第五章、链路层

数据链路层在物理层提供的服务的基础上向网络层提供服务，其最基本的服务是将源自网络层来的数据可靠地传输到相邻节点的目标机网络层。**数据链路层在不可靠的物理介质上提供可靠的传输。**

该层的作用包括：**物理地址寻址、数据的成帧、流量控制、数据的检错、重发等。**

有关数据链路层的重要知识点：

1. **数据链路层为网络层提供可靠的数据传输；**
2. **基本数据单位为帧；**
3. **主要的协议：以太网协议；**
4. **两个重要设备名称：网桥和交换机。**

5.1 链路层概述

运行链路层协议的任何设备均称为结点，沿着通信路径连接相邻结点的通信信道称为**链路。**通过特定链路时，传输结点将数据报封装在链路层**帧**中，将该帧传入链路

举例：游客想从苏州到临汾玩，旅行社安排的线路是：第一段线路是从苏州乘火车到上海，第二段线路是在上海坐飞机到太原，第三段线路是太原坐大巴到临汾

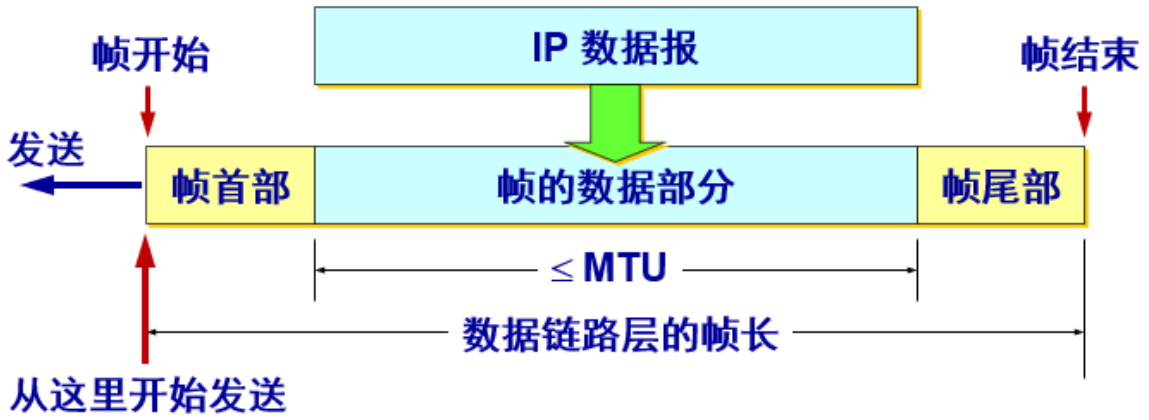
游客相当于数据报
每个运输段相当于一条链路
每种运输方式相当于一种链路层协议
旅行社相当于一个路由选择协议

- 数据链路层协议有许多种，但有三个基本问题则是共同的。这三个基本问题是：

1. 封装成帧
2. 透明传输
3. 差错控制

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**。



用帧首部和帧尾部封装成帧

用控制字符进行帧定界的方法举例

- 当数据是由可打印的 ASCII 码组成的文本文件时，帧定界可以使用特殊的**帧定界符**。
- 控制字符 SOH (Start Of Header) 放在一帧的最前面，表示帧的首部开始。另一个控制字符 EOT (End Of Transmission) 表示帧的结束。

用控制字符进行帧定界的方法举例

透明传输

SLIP

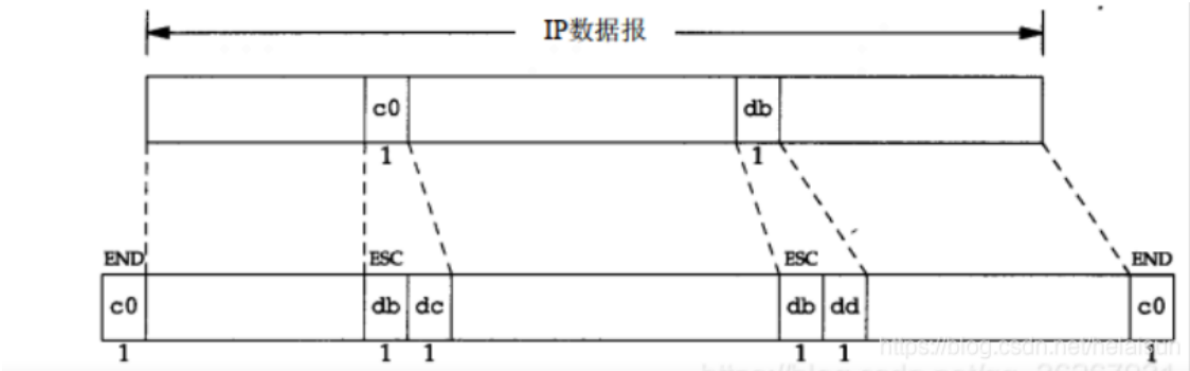
SLIP协议

SLIP协议全程 Serial Line IP。它是一种在串行线路上对IP数据报进行封装的简单形式，在RFC 1055中有详细描述。

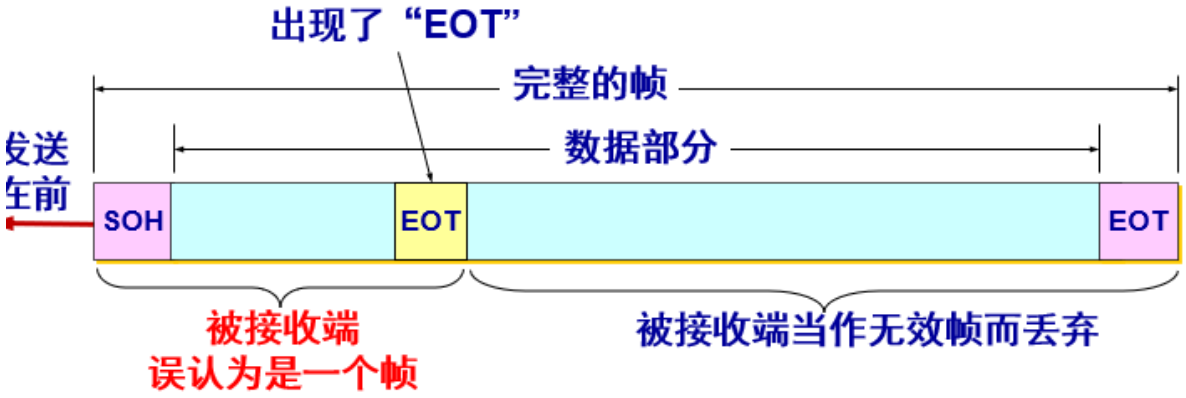
SLIP提供了两个特殊字符，END（0xc0）和 ESC(0xdb)

SLIP报文的头尾都有一个END字符，头部的END是用来结束之前的噪声，这些噪声传到上一层后会被丢弃，尾部END标志当前SLIP报文结束。

- 如果IP数据报中有END字符，则需要用ESC字符加0xdc替代。
- 如果IP数据报中有ESC字符，则需要用ESC加0xdd替代。



- 如果数据中的某个字节的二进制代码恰好和 SOH 或 EOT 一样，数据链路层就会错误地“找到帧的边界”。

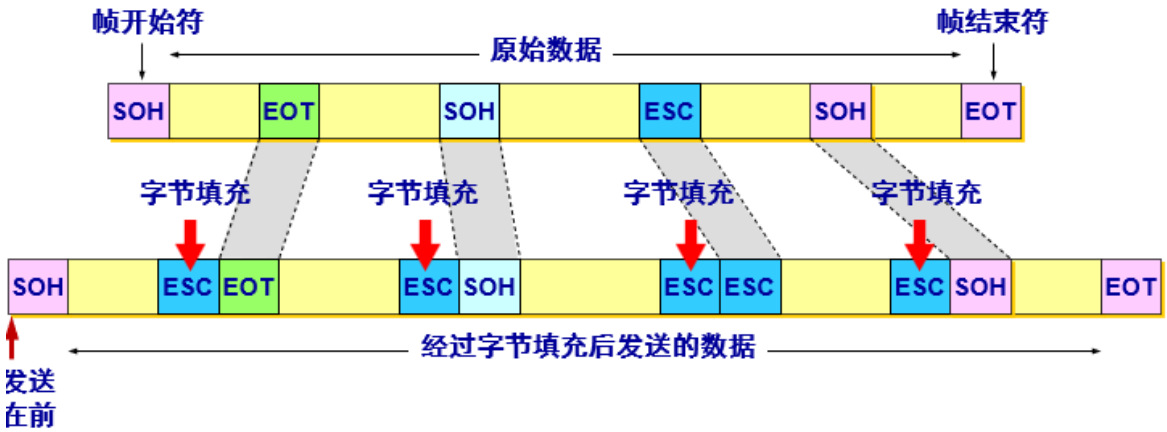


数据部分恰好出现与 EOT 一样的代码

解决透明传输问题

- 解决方法：字节填充 (byte stuffing) 或字符填充 (character stuffing)。
- 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)。
- 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在数据当中，那么应在转义字符前面插入一个转义字符 ESC。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



用字节填充法解决透明传输的问题

差错检验

CRC

- 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）。
- 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。
- 应当明确，“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，但这还不是可靠传输。
- 本章介绍的数据链路层协议都不是可靠传输的协议。

丢失、失序、重复---所以有的在CRC的基础之上增加了帧编号、确认和重传机制

5.1.1链路层提供的服务

能够提供的协议可能包括：（不同协议包括不同，细节不同）

成帧

网络层数据报经链路传送前，链路层协议要将其用链路层帧封装起来

帧的结构由链路层协议规定

链路接入

媒体访问控制MAC协议，规定帧在链路上传输的规则，协调多个结点的帧传输

可靠交付

保证无差错经链路层移动每个网络层数据报

确认和重传，类似TCP

通常用于高差错率链路，如无线链路，同轴电缆、光纤、双绞线等链路不需要

目的是在差错发生的链路上纠正差错，而不是通过运输层或应用层进行端到端数据重传

链路层可靠交付可能会被认为是一种不必要的开销。由于这个原因，许多有线的链路层协议**不提供可靠交付服务。**

差错检测和纠正（硬件）

- 奇偶校验

- 检验和
- 循环冗余检测

5.1.2 链路层在何处实现（较硬件）

路由器中：在线路卡中实现

端主机中：**网络适配器（网络接口卡，网卡）**，位于其核心的是**链路层控制器**，一个实现了许多链路层服务（成帧、链路介入、差错检测）的专用芯片。之前是物理分离的卡，现在网卡直接焊在了主板上

