

B站链接: <https://www.bilibili.com/video/BV1c4411d7jb?p=1>

关于其中各种协议的实现可看: [经典实验整理总结](#)

第1章 计算机概述

1. 各种网络

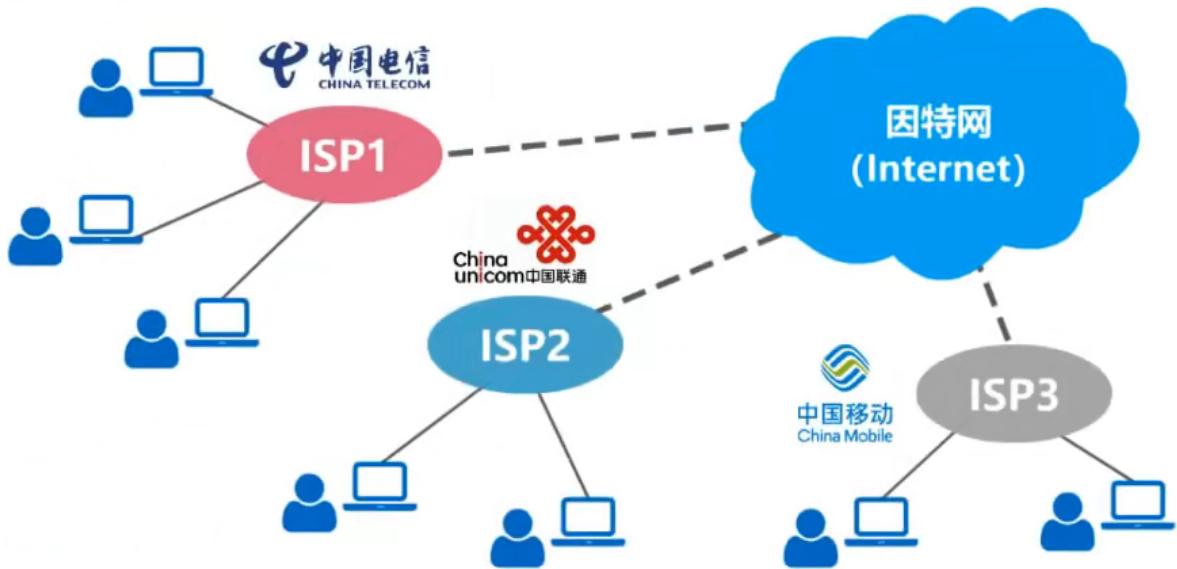
- 网络(Network)由若干结点(Node)和连接这些结点的链路(Link)组成
- 多个网络还可以通过路由器互连起来, 这样就构成了一个覆盖范围更大的网络, 即互联网(互连网)。因此, 互联网是"网络的网络(Network of Networks)"
- 因特网(Internet)是世界上最大的互连网络【小写i的internet是通用名词, 互连的网络都叫internet】

2. ISP

因特网服务提供者(Internet Service Provider)

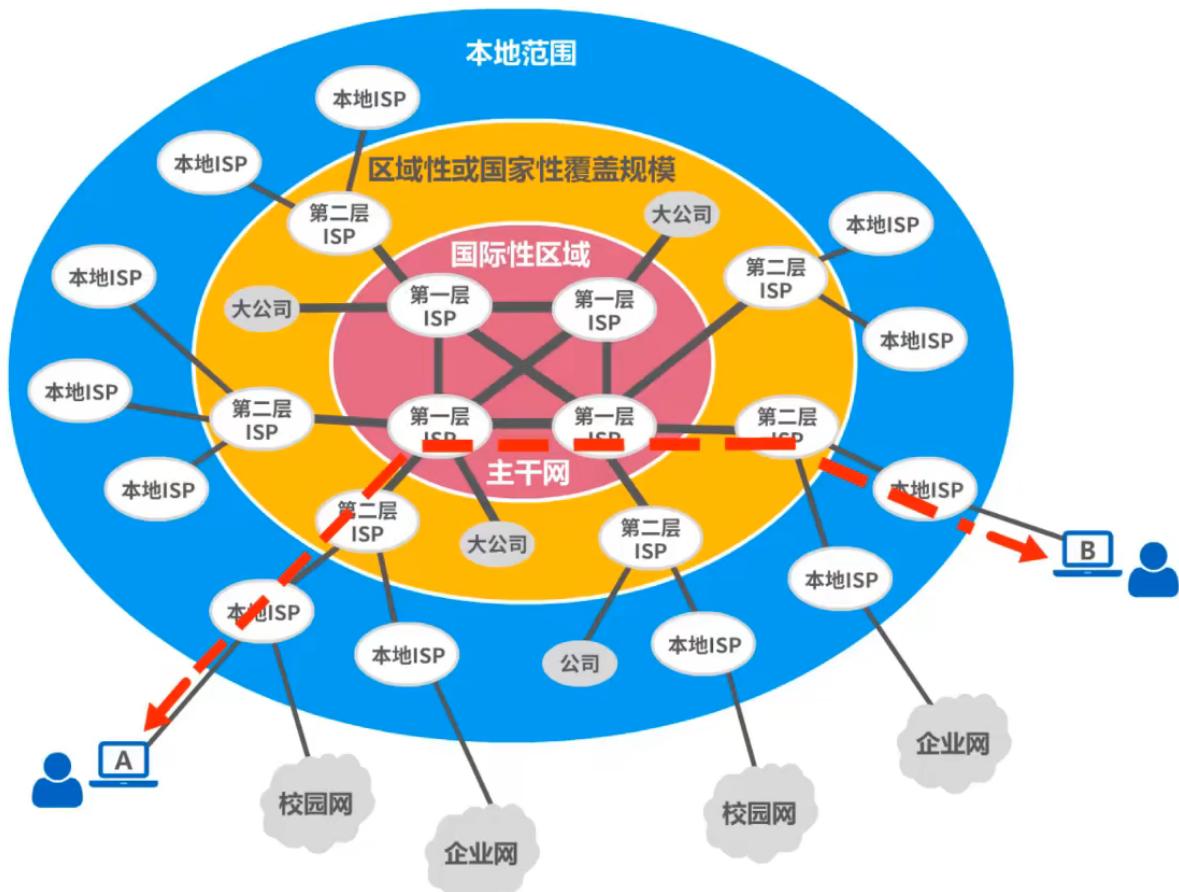
普通用户如何接入因特网?

通过ISP接入因特网。ISP可以从因特网管理机构申请到成块的IP地址, 同时拥有通信线路以及路由器等连网设备, 任何机构和个人只要向ISP交纳规定的费用, 就可以从ISP得到所需要的IP地址。互联网上的主机都必须有IP地址才能通信



基于ISP的三层结构的因特网

层数越小覆盖越多



3. 因特网的组成

- **边缘部分**: 由所有连接在因特网上的**主机**组成，这部分是**用户直接使用**的，用来进行**通信**(传送数据、音频或视频)和**资源共享**
- **核心部分**: 由**大量网络**和连接这些网络的**路由器**组成。这部分是**为边缘部分提供服务的**(提供连通性和交换)

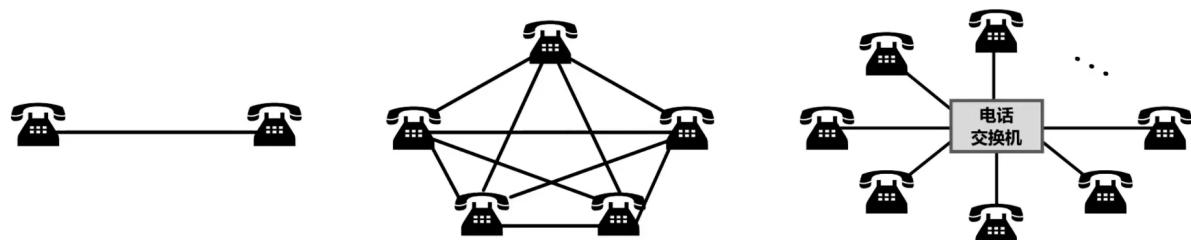


4. 三种交换方式

① 电路交换(Circuit Switching)

1. 电路交换怎么出现的?

电话问世后，人们发现所有电话之间都两两相连是不现实的。因此可以用一个中间设备将讲话接入，根据需要进行转发



2. 什么是电路交换?

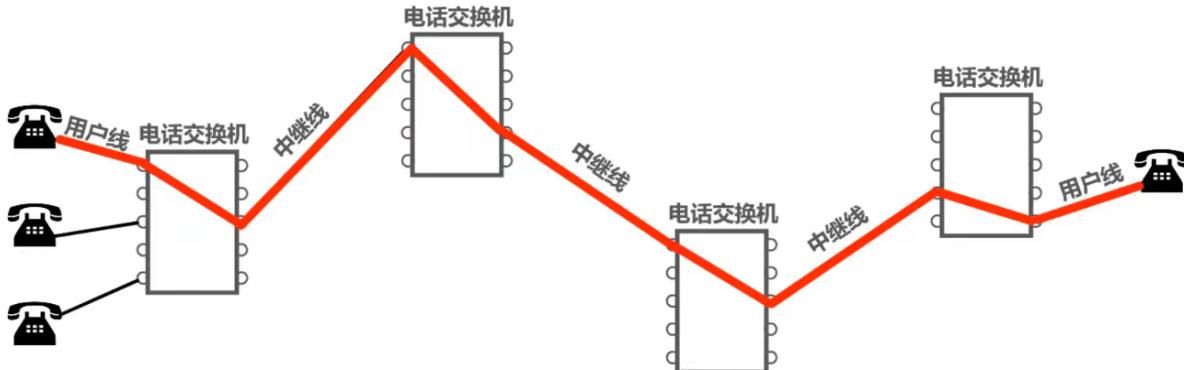
电话交换机接通电话线的方式称为电路交换
从通信资源分配角度来看，交换(switch)就是按照某种方式动态地分配传输线路的资源

3. 电路交换三个步骤

- 建立连接(分配通信资源)**: 例如在使用电路交换打电话前，必须先拨号请求建立连接，当被叫用户听到电话交换机送来的拨号音并拿起电话后，从主叫端到被叫端就建立了一条连接，也就是一条**专用的物理通**

路。这条连接保证了双方通话时所需的通信资源，而这些资源在双方通信中不会被其他用户占用

2. 通话(一直占用通信资源)：分配的资源始终被占用
3. 释放连接(归还通信资源)



优点

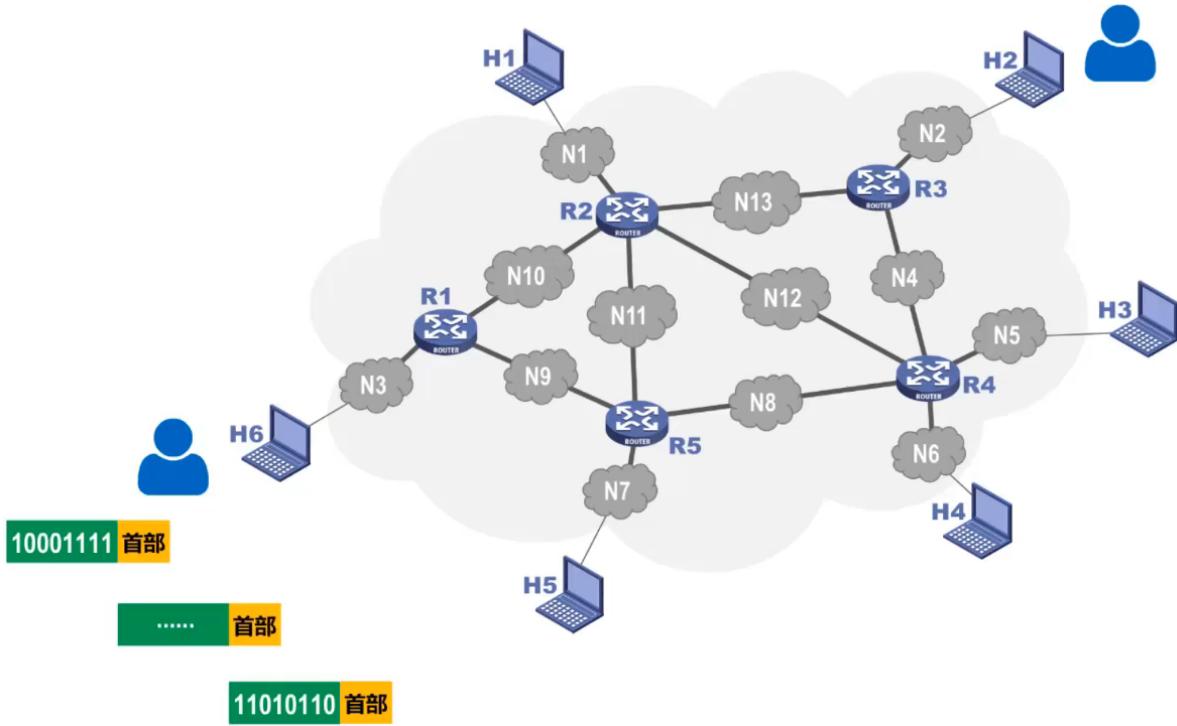
1. **通信时延小**：通信线路为通信双方专用的，数据直达
2. **有序传输**：通信双方只有一条专用通信线路，数据只在这一线上上传输，不存在失序问题
3. **没有冲突**：不同的通信双方拥有不同的信道，不会出现争用物理通道的问题
4. **适用范围广**：适用于传输模拟信号，也适用于传输数字信号
5. **实时性强**：时延小所以实时性强
6. **控制简单**

