

计算机网络应用技术

第三章：数据链路层

信息技术学院互联网技术教学团队

阮晓龙 许成刚 高海波 李瑞昌

<http://xslt.it.hactcm.edu.cn>

2022.9



扫码访问课程学习平台

本章教学计划

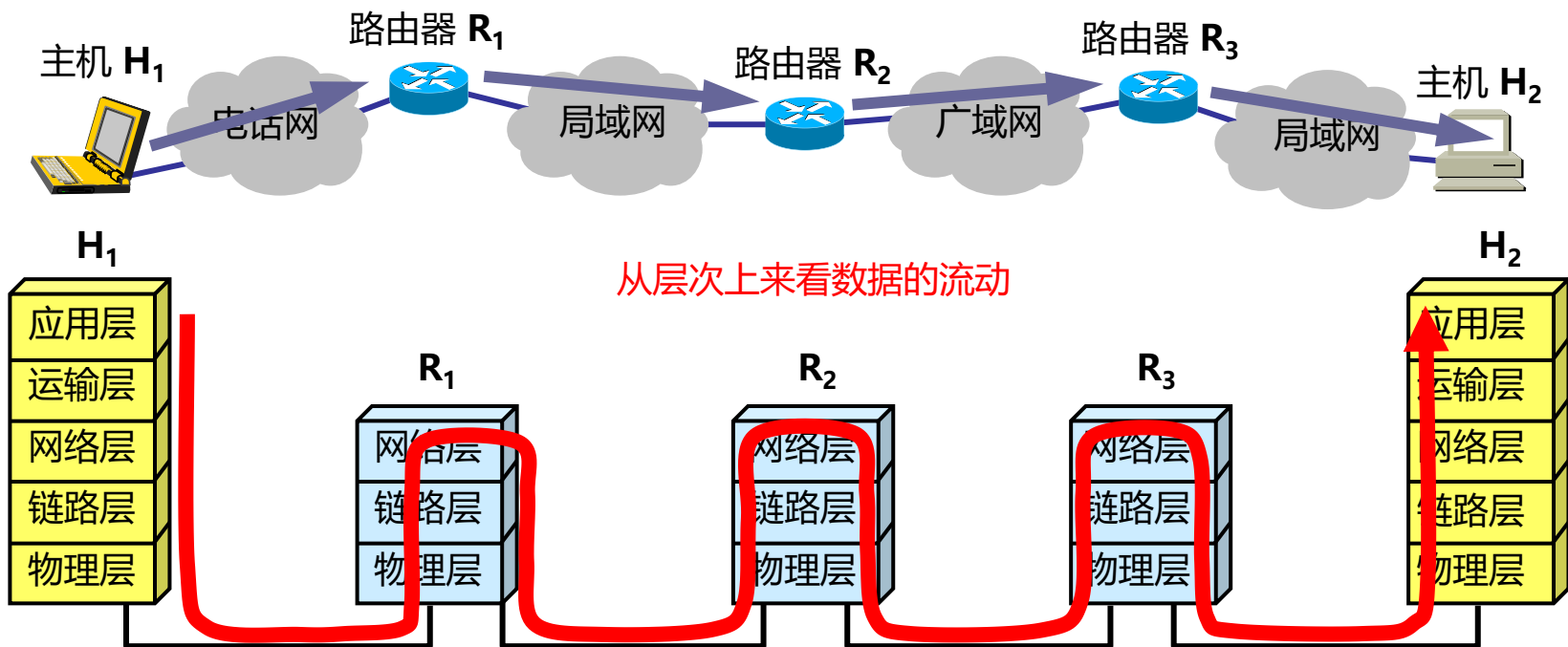
- 使用点对点信道的数据链路层
 - 点对点协议 (PPP)
 - 使用广播信道的数据链路层 (CSMA/CD)
- } 数据链路层原理
-
- 使用广播信道的以太网
 - 扩展的以太网
 - 高速以太网
- } 局域网通信

本章教学计划

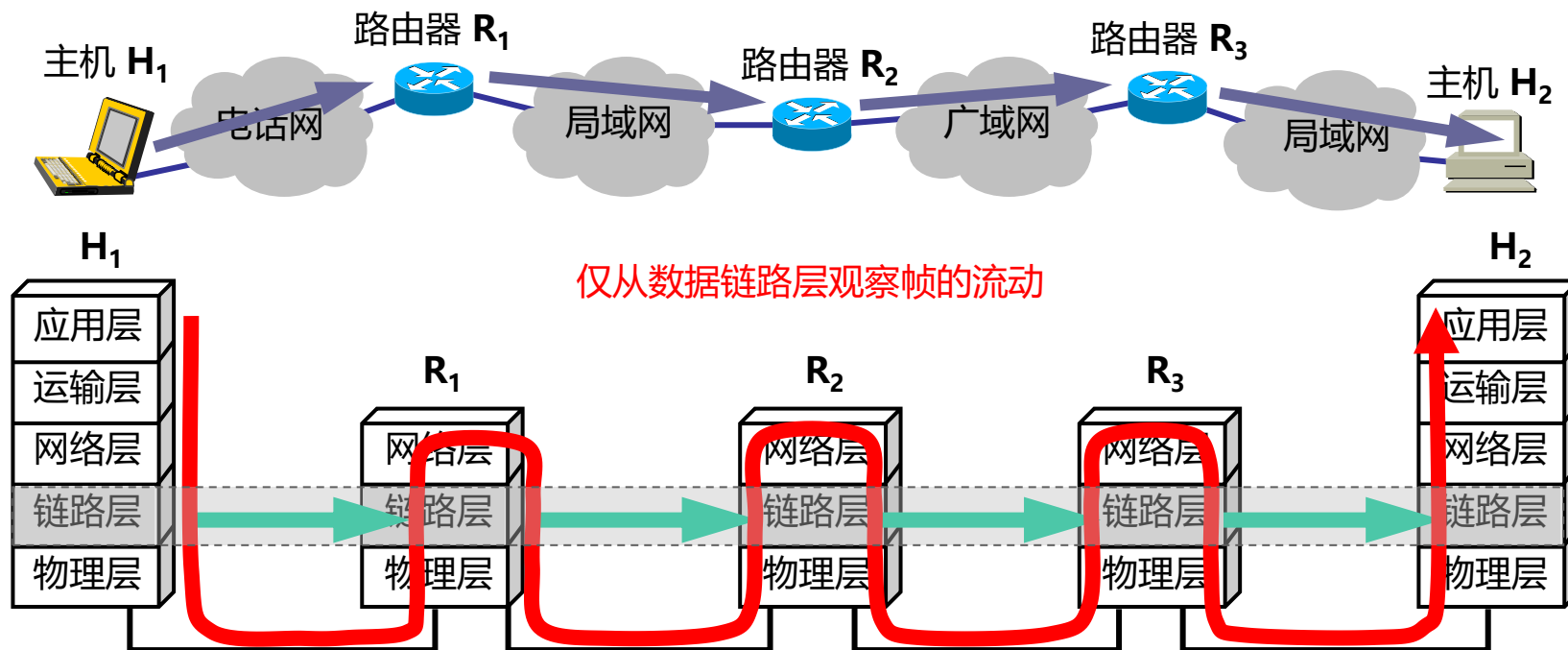
- 数据链路层属于计算机网络的低层。
- 数据链路层使用的信道主要有两种类型：
 - 点对点信道：一对一通信方式
 - 广播信道：一对多通信方式

- 由于局域网内通常不包括路由选择和分组转发，因此局域网的内容也在本章进行讨论。

本章教学计划



本章教学计划



1.使用点到点信道的数据链路层

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。
 - 有些情况下，链路也被称为“物理链路”。

1.使用点到点信道的数据链路层

1.1数据链路和帧

- 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。
 - 有些情况下，数据链路又被称为“逻辑链路”。

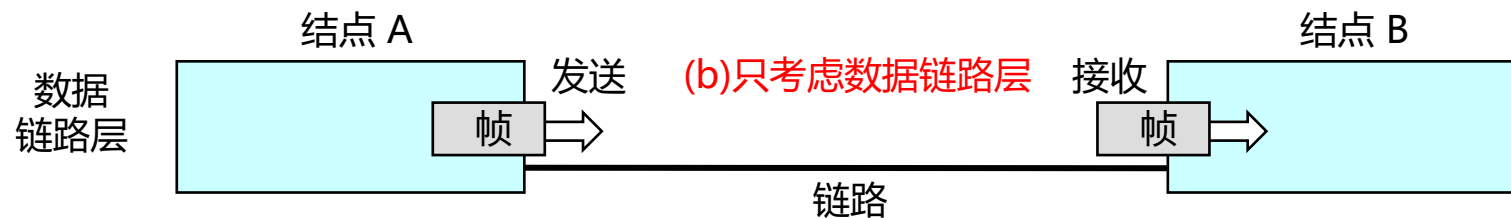
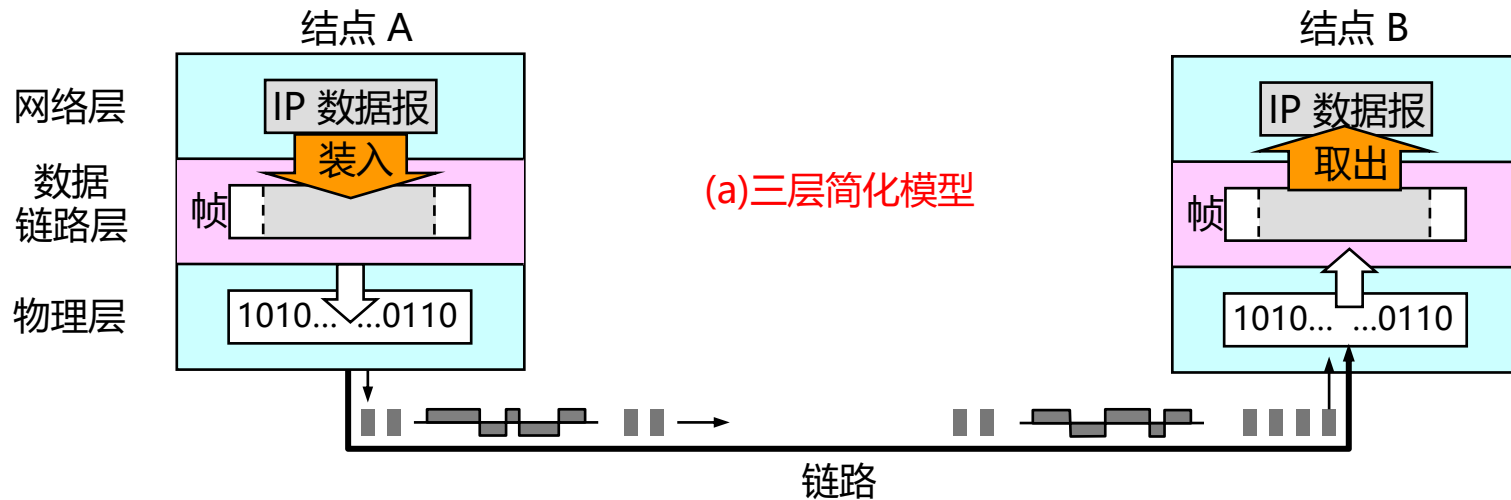
1.使用点到点信道的数据链路层

1.1数据链路和帧

- 数据链路层把网络层交下来的数据构成帧发送到链路上，以及把接收到的帧中的数据取出并上交给网络层。
 - 数据链路层协议数据单元是数据帧（帧）。
 - 网络层协议数据单元是IP数据报（数据报、分组、包）。
- 把重点放到数据链路层，只考虑数据链路层的通信，把网络体系简化为三层。

1.使用点到点信道的数据链路层

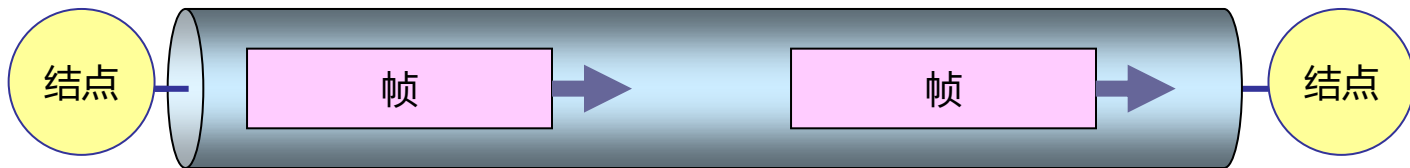
1.1数据链路和帧



1.使用点到点信道的数据链路层

1.1数据链路和帧

- 常常在两个对等的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。



1.使用点到点信道的数据链路层 1.2数据链路层的基本功能

封装成帧

1

形成帧

在一段数据的前后分别添加首部和尾部。

数据链路层的基本功能

透明传输

2

透明传送数据

在数据链路层上能够传送任何数据，任意组合的比特流能够在数据链路层传送。

数据链路层的基本功能

差错检测

3

可靠传输

解决比特差错。
不能够解决传输擦差错。

数据链路层的基本功能

1.使用点到点信道的数据链路层

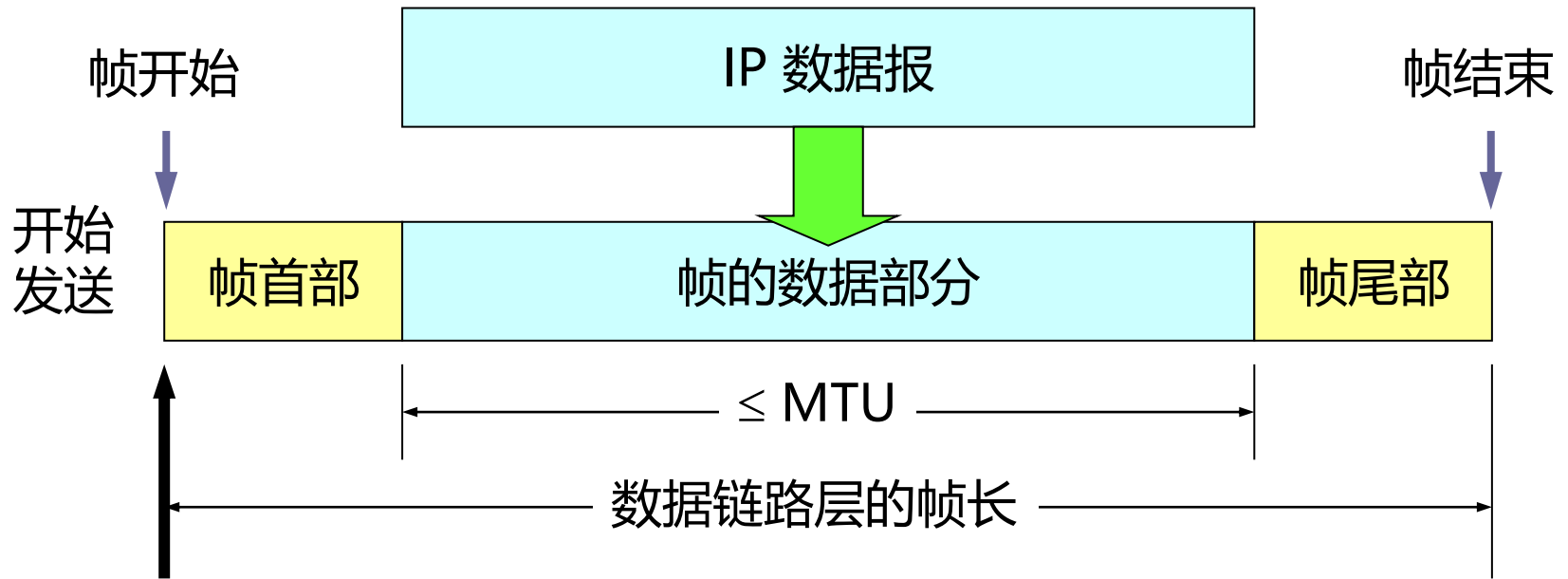
1.2数据链路层的基本功能

□ 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。
- 数据帧的长度就是数据加上首部和尾部的总长度。
- 首部和尾部的重要作用之一就是进行帧定界，确定数据帧的界限。

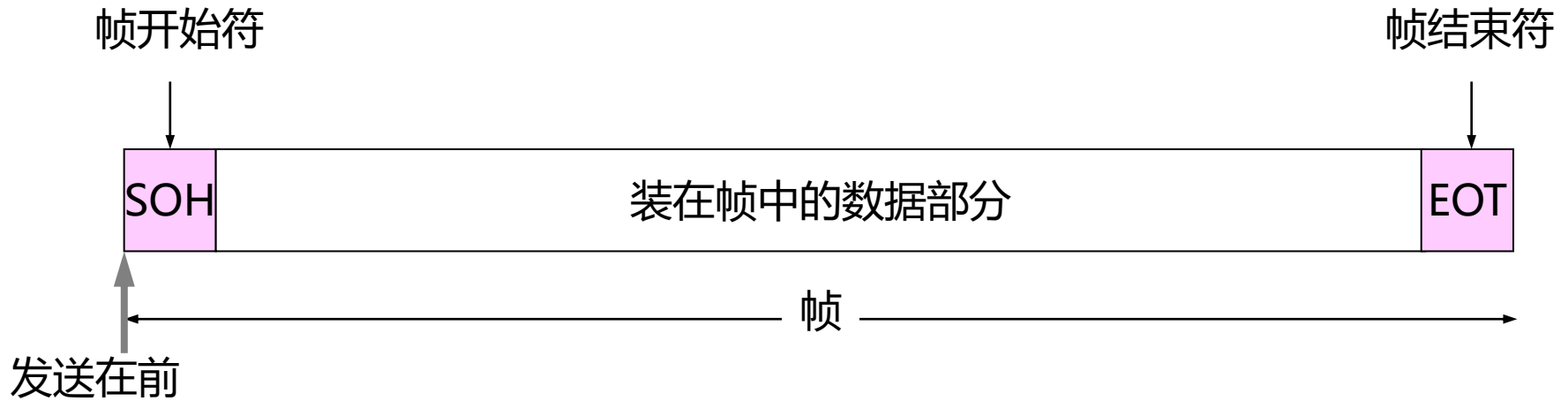
1.使用点到点信道的数据链路层

1.2数据链路层的基本功能



1.使用点到点信道的数据链路层

1.2数据链路层的基本功能



1.使用点到点信道的数据链路层

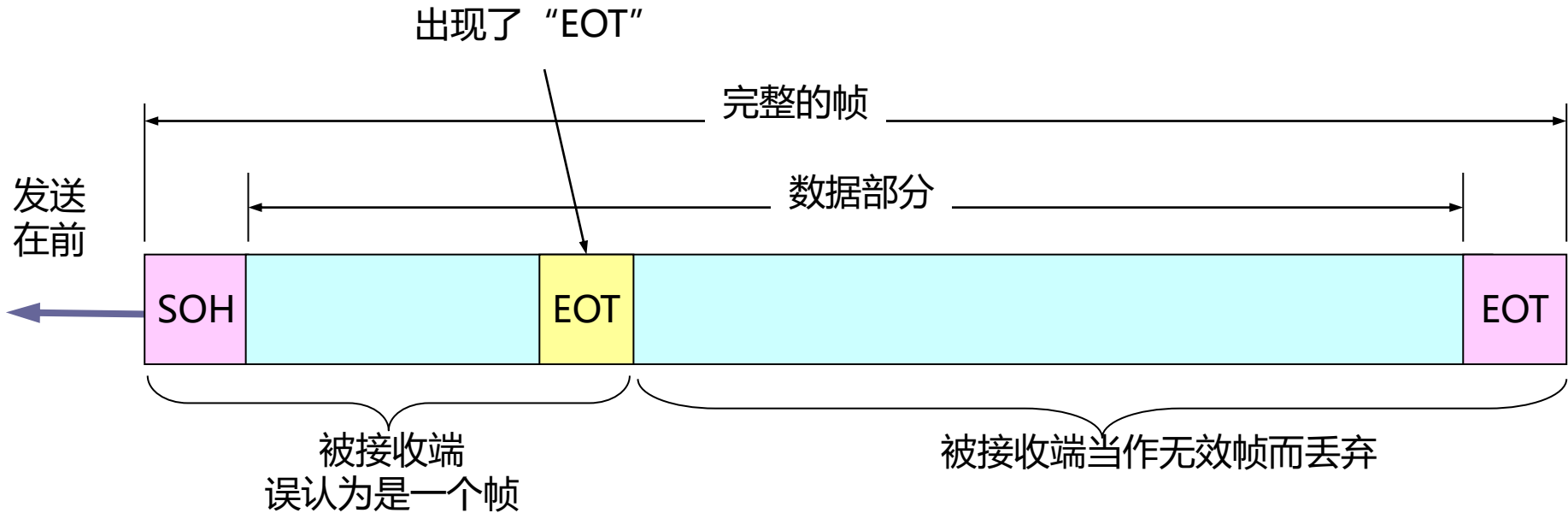
1.2数据链路层的基本功能

□ 透明传输：

- 由于帧的开始和结束的标记是专门指明的字符，因此所传输的数据中不能够有任何8比特的字符和用作帧定界的控制字符的比特编码一样。
- 也就是说传输的数据中，不能够有任何和SOH、EOT一致的8比特的字符。
- 为了解决这个问题，就采用了转义字符“ESC”。
 - ESC值是确定的，十六进制1B，或二进制00011011。