大部分链路层是在**硬件**中实现的，但部分链路层是在运行于主机CPU上的软件中实现的，软件实现了高级功能，如组装链路层寻址信息和激活控制器硬件，响应控制器中断

链路层是协议栈中软件和硬件交接的地方

5.2 差错检测和纠正技术

比特级差错检测与纠正，即对从一个结点发送到另一个物理上连接的邻近结点的链路层帧中的比特损伤进行检测与纠正，它们通常是链路层提供的两种服务。

为了保护比特免受差错，使用**差错检测和纠正比特（EDC）**。通常，要保护的数据不仅包括从网络层传递下来需要通过链路传输的数据报，而且包括链路帧首部中的链路级的寻址信息、序号和其它字段。

即使采用差错检验比特，也还是可能有未检出比特差错。

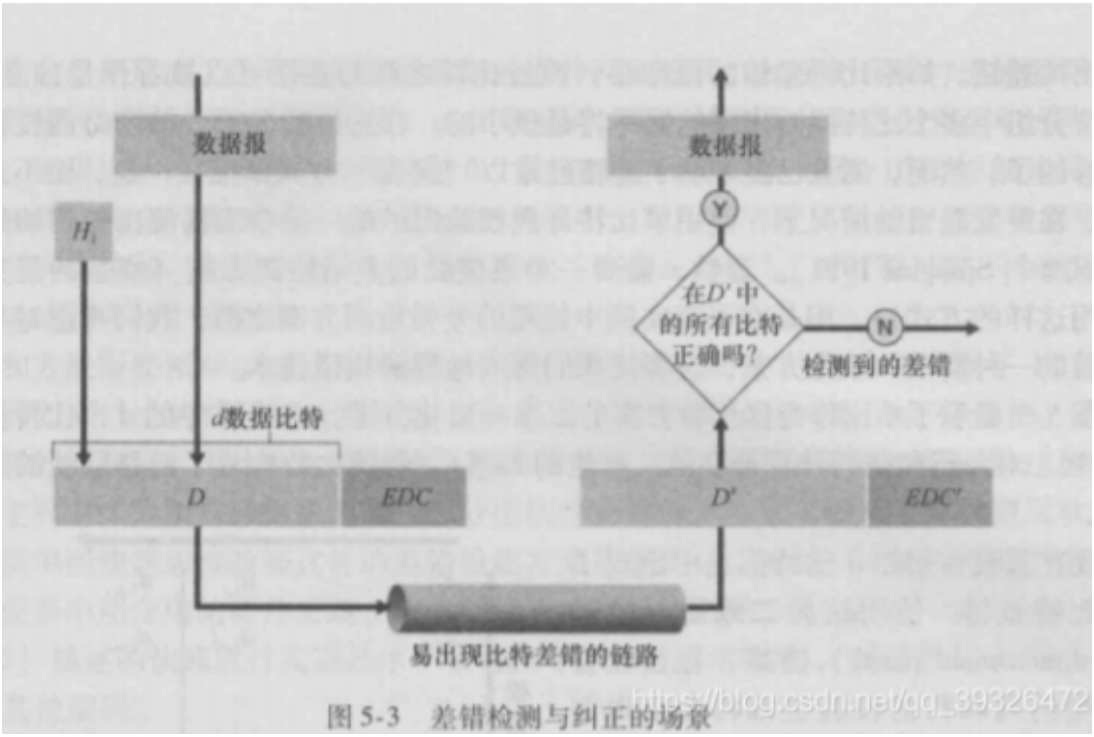


图 5-3　差错检测与纠正的场景

传输数据中检测差错的三种技术

奇偶校验（描述差错检测和纠正背后的思想）

检验和方法（应用于运输层）

循环冗余检测（应用在适配器中的链路层）

奇偶校验

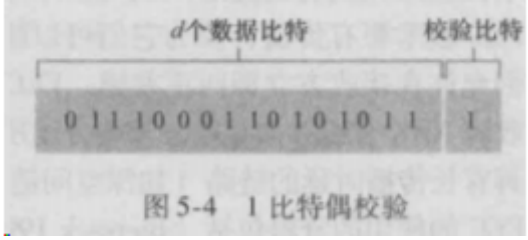
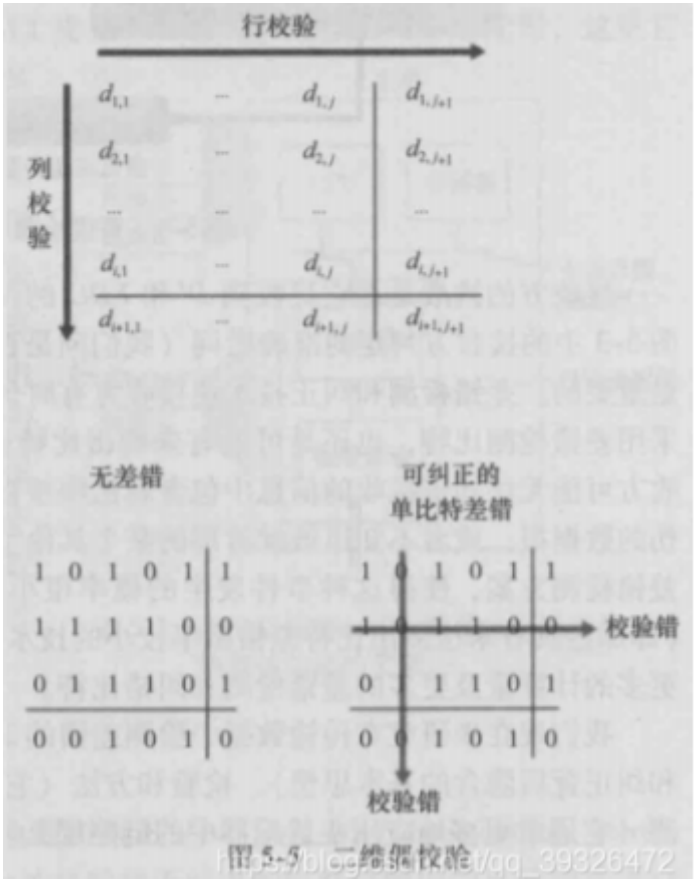


图 5-4　1 比特偶校验

单个奇偶校验位

单个比特的奇偶校验是指在要发送的数据最后附加一个奇偶校验位.奇校验的意思就是整个编码中的1的个数要是奇数.偶校验就是1的个数是偶数.显然如果有偶数个比特发生错误,那么奇偶校验就检测不出来了.

二维奇偶校验



D中的d个比特被划分为i行j列.对每行没咧计算奇偶值.产生的i + j + 1奇偶比特就构成了链路层帧的差错检测比特.

当出现单个比特差错时,发生错误的行和列都会出现差错.接收方不仅可以检测差错,还可以根据行列索引来纠正它.

二维奇偶校验也可以检测(但不能纠正)两个比特错误的任何组合.

如果在同一行两个比特错误,则那一行的奇偶校验正确,但是会有两列的奇偶校验失败.如果不同行,则会有四列出错.这两种情况都无法纠错,只能检测.但是还是有一些偶数个错误的情况是二维奇偶校验无法检测的.

接收方检测和纠错的能力被称为**前向纠错（FEC）--海明码？？？.**

校验和方法

在校验和方法中,数据被切成k比特的序列,这些序列全部相加之后取反码就是校验和.接收方收到数据之后,把所有数据加起来(包括校验和).用结果是否全为1来作为判断数据是否出错的标准.

和CRC相比,校验和提供较弱的保护.

为什么传输层使用校验和而链路层使用CRC呢?

1. 传输层使用软件实现,采用简单快速的方案是必须的(校验和)
2. 链路层的CRC用硬件实现,能够快速执行CRC操作.

循环冗余检测(CRC)

循环冗余检测(CRC)

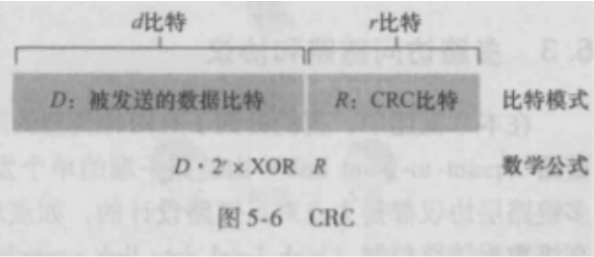


图 5-6 CRC

编码解码步骤：
CRC编码也称为多项式编码,因为该编码能够将要发送的比特串看成系数是0或1的一个比特串,对比特串的操作被解释为多项式算数.
这里不过多讨论多项式理论.
编码步骤如下：
1. 发送方和接收方实现协商一个r + 1的比特模式(比特串)G,叫做生成**多项式**.要求G的最高位是1
2. 对于一个给定的数据段D,发送方选择r个附加比特R,并将它们附加到D上.
3. 使得得到的d + r比特模式用模2算数恰好能被G整除.(模2算数就是异或)
接收方的解码步骤很简单,用G去除收到的d + r比特.如果余数非0,接收方知道出了差错.否则认为数据被正确接收.
怎么计算R
要使得R对于n有: $D * 2^r$XOR R = nG.
两边同异或R得: $D * 2^r$ = nG XOR
所以 R = remainder {$D * 2^r$ / G}

5.3 多路访问链路和协议

有两种类型的网络链路
点对点链路
点对点协议PPP
高级数据链路控制协议HDLC

HDLC

FCS（帧检验序列）是在以太网数据帧的尾部的4个字节的序列，而**CRC是循环冗余校验码**，也就是说FCS是真正位于以太网数据帧里面用于检验数据是否出错的序列，而CRC是一种给出FCS检测序列的检验方法。