缺点

1. **建立连接时间长**
2. **线路独占，适用效率低**
3. **灵活性差**：只要连接所建立的物理通路中的任何一点出现了故障，就必须重新拨号建立新的连接
4. **难以规格化**：不同类型、不同规格、不同速率的终端很难互相进行通信，也难以差错控制

②★分组交换(Packet Switching)

- **发送方**：①构建分组 ②发送分组
- **路由器**：①缓存分组 ②转发分组
- **接收方**：①接收分组 ②还原报文



优点

1. **无需建立连接**
2. **线路利用率高**
3. **简化存储管理**: 因为分组大小固定，管理起来就容易一些
4. **加速传输**: 因为分组是逐个传输，这样前一个交换机的转发操作与后一个交换机的存储操作可同时进行
5. **减少出错率和重复数据量**: 分组比报文小，因此出错概率也会比较小，即使出错也只需要重传出错的这一小部分即可

缺点

1. 引起了**转发时延**
2. 需要传输**额外信息量**(分组头部信息)
3. 当**分组交换采用数据报服务时**，可能会出现**失序、丢失、重复分组**。分组到达目的结点时，需要**重新还原成原始报文**，比较麻烦。若分组交换**采用虚电路服务**，虽然没有分组失序问题，但**有呼叫建立，数据传输和虚电路释放三个过程**

③报文交换(Message Switching)

报文交换与分组交换类似，不过对报文没有限制大小，现如今多使用分组交换。

优点

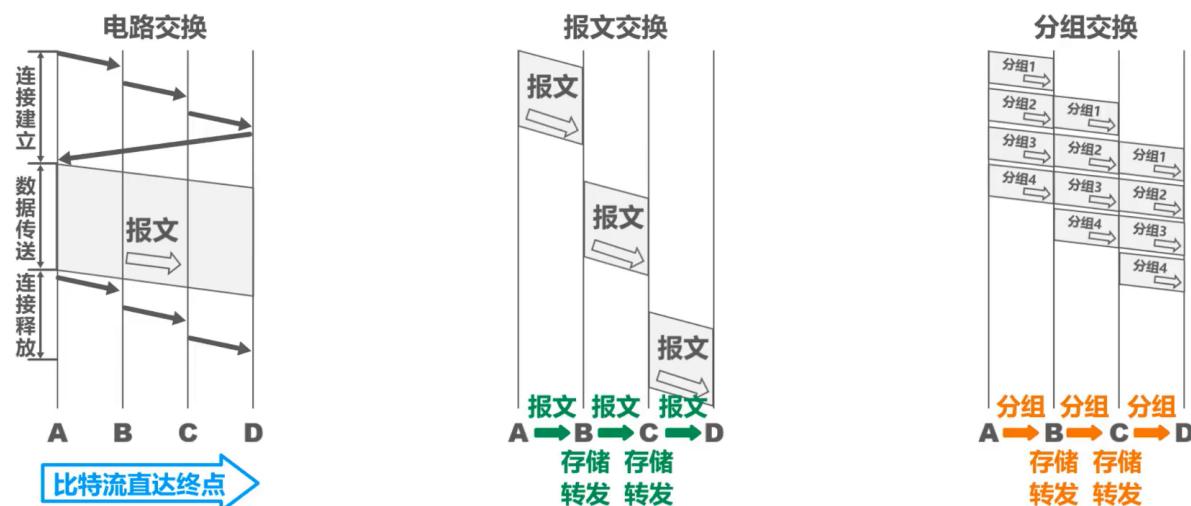
1. 无需建立连接
2. 动态分配线路
3. 提高线路可靠性：如果某条线路出现故障，会重新选择另一条线路
4. 提高线路利用率：通信双方在不同的时间分段占用物理线路
5. 提供多目标服务：一个报文可以同时发送给多个目的地址

缺点

1. 引起转发时延：报文在节点交换机上要经历存储转发的过程
2. 需要较大的存储转发空间：因为报文大小没有限制
3. 需要传输额外信息量：有头部等信息

三种交换对比

纵坐标为时间，分组交换相对报文交换分的更小，可以减少时延，防止过长时间占用线路以及方便排错



5. 计算机网络

定义

一些互相连接的、自治的计算机的集合

- **互连：**指计算机之间可以通过有线或无线的方式进行数据通信
- **自治：**指独立的计算机，它有自己的硬件和软件，可以单独运行使用
- **集合：**指至少需要两台计算机

分类

按交换技术：①电路交换网络 ②报文交换网络 ③分组交换网络

按使用者：①公用网 ②专用网

按传输介质：①有线网络 ②无线网络

按覆盖范围：①广域网WAN ②城域网MAN ③局域网LAN ④个域网PAN

按拓扑结构：①总线型网络 ②星型网络 ③环型网络 ④网状型网络

性能指标

性能指标可以从不同的方面来度量计算机网络的性能

1. 速率

连接在计算机网络上的主机在数字信道上传送比特的速率，也称为比特率或数据率

常用数据率单位

bit/s 可缩写为 b/s 或 bps

$$1 kb/s = 10^3 b/s$$

$$1 Mb/s = 10^6 b/s$$

$$1 Gb/s = 10^9 b/s$$

$$1 Tb/s = 10^{12} b/s$$

比特

计算机中的数据量单位，也是信息论中信息量的单位。一个比特就是二进制数字中的一个 1 或 0。

常用数据量

$$8 bit = 1 Byte$$

$$1 KB = 2^{10} B$$

$$1 MB = 2^{20} B$$

$$1 GB = 2^{30} B$$

$$1 TB = 2^{40} B$$

2. 带宽

带宽在模拟信号系统中的意义

信号所包含的各种不同频率成分所占据的频率范围

单位: Hz (kHz, MHz, GHz)

带宽在计算机网络中的意义

用来表示网络的通信线路所能传送数据的能力, 因此网络带宽表示在单位时间内从网络中的某一点到另一点所能通过的"最高数据率"

单位: b/s (kb/s, Mb/s, Gb/s, Tb/s)

一条通信线路的"频带宽度"越宽, 其所传输数据的"最高数据率"也越高

3. 吞吐量

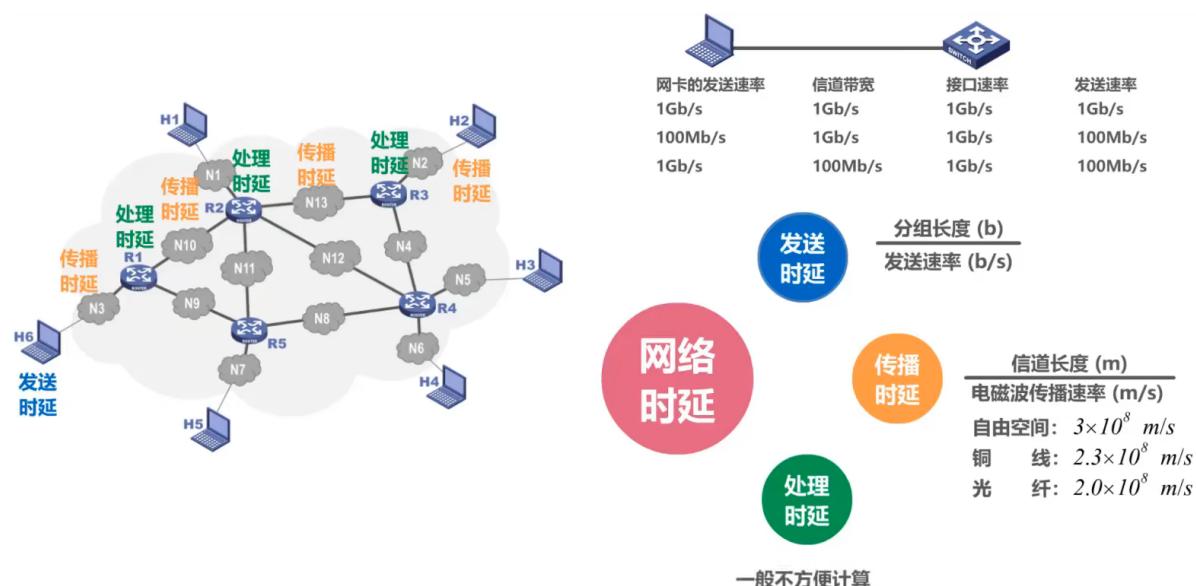
吞吐量表示在单位时间内通过某个网络(或信道、接口)的数据量。

吞吐量被经常用于对现实世界中的网络的一种测量, 以便知道实际上到底有多少数据量能够通过网络。

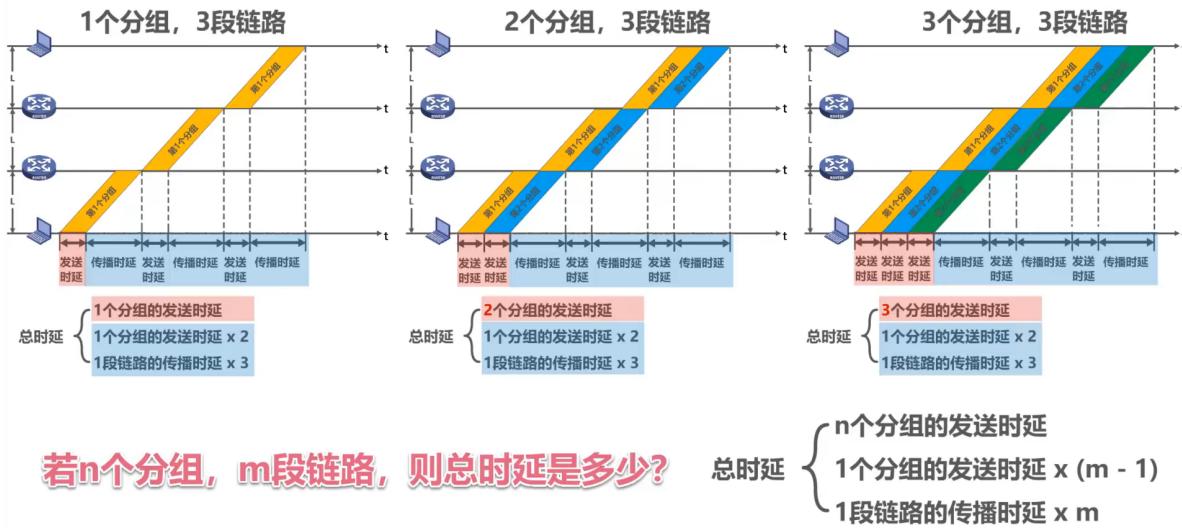
吞吐量受网络的带宽或额定速率的限制

4. ★时延

在处理过程中所需要的时间



假设：分组等长，各链路长度相同、带宽也相同，忽略路由器的处理时延



5. 时延带宽积

$$\text{时延带宽积} = \text{传播时延} \times \text{带宽}$$



- 若发送端连续发送数据，则在所发送的第一个比特即将到达终点时，发送端就已经发送了时延带宽积个比特；
 - 链路的时延带宽积又称为**以比特为单位的链路长度**。