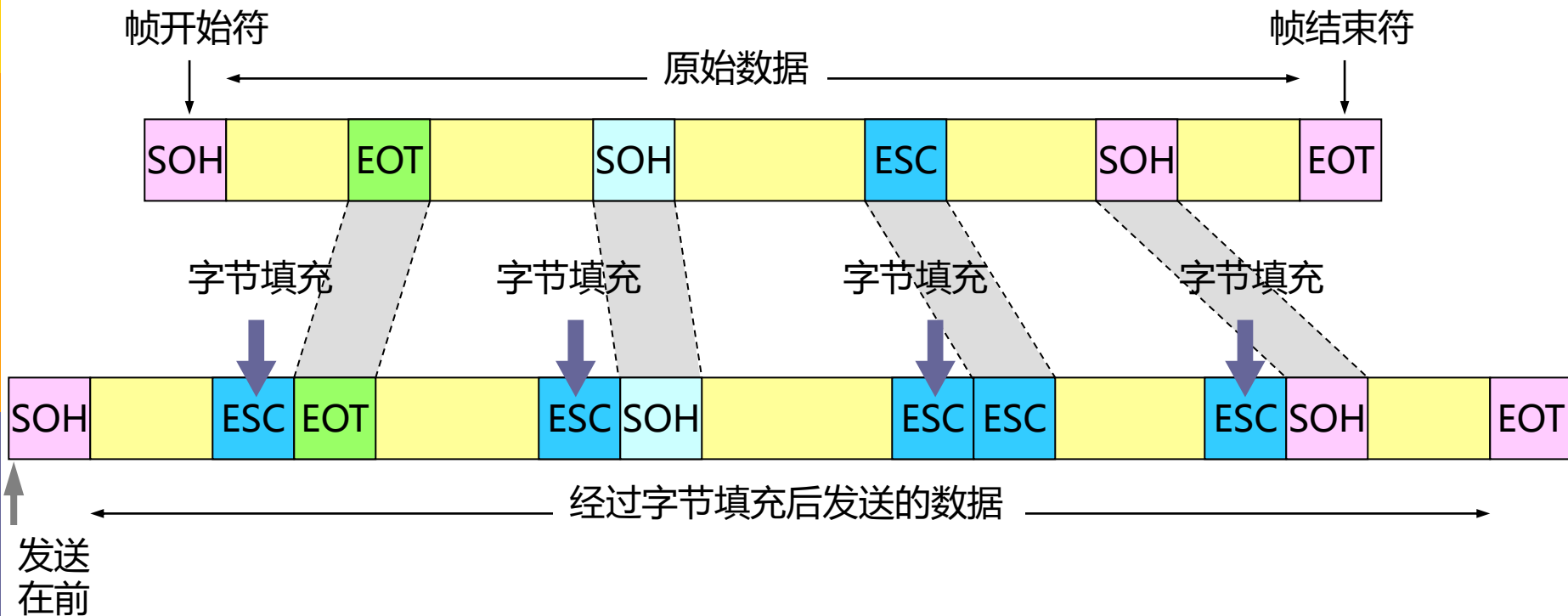
1.使用点到点信道的数据链路层

1.2数据链路层的基本功能



1.使用点到点信道的数据链路层

1.2数据链路层的基本功能



1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测:

- 数据帧在传输中可能发生两种错误:

- **比特差错**: 1可能变成0 或者 0可能变成1。
- **传输差错**: 收到的帧没有出现比特差错, 但出现帧丢失、帧重复和帧失序。

发送方

0	0	0	1	1	0	1	1
---	---	---	---	---	---	---	---



接收方

0	0	0	0	1	0	1	1
---	---	---	---	---	---	---	---

一位比特错

发送方

0	0	0	1	0	0	1	1	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---



接收方

0	0	0	0	1	0	1	1	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---

多位比特错

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

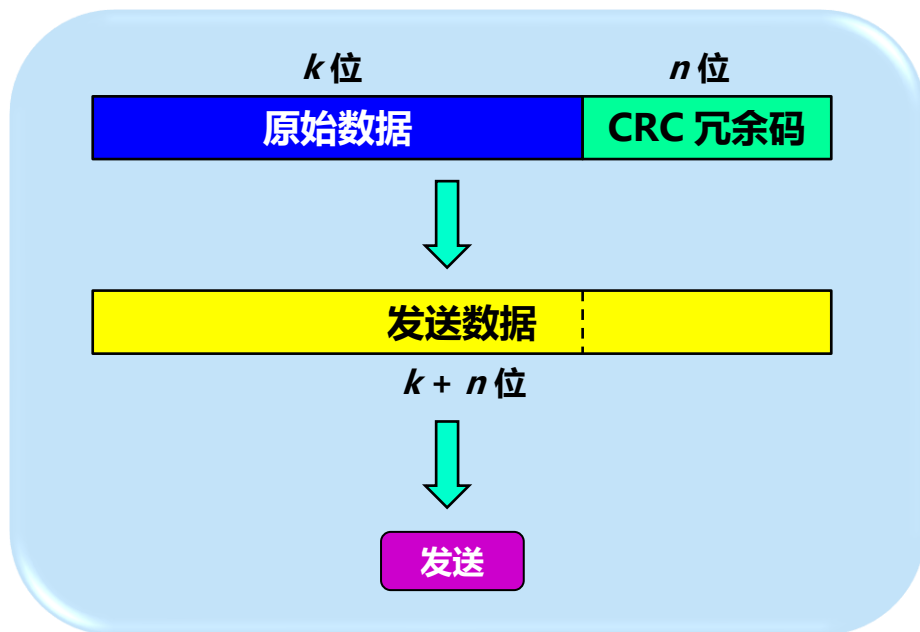
- 在一段时间内，传输错误的比特占所传输比特总数的比率称为**误码率BER**（Bit Error Rate）。
- 误码率通常和信噪比关系密切。
 - 信噪比越高，误码率越低。
- 为了保证数据传输的可靠性，在计算机网络传输数据时，必须采用各种**差错检测措施**。
- 在数据链路层传送的帧中，广泛使用了**循环冗余检验 CRC** 的检错技术。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

■ CRC的基本原理：



- 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。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

■ CRC的基本原理：

□ 在发送端：

- 先把数据划分为组，假定每组 k 个比特。
- 通过CRC运算，在数据 M 的后面添加供差错检测用的 n 位冗余码，然后构成一个帧发送出去，一共发送 $(k+n)$ 位。

□ 在接收端：

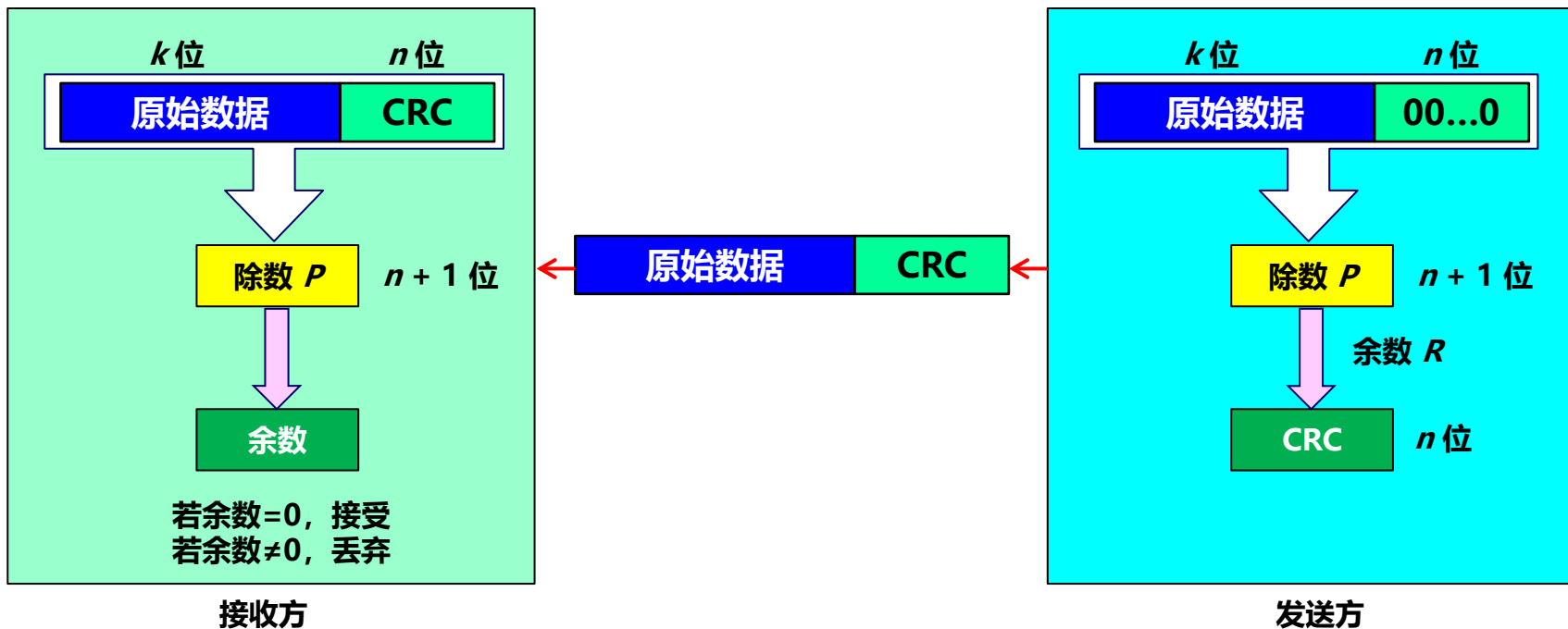
- 先从接收的数据帧中提取出 n 位冗余码，进行CRC检验。
- 如果检验的结果正确，则接受 (accept) 数据帧。
- 如果检验的结果不正确，则直接丢弃。
- CRC检验不能确定错误发生的原因，只能判断数据帧是否正确。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测:

■ CRC的基本原理:



1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测:

- 循环冗余校验码 (CRC) 的基本原理是:
 - 在K位信息码后再拼接R位的校验码, 整个编码长度为N位, 因此, 这种编码也叫 (N, K) 码。
 - 对于一个给定的 (N, K) 码, 可以证明存在一个最高次幂为 $N-K=R$ 的多项式 $G(x)$ 。
 - 根据 $G(x)$ 可以生成K位信息的校验码, 而 $G(x)$ 叫做这个CRC码的生成多项式。
 - 假设要发送的信息用多项式 $C(x)$ 表示, 将 $C(x)$ 左移R位 (可表示成 $C(x)*2^R$), 这样 $C(x)$ 的右边就会空出R位, 这就是校验码的位置。
 - 用 $C(x)*2^R$ 除以生成多项式 $G(x)$ 得到的余数就是校验码。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测:

■ 循环冗余校验码 (CRC) 的两个计算:

- 发送端: 如何计算出冗余码
- 接收端: 如何通过冗余码判断数据帧是否错误

■ 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。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

- 循环冗余校验码（CRC）的两个计算：
 - 发送端：如何计算出冗余码
 - 接收端：如何通过冗余码判断数据帧是否错误
- 通过冗余码判断数据帧是否错误：
 - 若得出的余数 $R = 0$ ，则判定这个帧没有差错，就接受 (accept)。
 - 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。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

■ 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 $2^n M = 101001000$
- 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： $2^n M + R$, 即： **101001001**
- 发送的数据共： $(k + n)$ 位

1.使用点到点信道的数据链路层 1.2数据链路层的基本功能

□ 差错检测：

■ 冗余码的计算举例：

原始数据 $M = 101001$

除数 $P = 1101$

得到：

发送数据 = 101001**001**

$$\begin{array}{r}
 \begin{array}{l} P(\text{除数}) \rightarrow 1101 \end{array} \begin{array}{l}) \\ \hline \end{array} \begin{array}{l} 101001000 \\ \hline 1101 \\ \hline 1110 \\ \hline 1101 \\ \hline 0111 \\ \hline 0000 \\ \hline 1110 \\ \hline 1101 \\ \hline 0110 \\ \hline 0000 \\ \hline 1100 \\ \hline 1101 \\ \hline 001 \end{array} \begin{array}{l} \leftarrow Q(\text{商}) \\ \leftarrow 2^n M(\text{被除数}) \\ \\ \\ \\ \\ \\ \\ \\ \\ \leftarrow R(\text{余数}), \text{ 作为 FCS} \end{array}
 \end{array}$$

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

- 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 不是获得 FCS 的唯一方法。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

- 仅用循环冗余检验 CRC 差错检测技术只能做到**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），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，因为有差错的帧被丢弃。
- 单纯使用 CRC 差错检测技术不能实现“无差错传输”或“可靠传输”

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

□ 差错检测：

- 【无比特差错】与【无传输差错】是不同的概念。
 - 在数据链路层使用 CRC 检验，能够实现无比特差错的传输，但这不是可靠传输。
 - 要做到无差错传输，即发送什么就收到什么，就必须再加上**确认和重传机制**。
 - 数据链路层协议都不是可靠传输的协议。

1.使用点到点信道的数据链路层

1.2数据链路层的基本功能

关于CRC的学习，请通过查阅资料，了解详细的计算方法。
http://www.repairfaq.org/filipg/LINK/F_crc_v31.html

2.点到点协议 (PPP)

2.1 PPP协议

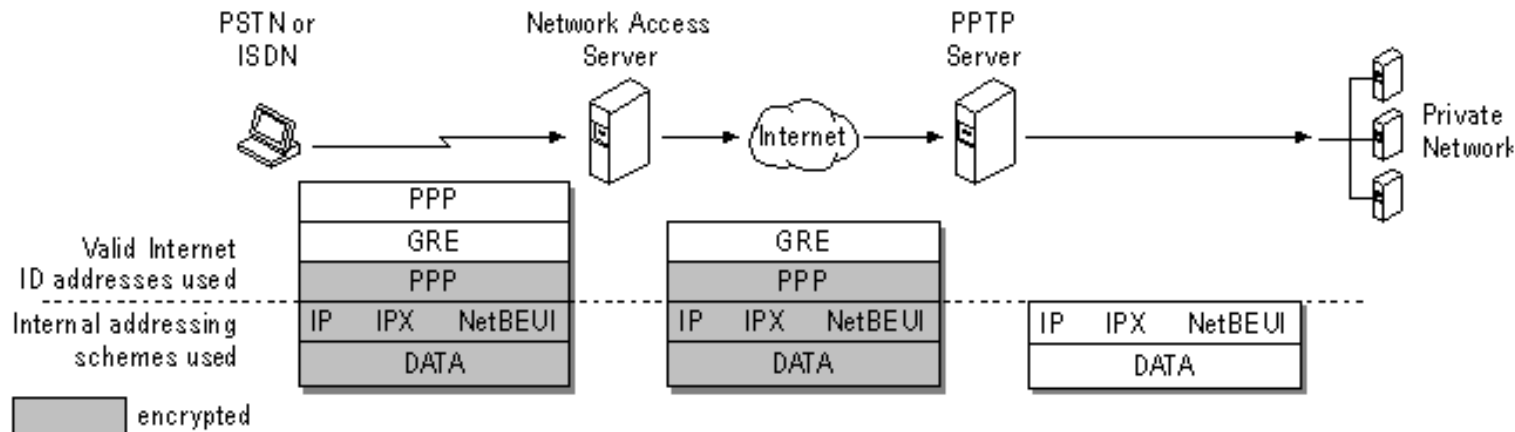
- 点对点协议 PPP (Point-to-Point Protocol): 点对点链路上, 使用最为广泛的数据链路层协议。
- PPP协议是 IETF 在1992年制定, 经过1993年和1994年的修订, 于1994年成为因特网的正式标准【RFC 1661】。

2.点到点协议 (PPP)

2.1 PPP协议

□ 什么情况下使用点对点协议 PPP?

- 用户使用拨号电话线接入因特网时，一般都是使用 PPP 协议。



2.点到点协议 (PPP)

2.1 PPP协议

□ PPP协议的基本需求:

■ 简单:

- IETF在设计因特网体系结构时把其中最复杂的部分放在TCP协议中，而网际协议IP则相对比较简单，它提供的是不可靠的数据报服务。
- 数据链路层就应该设计的比网络层更加简单，因此数据链路层的帧，不需要纠错，不需要序号，也不需要进行流量控制。
- “简单”是首要追求。对简单的追求还使得协议不易出错，不同厂商在协议的不同实现上提高了互操作性，促进了协议标准化工作。

■ 封装成帧:

- PPP协议规定特殊的字符作为帧定界符，便于在比特流中方便地确定帧的开始和结束位置。

2.点到点协议 (PPP)

2.1 PPP协议

□ PPP协议的基本需求:

- 透明性:
 - PPP协议要保证数据传输的透明性,也就是转义符。
- 多种网络层协议:
 - PPP协议必须能够在同一条物理链路上同时支持多种网络层协议的运行。
- 多种类型链路:
 - PPP协议必须能够在多种类型的链路上运行,要支持并行和串行通信、同步和异步通信、低速和高速通信、电或者光通信、交换和静态通信。
- 差错检测:
 - PPP协议要对接收的帧进行检测,对检测错误的帧立即丢弃。

2.点到点协议 (PPP)

2.1 PPP协议

□ PPP协议的基本需求:

- 检测连接状态:
 - 能够及时自动检测出链路是否处于正常工作状态。
- 最大传送单元:
 - 必须对每一种类型的点对点链路设置最大传送单元 MTU 的标准默认值, 促进各种实现之间的互操作性。
- 网络层地址协商:
 - 必须提供一种机制使通信的两个网络层实体能够通过协商知道或能够配置彼此的网络层地址。
- 数据压缩协商:
 - 必须提供一种方法来协商使用数据压缩算法。

2.点到点协议 (PPP)

2.1 PPP协议

- PPP协议的需要实现的功能：
 - 不支持可靠传输：
 - 可靠传输由TCP协议负责，PPP协议不需要进行纠错、不需要设置序号来解决传输错误、不需要进行流量控制。
 - PPP协议不支持多点线路，仅支持点对点链路通信，且只支持全双工通信。

2.点到点协议 (PPP)

2.1 PPP协议

□ PPP协议的组成三要素：

■ 封装方法：

- PPP协议要包含一个将IP数据报封装成串行链路的方法。
- 封装方法既要支持异步链路，也要支持同步链路。

■ 链路控制协议 (LCP, Link Control Protocol)：

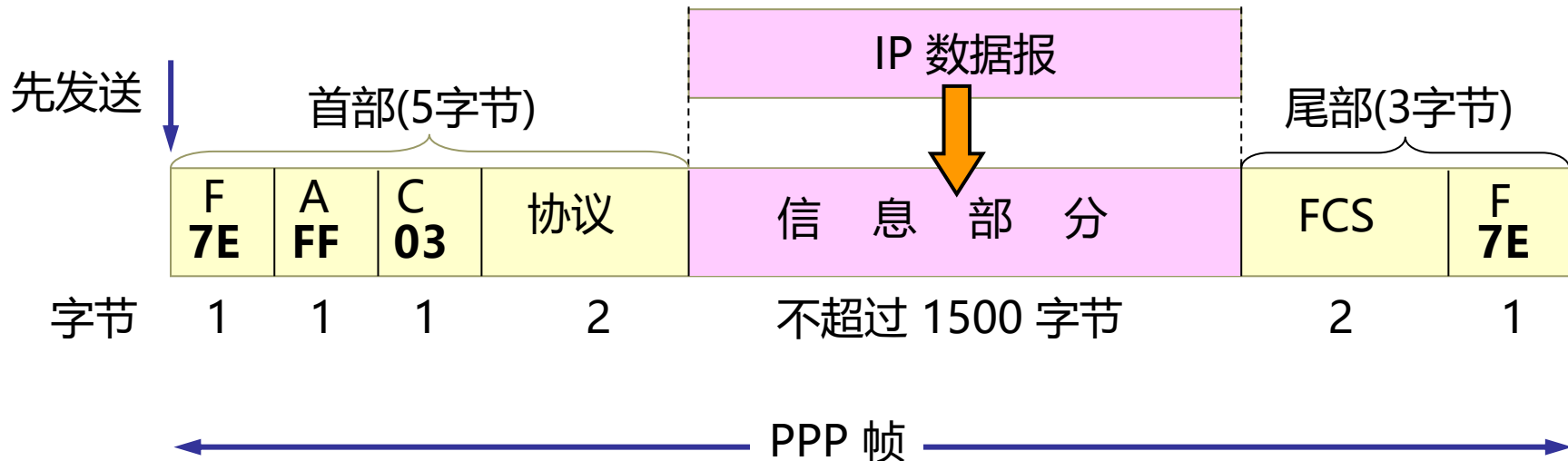
- 用来定义建立、配置和测试数据链路链接的协议。

■ 网络控制协议 (NCP, Network Control Protocol)：

- PPP包含了一套NCP，每个NCP协议支持不同的网络层协议。
- 例如IP、OSI的网络层、DECnet、AppleTalk等，实现向上兼容。

2.点到点协议 (PPP)