HDLC帧类型

HDLC定义了很多种帧，每种类型的帧有不同的功能，判断一个HDLC的帧是何种类型，是根据帧的控制字段的值来判断的。

我们在这里只介绍其中最主要的三种类型：

信息帧（I帧）：用来传输数据信息，上面说的主站对从站发出的命令，以及从站对主站的应答，就属于信息帧。

监督帧（S帧）：它的作用是流量控制，以及差错检测和控制等功能。（流量控制、差错控制这些功能其实是非常重要的，这些我们统一放到后面学习传输层TCP的时候会详细介绍）

无编号帧（U帧）：这种帧是执行对数据链路的建立和拆除的功能。

广播链路

让多个发送和接收结点都连接到相同的、单一的、共享的广播信道上

当任何一个结点传输一个帧时，信道广播该帧，其他结点都收到一个副本

如以太网和无线局域网

多路访问问题

如何协调多个发送和接收结点对一个共享广播信道的访问

所有结点都能传输帧，多个结点可能会同时传输帧，所有结点同时接到多个帧，传输的帧在所有接收方出碰撞了，发生碰撞时，所有帧丢失

多路访问协议：结点通过协议规范它们在共享的广播信道上的传输行为

信道划分协议

随机接入协议

轮流协议**

协议希望有的特性，理想情况下对速率R bps的广播信道

仅有一个结点发送数据，结点具有R bps的吞吐量

M个结点发送数据时，每个结点平均吞吐量R/M bps

协议分散，不会因为主结点故障似整个系统崩溃

协议简单不昂贵

5.3.1信道划分协议

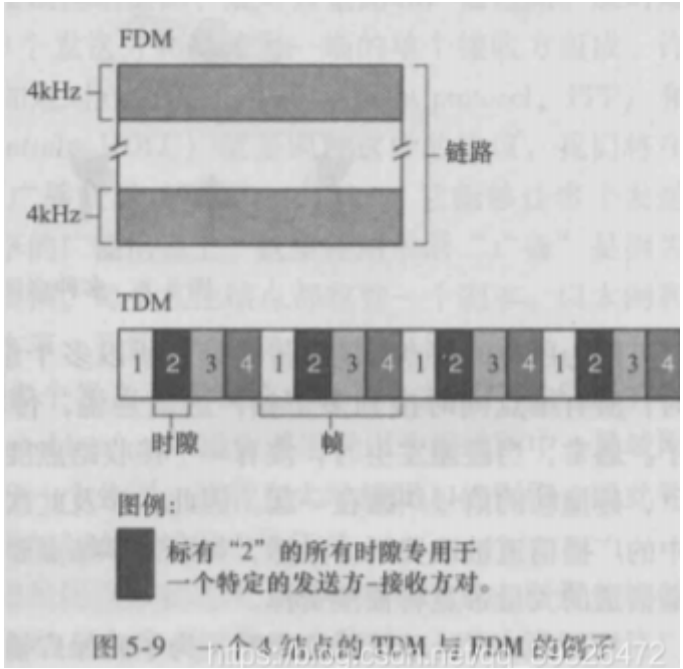


图 5-9 一个4结点的TDM与FDM的例子

TDM(时分多路复用)

TDM把时间划分为时间帧,并进一步把时间帧划分为N个时隙(slot).(时间帧和链路层交换的单元帧不是一个意思)
然后把每个时隙分给N个节点中的一个.无论何时某个节点在有分组想要发送的时候,他在循环的TDM帧中指派给它的时隙内传输分组比特.时隙长度一般应是一个时隙内能传输一个分组

TDM的缺点

- 最高速率只能达到R/N bps ,即使只有一个人使用信道
- 节点总是总是要等待它的时隙,可能会对缓存等造成压力
- 消除了碰撞且十分公平

FDM(频分多路复用)

FDM将Rbps的信道划分为不同的频段(每个频段具有R/N带宽),并把每个频段分给N个结点中的一个.因此FDM在N个较大的信道中创建了N个较小的R/N信道.

FDM的缺点和TDM相同,限制了每个结点只能使用R/N带宽..

码分多址CDMA

TDM和FDM分别为结点分配时隙和频率，CDMA对每个结点分配不同的编码
每个结点用其唯一编码对发送数据进行编码，使得不同结点能同时传输，接收方仍能正确而接收
抗干扰，军用系统，民用蜂窝电话

5.3.2随机接入协议

传输结点总是以信道**全部速率R** bps进行发送

有碰撞时，涉及碰撞的每个节点反复重发它的帧（等待一个随机时延），直到该帧无碰撞的通过

常用随机接入协议：
1.时隙ALOHA
当结点有新帧发送时，等到下一个时隙开始传输整个帧（设一个时隙传一个帧）。时隙开始时传输。
如果有碰撞，结点在时隙结束之前检测到这次碰撞，以p的概率在后序的每个时隙重传它的帧，直到无碰撞。

时隙ALOHA的确需要在结点中对时隙同步。刚好有一个结点传输的时隙称为一个成功时隙。时隙多路访问协议的效率定义为：当有大量的活跃结点且每个结点总有大量的帧要发送时，长期运行中成功时隙的份额。
效率：当活跃结点数量趋向无穷大时，最大效率1/e，即37%。一个给定的结点成功传送的概率是,因为有N个结点，任意一个结点成功传送的概率是N。

2.（纯）ALOHA
碰撞时，立即以概率p重传该帧，否则等待一个帧传输时间
效率：仅为时隙ALOHA的一半：.一个给定结点成功传输一次的该概率是

3.载波侦听多路访问CSMA
在时隙和纯ALOHA种，一个结点传输的决定独立于其他结点，不关心自己传输时别人是不是在传输
举例：有礼貌的人类谈话有两个重要规则
说话之前先听。如果在说话，等他们说完话再说，网络中称为载波侦听 ，结点等待直到一小段时间没有传输，然后开始传输
如果与他人同时开始说话，停止说话。称为碰撞检测，当一个传输结点在传输时一直侦听此信道，如果检测到另一个结点正在传输，它就停止，等待一段随机事件，重复『侦听=当空闲时传输』动作
这两个规则包含在CSMA和具有碰撞检测的CSMA/CD协议族中
所有结点都载波侦听了，为何当初会发生碰撞?
B的比特沿着广播媒体传播所实际需要的时间不是0（即使2*10^8），在还没到D时，尽管B正在发，但D侦听的信道空闲，D就开始传输，于是发生了碰撞

广播信道端到端信道传播时延决定了性能，时延越大，不能侦听到已传输结点的可能就越大，碰撞越多，性能越差

4.具有碰撞检测的载波侦听多路访问CSMA/CD

半双工通信

与广播信道相连的适配器：

适配器从网络层一条获得一条数据报，准备链路层帧，并将其放入帧适配器缓存中

如果适配器侦听到信道空闲，开始传输帧；如果侦听到信道在忙，等待，直到空闲

传输过程中，适配器监视信道

如果适配器传输整个帧而未检测到其他信号，该适配器完成了该帧，否则停止传输帧

中止传输后，适配器等待一个随机时间量，继续侦听

选择随机回退时间间隔太大，信道会空闲，太小会再次碰撞。当碰撞结点数量较少时时间应该间隔较短，否则较长。二进制指数后退算法解决这个问题

帧经历一连串n次碰撞，结点随机从{0，1，2，...，2^n-1}选择一个K值

一个帧经历碰撞越多，K选择的间隔越大.该算法称为二进制指数倒退。

以太网中，一个结点等待的实际时间量是K*512bit的时间.

效率

信道在大多数时间都会有效地工作。

5.3.3 轮流协议

轮询协议

指定一个主结点，以循环的方式轮询每个结点

主结点首先向结点A发送一个报文，告知A能传输帧的最大数量，A传完后主结点告诉B能传帧的最多数量，如此循环

缺点：有轮询时延；主结点故障，整个信道就GG

令牌传递协议

没有主结点，一个叫令牌token的特殊帧在结点之间以固定次序交换，如1发给2，2发给3，N发给1，就像网络拓扑结构中的环状网络令牌

当一个结点收到令牌时，有帧发送，则发送最大数量的帧，然后转发令牌；没帧发送，直接把令牌转发.

缺点：单点故障。

DOCISIS：用于电缆因特网接入的链路层协议

局域网和广域网

[]: https://blog.csdn.net/vavid317/article/details/126038959?ops_request_misc=%257B%2522request%255Fid%2522%253A%252216844188281678242749907 7%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=168441882816782427499077&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_ecpm_v1~rank_v31_ecpm-3-126038959-null-null.142v87control_2,239v2insert_chatgpt&utm_term=%E5%B1%80%E5%9F%9F%E7%BD%91%E5%92%8C%E5%B9%BF%E5%9F%9F%E7%BD%91%E7%9A%84%E5%8C%BA%E5%88%AB&spm=1018.2226.3001.4187

5.4 交换局域网

交换机运行在链路层，它们**使用链路层地**址而不是IP地址来转发链路层帧通过交换机网络

5.4.1 链路层寻址和ARP

1.MAC地址

并不是主机或路由器具有链路层地址，**而是它们的适配器（网络接口）具有链路层地址。**具有多个网络接口的主机或路由器也有多个链路层地址，就像它也有多个IP地址一样。

链路层交换机并没有链路层地址，交换机透明地执行在主机与路由器之间承载数据报的任务

链路层地址也叫LAN地址、物理地址、MAC地址

MAC地址长度6字节，2^48个可能的MAC地址，通常用**十六进制**表示法，如5C-66-AB-90-75-B1

MAC地址一般是固定的（也有软件改变适配器MAC地址的可能）

没有两块适配器有相同的MAC地址，MAC地址空间由IEEE管理，IEEE给公司固定前24个比特，后面24个比特让公司自己去生成

MAC地址具有扁平接口。比如具有802.11接口的手机总是有相同mac地址，而当主机移动时，IP地址会改变（IP地址是层次结构）

MAC地址像身份证号，IP地址像邮政地址，有层次，会改变

当某适配器要向目的适配器发送一个帧时，发送适配器将目的适配器的MAC地址插入该帧，发送到局域网上，适配器可以接受一个并非向它寻址的帧，当适配器接受一个帧时，检查帧中的目的MAC地址与自己的MAC地址是否匹配，若匹配则取出数据报，向上传递，否则丢弃