6. 往返时间

在许多情况下，因特网上的信息不仅仅单方向传输，而是双向交互。我们有时很需要知道双向交互一次所需要的时间。因此，往返时间 RTT (Round-Trip Time) 也是一个重要的性能指标。

7. ★利用率

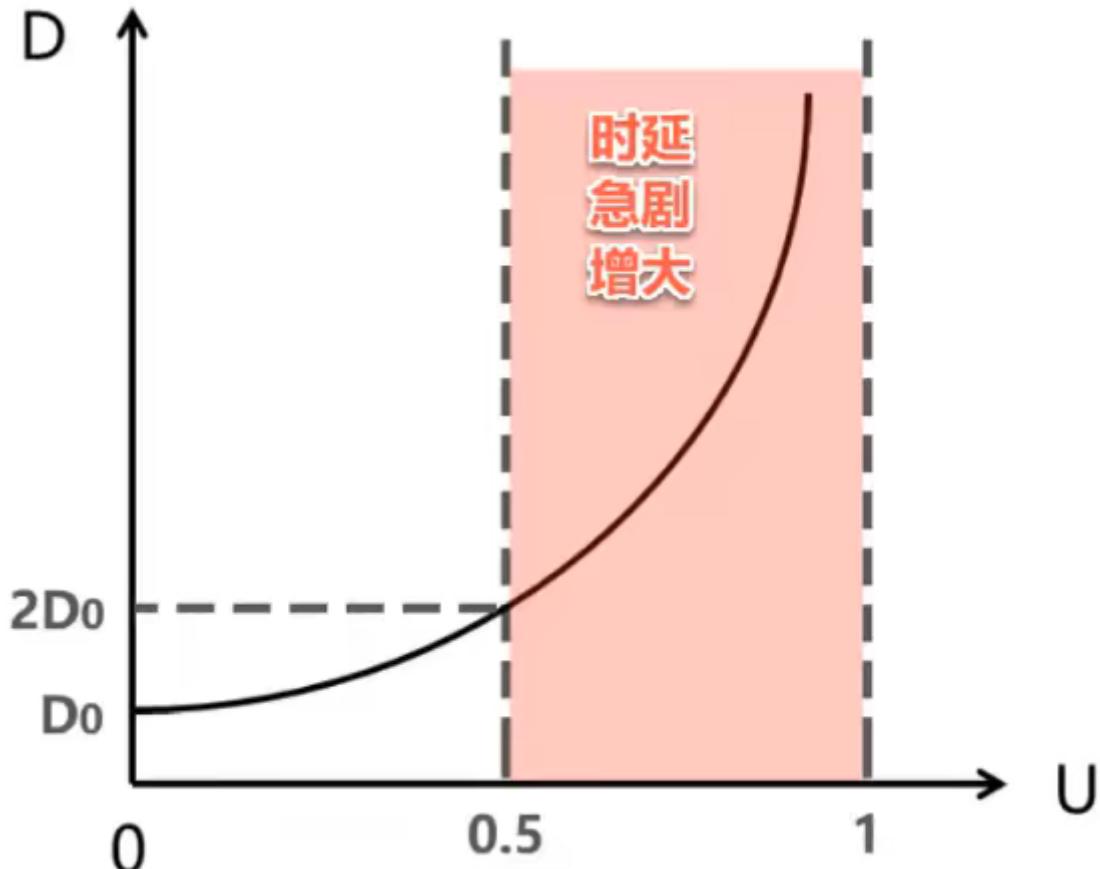
信道利用率：用来表示某信道有百分之几的时间是被利用的(有数据通过)

网络利用率: 全网络的信道利用率的加权平均

- 根据排队论，当某信道的利用率增大时，该信道引起的时延也会迅速增加。因此**信道利用率不是越高越好**

- 如果令 D_0 表示网络空闲时的时延， D 表示网络当前的时延，那么在适当的假定条件下，可以用下面的简单公式来表示 D 、 D_0 和 利用率 U 之间的关系

$$D = \frac{D_0}{1 - U}$$



当网络利用率 U 到 50% 时，时延急剧增大。

当网络利用率接近 100% 时，时延趋于无穷大

但是也不能使信道利用率过低，这回使宝贵的通信资源被浪费

因此一些拥有较大主干网的 ISP 通常会控制它们的信道利用率不超过 50%。

如果超过了，就要准备扩容，增大线路带宽

8. 丢包率

丢包率即分组丢失率，是指在一定的时间范围内，传输过程中 **丢失的分组数量与总分组数量的比率**

具体可分为：接口丢包率、结点丢包率、链路丢包率、路径丢包率、网络丢包率等

丢包率是网络运维人员非常关心的一个网络性能指标，但对于普通用户来说往往并不关心这个指标，因为他们意识不到丢包

分组丢失的两种情况

- 分组在传输过程中出现误码，被结点丢弃
- 分组到达一台队列已满的分组交换机时被丢弃，在通信量较大时就可能造成网络拥塞

丢包率反映了网络的拥塞情况

- **无拥塞**时路径丢包率为 0
- 轻度拥塞时路径丢包率为 1%~4%
- **严重拥塞**时路径丢包率为 5%~15%

★体系结构

原理体系结构

5	应用层	解决通过应用进程的交互来实现特定网络应用的问题
4	运输层	解决进程之间基于网络的通信问题
3	网络层	解决分组在多个网络上传输（路由）的问题
2	数据链路层	解决分组在一个网络（或一段链路）上传输的问题
1	物理层	解决使用何种信号来传输比特的问题

1. 分层的必要性

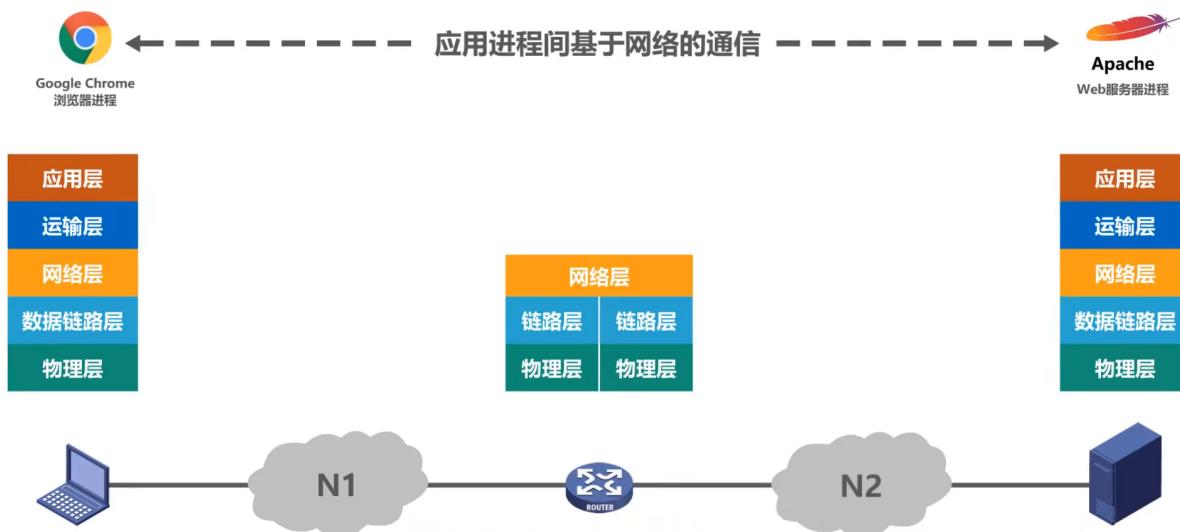
在平常编程时，我们总是喜欢利用不同的类实现不同的功能，最后进行整合实现真正的功能。这样的好处是让结构更加清晰，维护也更加简单。计算机网络分层同理，在计算机网络上实现不同进程的通信需要解决众多问题，分层便于维护与管理。

原理体系结构

5	应用层	解决通过应用进程的交互来实现特定网络应用的问题
4	运输层	解决进程之间基于网络的通信问题
3	网络层	解决分组在多个网络上传输（路由）的问题
2	数据链路层	解决分组在一个网络（或一段链路）上传输的问题
1	物理层	解决使用何种信号来传输比特的问题

2. 分层举例

当我们利用浏览器发送网页请求到服务器并发生响应的过程中，数据会怎么变化呢？



浏览器发送

- 应用层按照 HTTP 协议的规定构建一个 **HTTP请求报文(请求要干什么)**。
应用层将 **HTTP请求的报文交给运输层处理**
- 运输层给 **HTTP请求报文添加一个TCP首部(区分应用进程)**，使之成为 **TCP报文段**。 **运输层将TCP报文段交给网络层处理**
- 网络层给 **TCP报文段添加一个IP首部(使之可以在互联网上传输)**，使之成为 **IP数据报**。 **网络层将IP数据报交付给数据链路层处理**
- 数据链路层给 **IP数据报添加一个首部(让其能在一段链路上传输，能被相应主机接收)**和一个尾部(**让目的主机检查所接收到的帧是否有误码**)，使之成为 **帧**。 **数据链路层将帧交给物理层**
- 物理层将 **帧看作比特流**，如果网络是以太网，它还会在帧上加 **前导码(让目的主机做好接收帧的准备)**。接着将 **比特流变成相应信号发送到传**

输出媒体

路由器转发

- 物理层收到信号将其变为比特流，去掉前导码后，将其交付给数据链路层(交付的实际是帧)
- 数据链路层将帧去掉首部和尾部后，将其交付给网络层(交付的实际是IP数据报)
- 网络层解析IP数据报首部，从中提取目的网络地址，然后查找自身路由表，确定转发端口。接着数据链路层封装，物理层再封装，将比特流变成信号发送出去。

服务器接收

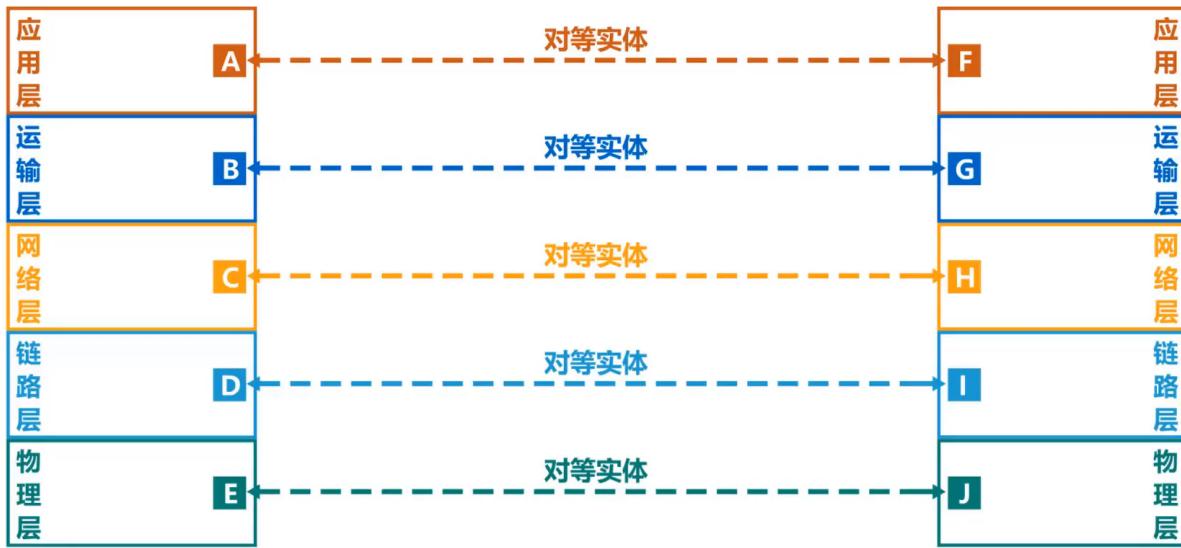
- 物理层收到信号将其变为比特流，去掉前导码后，将其交付给数据链路层(交付的实际是帧)
- 数据链路层收到帧后，去除首部和尾部，将其交付给网络层(交付的实际是IP数据报)
- 网络层收到IP数据报后，去除IP首部，将其交付给运输层(交付的实际是TCP报文)
- 运输层收到TCP报文后，从中得知是与哪个端口上的进程通信，去除TCP头部后，交付给应用层(交付的实际是HTTP请求报文)
- 应用层收到HTTP请求报文后，将其解析给对应进程，并执行相关操作，返回HTTP响应报文