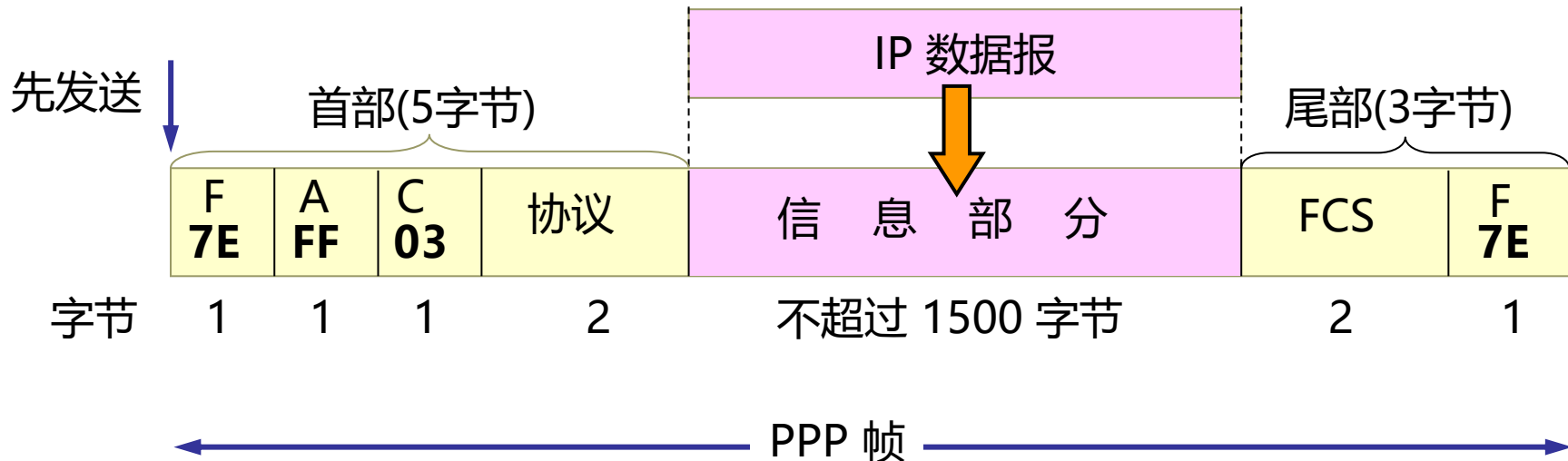
2.2 PPP协议的帧格式



- F字段:** 标志字段, 规定为0x7E (十六进制下的7E)
二进制表示为01111110, 表示PPP数据帧的开始和结束。
- A字段:** 规定为0xFF, 没有任何意义, 预留使用。
- C字段:** 规定为0x03, 没有任何意义, 预留使用。

2.点到点协议 (PPP)

2.2 PPP协议的帧格式



协议字段, 2字节:

0x0021: 信息字段为IP数据报

0xC021: 信息字段为PPP链路控制数据

0x8021: 信息字段为网络控制数据

2.点到点协议 (PPP)

2.2PPP协议的帧格式

异步传输：字节填充

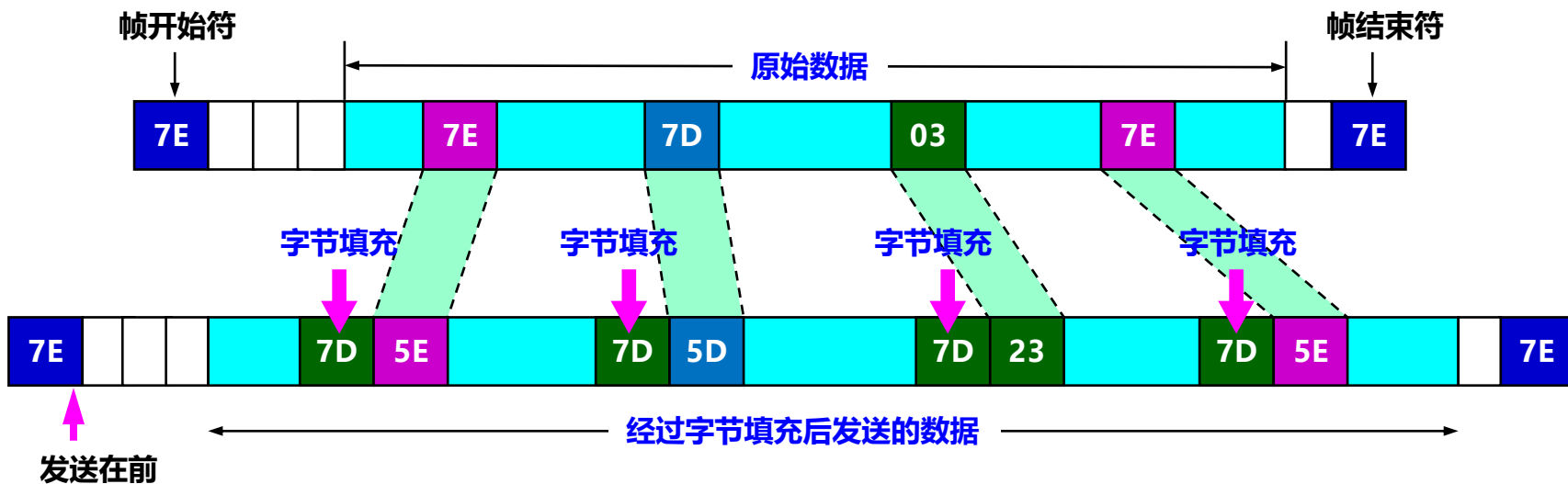
- 对信息字段内出现标志字段的组合时，使用转义字符进行字节填充。
- (0x7E) -> (0x7D,0x5E)
- (0x7D) -> (0x7D,0x5D)
- ASCII码的控制字符，即<(0x03)在字符前增加0x7D，并对该字符的编码加以改变。

同步传输：零比特填充

- 同步传输通过硬件或者软件扫描整个信息字段。
- 对于五个连续的1，在后面填入一个0，避免六个连续的1的出现。

2.点到点协议 (PPP)

2.2 PPP协议的帧格式



2.点到点协议 (PPP)

2.2 PPP协议的帧格式



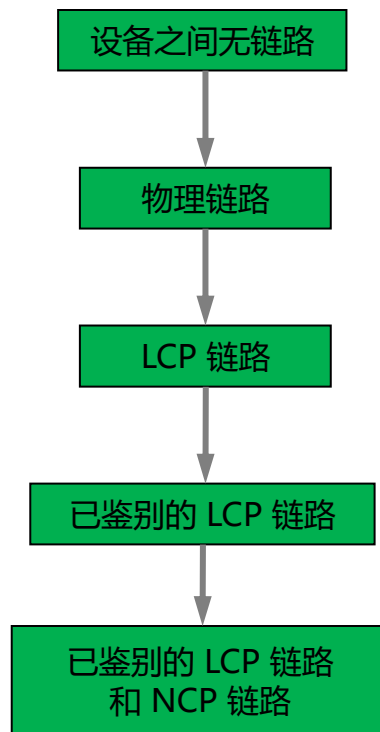
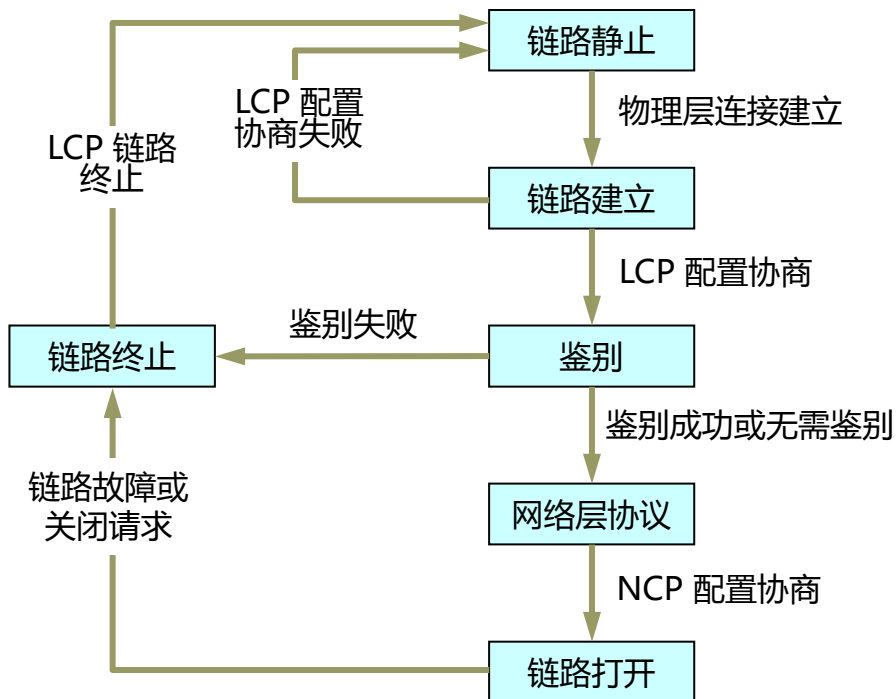
2.点到点协议 (PPP)

2.2 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。
 - LCP分组及其响应选择确定PPP参数，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。
 - 通信完毕后，首先NCP释放网络层连接，收回原来分配出去的 IP 地址，其次NCP释放数据链路层连接，最后释放物理层连接。

2.点到点协议 (PPP)

2.2 PPP协议的工作状态



3.使用广播信道的数据链路层

3.1局域网

- 局域网是20世纪70年代末发展起来的。
- 局域网技术在计算机网络中占有非常重要的意义，对于今天的国际互联网更是意义深远。
- 局域网最主要的特点是：
 - 网络为一个单位所拥有
 - 地理范围和站点数目均有限
- 学校、企业、政府、小区等的网络，都是局域网。

3.使用广播信道的数据链路层

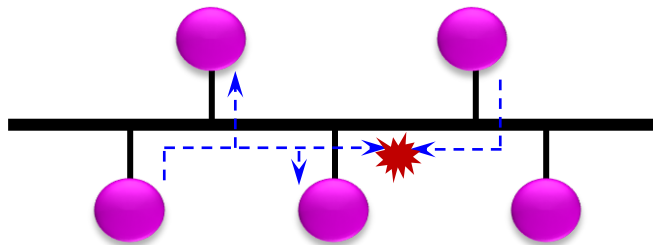
3.1局域网

- 局域网具有的主要优点：
 - 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。
- 局域网常使用的拓扑结构三种：
 - 星形拓扑、环形拓扑和总线拓扑。
- 局域网可以使用多种传输介质，以双绞线最为常见。

3.使用广播信道的数据链路层

3.1局域网

- 局域网的工作层次跨越了数据链路层和物理层。
 - 由于局域网中的大部分内容都在数据链路层，因此在数据链路层部分来讨论局域网技术。
- 局域网是一对多的通信，因此局域网内必须要共享信道。



3.使用广播信道的数据链路层

3.1局域网

- 共享信道要着重考虑的一个问题就是：
 - 如何使众多用户能够合理而方便的共享通信媒体资源。
- 共享信道的常用方法有两种：
 - 静态划分信道：物理层中介绍的频分复用、时分复用、波分复用和码分复用，主要用户分配了信道就不会和其他用户发生冲突。成本较高，不适合局域网使用。
 - 动态媒体接入控制：又称为**多点接入**。通常分为**随机接入**和**受控接入**。随机接入说的是用户随机发送数据，受控接入说的是用户服从一定的控制进行信息发送，例如令牌网。
- **局域网最为常用的共享信道为随机接入。**

3.使用广播信道的数据链路层

3.2以太网标准

- 以太网是一种基带总线局域网，其数据率经历了10Mbps、100Mbps、1000Mbps、10Gbps等。
 - 通常：
 - 标准以太网：10Mbps速率
 - 快速以太网：100Mbps速率
 - 千兆以太网：1000Mbps速率
 - 万兆以太网：10Gbps速率叫做
- 以太网有两种标准：DIX Ethernet V2和IEEE 802.3。
 - DIX是DEC、Xerox（美国施乐）、Intel（英特尔）三家公司的名称缩写。

3.使用广播信道的数据链路层

3.2以太网标准

- 诞生于1982年的DIX Ethernet V2是世界上第一个局域网产品（以太网）的规约，定义传送速率为10Mbps。
- IEEE 802委员会的802.3工作组在1983年制定了第一个IEEE的以太网标准，定义传送速率为10Mbps。
 - DIX Ethernet V2标准与IEEE 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。

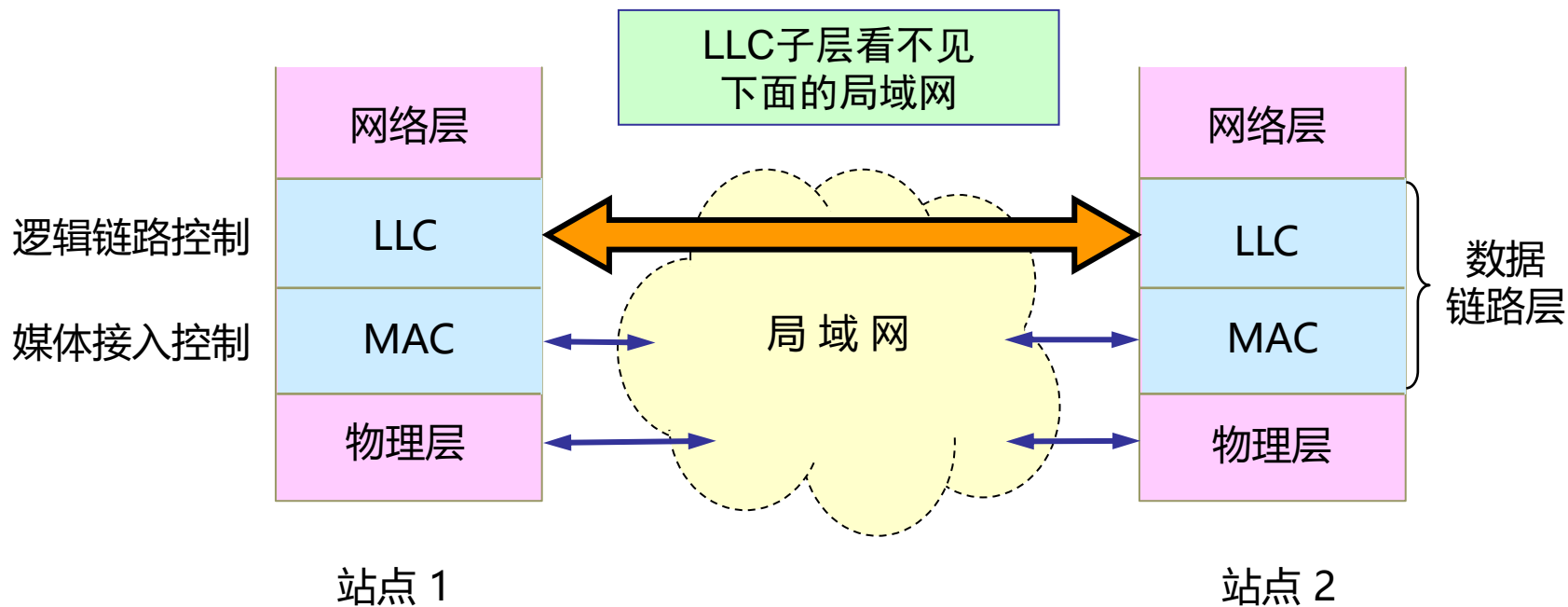
3.使用广播信道的数据链路层

3.2以太网标准

- 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。
- IEEE的做法，主要是由于商业的激烈竞争。

3.使用广播信道的数据链路层

3.2以太网标准



3.使用广播信道的数据链路层

3.2以太网标准

- 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。
 - 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。

3.使用广播信道的数据链路层

3.3网络适配器

- 计算机与局域网的连接是通过通信适配器（adapter）。
 - 适配器本来是在主机内的一块网络接口板，又称为网络接口卡NIC（Network Interface Card），俗称为网卡。



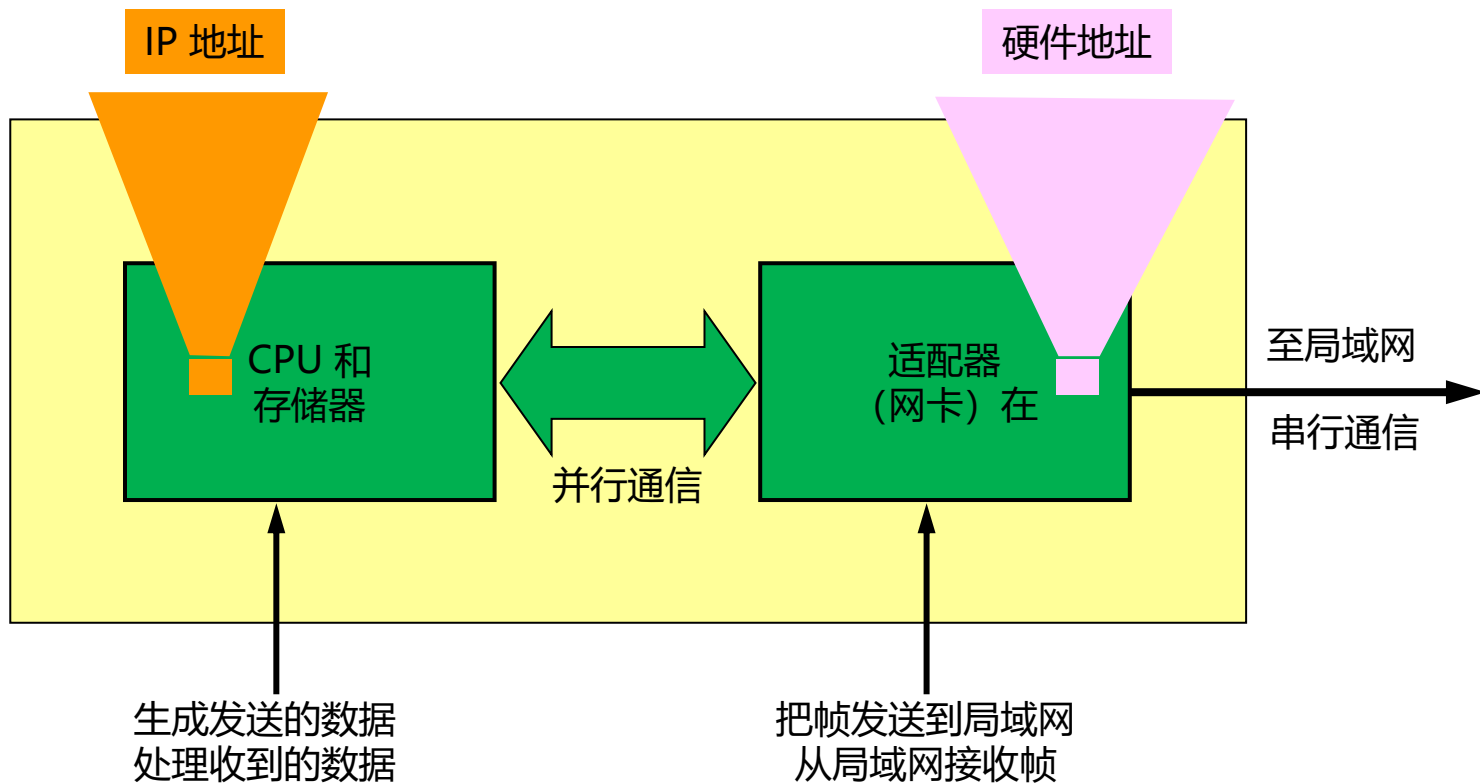
3.使用广播信道的数据链路层

3.3网络适配器

- 网络接口卡NIC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
 - 对帧进行计算

3.使用广播信道的数据链路层

3.3网络适配器



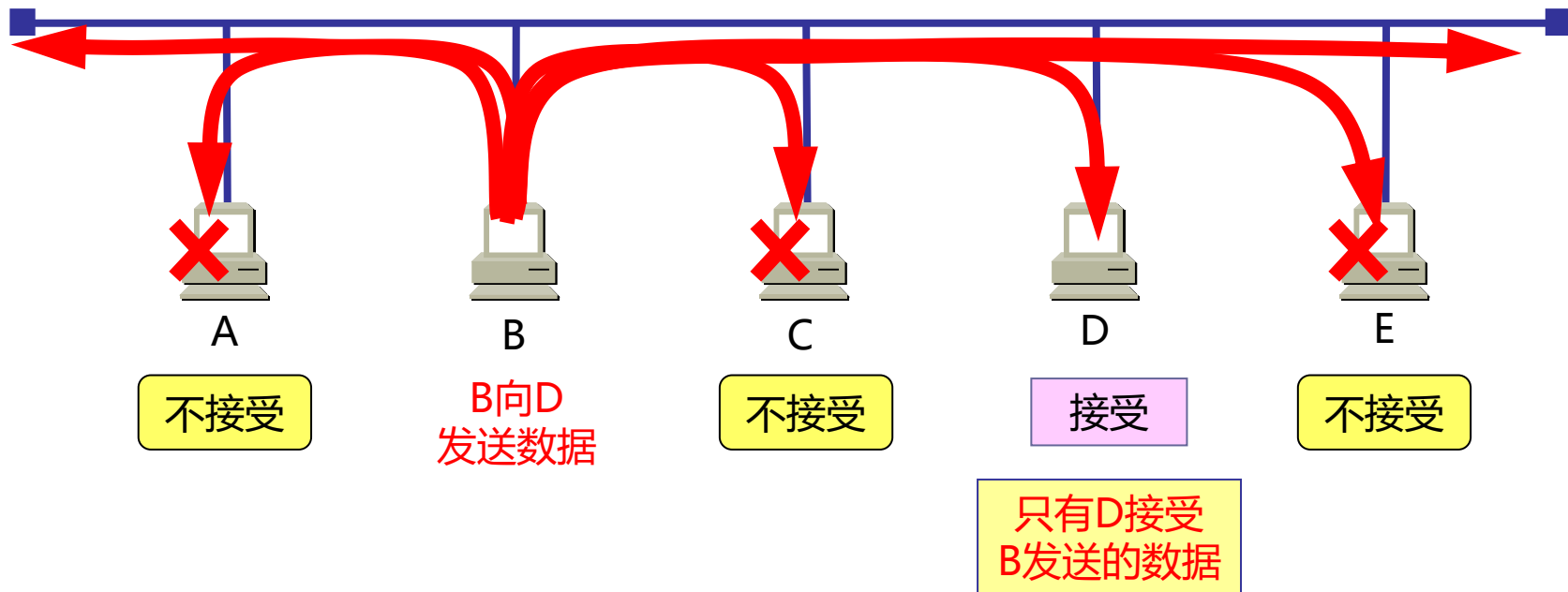
3.使用广播信道的数据链路层

3.4CSMA/CD

总线结构的以太网

匹配电阻 (用来吸收总线上传播的信号)