适**配器通过MAC广播地址FF-FF-FF-FF-FF-FF来广播**

2. **地址解析协议ARP（即插即用的）**

   转换网络层地址和链路层地址，如IP地址和MAC地址的转换

   DNS为因特网中任何地方的主机解析主机名，而ARP只为在同一个子网上的主机和路由器接口解析IP地址

   每台主机和路由器在内存中有一个ARP表，包含IP地址到MAC地址的映射关系，过期时间20分钟

   若发送方的ARP表没有目的主机的表项，发送反公用ARP协议来解析这个地址

   首先发送方构造一个ARP分组，字段包括发送和接受IP地址和MAC地址，ARP查询分组和响应分组格式相同

   适配器用MAC广播地址发送该ARP查询分组，每个适配器都把ARP分组向上传递给ARP模块，检查自己的IP地址和分组中的目的IP地址是否一致

   匹配的主机发送回一个ARP响应分组，然后查询主机更新它的ARP表，并发送它的IP数据报

   ARP协议是一个跨越链路层和网络层的协议

3. **发送数据报到子网以外**

   路由器有几个接口，就有几个IP地址、ARP模块和适配器，假设一个路由器连着两个子网A、B

   子网A中的适配器要发往子网B中的适配器，先通过子网A的ARP把数据报发到子网A跟子网B相连的路由器（目的地址是路由器的MAC），路由器通过子网B的ARP将该数据报转发给目的适配器（目的地址是最终目的地的MAC）。

5.4.2 以太网（无连接不可靠）

[]: https://blog.csdn.net/weixin_40274679/article/details/105995323?ops_request_misc=%257B%2522request%255Fid%2522%253A%252216844192151678242743081 8%2522%252C%2522scm%2522%253A%252220140713.130102334..%2522%257D&request_id=168441921516782427430818&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~top_positive~default-1-105995323-null-null.142^{v87}control_2,239^{v2}insert_chatgpt&utm_term=%E4%BB%A5%E5%A4%AA%E7%BD%91&spm=1018.2226.3001.4187

3.4 以太网协议详解

MAC地址：每一个设备都拥有唯一的MAC地址，共48位，使用十六进制表示。

以太网协议：是一种使用广泛的局域网技术，是一种应用于数据链路层的协议，使用以太网可以完成**相邻设备**的数据帧传输：

目的地址	源地址	类型	帧数据	CRC
6	6	2	46~1500	4

局域网分类：

Ethernet以太网IEEE802.3：

1. 以太网**第一个**广泛部署的高速局域网
2. 以太网**数据速率快**
3. 以太网硬件价格**便宜**，网络造价成本低

以太网帧结构：

1. **类型**：标识上层协议（2字节）
2. **目的地址和源地址**：MAC地址（每个6字节）
3. **数据**：封装的上层协议的分组（46~1500字节）
4. CRC：循环冗余码（4字节）
5. **以太网最短帧**：以太网帧最短64字节；以太网帧除了数据部分18字节；数据最短46字节；

MAC地址（物理地址、局域网地址）

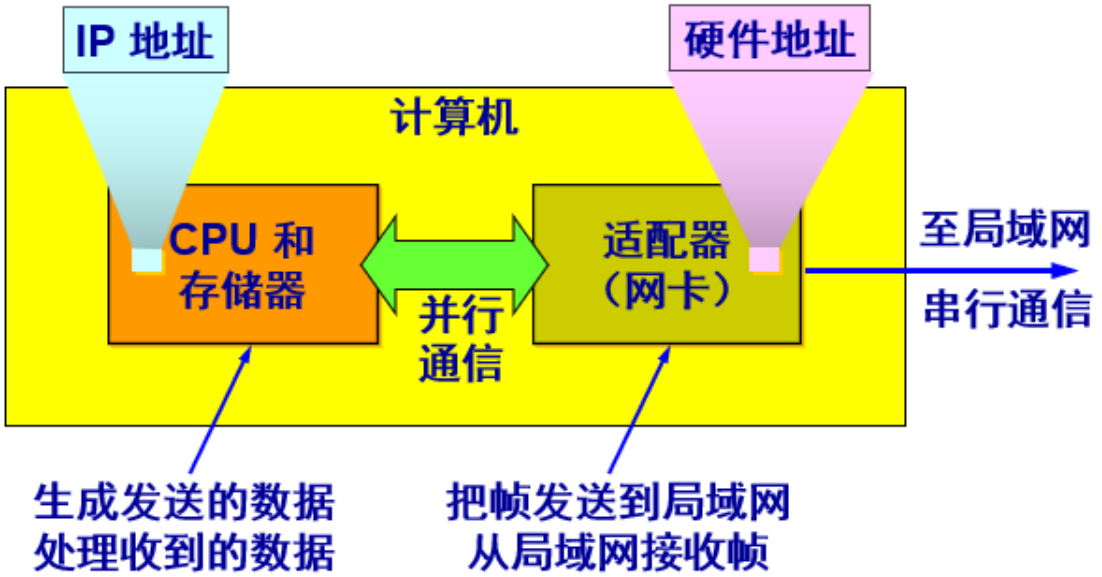
1. MAC**地址长度为6字节**，48位；
2. MAC地址**具有唯一性**，每个**网络适配器**对应一个MAC地址；
3. 通常采用十六进制表示法，每个字节表示一个十六进制数，用 - 或 : 连接起来；
4. MAC广播地址：FF-FF-FF-FF-FF-FF。

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



以太网占领了现有的**有线局域网**市场，就像因特网之于全球联网的地位

集线器是一种物理层设备，作用于比特而不是帧。当0或1的比特到达一个接口时，集线器只是重新生成这个比特，将其能量强度放大，并将该比特向其他所有接口传输出去

早期基于集线器**星形拓扑以太网**，现在位于中心的集线器被交换机所取代。交换机是无碰撞的存储转发分组交换机，运行在链路层

以太网帧结构

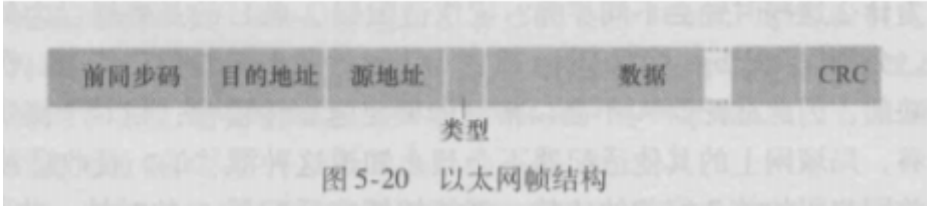


图 5-20　以太网帧结构

数据字段（46~1500字节）：承载了IP数据报（如），超过1500字节的数据报需要分片；若小于46字节，需要填充到46字节

目的地址（6字节）：目的适配器的MAC地址。当目的适配器收到一个以太网帧，**若目的地址是自己的MAC地址或广播地址**，将数据字段传给网络层，其他则丢弃

源地址

类型字段：允许以太网复用多种网络层协议
CRC（4字节）：差错检测
前同步码（8字节）：以太网帧以前同步码开始，前7个字节用于唤醒接收适配器，同步发送方接收方时钟，第8个字节最后两个比特（11）警告目的适配器，重要内容来了
以太网技术向网络层提供不可靠、无连接服务。没有通过CRC校验只是丢弃。

以太网有时候的确重传了数据，但并不知道正在传输新数据还是旧数据。

以太网技术
早期10BASE-2和10BASE-5标准规定两种类型的同轴电缆的10Mbps以太网，每种标准限制在500米，通过转发器得到更长运行距离。

BASE表示基带以太网。前面的数字代表速率。T代表双绞线。F代表光纤。 100BASE-FX
今天的以太网，结点经点对点由双绞铜线或光纤构成的线段与一台交换机相连
10Gbps以太网，5类UTP线缆
线代交换机是全双工的，一台交换机和一个结点能同时向对方发送帧而没有干扰。在基于交换机的以太局域网中，没有必要使用MAC协议了！

5.4.3 链路层交换机

全双工

交换机的任务：接收入链路层帧，转发到出链路
交换机自身对子网中的主机和路由器是透明的，主机/路由器向另一个主机/路由器寻址一个帧，顺利将帧发送进局域网，并不知道交换机干嘛

交换机输出接口设有缓存
交换机是即插即用设备，管理员无需配置
交换机是双工的，任何交换机接口能同时发送和接收

转发和过滤
借助于交换机表，包含局域网上某些主机和路由器的表项（MAC地址，通向该地址的交换机接口，表项放置的时间）
假定目的地址为DD-DD-DD-DD-DD-DD的帧从交换机接口x到达，交换机用该MAC地址索引交换机表，有三种可能：
表中没有该地址，交换机广播该帧
表中有表项将该地址与接口x联系起来，过滤掉，因为该帧从x来，DD也通过x去，说明该帧跟DD适配器在同一个局域网段，该帧已经在包含目的地的局域网网段广播过了
表中有表象将该地址与接口y≠x联系起来，该帧需要被转发到与接口y相连的局域网段，放到接口y前的输出缓存，完成转发功能
自学习：表是自动、动态建立的
交换机表初始为空
对于每个接口接收到的每个入帧，交换机在其表中存储
该帧源MAC地址
帧到达的接口
当前时间
一段时间后，交换机没有接受到以该地址作为源地址的帧，在表中删除该地址。如果一台PC被另一台PC代替，原来PC的MAC地址将被清除
链路层交换机的性质
消除碰撞
交换机缓存帧并且不会在网段上同时传输多于一个帧，交换机提供了比广播链路局域网高的多的性能改善
异质的链路

交换机将链路彼此隔离，因此局域网中的不同链路能够以不同速率运行，在不同媒介上运行

网络管理

主动断开异常适配器

收集带宽使用的统计数据、碰撞率和流量类型，这些信息用来调试解决问题

安全性

交换机毒化：向交换机发送大量不同伪造源MAC地址的分组，用伪造表项填满了交换机表，没有为合法主机留下空间，导致交换机广播大多数帧，被嗅探器俘获到

交换机和路由器比较

表 5-1 流行的互联设备的典型特色的比较

	集线器	路由器	交换机
流量隔离	无	有	有
即插即用	有	无	有
优化路由	无	有	无

路由器是第三层的分组交换机，交换机是第二层的分组交换机

交换机：

- 交换机即插即用，相对高的分钟过滤和转发速率
- 防止广播帧循环，交换网络的活跃拓扑限制为一颗生成树
- 大型交换网络要求在主机和路由器中有大的ARP表，生成大量ARP流量和处理量
- **对广播风暴不提供任何保护，使得以太网崩溃**