专用术语

① 实体

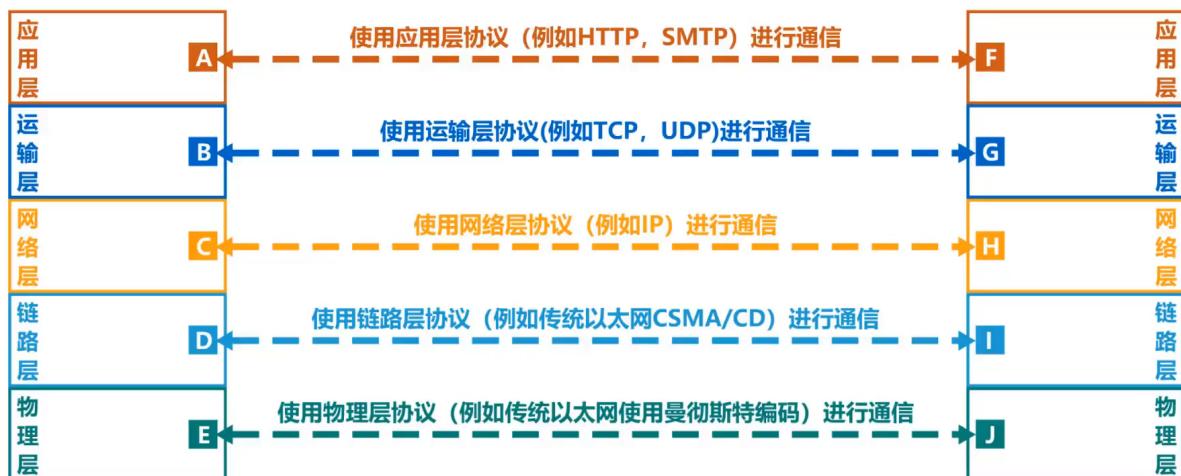
任何可发送或接收信息的硬件或者软件进程

对等实体：收发双方相同层次中的实体



②协议

控制两个对等实体进行逻辑通信(这种通信实际上不存在，只是便于我们考虑问题)的规则的集合



三要素

- **语法**: 定义所交换信息的格式(即报文格式)
- **语义**: 定义收发双方所要完成的操作(即收，发任务需要各自定义)
- **同步**: 定义收发双方的时序关系(如先建立侦听才可以通信是有先后顺序的)

③服务

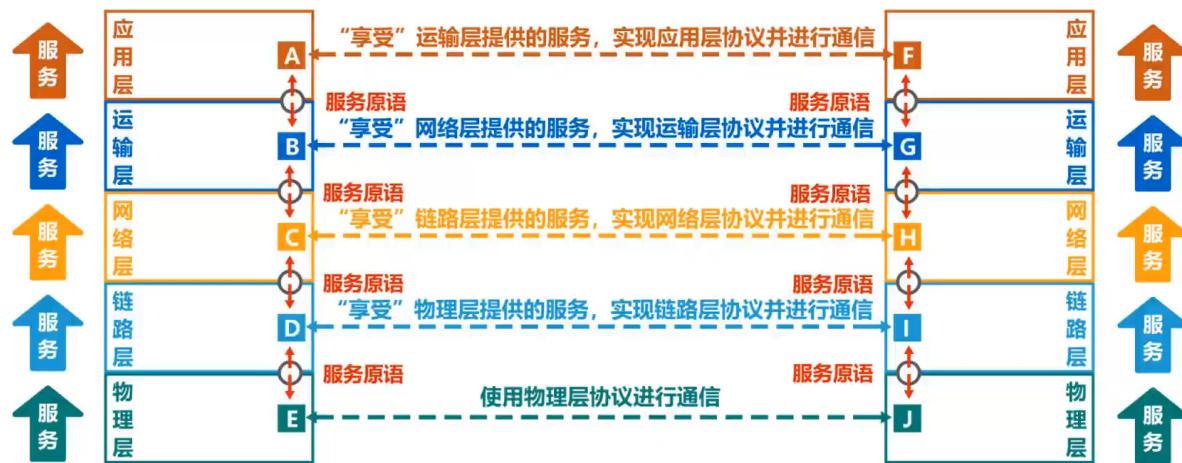
- 在协议控制下，两个对等实体间的逻辑通信使得本层能够**向上一层提供服务**(也就是说通过协议完成本层的内容后就可以向上提供服务)
- 要实现本层协议，还需要使用下面一层所提供的服务
- 协议是“**水平的**”，服务是“**垂直的**”

- 实体看得见相邻下层所提供的的服务，但是并不知道实现该服务的具体协议。也就是说，下面的协议对上面的实体是“透明”的（就像手机为我们提供服务，但是我们并不知道具体是如何实现的）

服务访问点：在同一系统中相邻两层的实体交换信息的逻辑接口（就像Web里的request域，后端前端都能取到），用于区分不同的服务类型

- 数据链路层的服务访问点为帧的“类型”字段
- 网络层的服务访问点位IP数据报首部中的“协议字段”
- 运输层的服务访问点为“端口号”

服务原语：上层使用下层所提供的服务必须通过与下层交换一些命令，这些命令称为服务原语



协议数据单元PDU(横向)：对等层次之间传送的数据包称为该层的协议数据单元

- 物理层是比特流；数据链路层是帧；网络层是IP数据报或分组.....

服务数据单元SDU(竖向)：同一系统内，层与层之间交换的数据报称为服务数据单元

- 物理层往上送是比特流；数据链路层往下送是帧.....

多个SDU可用合成为一个PDU；一个SDU页可以划分为几个PDU

第2章 物理层

考虑怎样才能在连接各种计算机的传输媒体上上传输数据比特流
物理层为数据链路层屏蔽了各种传输媒体的差异，使数据链路层只需要
考虑如何完成本层的协议和服务，而不必考虑网络具体的传输媒体是什
么

1. ★物理层协议主要任务

- **机械特性**: 指明接口所用接线器的形状和尺寸、引脚数目和排列、固定和锁定装置
- **电气特性**: 指明在接口电缆的各条线上出现的电压范围
- **功能特性**: 指明某条线上出现的某一电平的电压表示何种意义
- **过程特性**: 指明对于不同功能的各种可能事件的出现顺序

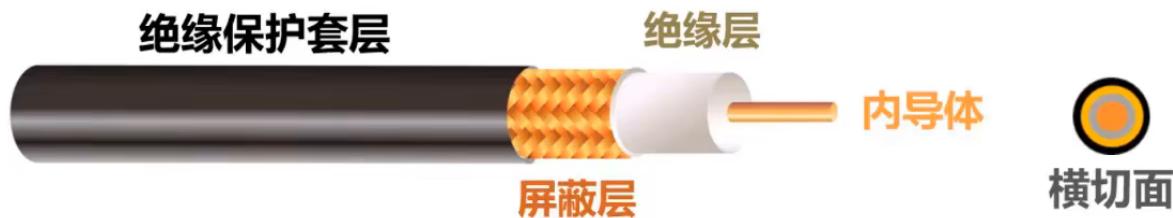
2. 传输媒体

导引型传输媒体

有摸得到的实物进行传导信号的方式

① 同轴电缆

电缆各层都是同轴心的，因此称同轴电缆



基带同轴电缆(50Ω): 数字传输，过去用于局域网

宽带同轴电缆(75Ω): 模拟传输，目前主要用于有线电视

同轴电缆**价格较贵且布线不够灵活和方便**，随着集线器的出现，在局域网领域基本上都是采用双绞线作为传输媒体

② ★双绞线

把两根互相绝缘的铜导线并排放在一起，然后按照一定规则绞合起来就构成了双绞线，这是一种古老且常用的传输媒体

常用绞合线类别、带宽和典型应用

绞合线类别	带宽	线缆特点	典型应用
3	16MHz	2对4芯双绞线	模拟电话；曾用于传统以太网 (10Mbit/s)
4	20MHz	4对8芯双绞线	曾用于令牌局域网
5	100MHz	与4类相比增加了绞合度	传输速率不超过100Mbit/s的应用
5E (超五类)	125MHz	与5类相比衰减更小	传输速率不超过1Gbit/s的应用
6	250MHz	与5类相比改善了串扰等性能	传输速率高于1Gbit/s的应用
7	600MHz	使用屏蔽双绞线	传输速率高于10Gbit/s的应用

UTP

无屏蔽双绞线 UTP 电缆

- 蓝线和蓝白线绞合
- 橙线和橙白线绞合
- 绿线和绿白线绞合
- 棕线和棕白线绞合

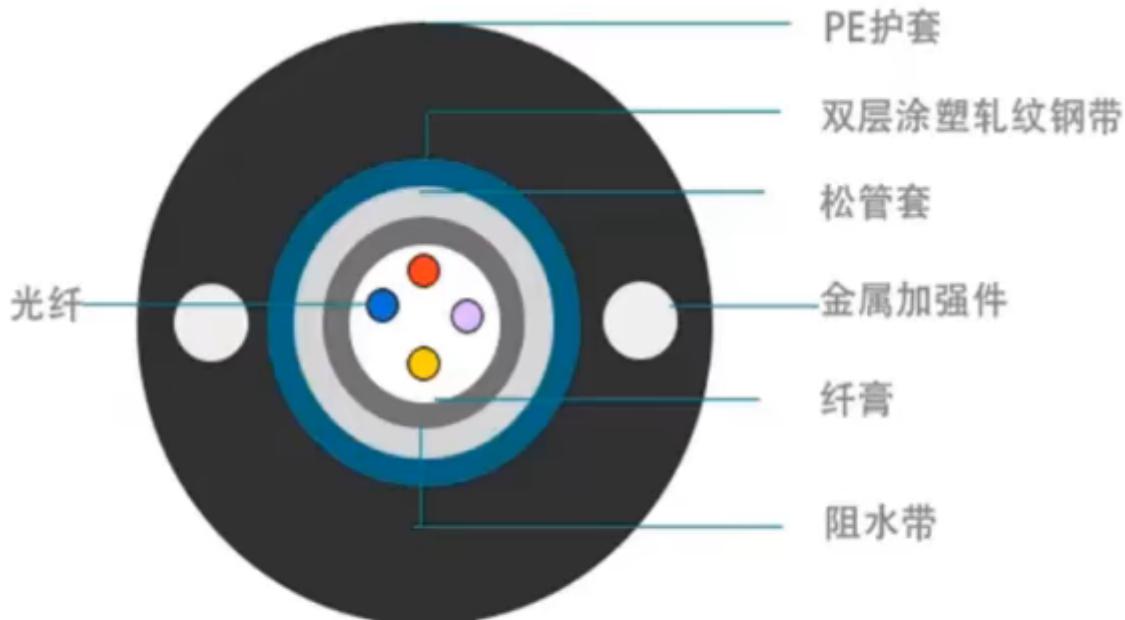
作用：①抵御部分来自外界的电磁波干扰 ②减少相邻导线的电磁干扰

STP

屏蔽双绞线 STP 电缆，其与 UTP 相比增加了金属丝编织的屏蔽层，提高了抗电磁干扰能力

③★光纤

光纤很细，因此必须将其做成结实的光缆。一根光缆少则一根光纤，多则可包括数百根



光缆内部结构

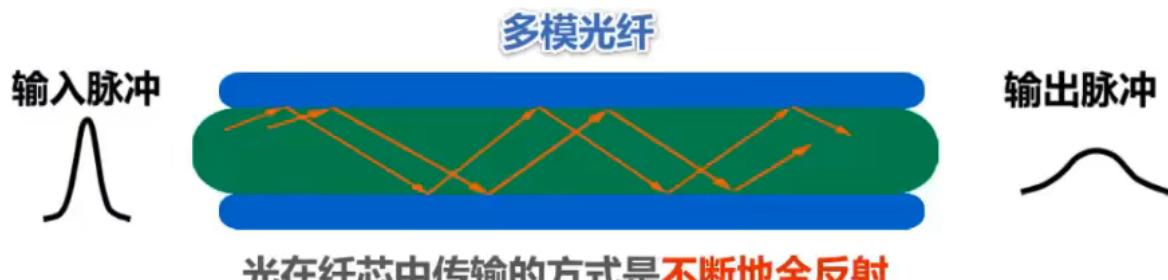
原理



- 当光从高折射率的媒体射向低折射率的媒体时，其折射角将大于入射角；
- 因此，如果入射角足够大，就会出现全反射，即光碰到包层时，就会反射回纤芯。

如果全反射一直进行，则光就会沿着光纤一直传输下去。

实际上只要入射角大于某个临界角度就可以发生全反射，因此多条不同角度的光可以在光线里一起传输，这种光纤称作**多模光纤**



- 由于色散(模式、材料、波导色散)，光在多模光纤中传输一定距离后必然产生失真(脉冲展宽)
- 因此**多模光纤只适合近距离传输**(建筑物内)
- 发送光源可使用**发光二极管(便宜)**；接收检测可用光电二极管

