.6.3 10 吉比特以太网和100 吉比特以太网

- 10 吉比特以太网与 10 Mb/s，100 Mb/s 和 1 Gb/s 以太网的帧格式完全相同。
- 10 吉比特以太网还保留了 802.3 标准规定的以太网最小和最大帧长，便于升级。
- 10 吉比特以太网不再使用铜线而只使用光纤作为传输媒体。
- 10 吉比特以太网只工作在全双工方式，因此没有争用问题，也不使用 CSMA/CD 协议。

吉比特以太网的物理层

- 局域网物理层 LAN PHY。局域网物理层的数据率是 10.000 Gb/s。
- 可选的广域网物理层 WAN PHY。广域网物理层具有另一种数据率，这是为了和所谓的“Gb/s”的 SONET/SDH（即OC-192/STM-64）相连接。
  - 为了使 10 吉比特以太网的帧能够插入到 OC-192/STM-64 帧的有效载荷中，就要使用可选的广域网物理层，其数据率为 9.95328 Gb/s。

端到端的以太网传输

- 10 吉比特以太网的出现，以太网的工作范围已经从局域网（校园网、企业网）扩大到城域网和广域网，从而实现了端到端的以太网传输。
- 这种工作方式的好处是：
  - 成熟的技术
  - 互操作性很好
  - 在广域网中使用以太网时价格便宜。
  - 统一的帧格式简化了操作和管理。

10 G以太网的物理层标准

- 10GBASE-SR 光缆，300 m，多模光纤（0.85 μm）
- 10GBASE-LR 光缆，10 km，单模光纤（1.3 μm）
- 10GBASE-ER 光缆，40 km，单模光纤（1.5 μm）
- 10GBASE-CX4 铜缆，15 m，4 对双芯同轴电缆(twinax)
- 10GBASE-T 铜缆，100 m，4 对 6A 类UTP双绞线

40GB/100GB 以太网的物理层标准

物 理 层	40GB 以太网	100GB 以太网
在背板上传输 至少超过1 m	40GBASE-KR4	
在铜缆上传输 至少超过7 m	40GBASE-CR4	100GBASE-CR10
多模光纤上传输 至少100 m	40GBASE-SR4	100GBASE-SR10
单模光纤上传输 至少10 km	40GBASE-LR4	100GBASE-LR4
单模光纤上传输 至少40 km		100GBASE-ER4

以太网从 10 Mb/s 到100 Gb/s 的演进

- 以太网从 10 Mb/s 到 100 Gb/s 的演进证明了以太网是:
- 可扩展的（从 10 Mb/s 到 100 Gb/s）。
- 灵活的（多种传输媒体、全/半双工、共享/交换）。
- 易于安装。
- 稳健性好。

3.6.4 使用高速以太网进行宽带接入

- 以太网已成功地把速率提高到 1 ~ 10 Gb/s，所覆盖的地理范围也扩展到了城域网和广域网，因此现在人们正在尝试使用以太网进行宽带接入。
- 以太网接入的重要特点是它可提供双向的宽带通信，并且可根据用户对带宽的需求灵活地进行带宽升级。
- 采用以太网接入可实现端到端的以太网传输，中间不需要再进行帧格式的转换。这就提高了数据的传输效率和降低了传输的成本。

本章小结

- 数据链路层的功能
- 组帧
- 差错控制
  - 检错编码
  - 纠错编码
  - 常见数据链路层协议
  - PPP协议、HDLC协议
  - 随机访问介质访问控制
  - CSMA/CD协议。
  - 局域网
  - 局域网的基本概念与体系结构
  - 以太网与IEEE 802.3
  - 数据链路层设备
  - 网桥：网桥的概念；透明网桥与生成树算法；源选径网桥与源选径算法
  - 局域网交换机及其工作原理。

END!