路由器：

- 分组不会被限制到生成树上，可以使用源到目的地的最佳路径，拓扑结构更加丰富
- 对第二层的广播风暴提供了防火墙保护
- 不是即插即用，需要人为配置IP地址
- 对分组处理时间较长，因为必须处理第三层字段

点对点PPP协议

> **点对点协议PPP**是目前 使用最广泛的 点对点数据链路层协议。

PPP协议为在点对点链路传输各种协议数据报提供了一个标准方法，主要由以下三部分构成：

> (1) 对各种协议数据报的 封装方法 （封装成帧）；
> (2) 链路控制协议LCP：用于建立，配置以及测试数据链路的连接；
> (3) 一层网络控制协议NCPs：其中每一个协议支持不同的网络层协议；

帧格式：

PPP帧的首部和尾部分别为四个字段和两个字段。

首部

首部中的标志字段F(Flag)，规定为0x7E(符号0x表示它后面的字符是用十六进制表示的。十六进制的7E的二进制表示是01111110)，标志字段 表示一个帧的开始。

首部中的地址字段A规定为0xFF(即11111111)。

首部中的控制字段C规定为0x03(即00000011)。

首部中的2字节的协议字段：

1. 当协议字段为0x0021时，PPP帧的信息字段就是IP数据报。


2. 当协议字段为0xC021时，PPP帧的信息字段就是PPP链路控制协议LCP分组。


3. 当协议字段为0x8021时，PPP帧的信息字段就是网络层的控制数据NCP分组。


信息字段

信息字段的长度是可变的，不超过1500字节

尾部

尾部中的第一个字段(2个字节)是使用CRC的帧检验序列FCS。

尾部中的标志字段F(Flag)，规定为0x7E(符号0x表示它后面的字符是用十六进制表示的。十六进制的7E的二进制表示是01111110)，标志字段 表示一个帧的结束。

当信息字段中出现和标志字段一样的比特(0x7E)组合时，就必须采取一些措施使这种形式上和标志字段一样的比特组合不出现在信息字段中。也就是之前提到过的 实现透明传输。

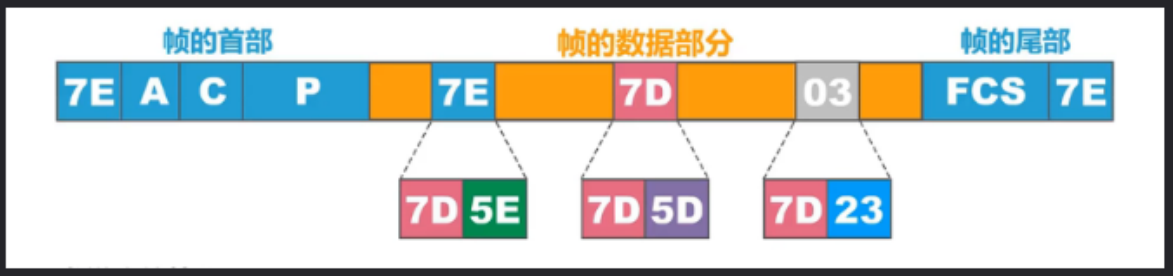
我们针对面向字节的异步链路和面向比特的同步链路进行不一样的处理。

面向字节的异步链路

> 当PPP使用异步传输时，它把转移符定义为0x7D，并使用 字节填充。

RFC1662规定了如下填充方法：

1. 把信息字段中出现的每一个0x7E字节转变为2字节序列(0x7D，0x5E)。
2. 若信息字段中出现一个0x7D的字节(即出现了和转义字符一样的比特组合)，则把转义字符0x7D转变为2字节序列(0x7D，0x5D)。
3. 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。例如，出现0x03(在控制字符中是"传输结束"ETX)就要把它转变为2字节序列的(0x7D，0x31)。



由于在发送端进行了字节填充，因此在链路上传送的信息字节数就超过了原来的信息字节数。但接收端在接收到数据后再进行与发送端字节填充`相反的变换`，就可以正确地恢复出原来的信息。

面向比特的同步链路

> 面向比特的同步链路我们一般使用`比特填充法`插入比特0。

零比特填充的具体方法：

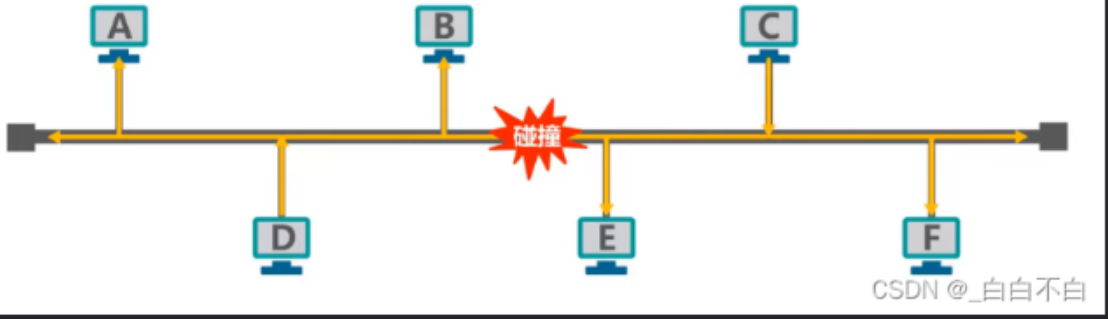
1. 在发送端先扫描整个信息字段(通常使用硬件实现，但也可以用软件实现，但是会慢一些)。
2. 只要发现有5个连续的1，则立即填入一个0。
3. 接收端在收到一个帧时，先找到标志字段F以确定帧的边界，接着再用硬件对其中的比特流进行扫描，每当发现5个连续1时，就把5个连续1后的一个0删除，以还原成原来的信息比特流。

工作状态

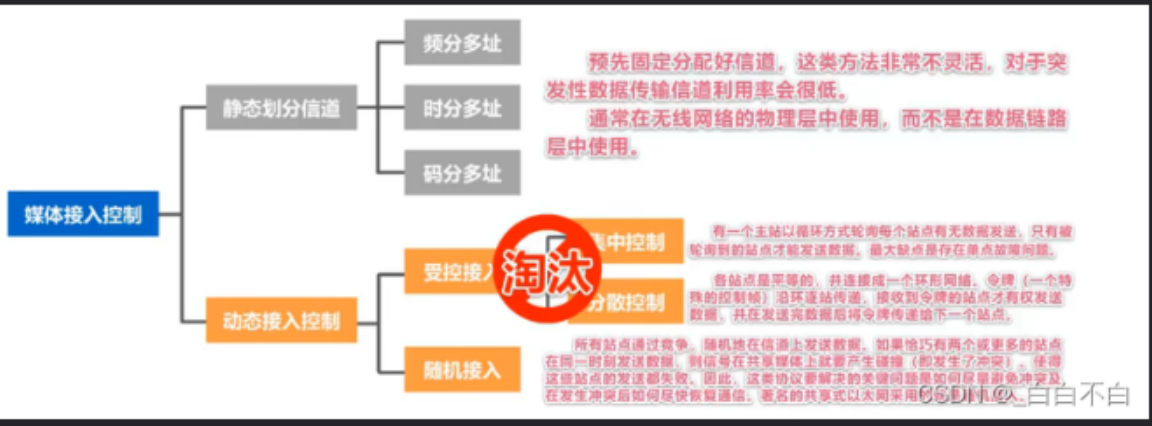
媒体接入控制

媒体接入控制

共享信道要着重考虑的一个问题就是如何协调多个发送和接受站点对一个共享传输媒体的占用，即媒体媒体接入控制 MAC 。



媒体接入控制分为 静态划分信道 和 动态接入控制 。具体如下：



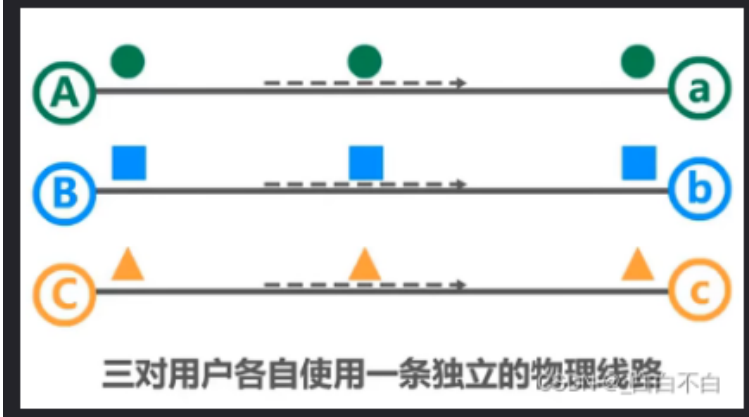
随着技术的发展，交换技术的成熟和成本的降低，具有更高性能的 使用点对点链路和链路层交换机的交换式局域网 在有线领域已完全 取代了共享式局域网 ，但由于无线信道的广播天性， 无线局域网仍然使用的式共享媒体技术 。

静态划分信道

静态划分信道

> **信道复用**是通信技术中的一个重要概念。复用就是通过一条物理线路同时传输多路用户的信号。

当网络中传输媒体的`传输容量大于多条的单一信道传输的总通信量`时，可利用复用技术在一条物理线路上建立多条通信信道来充分利用传输媒体的带宽。



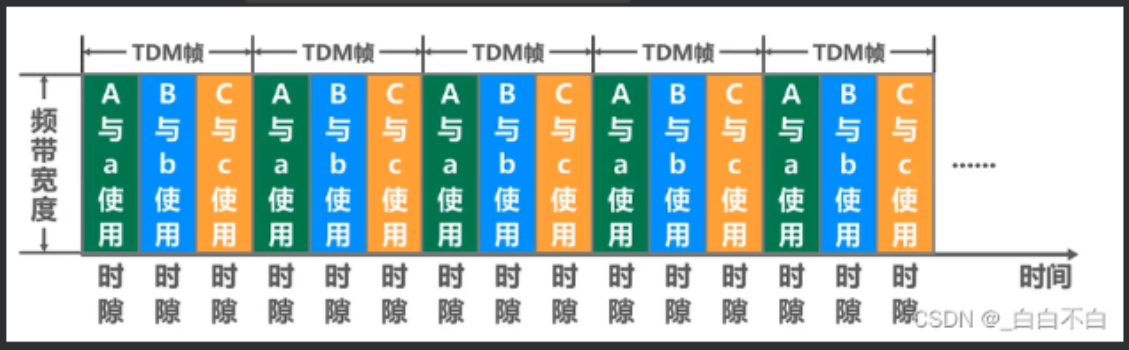
> 信道复用又分为：`频分复用FDM`，`时分复用TDM`，`波分复用WDM`，`码分复用CDM`