匹配电阻



3.使用广播信道的数据链路层

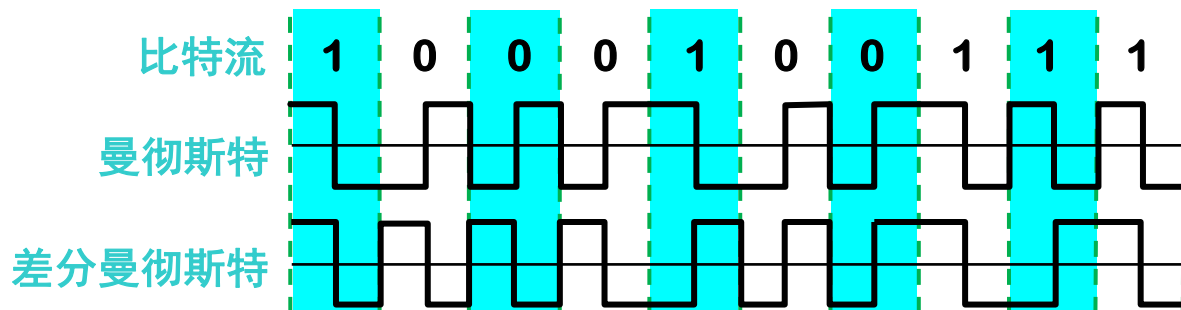
3.4CSMA/CD

- 为了通信的简便，以太网使用了两种重要的措施：
 - 措施一：无连接的工作方式
 - 不必建立连接就可以直接发送数据。
 - 网卡发送的数据帧不进行编号，也不要求进行确认。
 - 最大程度交付，但不保证可靠交付。
 - 对差错帧的是否重传由上层决定。
 - 使用CSMA/CD进行冲突协调。
 - 措施二：曼彻斯特编码

3.使用广播信道的数据链路层

3.4CSMA/CD

- 为了通信的简便，以太网使用了两种重要的措施：
 - 措施一：无连接的工作方式
 - 措施二：曼彻斯特编码
 - 特点：从高到低表示1，从低到高表示0。
 - 优点：方便进行信息的提出，解决了位同步的困难。
 - 缺点：所占的频带宽度比原始的基带信号增加了一倍。



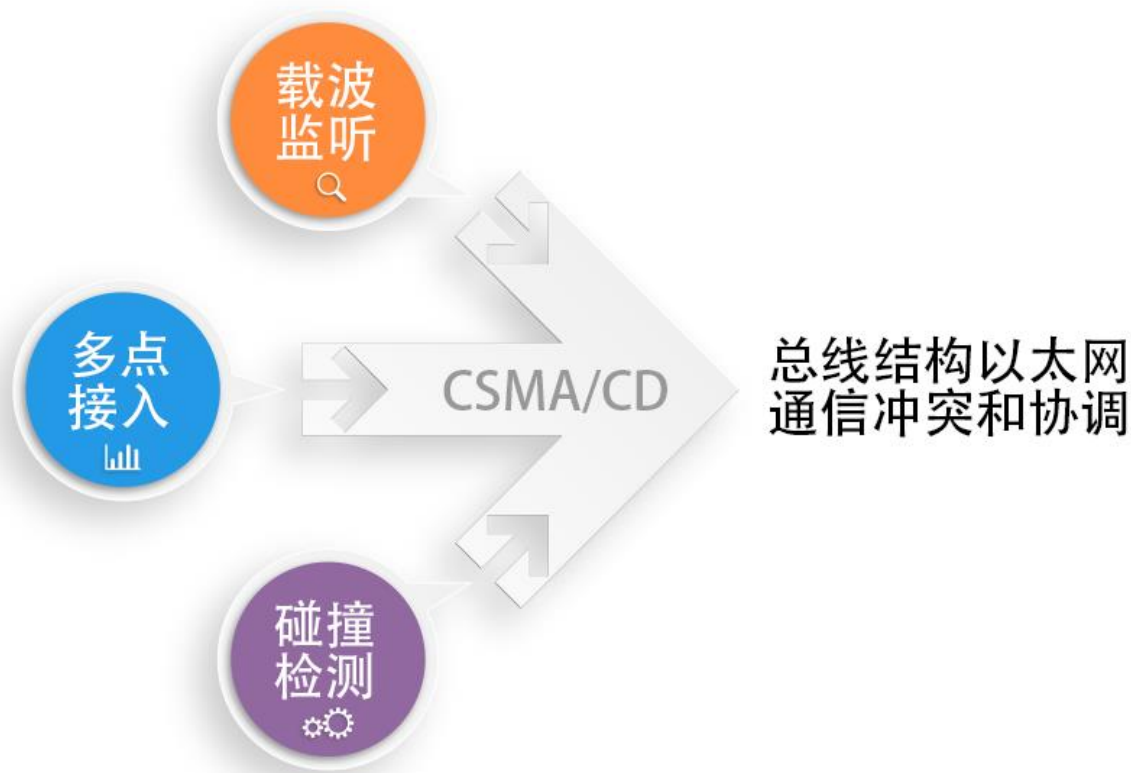
3.使用广播信道的数据链路层

3.4CSMA/CD

- CSMA/CD的意思是：**载波监听多点接入/碰撞检测**，Carrier Sense Multiple Access with Collision Detection。
 - CSMA/CD也被翻译为：载波侦听多路复用/碰撞检测。
 - 这是翻译的差异，两个说法是等同的。

3.使用广播信道的数据链路层

3.4CSMA/CD



3.使用广播信道的数据链路层

3.4CSMA/CD

- “多点接入”就说明这是总线型网络，许多计算机以多点接入的方式连接在一条总线上。
 - “多点接入”说明的是网络的结构，而不是协议的内容。
 - 协议的实质是“载波监听”和“碰撞检测”。

3.使用广播信道的数据链路层

3.4CSMA/CD

- “载波监听”就是用电子技术检测总线上有没有其他计算机也在发送。
 - “载波监听”就是检测信道。
 - 不管在发送前，还是在发送中，每个站都必须不停地检测信道。发送前检测信道是为了获得发送权。
 - 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。

3.使用广播信道的数据链路层

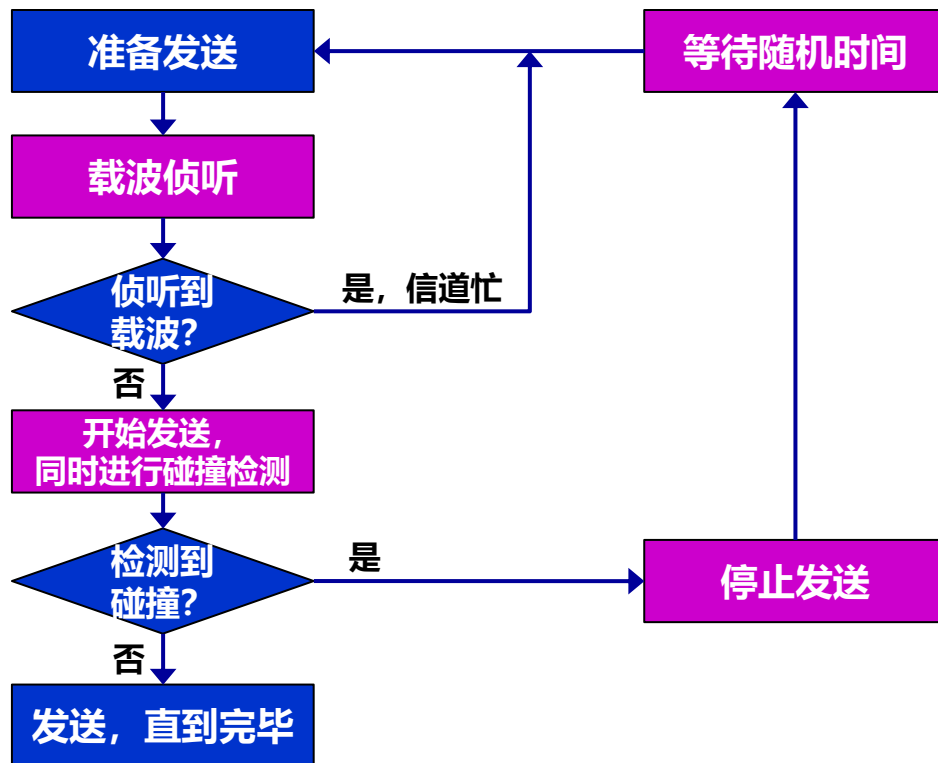
3.4CSMA/CD

- “碰撞检测”就是“边发送边监听”，即适配器边发送数据边检测信道上的信号电压的变化情况，以判断自己在发送数据的同时其他站是否也在发送数据。
 - 碰撞就是冲突，因此碰撞检测也被称为冲突检测。
 - 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。

3.使用广播信道的数据链路层

3.4CSMA/CD

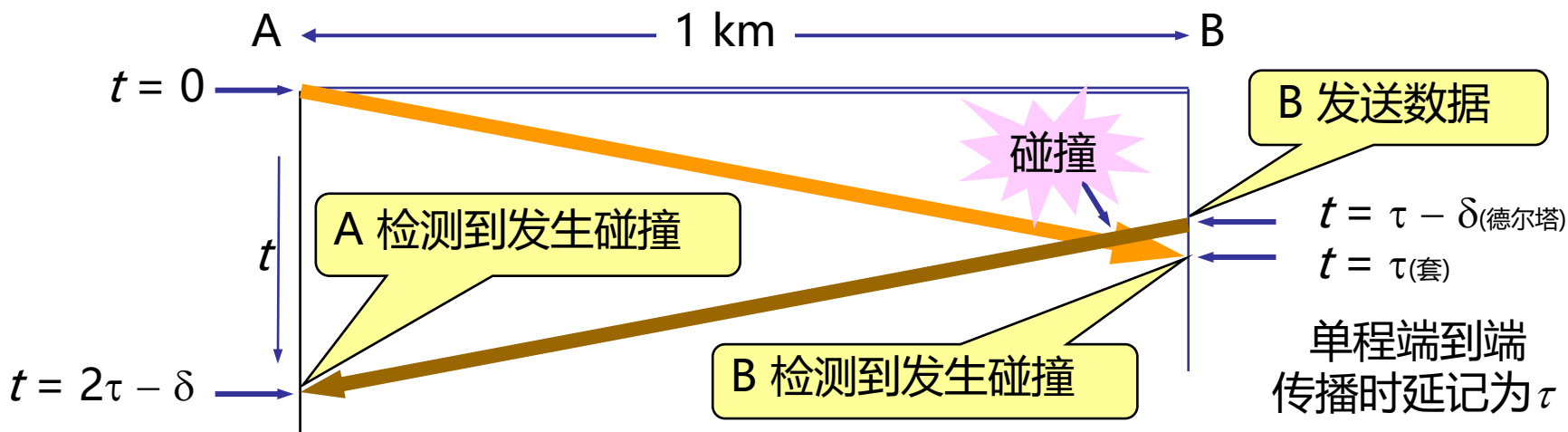
CSMA/CD 协议工作流程



3.使用广播信道的数据链路层

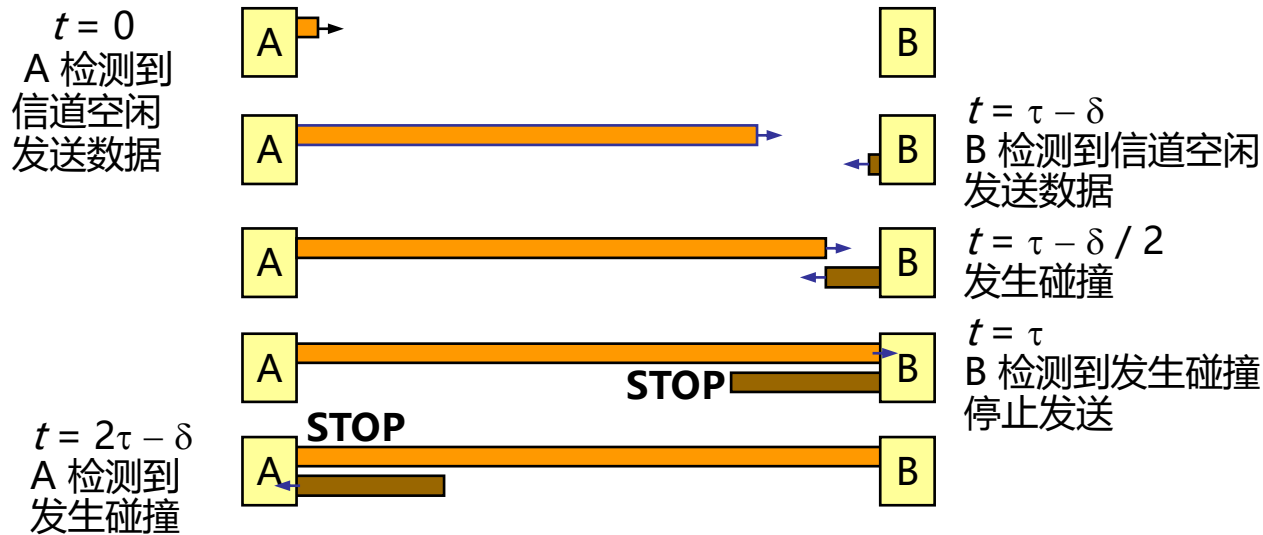
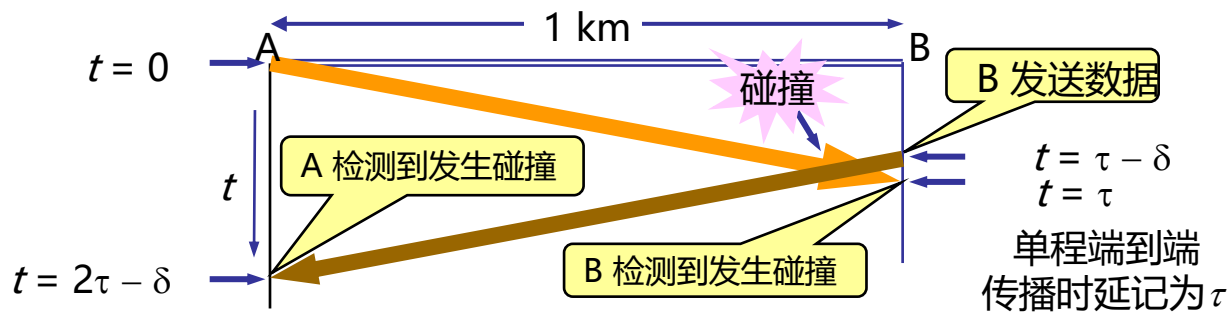
3.4CSMA/CD

- 尽管使用了“碰撞检测”机制，但是在以太网上还是会发生碰撞的情况。



3.使用广播信道的数据链路层

3.4CSMA/CD



3.使用广播信道的数据链路层

3.4CSMA/CD

- 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。

3.使用广播信道的数据链路层

3.4CSMA/CD

- 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。
- 以太网使用**二进制指数类型退避算法**来确定碰撞后重传的时机。
 - 基本退避时间取值为争用期 $2t$ 。
 - 从整数集合 $[0, 1, \dots, (2^k - 1)]$ 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： **$k = \text{Min}[\text{重传次数}, 10]$**
 - 当 k 不大于10时，参数 k 等于重传次数；当 k 大于10时，参数 k 等于10。
 - 当重传达16次仍不能成功时即丢弃该帧，并向高层报告。

3.使用广播信道的数据链路层

3.4CSMA/CD

- 以太网取 $51.2\mu\text{s}$ 为争用期的长度。 (这是经验值)
- 对于10Mbps以太网，在争用期内可发送512bit，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。
 - 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。

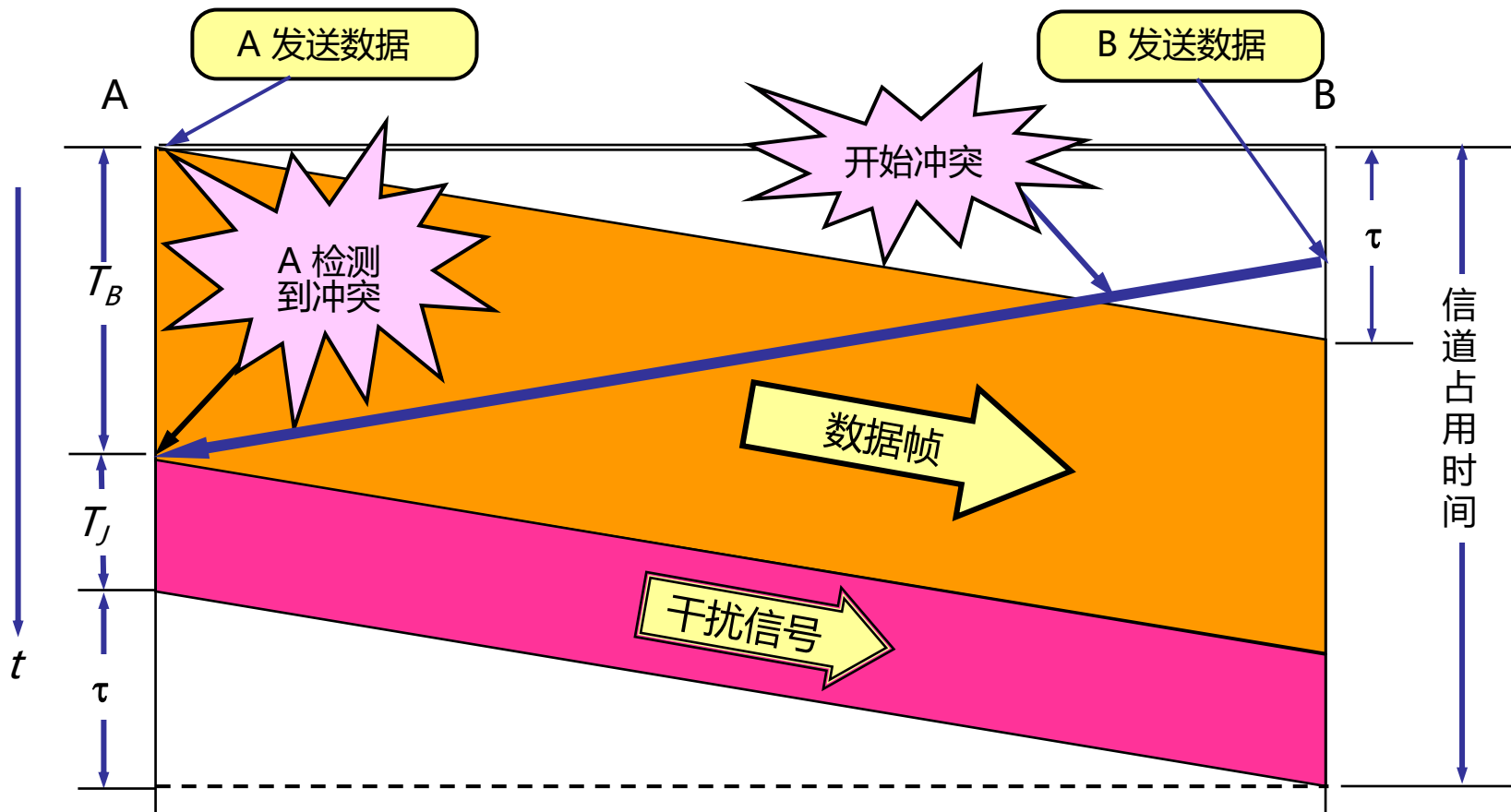
3.使用广播信道的数据链路层

3.4CSMA/CD

- 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；
 - 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。
- 这样的机制叫做“**强化碰撞**”。