若光纤直径减小到只有一个光的波长，则光纤就像一根波导一样，可使光线一直向前传播，而不会产生多次反射，这样的光纤称作**单模光纤**



- 没有模式色散，在 1.31微米 波长附近，材料色散和波导色散大小相等符号相反，两者正好抵消
- 单模光纤适合长距离传输且衰减小，但其制造成本高，对光源要求高
- 发送光源需要使用**激光发生器(贵)**；接收检测用激光检波器

纤芯直径

- **多模光纤**: $50\text{微米}, 62.5\text{微米}$
- **单模光纤**: 9微米
- 纤芯外包层: 125微米

工作波长

- **0.85微米(衰减较大)**
- **1.30微米(衰减较小)**
- **1.55微米(衰减较小)**

优点

- **通信容量大**($25000\sim30000\text{GHz}$ 的带宽)
- **传输损耗小**，远距离传输时更加经济
- **抗雷电和抗电磁干扰性能好**。这在大电流脉冲干扰环境下尤为重要
- **无串音干扰**，保密性好，不易被窃听
- **体积小，重量轻**

缺点

- **割接需要专用设备**
- **光电接口价格较贵**

非导引型传输媒体

传导信号的东西摸不到

电信领域使用的电磁波的频谱



ITU波段号	频段名称	缩写	频率范围	波段名称	波长范围	用途	电磁波谱对应名称
5	低频	LF	30KHz ~ 300KHz	长波	10km ~ 1km	国际广播, 全向信标	无线电波
6	中频	MF	300KHz ~ 3MHz	中波	1km ~ 100m	调幅(AM)广播, 全向信标, 海事及航空通讯	
7	高频	HF	3MHz ~ 30MHz	短波	100m ~ 10m	短波、民用电台	
8	甚高频	VHF	30MHz ~ 300MHz	米波	10m ~ 1m	调频(FM)广播, 电视广播, 航空通讯	
9	特高频	UHF	300MHz ~ 3GHz	分米波	1m ~ 100mm	电视广播, 无线电话通讯, 无线网络, 微波炉	微波
10	超高频	SHF	3GHz ~ 30GHz	厘米波	100mm ~ 10mm	无线网络, 雷达, 人造卫星接收	
11	极高频	EHF	30GHz ~ 300GHz	毫米波	10mm ~ 1mm	射电天文学, 遥感, 人体扫描安检仪	

①无线电波

低频和中频频段用地面波传播；高频和甚高频靠电离层(地球上空100~500千米高空的带电离子层)反射

②★微波

微波会穿透电离层进入宇宙，因此其不能通过电离层反射到很远的地方

地面微波接力通信

微波是直线传播的，而地球表面是个曲面，因此传播距离受到限制，一般只有50KM左右；如果采用100米高的天线塔，则传播距离可增大到100公里。

为实现远距离通信，必须在一个微波通信信道的两个终端之间建立若干个中继站，中继站把前一阵送来的信号经过放大后再发送到下一站

卫星通信

在地球站之间，利用位于约36000KM高空的人造同步地球卫星作为中继器的一种微波接力，其最大特点是通信距离远，传播时延大(约250~300ms)。低轨道卫星通信系统也已经正在部署

③红外线

- 点对点无线传输
- 直线传播，中间不能有障碍物，传输距离短
- 传输速率低(4Mb/s~16Mb/s)

④可见光

即光源作为信号源，前景好，暂时未被大范围应用

3. 传输方式

★串行/并行传输

串行传输是指数据是1个比特1个比特依次发送的，发送端与接收端之间只用1条数据传输线即可

并行传输是指一次发送n个比特而不是一个比特，在发送端和接收端之间要有n条传输线路

在计算机网络中，数据在传输线路上的传输时串行传输；而计算机内部(如CPU和内存)多使用并行传输

★同步传输

数据块以稳定的**比特流形式**传输，字节之间没有间隔。接收端在每个比特信号的中间时刻(有区分0,1的标志)进行检测，以判别接收到的是比特0还是1。

由于不同设备的时钟频率存在一定差异，不可能完全相同，在传输大量数据的过程中，所产生的判别时刻的累计误差会导致接收端对比特信号的判别错位。因此需要采取方法使双方的时钟保持同步

收发双方时钟同步方法

- **外同步**：在收发双方之间加一条单独的时钟信号线
- **内同步**：发送端将时钟同步信号编码到发送数据中一起传输(如曼彻斯特编码)

★异步传输

以字节为独立的传输单位，字节间的时间间隔不是固定的，接收端仅在每个字节的起始处对字节内的比特实现同步，为此通常传送前要在每个字节前后加上起始位和结束位。

- 异步是指**字节之间异步**(字节之间的时间间隔不固定)
- 字节中的每个比特仍然要同步(各比特的持续时间是相同的)

单工/半双工/全双工

- **单工通信**: 通信双方只有一个数据传输方向(无线电广播)
- **半双工通信**: 通信双方可以相互传输数据，但不能同时进行(对讲机)
- **全双工通信**: 通信双方可以同时发送和接收消息(电话)

单工需要一条信道；其他的需要两条(一个方向一条)

4. 编码与调制

- 消息(message)包括文字、图片、音频和视频
- **数据是运送消息的实体**：计算机中的网卡将比特1和0转换成相应电信号发送到网线，即信号。
- **信号是数据的电磁表现**，由信源(网卡)发出的原始电信号称为基带信号
- 基带信号又分为**数字基带信号(CPU和内存传输的信号)**和**模拟基带信号(麦克风收到声音后转变的电信号)**

信号需要在信道中进行传输，信道可分为数字信道和模拟信道

传输媒体



如果使用信道复用技术，传输媒体里可以有多个信道

在不改变信号性质的前提下，仅对基带信号的波形进行变换，称为**编码**。编码后产生的信号还是数字信号，可以在数字信道中传输

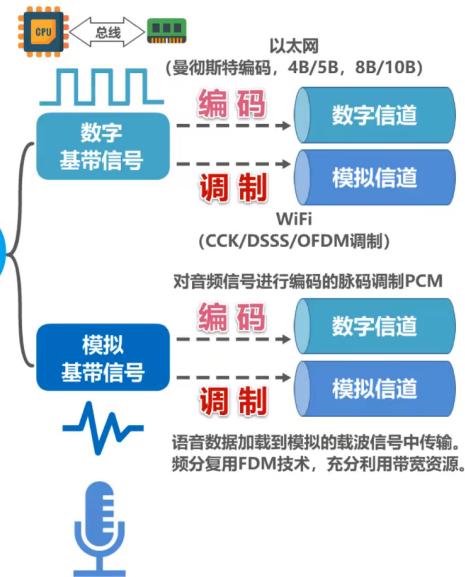
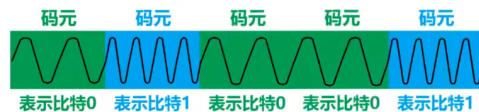
把基带信号的频率范围搬移到较高的频段，并转换为模拟信号，称为**调制**。调制后产生的信号还是模拟信号，可以在模拟信道中传输

2.4 编码与调制

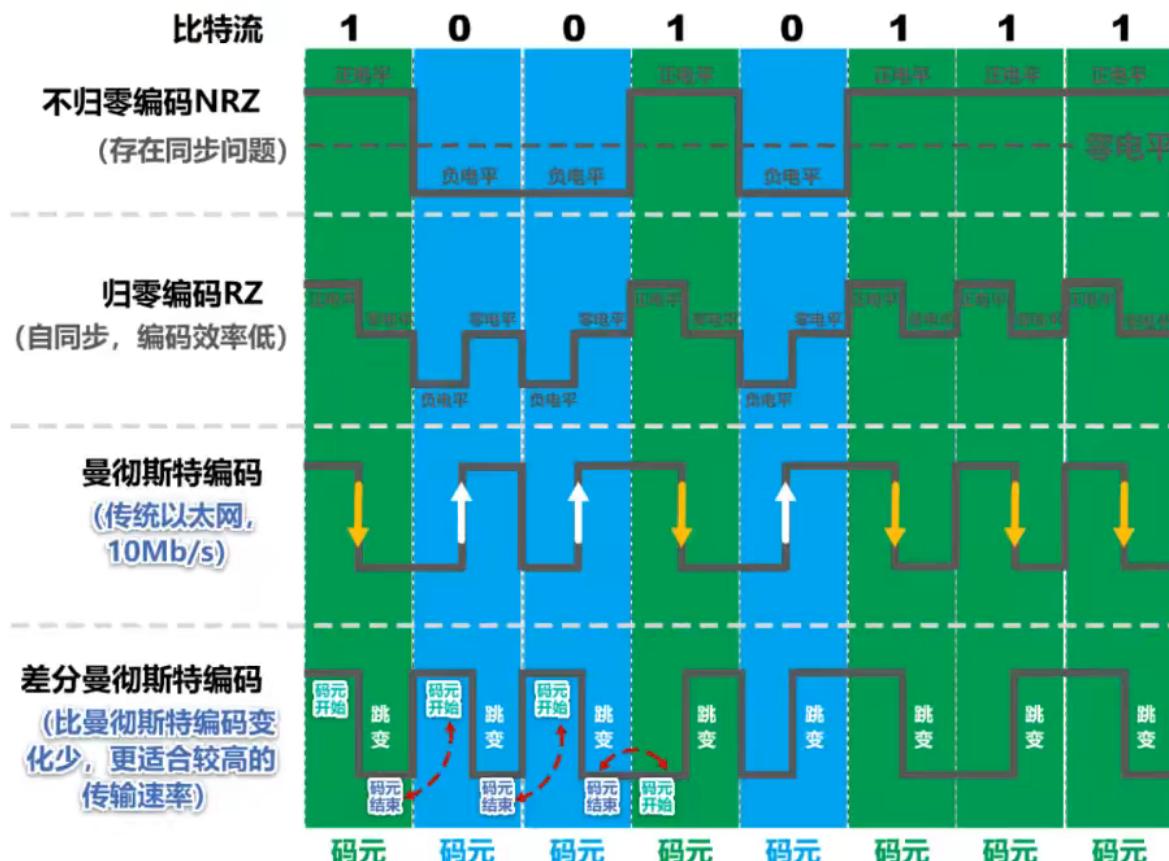


码元

在使用时间域的波形表示数字信号时，
代表不同离散数值的基本波形。



常用编码



①不归零编码

正电平代表比特1，负电平代表比特0。在整个码元时间内，电平不会出现零电平

这种编码方式如何区分连续几个相同电平呢？

这要求发送方发送和接收方接收严格同步，这就需要额外一根传输线来传输时钟信号。接收方按照时钟节拍逐个接收码元。但是对于计算机网络，多的线不如拿来传输数据，因此由于存在同步问题，**计算机中的数据传输不使用不归零编码**