频分复用FDM

频分复用的所有用户 占用不同的频带资源并行通信 。

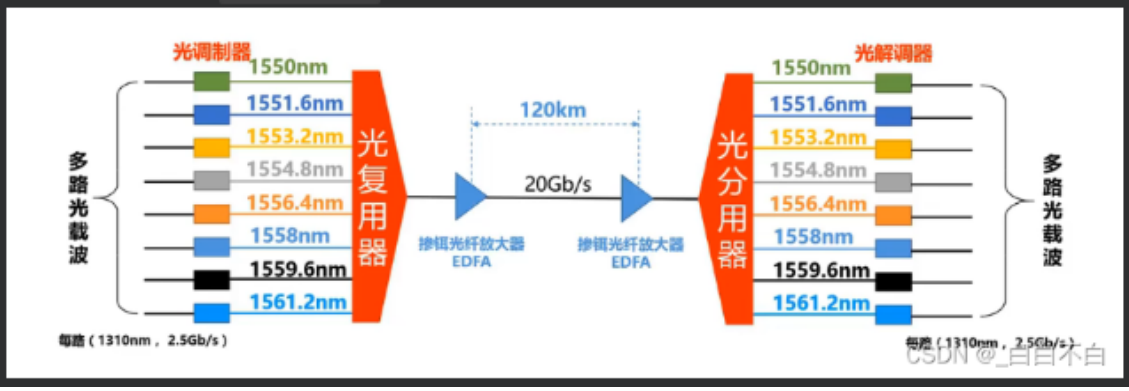
时分复用TMD

时分复用的所有用户 在不同的时间占用同样的频带宽度



波分复用WDM

波分复用其实就是 光的频分复用



码分复用CDM

> 与FDM和TDM不同，CDM的每一个用户 `可以在相同的时间使用同样的频带进行通信` 。

由于各用户使用经过特殊挑选的不同码型，因此各用户之间 `不会造成干扰` 。

码分复用CDM是另一种共享信道的方法，实际上，由于该技术主要用于多址接入，人们更常用的名词是码分多址CDMA。

> 在CDMA中，每一个比特时间在划分为m个短的间隔，称为 `码片` 。

使用CDMA的每一个站被指派一个唯一的m bit码片序列 。

1. 一个站如果要发送比特1，则发送它自己的m bit码片序列；
2. 一个站如果要发送比特0，则发送他自己的m bit码片序列的二进制反码

码片挑选原则如下：

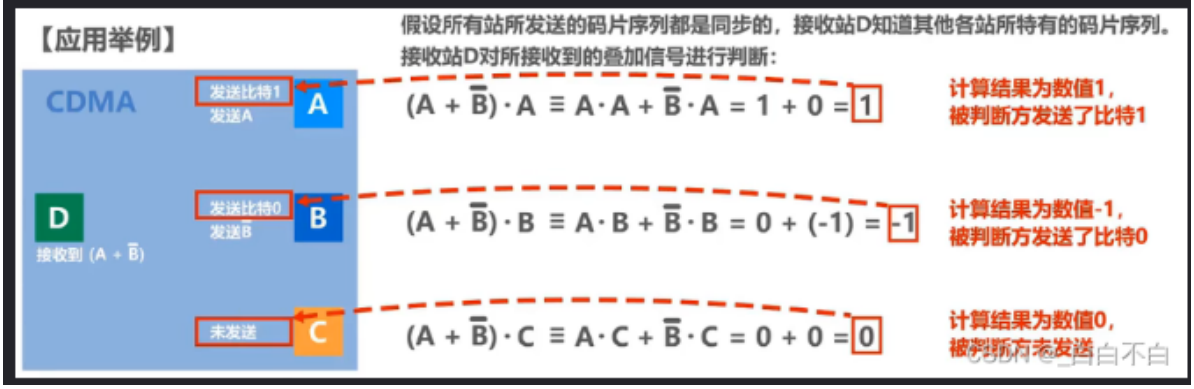
1. 分配给每一个站的码片序列必须各不相同，实际常采用伪随机码序列。
2. 分配给每个站的码片序列必须相互正交（规格化内积为0）。

规格化内积为0，参考：

令向量S表示站S的码片序列，令向量T表示其他任何站的码片序列。
两个不同站S和T的码片序列正交，就是向量S和T的规格化内积。

参考公式 $S \cdot T = 0 \quad S \cdot \bar{T} = 0$
$S \cdot S \equiv 1 \quad S \cdot \bar{S} \equiv -1$

应用举例



动态接入控制

载波监听，多点接入，碰撞检测

动态接入控制

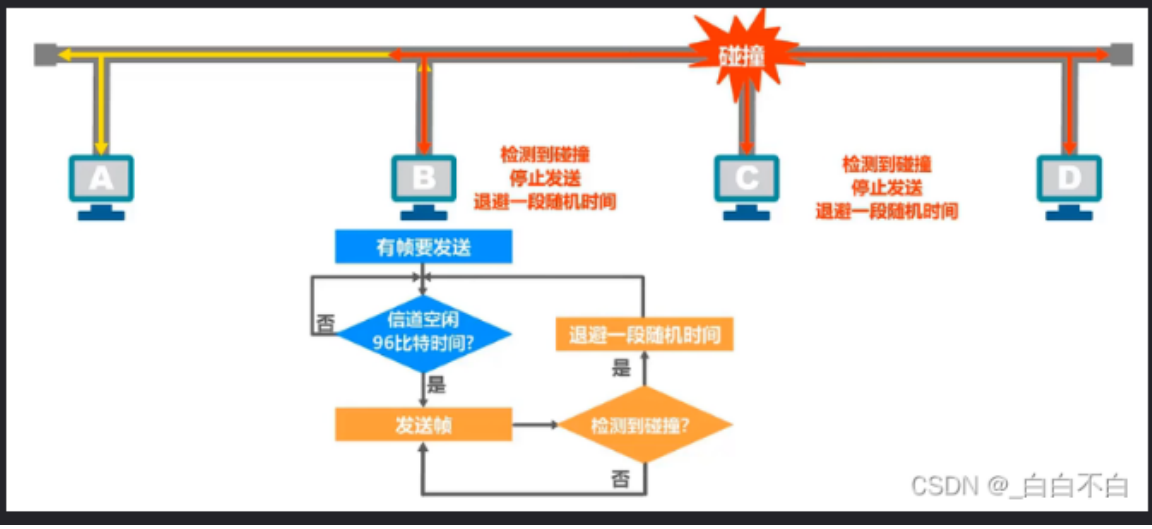
动态接入控制中的受控接入已经被淘汰，我们直接学习随机接入。

随机接入又分为两种：`载波监听多址接入/碰撞检测 CSMA/CD`和`载波监听多址接入/碰撞避免 CSMA/CA`

载波监听多址接入/碰撞检测 CSMA/CD

首先要清楚几个概念：

> （1）多址接入MA：多个站连接在一条总线上，竞争使用总线。
> （2）载波监听CS：每一个站在发送帧之前先要检测一下总线上是否有其他站点在发送帧（先听后说）。
>
> 1. 若检测到总线空闲96比特时间，则发送这个帧；
> 2. 若检测到总线忙，则继续检测并等待总线转为空闲96比特时间，然后再发送这个帧；
>
> （3）碰撞检测CD：每一个正在发送帧的站边发送边检测碰撞（边说边听）。
>
> 1. 一旦发现总线上出现碰撞，立即停止发送，退避一段时间再次发送。



最大帧长

为什么要有最大帧长?

如果一个帧的长度过大，一个站点不停地发送帧，让其他站点一直无法无法使用信道；另外如果帧的长度过大，接收方的缓冲区可能也装不下该帧产生溢出。

最小帧长

为什么要有最小帧长?

因为如果帧的长度过小，站点A在极短时间内将帧a全部发送成功。因为发送动作完成，A不再进行碰撞检测。那么当其他站点发送帧时，和该帧产生了碰撞。接收端检测帧a后，将其丢弃。此时站点A即不能知道帧a发生了碰撞也不会重传该帧。

最小帧长的作用?

保证了站点在帧在发送过程中，能够检测到帧是否发生了碰撞。

1. 若在争用期内没有检测到碰撞，那么后续发送的数据就一定不会发生碰撞（因为争用期中没有发生碰撞，表明无其他站点争用主线，那么只有单个站点进行数据帧的发送。）
2. 若在争用期内检测到碰撞，停止发送数据。(之前发送的帧被接收方进行插错检测后丢弃)

争用期

当帧发生碰撞后会向其发送站点返回碰撞信号，设该帧发送时间为t，则检测到碰撞信号的时间为2t，则在整个信道上，取该时间的最大值为2T为争用期。T为单程端对端的传播时延。

截断二进制指数

当发生碰撞时立即停止发送帧。隔一段时间后重新发送。那么需要隔多长时间呢?