3.使用广播信道的数据链路层

3.4CSMA/CD



3.使用广播信道的数据链路层

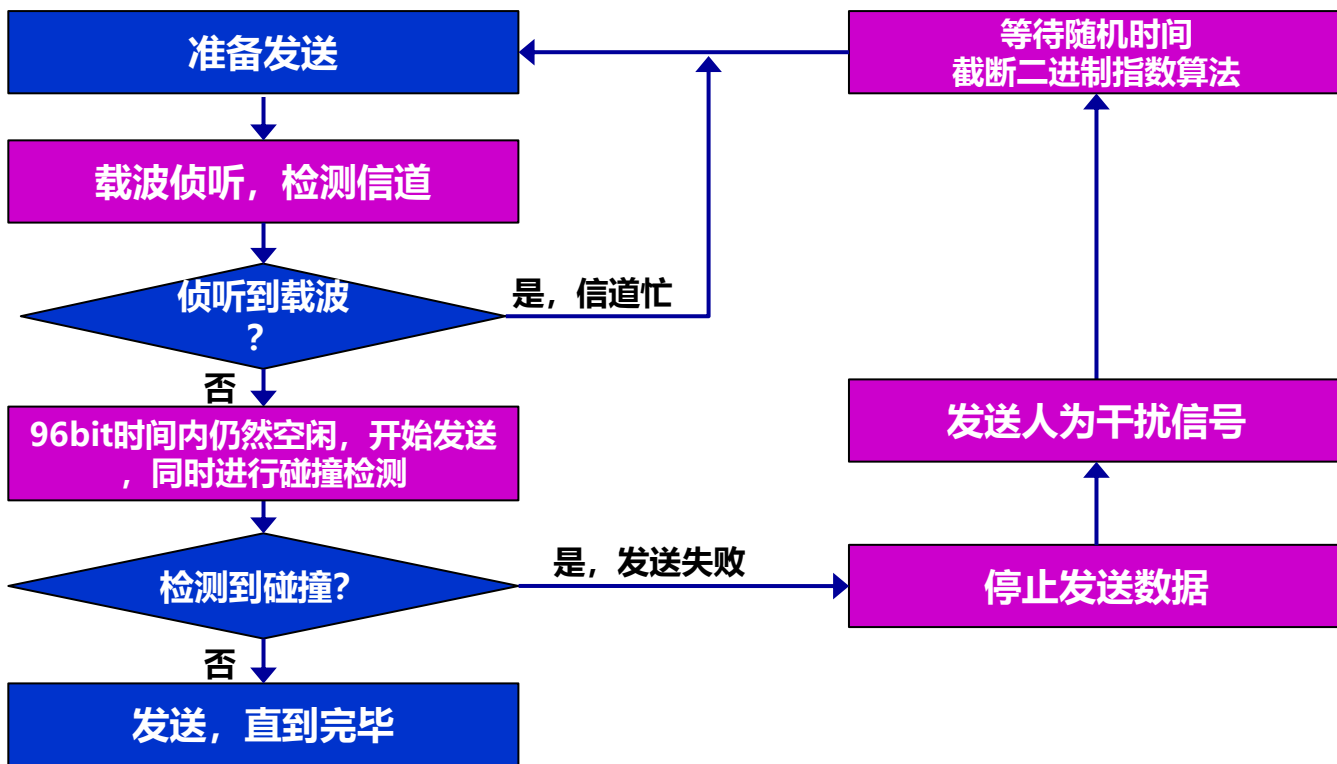
3.4CSMA/CD

- 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。

3.使用广播信道的数据链路层

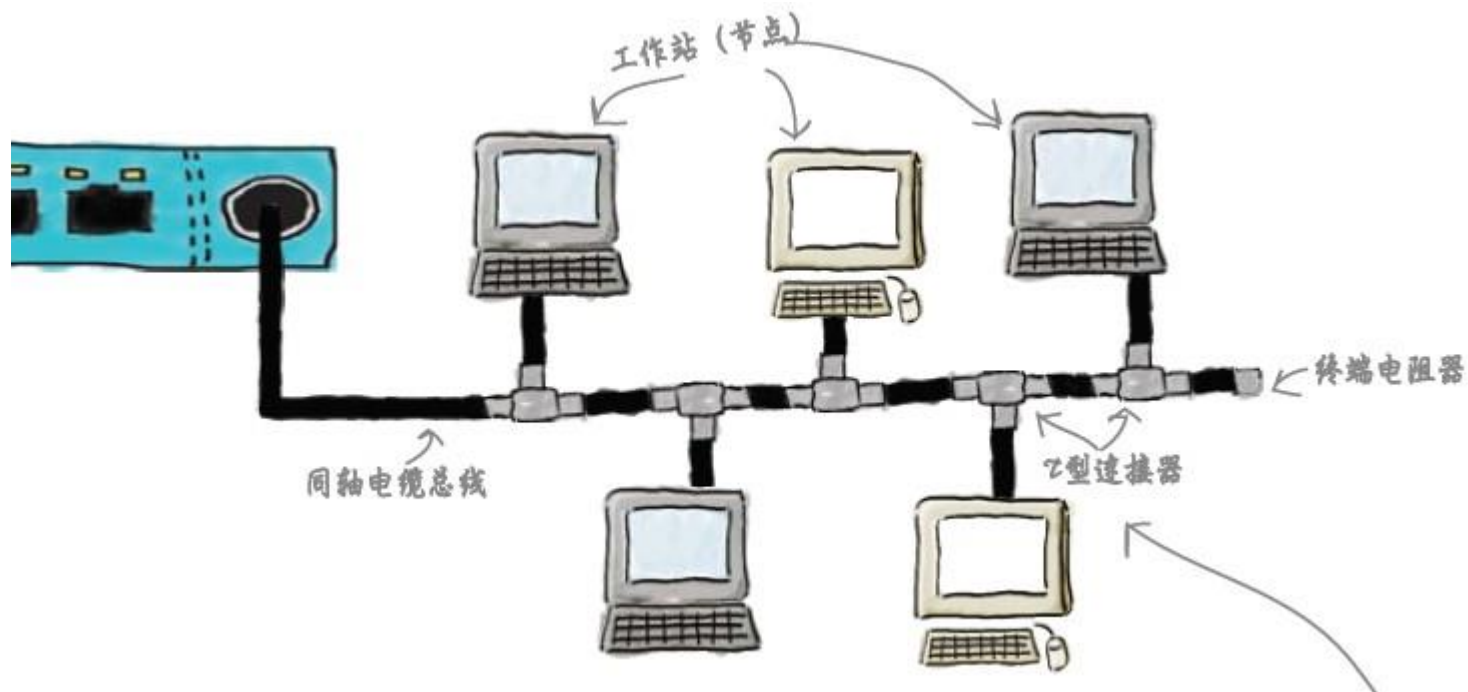
3.4CSMA/CD

CSMA/CD 协议的要点



4.使用广播信道的以太网

4.1使用集线器的以太网



4.使用广播信道的以太网

4.1使用集线器的以太网

- 标准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。
- 使用双绞线的标准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)**。

4.使用广播信道的以太网

4.1使用集线器的以太网



4.使用广播信道的以太网

4.1使用集线器的以太网

- 1990年，IEEE制定出星形以太网10BASE-T的标准802.3i。
 - 10 表示速率是10Mbps
 - BASE 表示连接线上的信号是基带信号
 - T 表示通过双绞线进行通信

- 由于使用双绞线和集线器的网络，其可靠性和易用性大大提升，并且使用双绞线电缆的以太网价格便宜，因此粗同轴电缆和细同轴电缆的以太网就逐步从市场上消失了。

4.使用广播信道的以太网

4.1使用集线器的以太网

- 10BASE-T 以太网中，每个站都需要用到两对无屏蔽双绞线，分别用于发送和接收。
- 10BASE-T 以太网的通信距离稍短，每个站到集线器的距离不超过100m。
 - 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。

4.使用广播信道的以太网

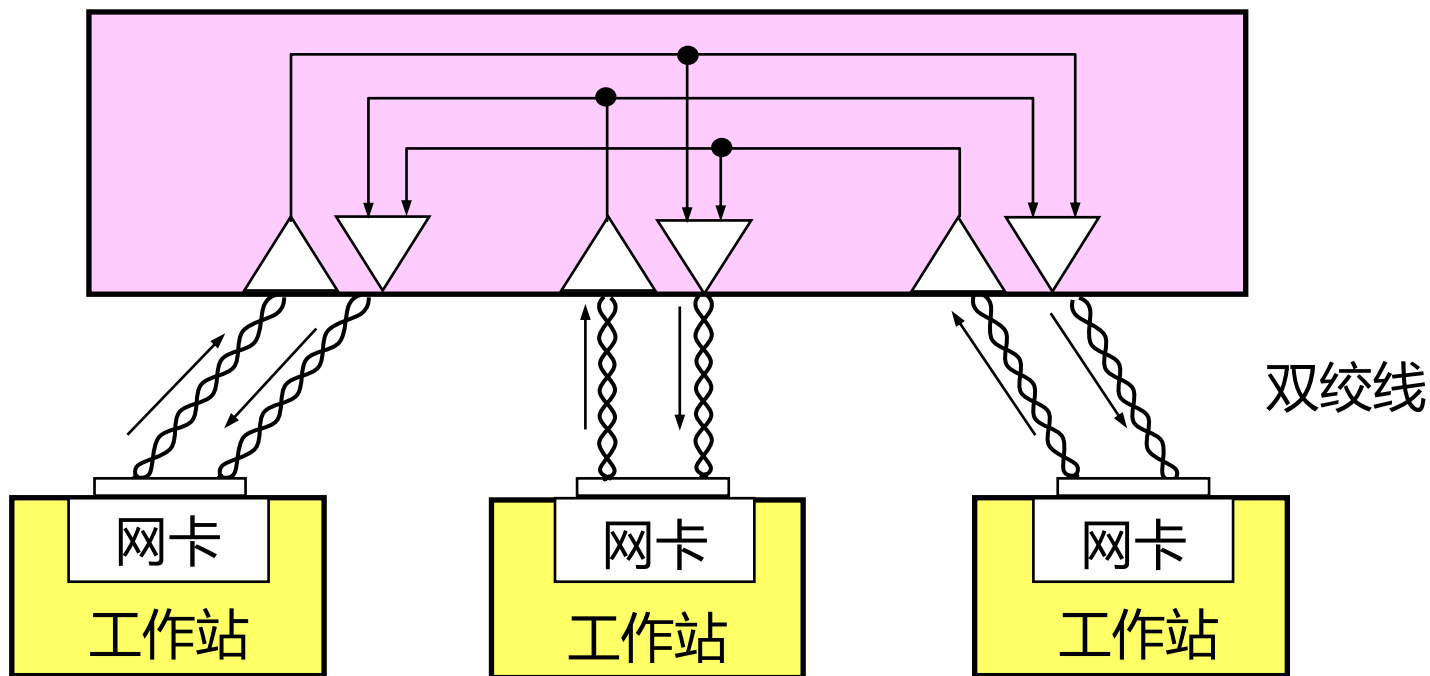
4.1使用集线器的以太网

- 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。
- 集线器采用了专门的芯片，进行自适应串音回波抵消，减少了近端串音。

4.使用广播信道的以太网

4.1使用集线器的以太网

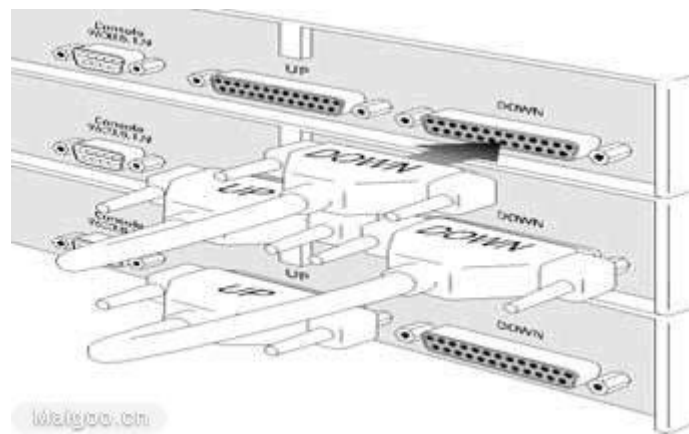
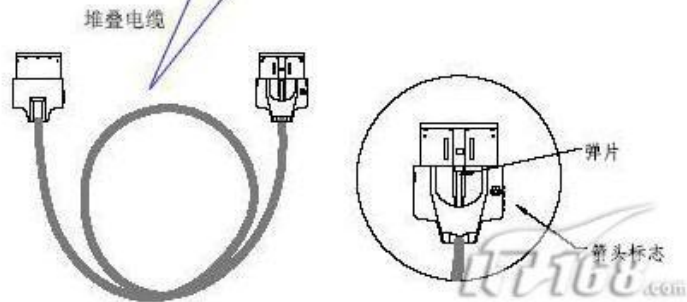
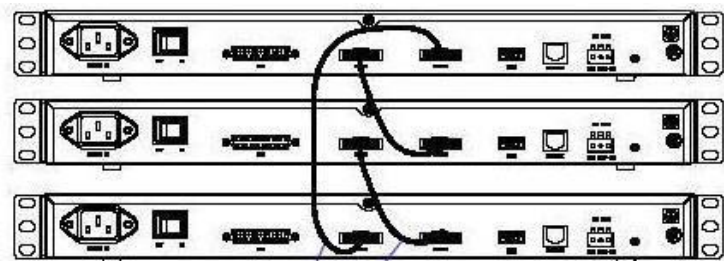
具有三个接口的集线器的逻辑结构



4.使用广播信道的以太网

4.1使用集线器的以太网

集线器的堆叠



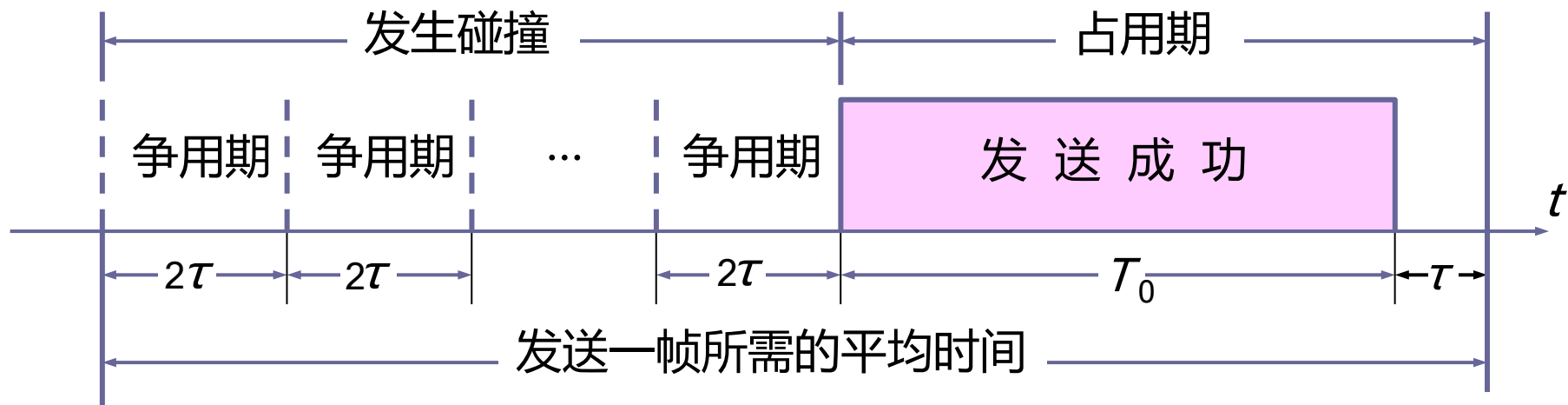
4.使用广播信道的以太网

4.2以太网信道利用率

- 以太网信道被占用的场景：
 - 发送数据的时间。
 - 发送数据结束后，等待最后一个bit传送到接收方的等待时间。
 - 发生碰撞的争用期。
 - 没有数据传送的空闲时间。
- 发送数据的时间在总时间的占用率，就是通俗意义上的信道利用率。

4.使用广播信道的以太网

4.2以太网信道利用率



- 一个站在发送帧时出现了碰撞。经过一个争用期 2τ 后，可能又出现了碰撞。
- 这样经过若干个争用期后，一个站发送成功了。
- 假定发送帧需要的时间是 T_0 。

4.使用广播信道的以太网

4.2以太网信道利用率

- 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：

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

- $a \rightarrow 0$ 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。

4.使用广播信道的以太网

4.2以太网信道利用率

- 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：

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

- 以太网的参数 a 的值应该尽量小一些。
- 以太网的连线长度受到了限制，线缆越长则 a 的值越大。
- 以太网的帧不能够太短，数据帧越大则 a 的值越小。

4.使用广播信道的以太网

4.2以太网信道利用率

- 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，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 。
 - 可计算出理想情况下的极限信道利用率 S_{\max} 为：

$$S_{\max} = \frac{T_0}{T_0 + \tau} = \frac{1}{1 + a}$$

4.使用广播信道的以太网

4.2以太网信道利用率

□ 极限信道利用率 S_{\max} 为:

$$S_{\max} = \frac{T_0}{T_0 + \tau} = \frac{1}{1 + a}$$

■ 说明:

- 只有当参数a远小于1才能得到尽可能高的极限信道利用率。
- 若参数a远大于1, 则极限信道利用率就远小于1, 而此时的实际信道利用率就更低。

4.使用广播信道的以太网

4.3以太网的MAC层

- 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。
 - 大家都早已习惯了将这种48位的“名字”称为“地址”，所以我们也采用这种习惯用法，尽管这种说法并不太严格。

4.使用广播信道的以太网

4.3以太网的MAC层

- MAC地址就是计算机系统中的一种标识系统 (identification System) 。
- 著名文献【SHOC78】对标识系统的定义：
 - 名字指出我们所要寻找的那个资源。
 - 地址指出那个资源在何处。
 - 路由告诉我们如何到达该处。

4.使用广播信道的以太网

4.3以太网的MAC层

- MAC层讨论两个问题：



4.使用广播信道的以太网

4.3以太网的MAC层

□ MAC地址:

- IEEE 802规定, MAC地址字段可以使用6字节(48位)或者2字节(16位)两种表示方式。
- 随着以太网的迅猛发展, 6字节的表示方法可以使得全世界所有的局域网都使用不相同的地址, 因此目前主要使用6字节的方式来表示MAC地址。

4.使用广播信道的以太网

4.3以太网的MAC层

□ 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。

4.使用广播信道的以太网

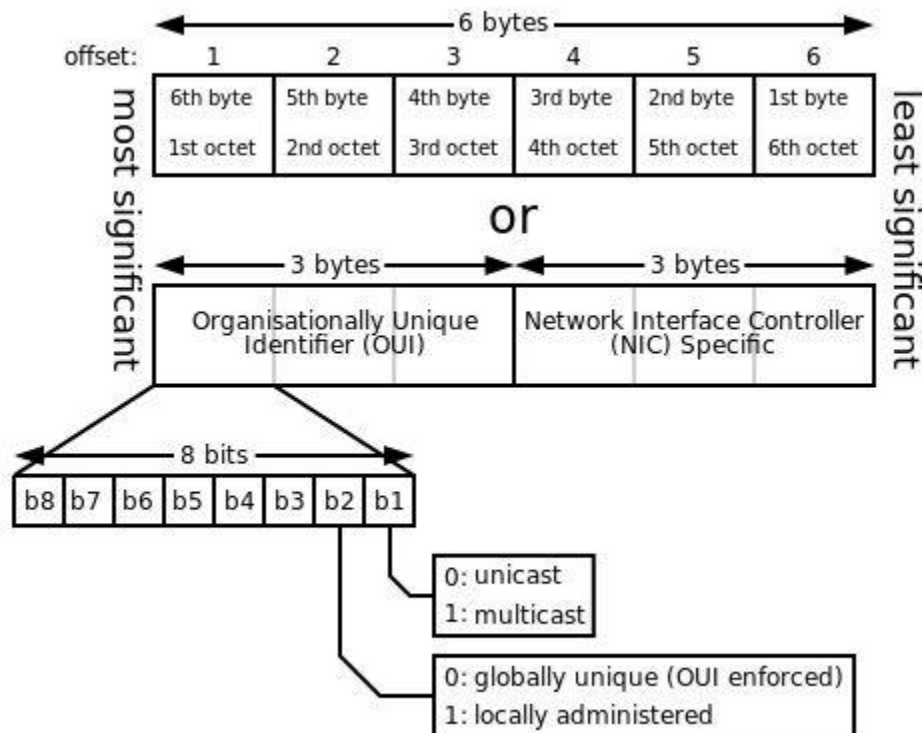
4.3以太网的MAC层

□ MAC地址:

- IEEE规定地址字段的**第一个字节的最低位**为I/G位。
 - I/G位为0, 表示一个单个站地址。
 - I/G位为1, 表示组地址, 用来进行多播。
- IEEE规定地址字段的**第一个字节的最低第二位**为G/L位。
 - G/L位为0, 表示地址是全球管理, 在全球没有相同的。
 - G/L位为1, 表示地址是本地管理, 用户可以任意分配网络上的地址。

4.使用广播信道的以太网

4.3以太网的MAC层



4.使用广播信道的以太网

4.3以太网的MAC层

- 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。
 - 如果不是就将此帧丢弃，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）

4.使用广播信道的以太网

4.3以太网的MAC层

- 以太网适配器还可以设置为一种特殊的工作方式，即“**混杂方式 (promiscuous mode)**”。
 - 工作在混杂方式的网卡只要“听到”有帧在以太网上传输就全部记录下来，而不管是发往哪里数据帧。
 - 如果网络上有混杂方式工作的网卡，那么就可以通过数据分析的方式窃取别人的信息。
 - 网络管理人员也可以通过此方式来监听和分析以太网的流量，已实现可量化的网络管理。
 - 最为常见的是“**嗅探器**”。

4.使用广播信道的以太网

4.3以太网的MAC层

□ 现场演示并讨论：

- 如何查看网卡的MAC地址呢？

分别演示Windows、Linux、UNIX、MAC OS X、手机的MAC地址查看方法。

- 可否自行修改网卡的MAC地址呢？

4.使用广播信道的以太网

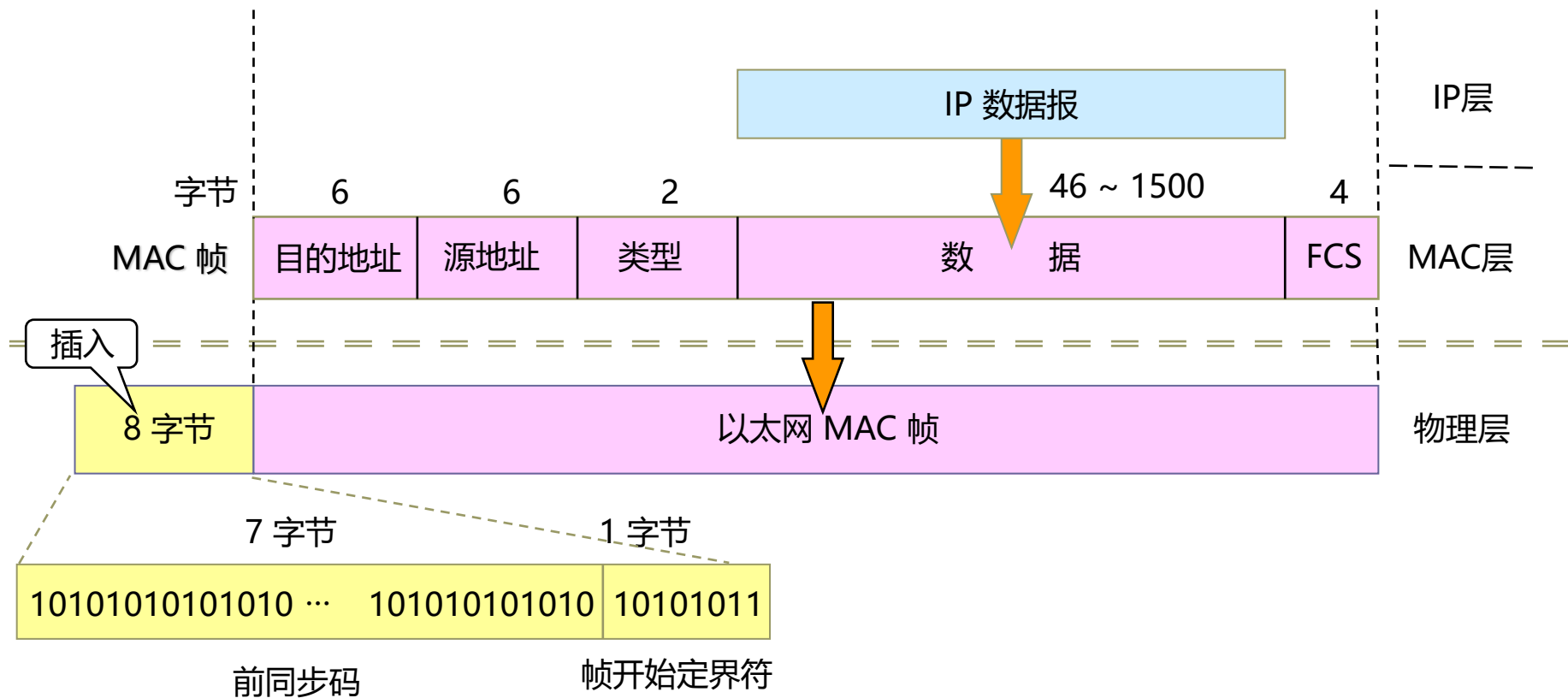
4.3以太网的MAC层

□ MAC帧格式:

- 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DIX Ethernet V2 的格式。

4.使用广播信道的以太网

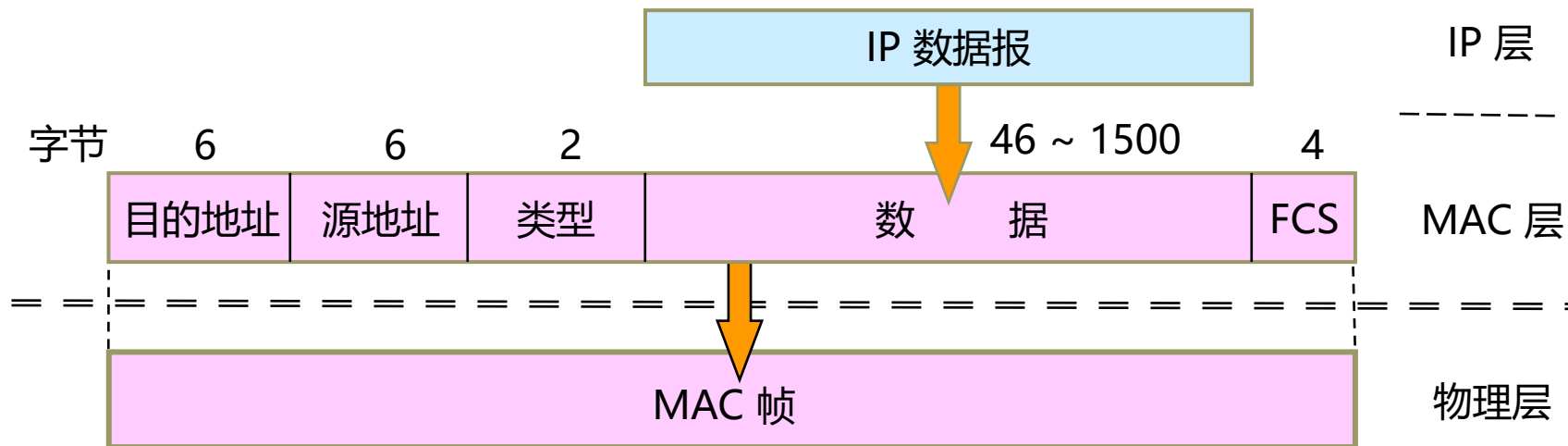
4.3以太网的MAC层



4.使用广播信道的以太网

4.3以太网的MAC层

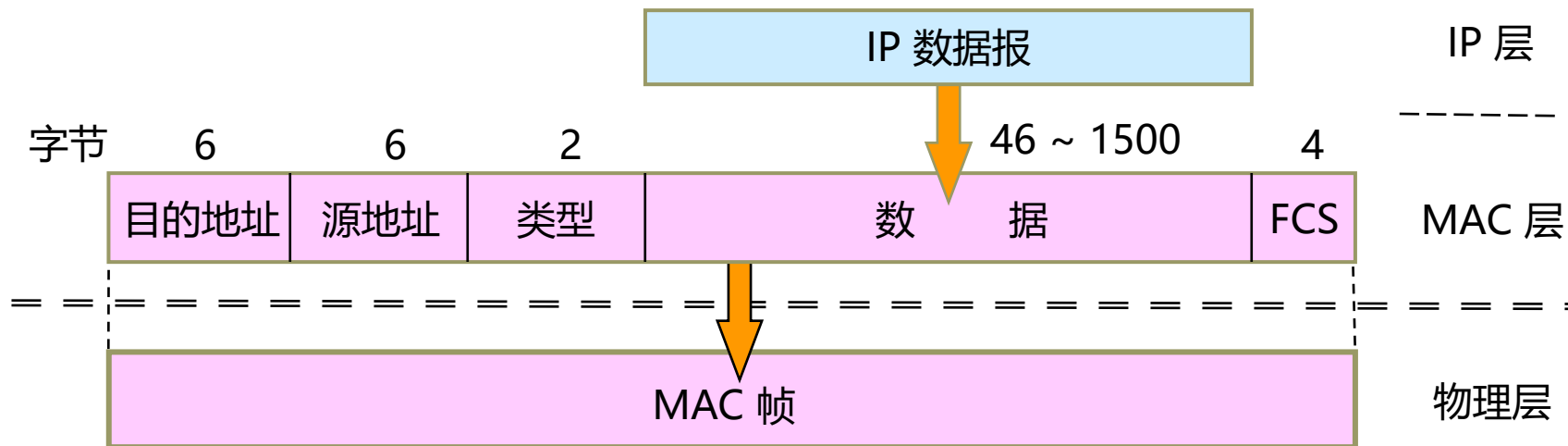
- 目的地址：6字节，记录数据帧的接收方的MAC地址。
- 源地址：6字节，记录数据帧的发送方的MAC地址。
- 类型：2字节，标志上一层使用的协议。例如0x8137表示IPX报文，0x0800表示IP数据报。



4.使用广播信道的以太网

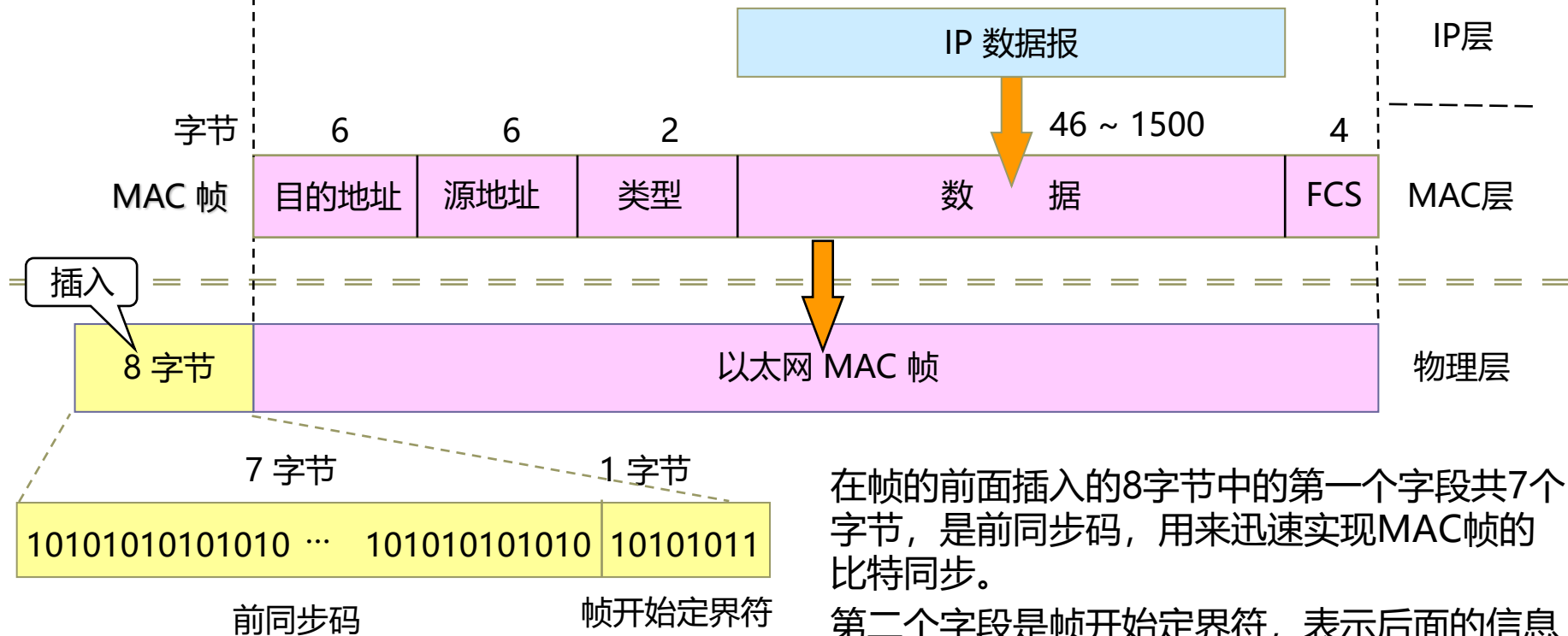
4.3以太网的MAC层

- 数据字段：46-1500字节之间。
- FCS：帧检验序列FCS，用于CRC校验。



4.使用广播信道的以太网

4.3以太网的MAC层



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。

4.使用广播信道的以太网

4.3以太网的MAC层

□ MAC帧格式:

- IEEE 802规定凡出现下述情况, 均为无效帧:
 - 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。
 - 数据字段长度不在46~1500字节区间, 即MAC帧长度不在64~1518字节区间。
- 对于检查出的无效MAC帧, 直接简单**丢弃**。

4.使用广播信道的以太网

4.3以太网的MAC层

□ 现场演示并讨论：

- 安装并使用嗅探器软件Wireshark。
- 使用Wireshark软件抓取数据包，分析数据帧结构。



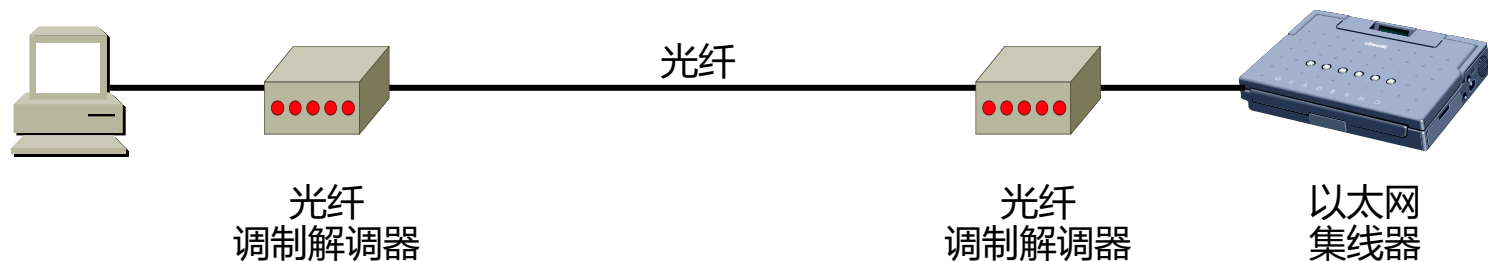
5.扩展的以太网

- 以太网技术的发展迅速，但是以太网技术存在**传输距离和规模**的限制，因此希望能够把以太网进行扩展。
- 扩展主要在两个层面：
 - 物理层扩展：支持更远距离的网络。
 - 数据链路层扩展：支持更大规模的网络。
- 扩展的主要手段：
 - 集线器/HUB：物理层扩展。
 - 网桥/交换机：数据链路层扩展。

5.扩展的以太网

5.1在物理层扩展以太网

- 以太网上的主机之间距离不能够太远，否则CSMA/CD会因为衰减而无法正常工作。
 - 10BASE-T以太网的两个主机之间的距离不得超过200m。
- 扩展主机和集线器之间距离的简单方法是使用光纤。
 - 通过光纤调制解调器，将传输的距离扩展。



5.扩展的以太网

5.1在物理层扩展以太网



5.扩展的以太网

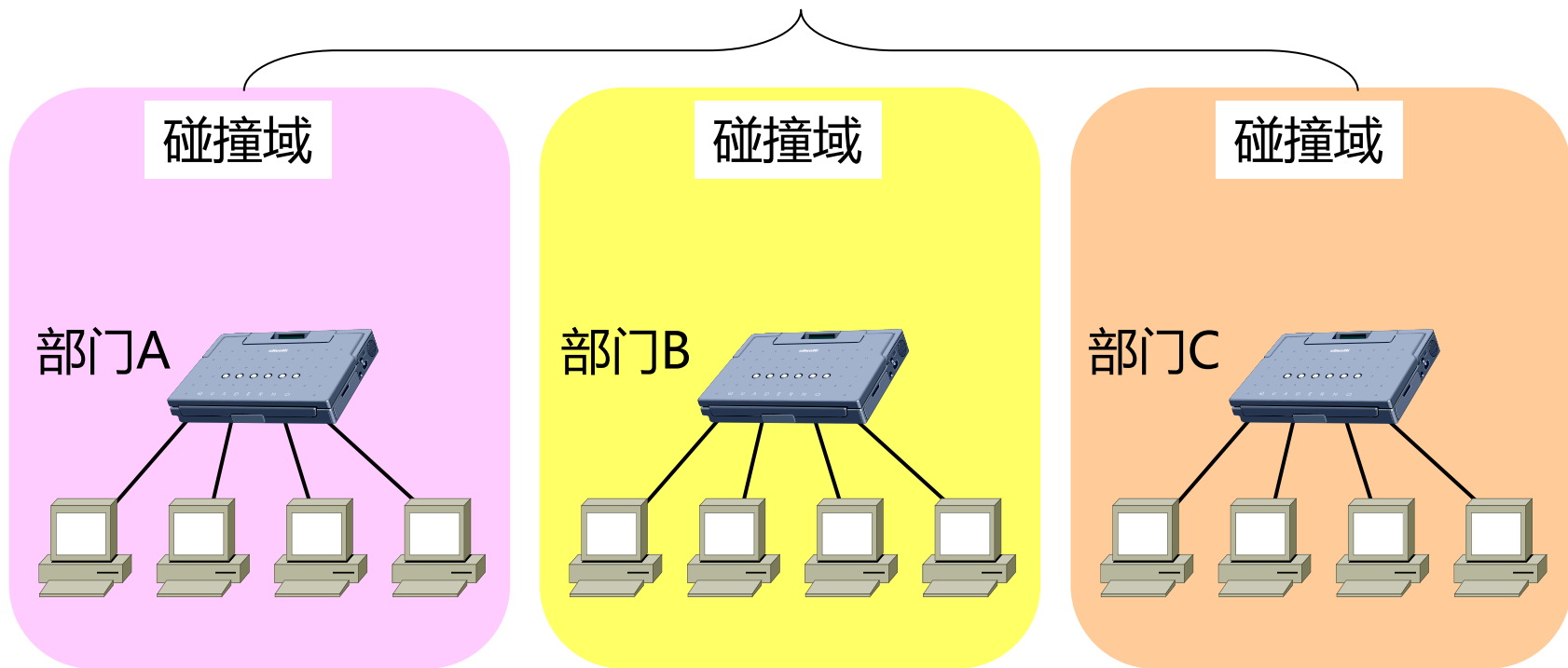
5.1在物理层扩展以太网

- 某单位有三个部门，每个部门都有一个局域网。
 - 通过主干集线器将三个部门的局域网连接起来，就变成了一个大的局域网。
 - 通过多级结构的集线器，将以太网进行了扩展。例如部门局域网的主机到集线器的距离为100m，主机间最远距离为200m，通过扩展后，最远主机间的距离为300m。
 - 扩展之后，三个局域网变为一个更大的局域网，扩展后的以太网是一个碰撞域（collision domain，冲突域），同时只能够有一个主机发送数据，传输效率降低了。