②归零编码

每个码元传输结束后信号都要“归零”，所以接收方只要在信号归零后进行采样即可，不需要单独的时钟信号。

实际上，归零编码相当于把时钟信号用“归零”方式编码在了数据之内，这称为“**自同步**”信号

归零编码中的大部分**数据带宽**都用来传输“归零”而浪费掉了(编码效率低)

③曼彻斯特编码

码元的中间时刻既表示时钟，又表示数据。根据**正负跳变**来区分比特具体如何根据跳变实现同步？

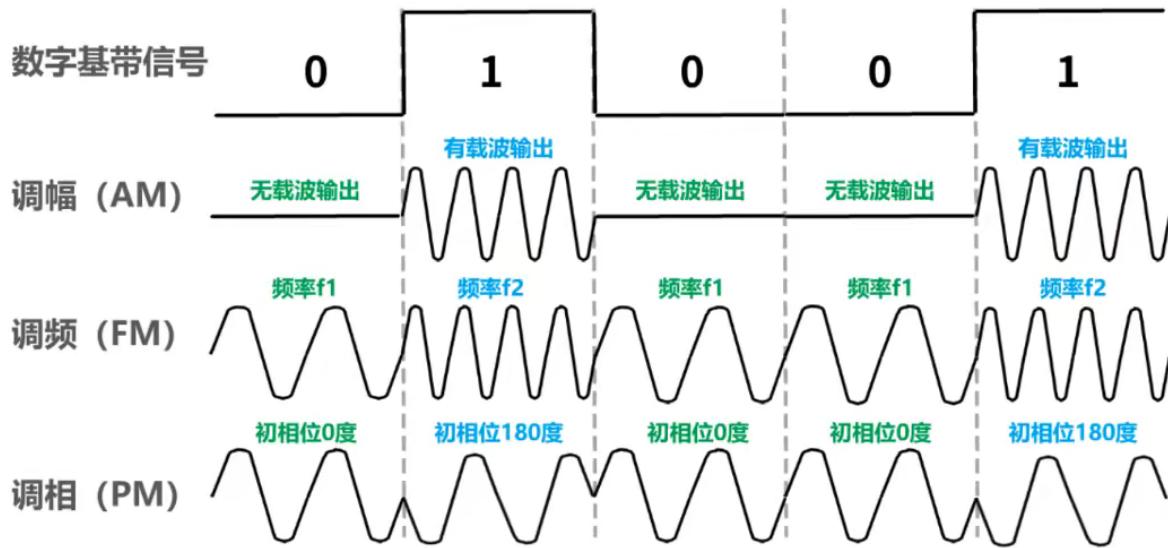
第一次数据跳变的时间记录下来【即半个码元的时间】，此后每过一个码元的时间就进行检测，根据跳变方向决定数据为0还是1。

④差分曼彻斯特编码

①跳变仅表示时钟 ②码元开始处电平是否发生变化表示数据。

比曼彻斯特变化少，更适合较高的传输速率

基本调制方法



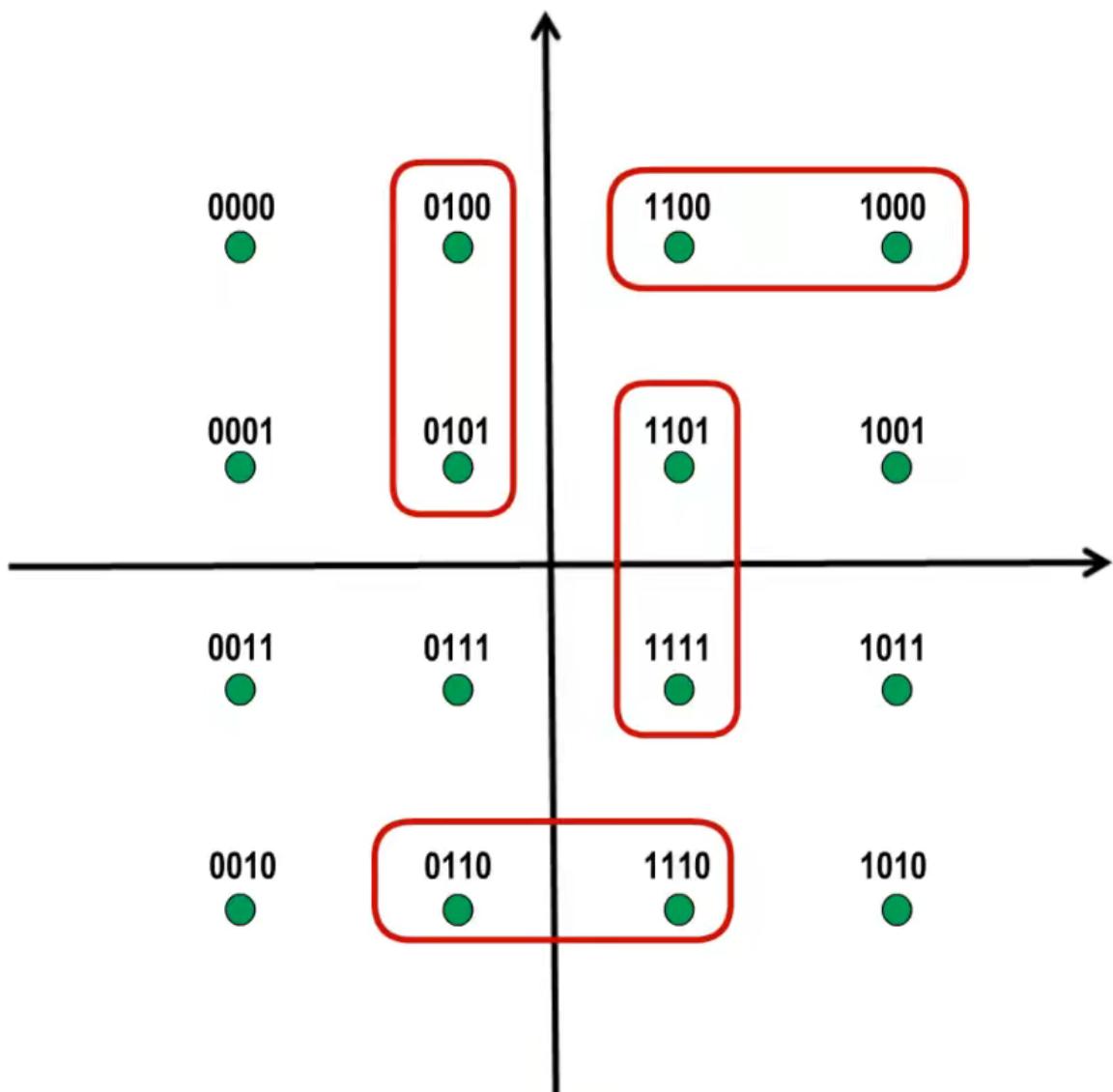
如上一个码元对应一个比特，如何能让1个码元包含多个比特呢？

可以使用混合调制。正弦信号 $A \sin(\omega x + \frac{\varphi}{\omega})$ ，相位和频率是相关的，因此二者不可同时做修改

通常情况下，相位和振幅可以结合起来其一调制，称为**正交振幅调制 QAM**

QAM-16

- 12 种相位
- 每种相位有 1 或 2 种振幅可选
- 可以调制出 16 种码元(波形)，即 16 个形状可以用 4 个二进制位排列组合表示，所以一个码元可以对应 4 比特
- 码元与 4 个比特的对应关系要采用**格雷码**【任意两个相邻码元只有一位不同】



5. 信道的极限容量

通信质量较差的信道在传输信号的过程中会发生**严重失真**(无法识别原信号)**【码间串扰】**

失真因素

- 码元传输速率
- 信号传输距离
- 噪声干扰
- 传输媒体质量

奈氏准则

在假定的理想条件下，为了避免码间串扰，码元传输速率是有上限的。

理想低通信道的最高码元传输速率 = $2W$ Baud = $2W$ 码元/秒

理想带通信道的最高码元传输速率 = W Baud = W 码元/秒

W：信道带宽（单位为Hz）

Baud：波特，即码元/秒

只要采用更好的调制方法，让码元可以携带更多的比特，岂不是可以无限制地提高信息的传输速率？

答案是否定的。因为信道的极限信息传输速率还要受限于实际的信号在信道中传输时的信噪比。

■ 码元传输速率又称为波特率、调制速率、波形速率或符号速率。它与比特率有一定关系：

□ 当1个码元只携带1比特的信息量时，则波特率（码元/秒）与比特率（比特/秒）在数值上是相等的；

□ 当1个码元携带n比特的信息量时，则波特率转换成比特率时，数值要乘以n。

■ 要提高信息传输速率（比特率），就必须设法使每一个码元能携带更多个比特的信息量。这需要采用多元制。

■ 实际的信道所能传输的最高码元速率，要明显低于奈氏准则给出的这个上限数值。

香农公式

带宽受限且有高斯白噪声干扰的信道的极限信息传输速率。

$$c = W \times \log_2(1 + \frac{S}{N})$$

■ 信道带宽或信道中信噪比越大，信息的极限传输速率越高。

C：信道的极限信息传输速率（单位：b/s）

■ 在实际信道上能够达到的信息传输速率要比该公式的极限传输速率低不少。这是因为在实际信道中，信号还要受到其他一些损伤，如各种脉冲干扰、信号在传输中的衰减和失真等，这些因素在香农公式中并未考虑。

W：信道带宽（单位：Hz）

S：信道内所传信号的平均功率

N：信道内的高斯噪声功率

S/N：信噪比，使用分贝（dB）作为度量单位

$$\text{信噪比 (dB)} = 10 \times \log_{10}(\frac{S}{N}) \text{ (dB)}$$

- 在信道带宽一定的情况下，根据奈氏准则和香农公式，要想提高信息的传输速率就必须采用**多元制**【更好的调制方法】和努力**提高信道中的信噪比**
- 自从香农公式发布后，各种新的信号处理和调制方法就不断出现，其目的都是为了**尽可能地接近香农公式给出的传输速率极限**

例题↓

【2016年题34】若连接R2和R3链路的频率带宽为8kHz，信噪比为30dB，该链路实际数据传输速率约为理论最大数据传输速率的50%，则该链路的实际数据传输速率约是 C

A. 8 kbps

B. 20 kbps

C. 40 kbps

D. 80 kbps

【解析】

$$\text{理论最大数据传输速率 } c = 8k \times \log_2(1 + \frac{S}{N})$$

$$30 \text{ (dB)} = 10 \times \log_{10}(\frac{S}{N}) \text{ (dB)} \text{ 解得 } \frac{S}{N} = 1000 \text{ 代入上式}$$

$$c = 8k \times \log_2(1 + 1000) \approx 80 \text{ kbps}$$

$$\text{该链路的实际数据传输速率 } c \times 50\% = 40 \text{ kbps}$$

【2017年 题34】若信道在无噪声情况下的极限数据传输速率不小于信噪比为30dB条件下的极限数据传输速率，则信号状态数至少是

A. 4

B. 8

C. 16

D. 32

【解析】

设信号状态数（可调制出的不同基本波形或码元数量）为X

则每个码元可携带的比特数量为 $\log_2 X$

信道在无噪声情况下的极限数据传输速率（用奈氏准则计算） = $2W$ (码元/秒) = $2W \log_2 X$ (比特/秒)

30dB信噪比条件下的极限数据传输速率（用香农公式计算） = $W \log_2(1 + 1000)$ (比特/秒)

根据题意列出不等式： $2W \log_2 X \geq W \log_2(1 + 1000)$ 解得 $X \geq 32$

【2009年 题34】在无噪声情况下，若某通信链路的带宽为3kHz，采用4个相位，每个相位具有4种振幅的QAM调制技术，则该通信链路的最大数据传输速率是 B

A. 12 kbps

B. 24 kbps

C. 48 kbps

D. 96 kbps

【解析】

(1) 根据奈氏准则可知，该通信链路的最高码元传输速率 = $2 \times 3k = 6k$ (Baud) = $6k$ (码元/秒)