> 退避时间 = 争用期（2t）* 随机次数r

CSMA/CD协议曾经用于各种总线结构以太网和双绞线以太网的早期版本中。
现在的以太网基于交换机和全双工连接，不会有碰撞，因此没有必要使用CSMA/CD协议。

虚拟局域网VLAN

避免广播风暴！！！

随着交换机以太网模式的扩大，广播域相应扩大。

但是巨大的广播域会带来很多弊端：广播风暴、难以管理和维护、潜在的安全问题。

那么我们就得对广播域进行分割，于是虚拟局域网VLAN技术应运而生。

> **虚拟局域网**：是局域网向用户提供的一种服务，虚拟局域网是用户和局域网资源的一种逻辑组合，而交换式局域网技术是实现虚拟局域网的基础。

虚拟局域网的基本概念

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属于哪一个VLAN`。

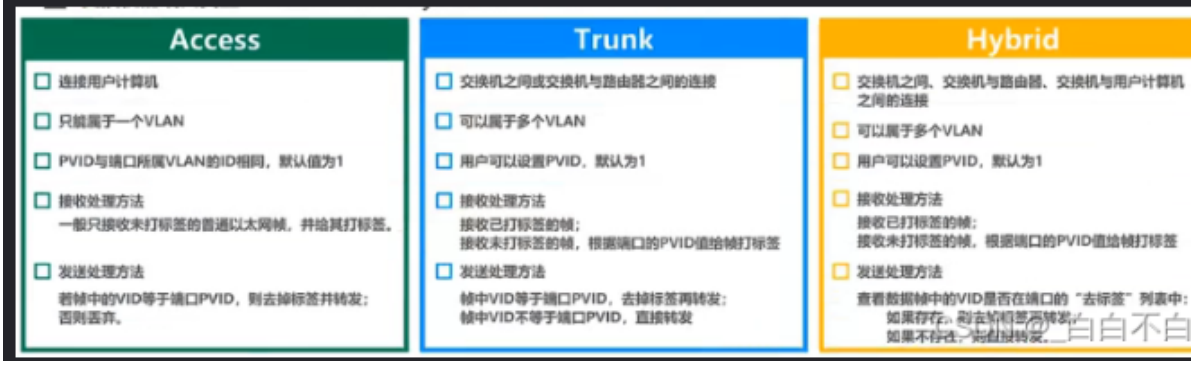
传统的局域网中的工作组通常在同一个网段上，多个工作组之间通过实现互联的网桥或者路由器来交换数据。当一个逻辑工作组的结点要转移到另一个逻辑工作组时，就需要将结点计算机从一个网段撤出，并将其连接到另外一个网段上，这时甚至需要重新进行布线。因此，逻辑工作组的组成受结点所在网段的物理位置限制。

1988年IEEE批准了802.3ac标准，这个标准定义了以太网的帧格式的扩展，以便支持虚拟局域网。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（见图），称为VLAN标记(tag)，用来指明发送该帧的计算机属于哪一个虚拟局域网。插入VLAN标记得出的帧称为802.1Q帧。显然，如果还使用原来的以太网帧格式，那么就无法区分是否划分了虚拟局域网。图3-27标注出在几个粗线链路上传输的帧是802.1Q帧。在其他链路上传输的仍然是普通的以太网帧。

802.1Q帧是由交换机来处理的不是用户主机来处理：

1. 当交换机收到普通的以太网帧时，会插入4字节的VLAN标记转变为802.1Q帧，简称“打标签”。
2. 当交换机转发802.1Q帧时，可能会删除其4字节的VLAN标记转变为普通以太网帧，简称“去标签”。

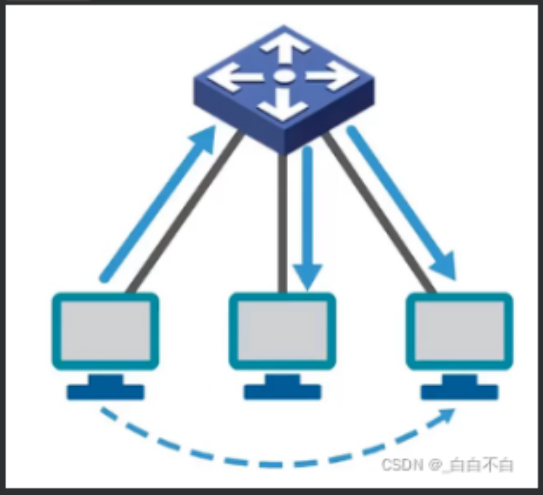
交换机的端口类型：Access、Trunk、Hybrid



集线器与交换机（前者是物理层，交换机和网桥是链路层）

集线器

集线器：（`Hub`）是指将多条以太网双绞线或光纤集合连接在同一段物理介质下的设备。发生在`物理层`。



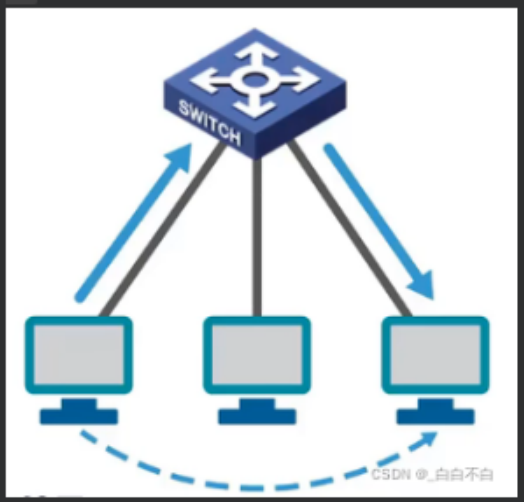
对于集线器我们要知道：

（1）集线器是早期以太网的`互连设备`。
（2）工作在OSI体系结构的物理层。
（3）对接收到的信号进行放大、进行`盲目转发`。
（4）使用集线器作为互连设备的以太网仍然属于`共享总线式以太网`，集线器互连起来的所有主机共享总线带宽，属于`同一个碰撞域和广播域`。

这中设备已经过时了，已经被时代所淘汰。现在大多使用的是交换机。

交换机

交换机：（`Switch`）是一种用于电（光）信号转发的网络设备。它可以为接入交换机的任意两个网络节点提供独享的电信号通路，把传输的信息送到符合要求的相应路由上。发生在`数据链路层`。