5.扩展的以太网

5.1在物理层扩展以太网

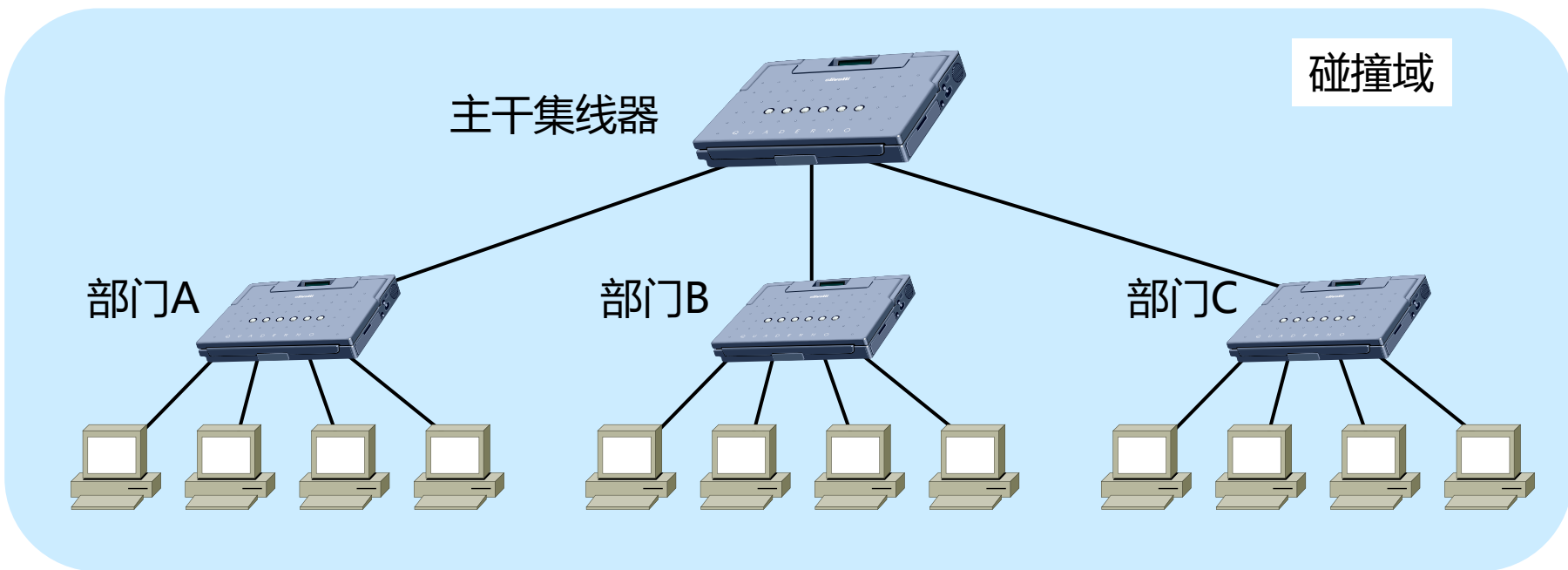
三个独立的碰撞域



5.扩展的以太网

5.1在物理层扩展以太网

一个更大的碰撞域



5.扩展的以太网

5.1在物理层扩展以太网

- 在物理层通过集线器扩展局域网的**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通信。
 - 扩大了局域网覆盖的地理范围。
- 在物理层通过集线器扩展局域网的**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。

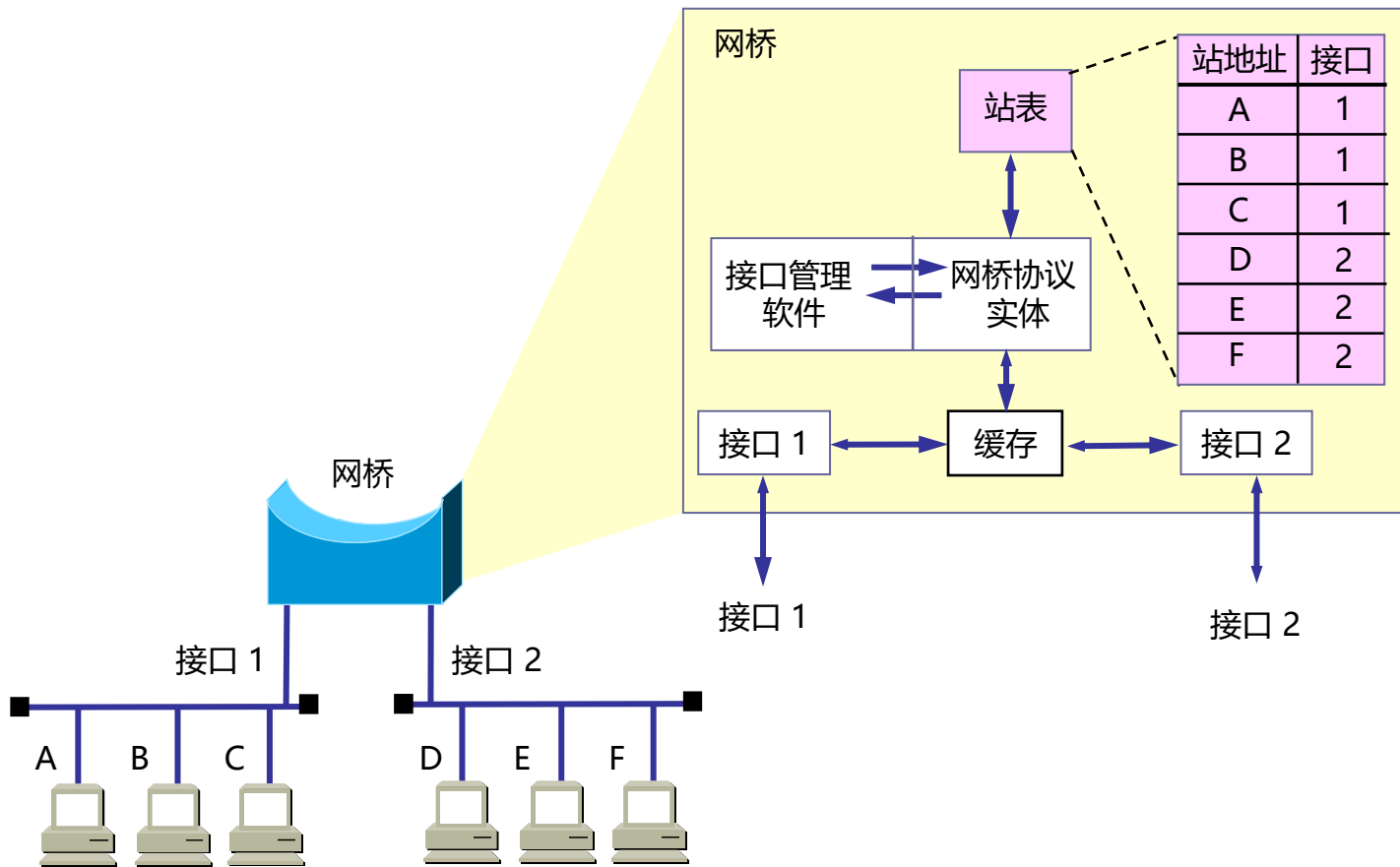
5.扩展的以太网

5.2在数据链路层扩展以太网

- 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，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，然后确定将该帧转发到哪个接口。

5.扩展的以太网

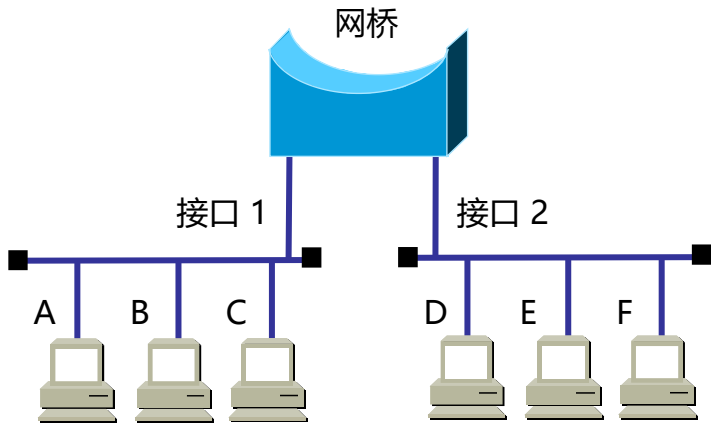
5.2在数据链路层扩展以太网



5.扩展的以太网

5.2在数据链路层扩展以太网

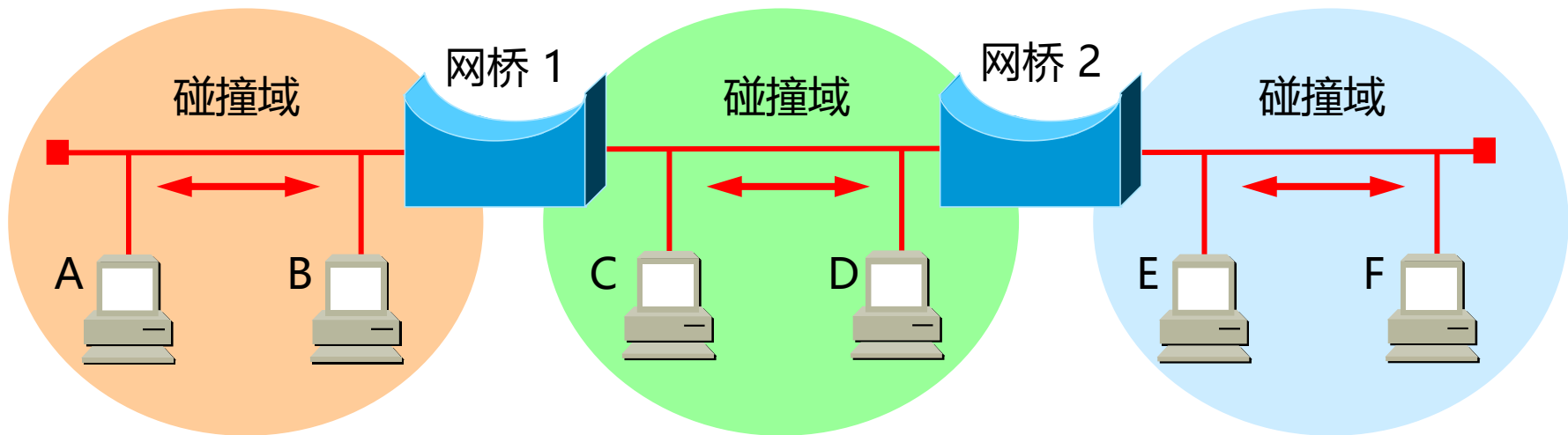
- 最简单的网桥有两个接口，复杂的网桥可以有更多的接口。
- 两个以太网通过网桥连接起来后，就成为了一个覆盖更为广泛的以太网，而原来的每个以太网都是一个网段(segment)。
 - 下图接口1、接口2分别接到的就是一个网段。



5. 扩展的以太网

5.2 在数据链路层扩展以太网

- 网桥使各网段成为隔离的碰撞域。



5.扩展的以太网

5.2在数据链路层扩展以太网

□ 使用网桥的优点：

- 过滤通信量，增大吞吐量。
- 扩大了以太网上的物理范围和主机数量。
- 提高了可靠性，让网段故障不影响全局。
- 可互联不同物理层、不同MAC子层和不同速率。

5.扩展的以太网

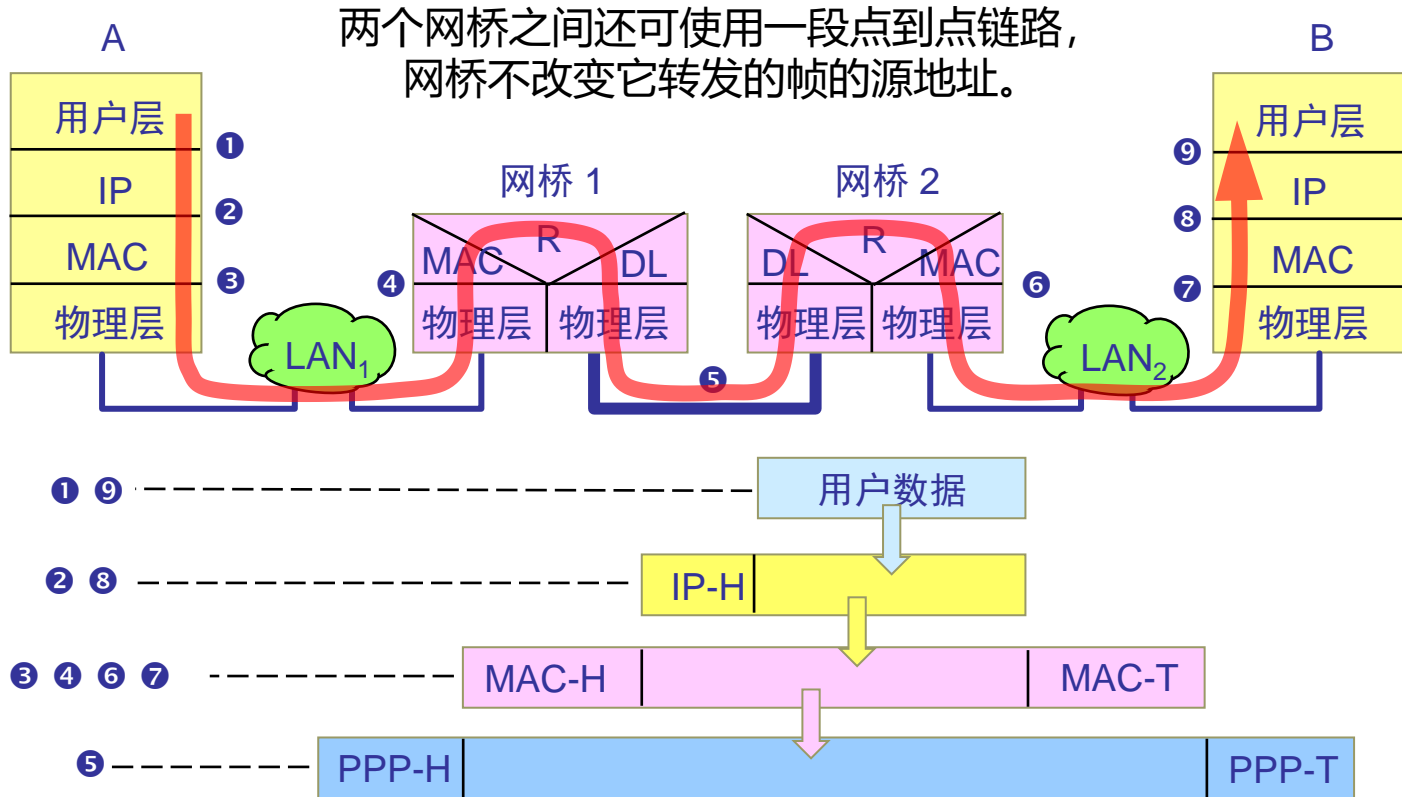
5.2在数据链路层扩展以太网

□ 使用网桥的缺点:

- 网桥对接收的帧要先存储和查找转发表, 然后才转发, 增加了时延。
- 在MAC子层并没有流量控制功能, 繁忙时提升了帧丢失的概率。
- 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。

5.扩展的以太网

5.2在数据链路层扩展以太网



5.扩展的以太网

5.2在数据链路层扩展以太网

- 常见的网桥有：
 - 透明网桥 (transparent bridge)
 - 源路由网桥 (source route bridge)
 - 多接口网桥 (switching hub)
 - 又叫做交换式集线器, 以太网交换机



5.扩展的以太网

5.2在数据链路层扩展以太网

□ 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。
- 透明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。

5.扩展的以太网

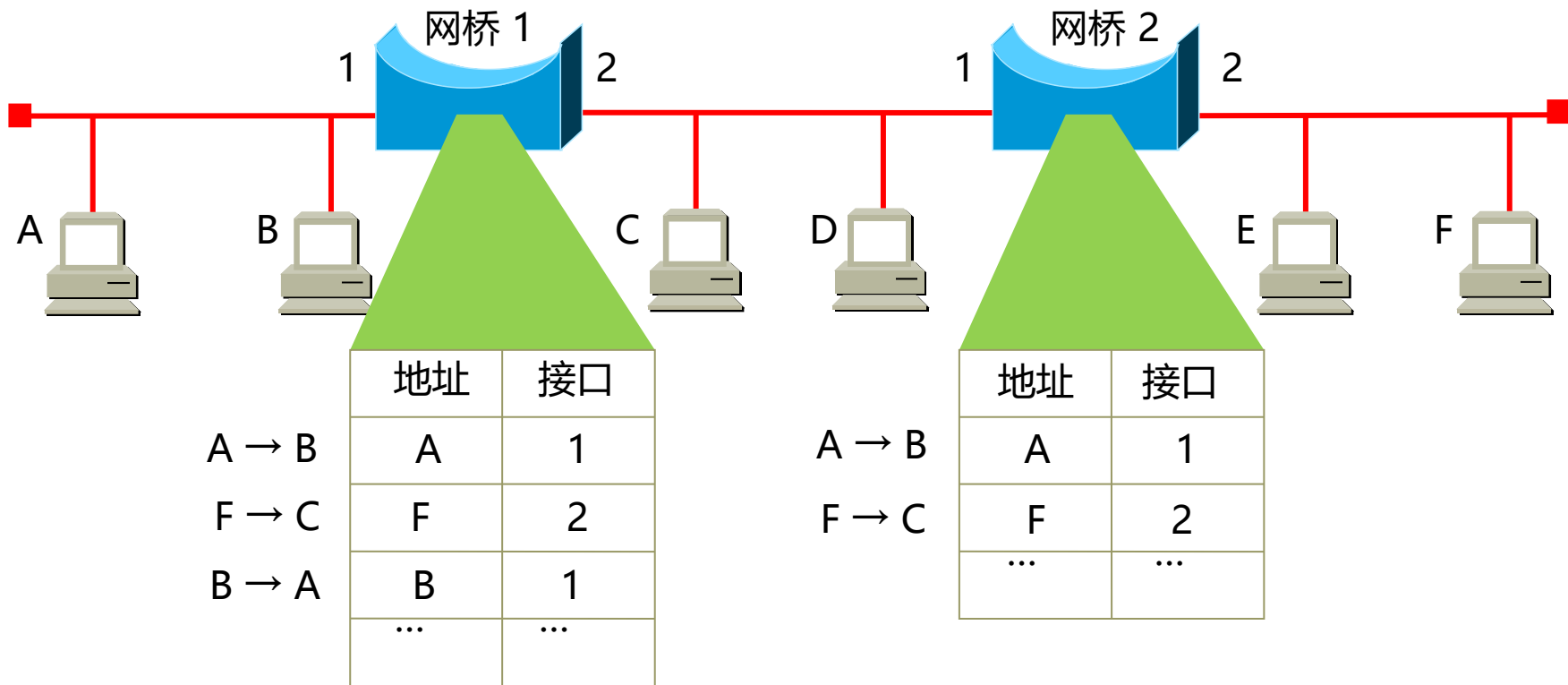
5.2在数据链路层扩展以太网

□ 透明网桥：

- 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。

5.扩展的以太网

5.2在数据链路层扩展以太网



5.扩展的以太网

5.2在数据链路层扩展以太网

□ 透明网桥：

- 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。

5.扩展的以太网

5.2在数据链路层扩展以太网

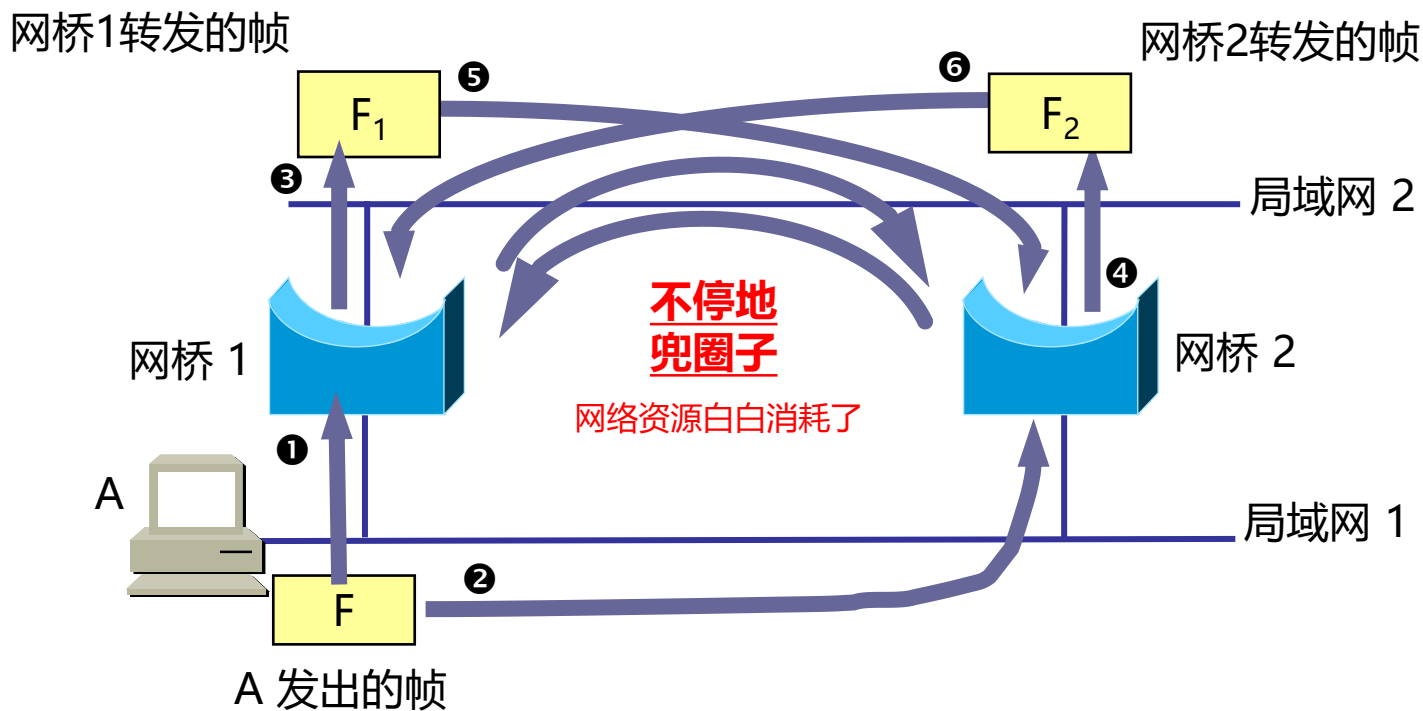
□ 透明网桥自学习和转发帧的步骤归纳：

- 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，在转发表中增加一个项目（源地址、进入的接口和时间）。
 - 如有，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，通过所有其他接口（但进入网桥的接口除外）进行转发。
 - 如有，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）。