(2) 采用4个相位，每个相位4种振幅的QAM调制技术，可以调制出 $4 \times 4 = 16$ 个不同的基本波形（码元）；

采用二进制对这16个不同的码元进行编码，需要使用4个比特 ($\log_2 16 = 4$)。换句话说，每个码元可以携带的信息量为4比特；

综合(1)和(2)可知，该通信链路的最大数据传输速率 = $6k$ (码元/秒) $\times 4$ (比特/码元) = $24k$ (比特/秒) = 24 kbps

第3章 数据链路层

1. 概述

物理层发出去的信号需要通过数据链路层才知道是否到达目的地；才知道比特流的分界线

- **链路(Link)**: 从一个结点到相邻结点的一段物理线路，中间没有任何其他交换结点
- **数据链路(Data Link)**: 把实现通信协议的硬件和软件加到链路上，就构成了数据链路
- 数据链路层以帧为单位传输和处理数据

封装成帧

数据链路层为数据加上帧首和帧尾使之成为帧的过程

以太网V2的MAC帧 (最大长度为1518字节)				
6字节	6字节	2字节	46 ~ 1500 字节	4字节
目的地址	源地址	类型	数据载荷	FCS
帧头		上层交付的协议数据单元		帧尾

- 帧头和帧尾中含有重要的控制信息

- 帧头帧尾的作用之一就是帧定界(变成比特传输后据此区分每个帧的起始和结束)
- 为了提高帧的传输效率，应当使帧的数据部分长度尽可能大
- 考虑到差错控制等多种因素，每一种数据链路层协议都规定了帧的数据部分的长度上限，即**最大传送单元MTU** (Maximum Transfer Unit)

透明传输是指**数据链路层对上层交付的传输数据没有任何限制**，就好像数据链路层不存在一样(即**保证接收方接到的数据是完整的数据**)

- **面向字节**的物理链路使用**字节填充**(或称字符填充)的方法实现**透明传输**

帧头帧尾有标志位用来划分一个个帧，如果帧内部恰好也出现了标志位，则在第一次扫描时在标志位前面加一个转义字符帮助区分哪个是真正的帧头。考虑到转义字符也可能在帧内部出现，因此在转义字符前也加转义字符。接收方接到消息后但凡看到转义字符开头就会去掉转义字符并且对其后一个字符不做特殊处理

- **面向比特的物理链路使用比特填充的方法实现透明传输**

零比特填充：在发送前，对帧的数据部分进行扫描，每5个连续的比特1后就插入1个比特0，防止其与首部尾部的标志位混淆。接收方接收时将每5个连续的比特1后面的0剔除即可

差错检测

实际的通信链路都不是理想的，比特在传输过程中可能会产生差错：**1可能变成0，0可能变成1**。这叫**比特差错**

在一段时间内，传输错误的比特占所传输比特总数的比率称为**误码率BER(Big Error Rate)**

使用**差错检测码**(如**Mac**帧尾的**FCS**)来检测数据在传输过程中是否产生了比特差错，是数据链路层所要解决的重要问题之一

奇偶校验

在待发送的数据后面添加1位奇偶校验位，使整个数据(包括所添加的校验位在内)中“1”的个数为奇数(奇校验)或为偶数(偶校验)

比如发送数据001

- 若是奇校验，则在数据后添加0，使其成为0001，1的个数为奇数。如果传输过程中发生了1位误码，则1的个数会变成偶数，据此判断是否发生误码；但是若发生了2个误码，1的个数依然为奇数，因此检查不出来
- 若为偶校验，则在数据后添加1，使其成为1001，1的个数为偶数，其他情况与奇校验类似

如果有奇数个位发生误码，则奇偶性发生变化，可以检查出误码

如果有偶数个位发生误码，则奇偶性不发生变化，不能检查出误码(漏检)

循环冗余校验CRC

- 收发双方约定好一个生成多项式 $G(x)$

【生成多项式举例】

$$\begin{aligned} G(x) &= x^4 + x^2 + x + 1 \\ &= 1 \cdot x^4 + 0 \cdot x^3 + 1 \cdot x^2 + 1 \cdot x^1 + 1 \cdot x^0 \end{aligned}$$

生成多项式各项系数构成的比特串：10111

【常用的生成多项式】

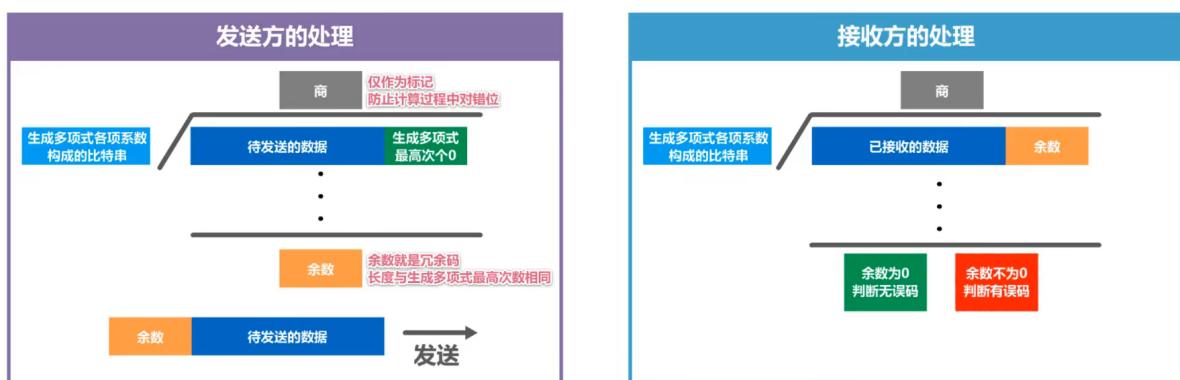
算法要求生成多项式必须包含最低次项

$$CRC-16 = x^{16} + x^{15} + x^2 + 1$$

$$CRC-CCITT = x^{16} + x^{12} + x^5 + 1$$

$$CRC-32 = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

- 发送方基于待发送的数据和生成多项式计算出差错检测码(冗余码)，将其添加到待传输数据的后面一起传输
- 接收方通过生成多项式来计算收到的数据是否产生了误码



除法内的相减实际是做异或运算，因此没有小的减不了大的这个说法

【循环冗余校验CRC举例】待发送的信息为101001，生成多项式为 $G(x) = x^3 + x^2 + 1$ ，计算余数。

1 构造被除数

待发送信息后面添加生成多项式最高次数个0

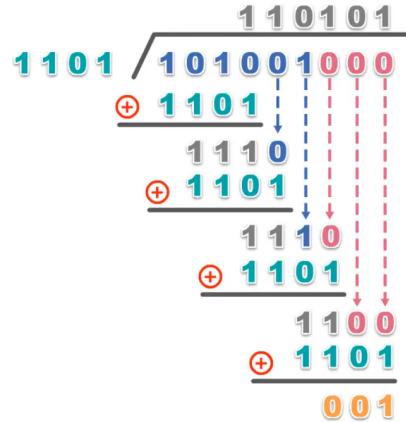
2 构造除数

生成多项式各项系数构成的比特串

3 做“除法”

4 检查余数

余数的位数应与生成多项式最高次数相同，如果位数不够，则在余数前补0来凑足位数。



← 101001001
发送

【循环冗余校验CRC举例】接收到的信息为101101001，生成多项式为 $G(x) = x^3 + x^2 + 1$ ，判断传输是否误码？

1 构造被除数

接收到的信息就是被除数

2 构造除数

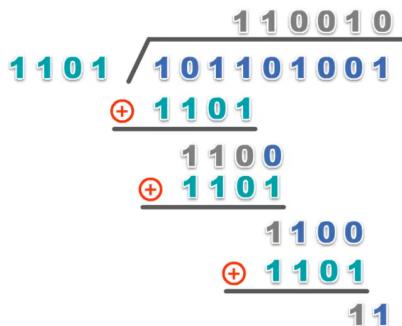
生成多项式各项系数构成的比特串

3 做“除法”

4 检查余数

余数为0，可认为传输过程无误码；

余数不为0，可认为传输过程产生误码。



余数不为0，表明传输过程产生误码！

- **检错码**只能检测出帧在传输过程中出现了差错，但并不能定位错误，因此**无法纠正错误**。
- 要想纠正传输中的差错，可以使用冗余信息更多的**纠错码**进行**前向纠错**。但纠错码的开销比较大，在**计算机网络中较少使用**
- **CRC**有很好的检错能力(**漏检率非常低**)，虽然计算比较复杂，但**非常易于用硬件实现**，因此**被广泛应用于数据链路层**
- 在**计算机网络中通常采用检错重传方式来纠正传输中的差错**，或者仅仅是**丢弃检测到差错的帧**，这取决于数据链路层向其上层提供的是**可靠传输服务还是不可靠传输服务**

可靠传输

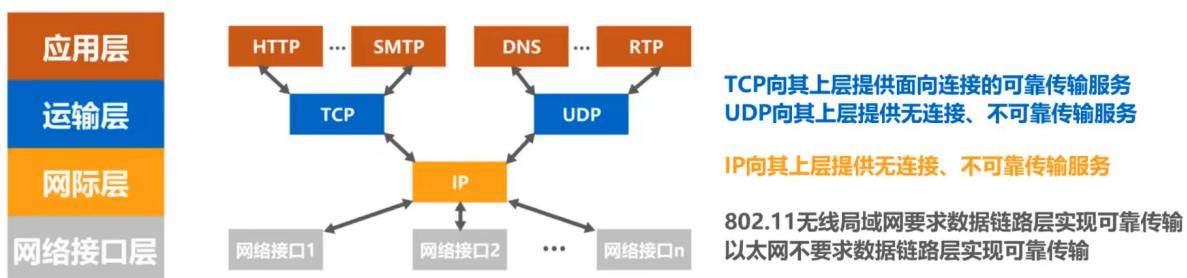
不可靠传输服务：仅仅丢弃有误码的帧，其他什么也不做

可靠传输：想办法实现发送端发送什么，接收端就收到什么

- 一般情况下，**有线链路**的误码率比较低，为了减小开销，并**不要求数据链路层向上提供可靠传输服务**。即使出现了误码，可靠传输的问题由其

上层处理

- **无线链路**易受干扰，误码率比较高，因此**要求数据链路层**必须向上层提供**可靠传输服务**
- 比特差错只是传输差错中的一种，从整个计算机网络体系结构来看，传输差错还包括**分组丢失、分组失序以及分组重复**
- 分组丢失、分组失序以及分组重复这些传输差错，一般不会出现在数据链路层，而会出现在其上层
- **可靠传输服务并不仅局限于数据链路层**，其他各层均可选择实现可靠传输
- 可靠传输的实现比较复杂，开销也比较大，是否使用可靠传输取决于应用需求



①停止-等待协议SW

发送方发送数据 DATA，接收方接收时进行差错检测

- ①如果没有出现误码，则接收信息并返回 ACK 确认分组给发送方，发送方收到 ACK 后，一次通信结束
- ②如果出现误码，则丢弃信息并返回 NAK 拒绝分组给发送方，发送方收到 NAK 后，重传 DATA，直至出现步骤①的情况

如果 DATA 传送过程中丢失了，即接收端一直等 DATA，发送端一直等 ACK，造成死锁，如何解决呢？

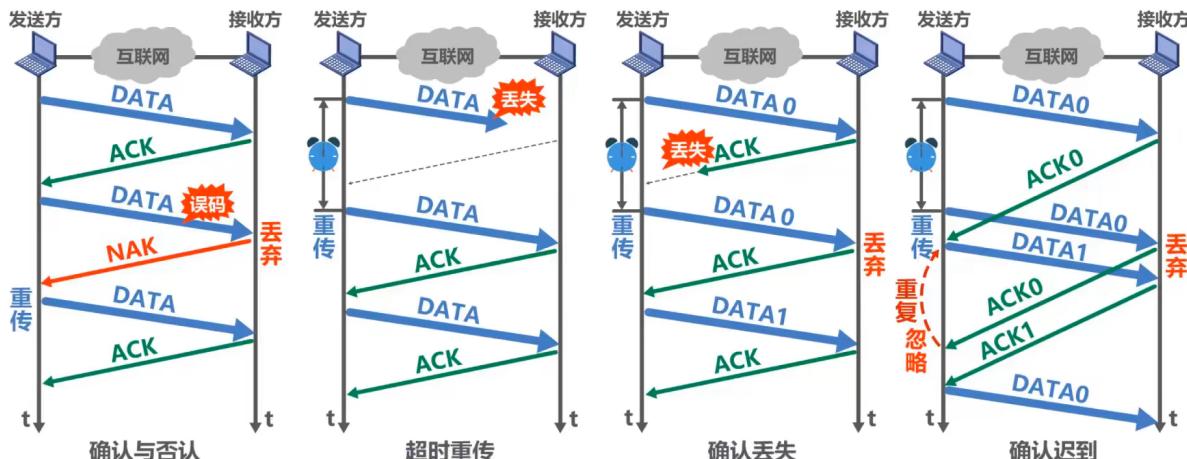
可以引入超时重传机制。可以在发送端设置一计时器（大约发送接收的平均时间），当发送端在这个时间内没有收到 ACK 或 NAK 时，就会判断 DATA 丢失，从而再次发送 DATA，打破死锁