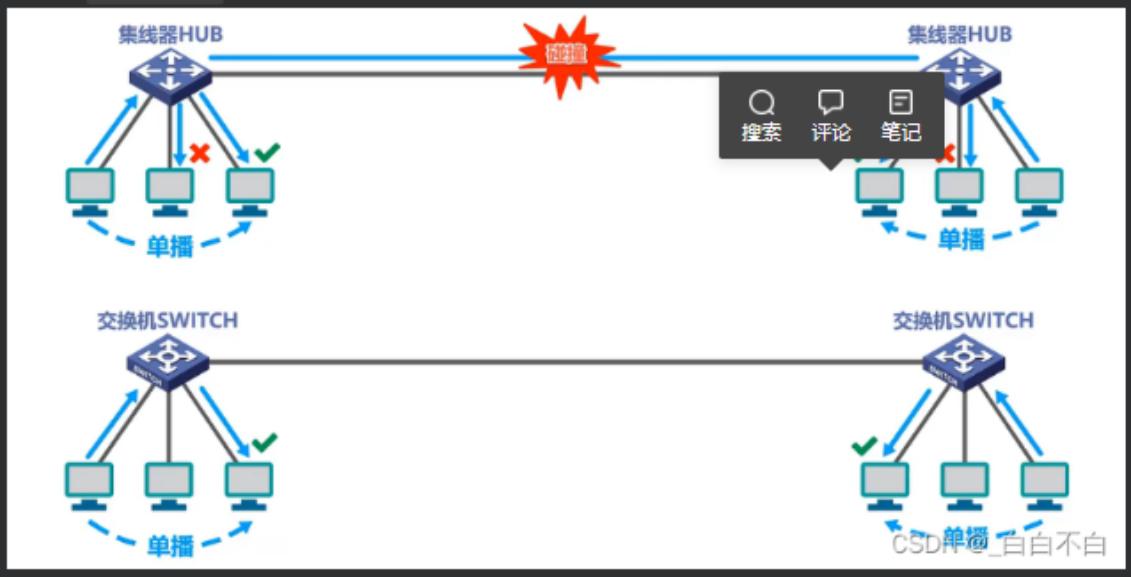


对于交换机我们要知道：
对于交换机我们要知道：

（1）交换机式目前在以太网中`使用最广泛`的`互连设备。`
（2）工作在OSI体系结构的数据链路层（也包括物理层）。

(3) 与集线器不同的是交换机对帧进行转发是根据其MAC地址进行转发的。

(4) 使用交换机做为互联设备的以太网，称为交换式以太网。交换机可以根据MAC地址过滤帧，即隔离碰撞域。



(5) 交换机的每一个接口是一个独立的碰撞域。

(6) 交换机隔离碰撞域但不隔离广播域（VLAN除外）。

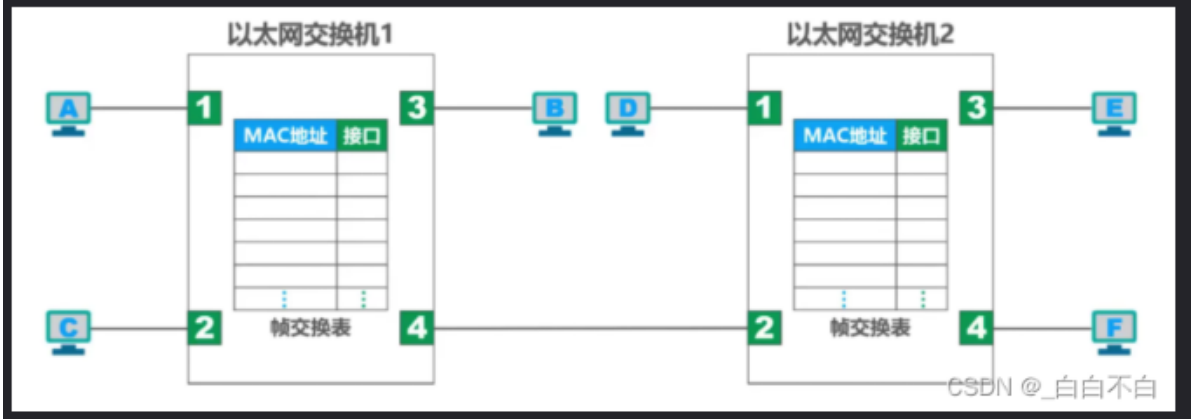
交换机自学习和转发帧

（1）以太网交换机工作在数据链路层（也包括物理层）。

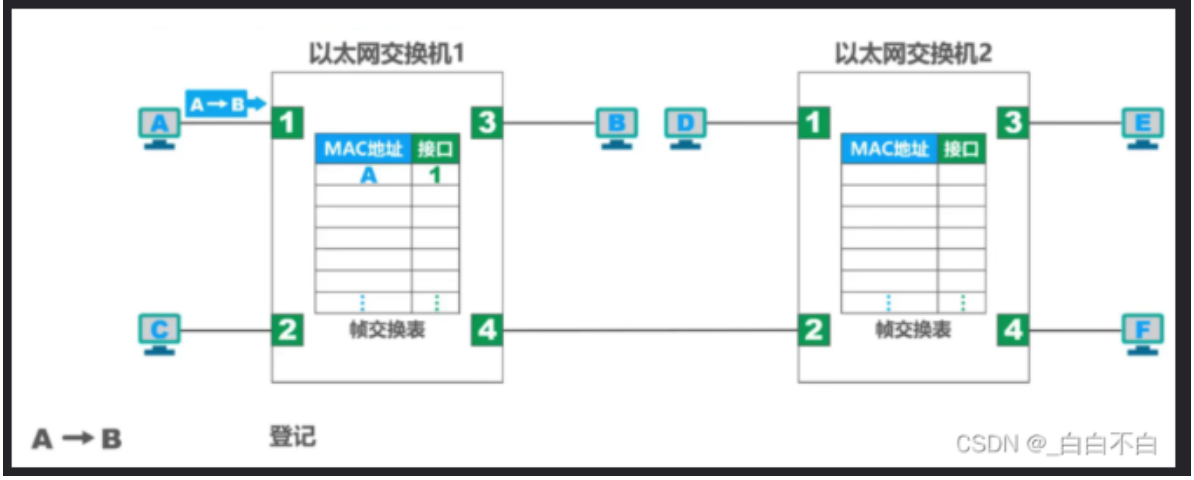
（2）以太网交换机收到帧后，在帧交换表中查找帧的目的MAC地址所对应的接口号，然后通过该接口转发帧。

（3）以太网交换机是一种即插即用设备，刚上电启动时其内部的帧交换表是空的，随着网络中各主机之间的通信，以太网交换机通过自学习算法自动逐渐建立起帧交换表。

下面我们来举例说明以太网交换机自学习和转发帧的过程。

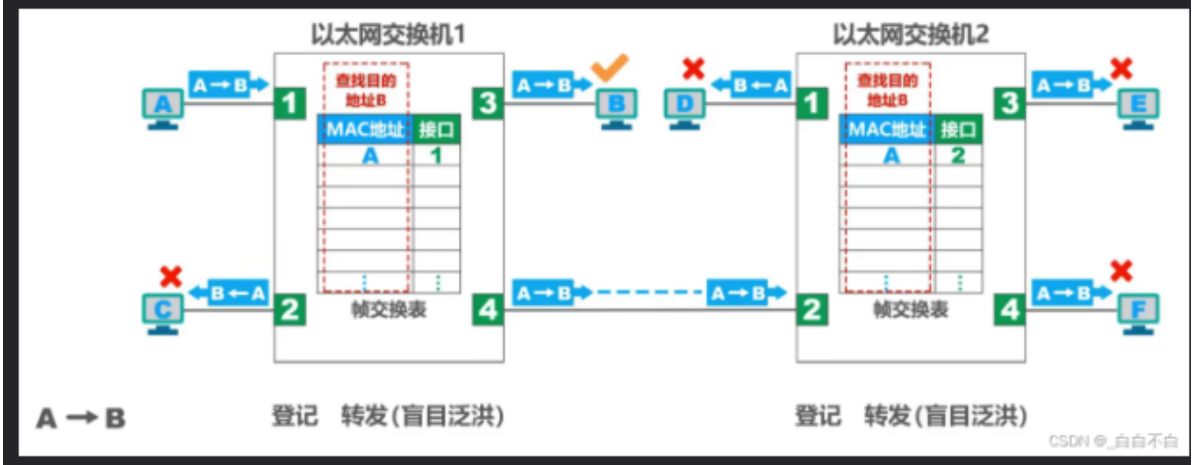


假设A给B发送帧，该帧从交换机1的接口1进入交换机1，交换机1首先进行登记的工作，将该帧的源MAC地址A记录到自己的帧交换表中。将该帧进入到自己的接口号1，相应的也记录到交换表中，上述登记工作就称为交换机的自学习。



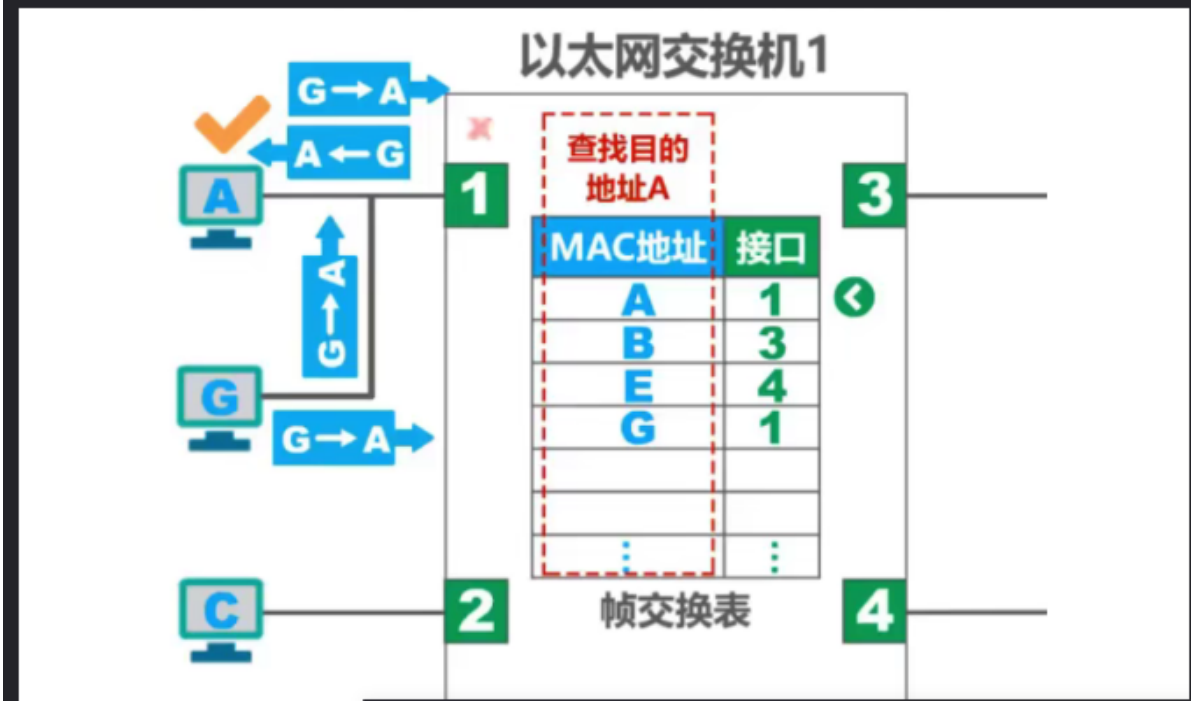
之后交换机1对该帧进行转发，该帧的目的MAC地址是B。在帧交换表中查找该帧的目的交换地址B，找不到，于是对该帧进行盲目的转发，发到除进入接口的其它接口。可以看出交换机一开始还是比较笨的，他还没有足够的知识去明确转发帧。主机B的网卡收到该帧后，根据该帧的目的MAC地址知道是发送给自己的，于是接收。主机C发现不是自己的，无情丢弃。之后

交换机1通过接口4把帧发送到交换机2中，交换机2重复上面的流程。



我们现在来看看B给A传帧的情况，现在交换机1的帧交换表中已经有A的MAC地址和接口了，所以这次发送是明确的发送，不涉及其它主机。

接下来我们看看丢弃的情况：



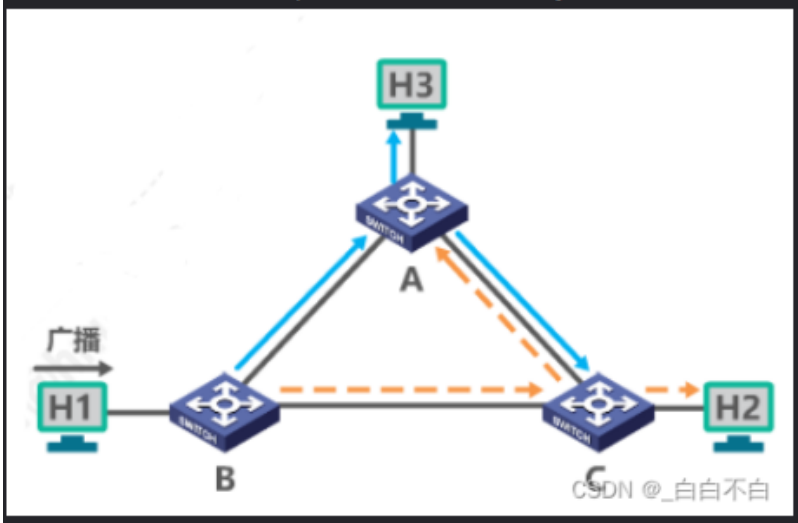
G给A发的帧到达A和交换机1，交换机1查表发现是A1，但是该帧就是从1来的，不会再转发回去。于是丢弃。

需要注意的是：帧交换表中的每条记录都有自己的存活时间，到期自动删除。为什么呢?

MAC地址和接口对应关系会改变.

交换机生成树协议STP

为了提高以太网的可靠性，我们添加了冗余链路。



但是冗余链路会生成网络链路。于是我们改用了生成树协议。

> **生成树协议**（`spanning-tree-protocol,stp`），就是在具有物理环路的交换机网络上生成没有回路的逻辑网络的方法。

1. 生成树协议使用生成树算法，在一个具有冗余路径的容错网络中`计算出一个无环路的路径`，使一部分端口处于转发状态，另一部分处于阻塞状态（备份状态），`从而生成一个稳定的、无环路的生成树网络拓扑`。
2. 一旦发现当前路径故障，生成树协议能立即激活相应的端口，打开备用链路，重新生成STP网络拓扑，从而保持网络的正常工作。
3. 最终生成的树型逻辑拓扑要保证连通整个网络。