5.扩展的以太网

5.2在数据链路层扩展以太网

- 透明网桥使用了生成树算法避免帧在网络中不断地兜圈子：



5.扩展的以太网

5.2在数据链路层扩展以太网

□ 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。



5.扩展的以太网

5.2在数据链路层扩展以太网

□ 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。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。
 - 以太网交换机的诞生和广泛应用，极大地提升了以太网技术的应用范围，促进了互联网的发展。

5.扩展的以太网

5.2在数据链路层扩展以太网

□ 多接口网桥（以太网交换机）：

- 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。

5.扩展的以太网

5.2在数据链路层扩展以太网

□ 多接口网桥（以太网交换机）：

- 对于普通10Mb/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(10Mb/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10Mb/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 $N*10\text{Mb/s}$** 。
- 这正是交换机的最大优点。

5.扩展的以太网

5.3VLAN

- VLAN (Virtual Local Area Network) 即虚拟局域网, 是一种通过将局域网内的设备逻辑地而不是物理地划分成一个个网段从而实现虚拟工作组的新兴技术。
- IEEE于1999年颁布了标准化VLAN实现方案的802.1Q协议标准草案。

5.扩展的以太网

5.3VLAN

- VLAN技术允许网络管理者将一个物理的LAN逻辑地划分成不同的广播域（或称虚拟LAN，即VLAN），每一个VLAN都包含一组有着相同需求的计算机工作站，与物理上形成的LAN有着相同的属性。
 - VLAN是逻辑地而不是物理地划分，所以同一个VLAN内的各个工作站无须被放置在同一个物理空间里，即这些工作站不一定属于同一个物理LAN网段。
 - 一个VLAN内部的广播和单播流量都不会转发到其他VLAN中，从而有助于控制流量、减少设备投资、简化网络管理、提高网络安全性。

5.扩展的以太网

5.3VLAN

- VLAN是为解决以太网的广播问题和安全性而提出的一种协议，它在以太网帧的基础上增加了VLAN头，用VLAN ID把用户划分为更小的工作组，限制不同工作组间的用户二层互访，每个工作组就是一个虚拟局域网。
- 虚拟局域网的好处是可以限制广播范围，并能够形成虚拟工作组，动态管理网络。

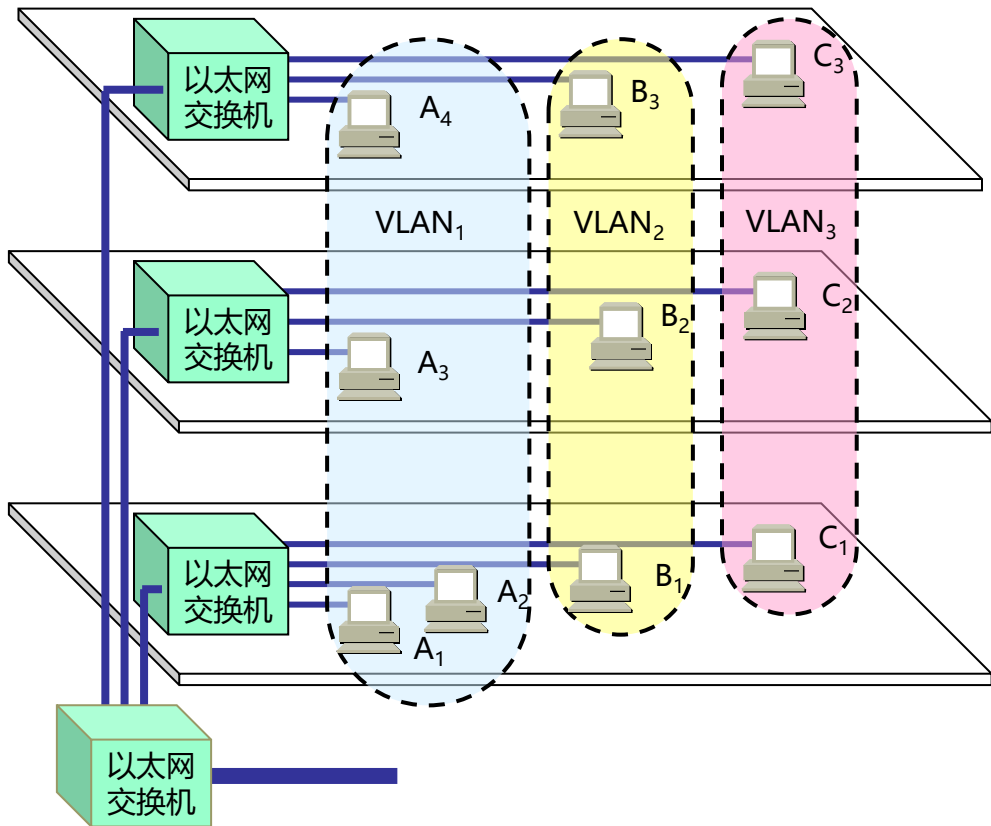
5.扩展的以太网

5.3VLAN

- VLAN是一个广播域，其中的成员仿佛共享同一物理网段一样。
- 不同VLAN成员不能直接访问。
 - 在VLAN中，划分在同一广播域中的成员并没有任何物理或地理上的限制，它们可以连接到一个交换网络中的不同交换机上。
 - 广播分组、未知分组及成员之间的数据分组都被限定在VLAN之内。
- 对VLAN的另一个定义是，它能够使单一的交换结构被划分成多个小的广播域。

5.扩展的以太网

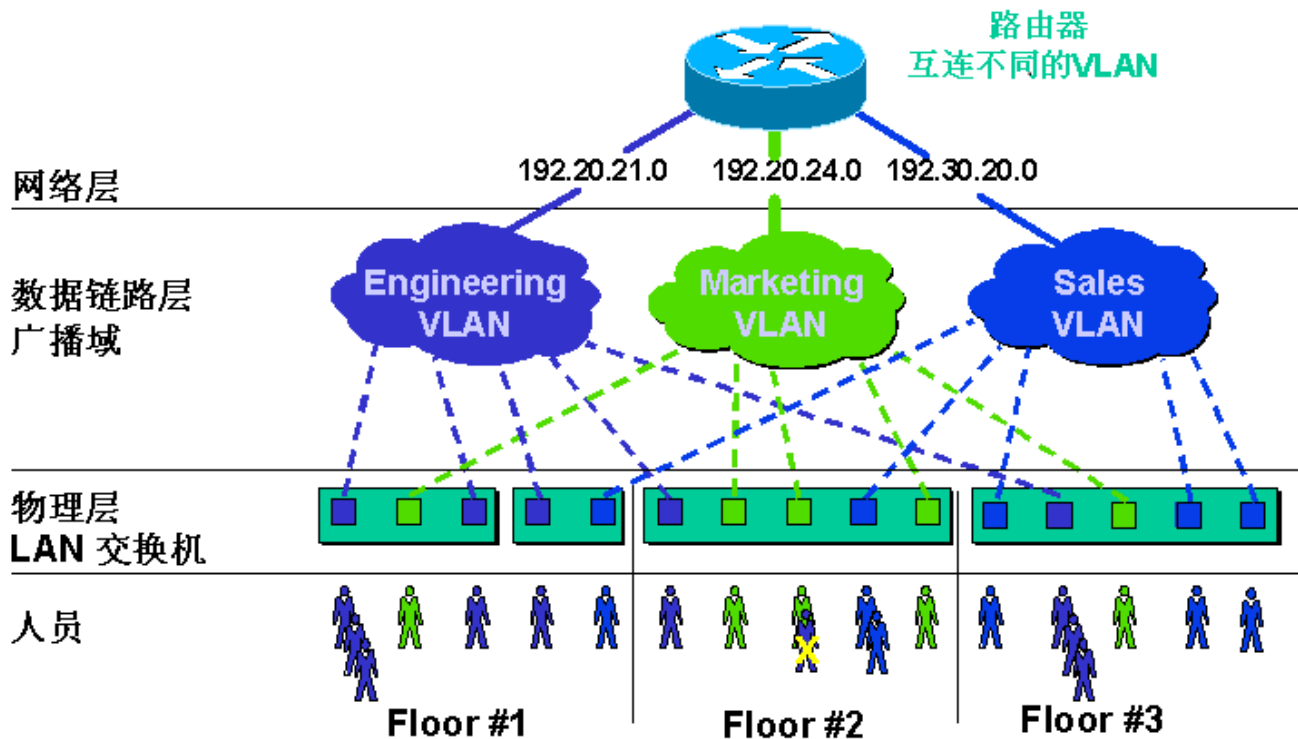
5.3VLAN



三个虚拟局域网:
VLAN₁ VLAN₂ VLAN₃

5.扩展的以太网

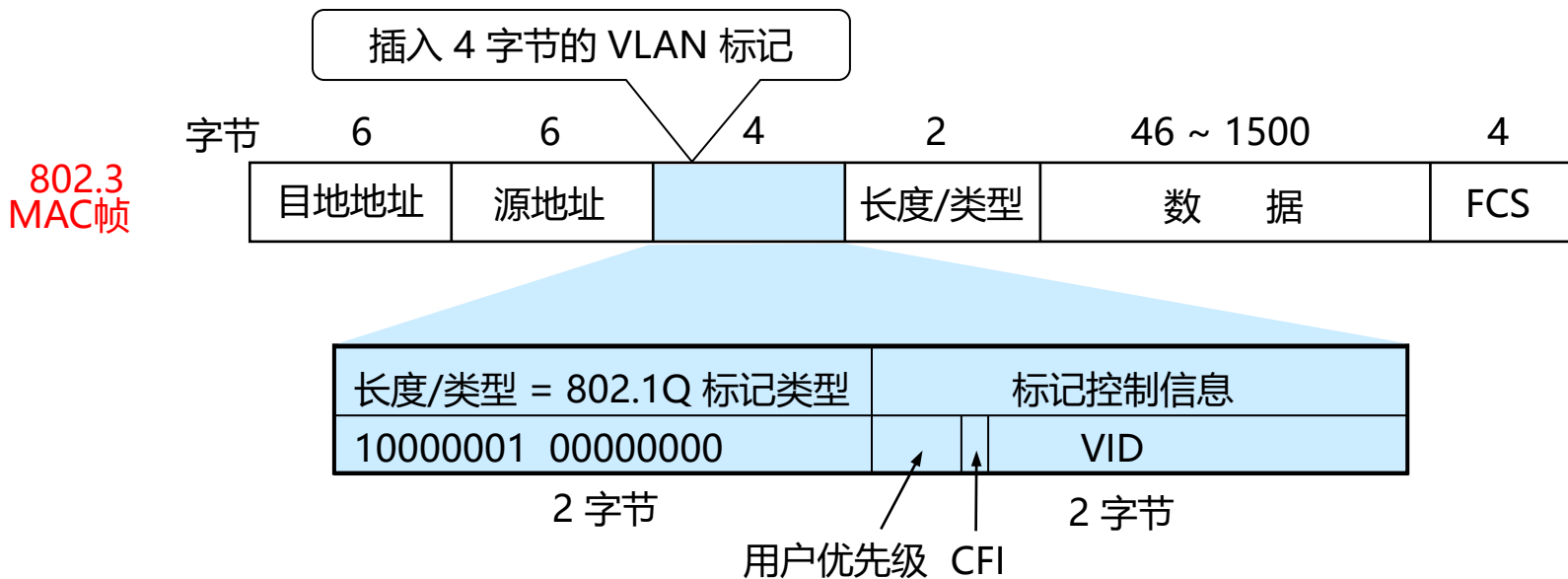
5.3VLAN



5.扩展的以太网

5.3 VLAN

- 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。



6. 高速以太网

- 速率达到或超过 100 Mb/s 的以太网称为**高速以太网**。
- 需要思考和总结的三个问题：
 - 从10Mb/s到100Mb/s是如何实现的？
 - 从100Mb/s到1000Mb/s是如何实现的？
 - 从1000Mb/s到10Gb/s是如何实现的？

6.高速以太网

6.1 100Base-T

- 在双绞线上传送100Mb/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)。

6.高速以太网

6.1 100Base-T

- 100BASE-T以太网的特点：
 - 可在全双工方式下工作而无冲突发生。
 - 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100m。
 - 帧间时间间隔从原来的 $9.6\mu\text{s}$ 改为现在的 $0.96\mu\text{s}$ 。

6.高速以太网

6.1 100Base-T

100Mb/s 以太网物理层标准

名称	媒体	网段最大长度	特点
100BASE-TX	铜缆	100m	两对 UTP5 类线或屏蔽双绞线 STP。
100BASE-T4	铜缆	100m	4 对 UTP3 类线或 5 类线
100BASE-FX	光缆	2000m	两根光纤，发送和接收各用一根。

6.高速以太网

6.2 1000BASE以太网

- 1996年夏季吉比特以太网的产品上市。
- IEEE在1997年通过了吉比特以太网标准802.3z，1998年成为正式标准。
- 目前，吉比特以太网已经成为以太网的主流产品。

6. 高速以太网

6.2 1000BASE以太网

- 吉比特以太网的特点：
 - 允许在1Gb/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。

6. 高速以太网

6.2 1000BASE以太网

1000Mb/s 以太网物理层标准

名称	媒体	网段最大长度	特点
1000BASE-SX	光缆	550m	多模光纤 (50 和 62.5 μm)
1000BASE-LX	光缆	5000m	单模光纤 (10 μm) 多模光纤 (50 和 62.5 μm)
1000BASE-CX	铜缆	25m	使用 2 对屏蔽双绞线电缆 STP
1000BASE-T	铜缆	100m	使用 4 对 UTP5 类线

6. 高速以太网

6.3 10GE和100GE以太网

- 10吉比特以太网与10Mb/s, 100Mb/s 和1Gb/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。
 - 没有争用问题, 不使用 CSMA/CD 协议。
 - 10吉比特以太网的传输距离由于没有争用问题而大大提高。

6. 高速以太网

6.3 10GE和100GE以太网

10GE 物理层标准

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

6.高速以太网

6.3 10GE和100GE以太网

- 以太网的技术发展很快，在10GE之后又制订了40GE/100GE的标准IEEE 802.3ba-2010。
- 40GE/100GE以太网的特点是：
 - 仅在全双工的模式下传输，没有争用问题。
 - 使用以太网的帧格式以及802.3标准规定的最小和最大帧长度。
 - 使用单模光纤传输时能够达到40km（使用4个波长通过波分复用技术实现）。

6.高速以太网

6.3 10GE和100GE以太网

40GB/100GB 以太网物理层标准

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

6.高速以太网

6.3 10GE和100GE以太网

- 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。
- 端到端的以太网传输，典型例子是中国教育与科研计算机网（CERNET）。

6.高速以太网

6.3 10GE和100GE以太网

□ 端到端的以太网传输的优势有：

- **成熟的技术**：以太网是一种经过实践证明的成熟技术。
- **互操作性很好**：不同厂商生产的以太网都能够可靠的进行操作。
- **在广域网中使用价格便宜**：在广域网中使用以太网时，其价格大约只有同步光网络SONET的五分之一和异步传递方式ATM的十分之一。以太网还能够适应多种的传输媒体，使得具有不同传输媒体的用户在进行通信时不需要重新布线。
- **统一的帧格式简化了操作和管理**：端到端都是用以太网技术时，数据帧的格式都是802定义的帧格式，而不需要进行帧格式转换。

6.高速以太网

6.3 10GE和100GE以太网

以太网从10Mbps到100Gbps的几十年来演进，充分说明了：

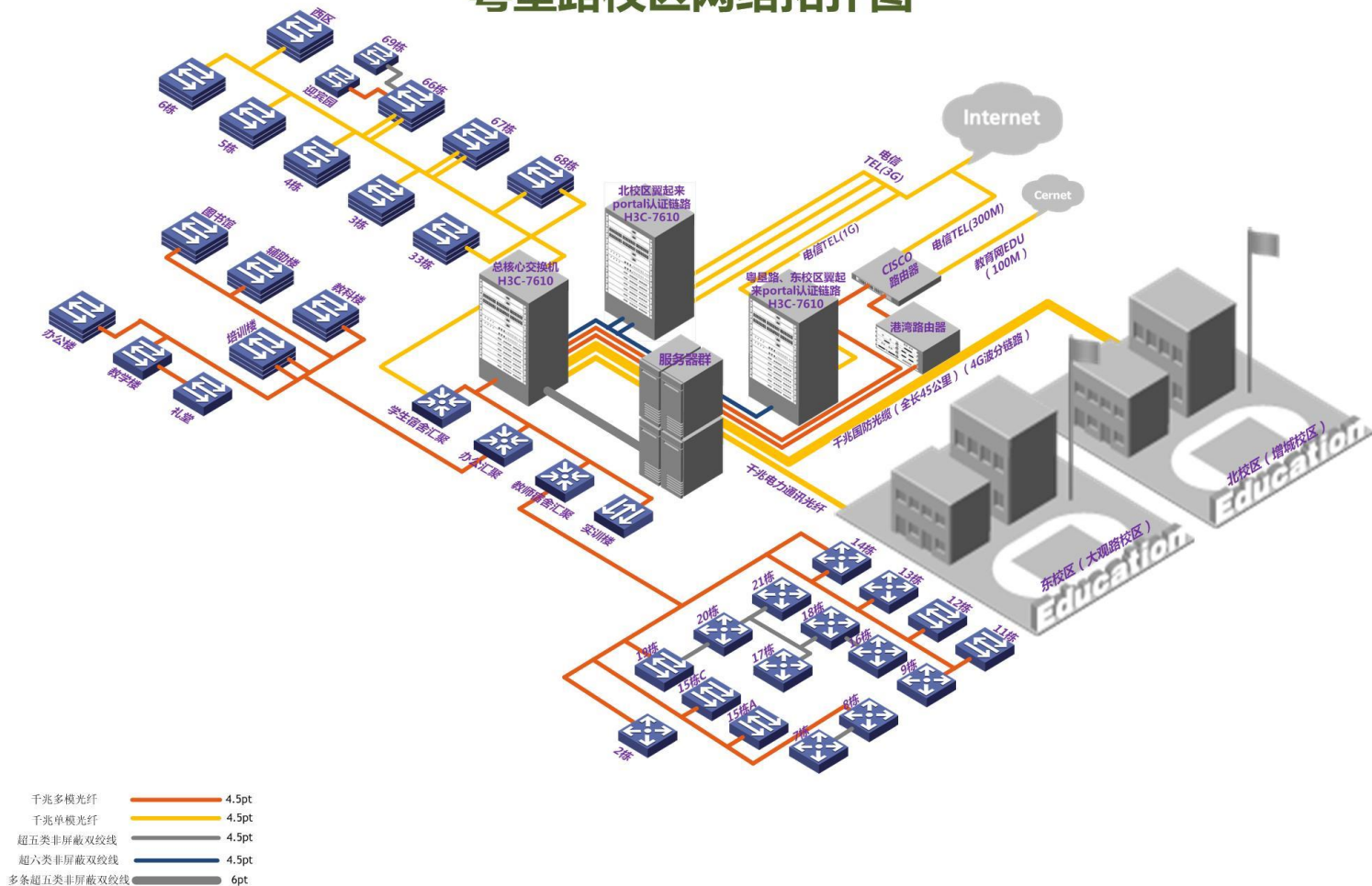


6.高速以太网

6.4使用以太网进行宽带接入

- 以太网已成功地把速率提高到1~10Gb/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。

广东农工商职业技术学院 粤垦路校区网络拓扑图



Thanks