如果 ACK 发送中丢失了，即接收方收不到 ACK 就将数据重新发送，而数据实际上接收方已经有了，因此重复接收，并返回 ACK，造成错误，怎么办？

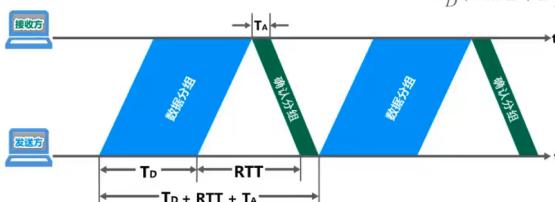
可以对每次发的 DATA 都加上序号，这样接收端就可以判断当前的数据是否有接收过，从而决定其去留

如果发送端发送 DATA 0，如果 ACK 由于某些原因使其到达接收端的时间变长了，那么根据超时重传，DATA 0 会再次发送，而此时 ACK 到达，则发送方会认为这是后一次 DATA 0 的确认分组，于是马上发送 DATA 1。而第二次发送的 DATA 0 此时返回 ACK，发送方误以为是 DATA 1 的 ACK，因此又会发送 DATA2，而实际上 DATA 1 的是否误码等情况还不知道，因此出现了错误，这种情况怎么办？

可以为 ACK 加上编号。则每个 ACK 的相互作用就不会互串了。【对于数据链路层点对点信道，往返时间比较固定，不会出现确认迟到的情况，因此可以不给确认分组编号】



$$\text{停止-等待协议的信道利用率 } U = \frac{T_p}{T_p + RTT + T_A}$$



假设信道长度2000km，数据分组长度1500B，发送速率10Mbit/s。
(忽略 T_A ，因为 T_A 一般都远小于 T_p)

$$U \approx \frac{T_p}{T_p + RTT} = \frac{\frac{1500 \times 8 \text{ bit}}{10 \times 10^6 \text{ bit/s}} \cdot 1.2 \text{ ms}}{\frac{1500 \times 8 \text{ bit}}{10 \times 10^6 \text{ bit/s}} + \frac{2000 \times 10^3 \text{ m} \times 2}{2 \times 10^8 \text{ m/s}} \cdot 20 \text{ ms}} \approx 5.66\%$$

若提高发送速率到100Mb/s

$$U \approx \frac{T_p}{T_p + RTT} = \frac{\frac{1500 \times 8 \text{ bit}}{100 \times 10^6 \text{ bit/s}} \cdot 0.12 \text{ ms}}{\frac{1500 \times 8 \text{ bit}}{100 \times 10^6 \text{ bit/s}} + \frac{2000 \times 10^3 \text{ m} \times 2}{2 \times 10^8 \text{ m/s}} \cdot 20 \text{ ms}} \approx 0.6\%$$

- 当往返时延RTT远大于数据帧发送时延 T_p 时（例如使用卫星链路），信道利用率非常低。
- 若出现重传，则对于传送有用的数据信息来说，信道利用率还要降低。
- 为了克服停止-等待协议信道利用率很低的缺点，就产生了另外两种协议，即后退N帧协议GBN和选择重传协议SR。

【2018年题36】主机甲采用停-等协议向主机乙发送数据，数据传输速率为3kbps，单向传播延时是200ms，忽略确认帧的传输延时。当信道利用率为40%时，数据帧的长度为 D

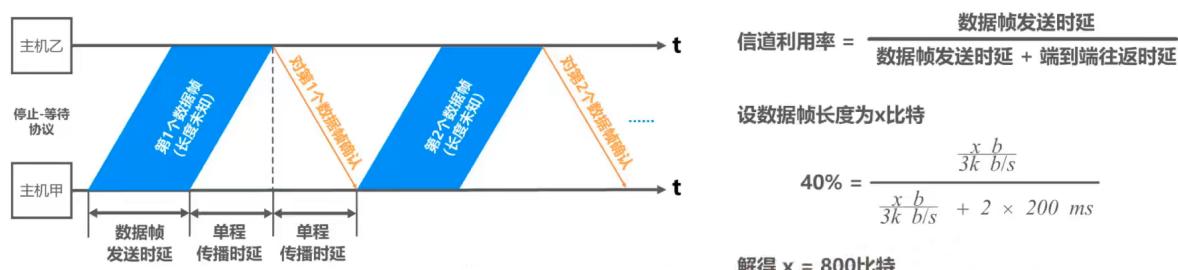
A. 240比特

B. 400比特

C. 480比特

D. 800比特

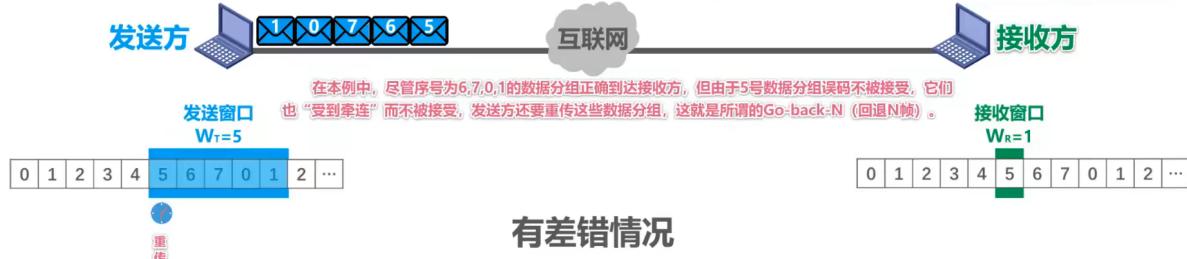
【解析】



②回退N帧协议GBN

相对停止-等待协议多个窗口的概念

接收窗口尺寸只能等于1，因此接收方只能按序接收正确到达的数据分组



发送方	接收方
<ul style="list-style-type: none">发送窗口尺寸W_T的取值范围是 $1 < W_T \leq 2^n - 1$ 其中，n是构成分组序号的比特数量。<input type="checkbox"/> $W_T = 1$ 停止-等待协议<input type="checkbox"/> $W_T > 2^n - 1$ 接收方无法分辨新、旧数据分组发送方可在未收到接收方确认分组的情况下，将序号落在发送窗口内的多个数据分组全部发送出去；发送方只有收到对已发送数据分组的确认时，发送窗口才能向前相应滑动；发送方收到多个重复确认时，可在重传计时器超时前尽早开始重传，由具体实现决定。发送方发送窗口内某个已发送的数据分组产生超时重发时，其后续在发送窗口内且已发送的数据分组也必须全部重传，这就是回退N帧协议名称的由来。	<ul style="list-style-type: none">接收方的接收窗口尺寸W_R的取值范围是 $W_R = 1$ 因此接收方只能按序接收数据分组。接收方只接收序号落在接收窗口内且无误码的数据分组，并且将接收窗口向前滑动一个位置，与此同时给发送方发回相应的确认分组。为了减少开销，接收方不一定每收到一个按序到达且无误码的数据分组就给发送方发回一个确认分组。<ul style="list-style-type: none">而是可以在连续收到好几个按序到达且无误码的数据分组后（由具体实现决定），才针对最后一个数据分组发送确认分组，这称为累积确认；或者可以在自己有数据分组要发送时才对之前按序接收且无误码的数据分组进行捎带确认；接收方收到未按序到达的数据分组，除丢弃外，还要对最近按序接收的数据分组进行确认；

- 接收端返回 ACK n 表示发送过来的分组x<=n的都收到了

③选择重传协议SR

与回退N帧相比，接收窗口允许多个，且发送窗口最大情况有所变化



发送方	接收方
<p>■ 发送窗口尺寸W_T的取值范围是 $1 < W_T \leq 2^{n-1}$ 其中，n是构成分组序号的比特数量。</p> <p><input type="checkbox"/> $W_T = 1$ 与停止-等待协议相同 <input type="checkbox"/> $W_T > 2^{n-1}$ 接收方无法分辨新、旧数据分组</p> <p>■ 发送方可在未收到接收方确认分组的情况下，将序号落在发送窗口内的多个数据分组全部发送出去；</p> <p>■ 发送方只有按序收到对已发送数据分组的确认时，发送窗口才能向前相应滑动；若收到未按序到达的确认分组时，对其进行记录，以防止其相应数据分组的超时重发，但发送窗口不能向前滑动。</p>	<p>■ 接收窗口尺寸W_R的取值范围是 $1 < W_R \leq W_T$</p> <p><input type="checkbox"/> $W_R = 1$ 与停止-等待协议相同 <input type="checkbox"/> $W_R > W_T$ 无意义</p> <p>■ 接收方可接收未按序到达但没有误码并且序号落在接收窗口内的数据分组； ■ 为了使发送方仅重传出现差错的分组，接收方不能再采用累积确认，而需要对每个正确接收到的数据分组进行逐一确认！ ■ 接收方只有在按序接收数据分组后，接收窗口才能向前相应滑动。</p>

- 发送窗口接收到ACK帧的部分不会超时重传

2. PPP协议

数据链路层协议，用于规定帧格式

帧的首部				帧的数据部分	帧的尾部	
F	A	C	P		FCS	F
1字节	1字节	1字节	2字节	不超过1500字节	2字节	1字节

标志 (Flag) 字段：PPP帧的定界符，取值为0x7E

地址 (Address) 字段：取值为0xFF，预留（目前没有什么作用）

控制 (Control) 字段：取值为0x03，预留（目前没有什么作用）

协议 (Protocol) 字段：指明帧的数据部分送交哪个协议处理

取值0x0021表示：帧的数据部分为IP数据报

7E FF 03 0021	IP数据报	FCS	7E
---------------	-------	-----	----

取值0xC021表示：帧的数据部分为LCP分组

7E FF 03 C021	LCP分组	FCS	7E
---------------	-------	-----	----

取值0x8021表示：帧的数据部分为NCP分组

7E FF 03 8021	NCP分组	FCS	7E
---------------	-------	-----	----

帧检验序列 (Frame Check Sequence) 字段：CRC计算出的校验位

透明传输

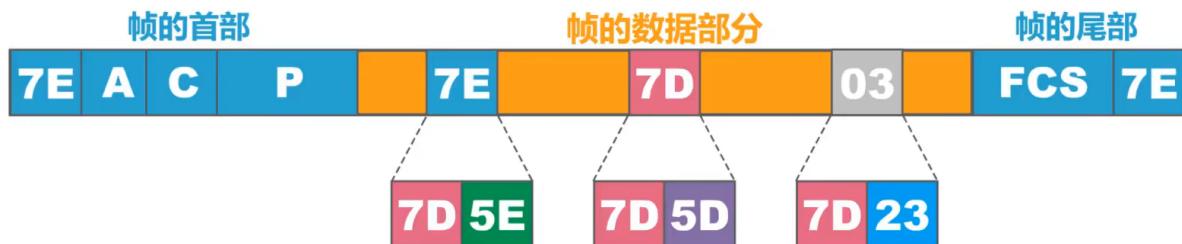
①字节填充法

面向字节的异步链路采用插入转义字符的字节填充法

- 如PPP帧的标志字段取值为7E(16进制)，如果数据中出现7E则需要在数据前插入转义字符7D(16进制)，并将原来的7E减20(16进制)，所以7E在数据中最终会变成7D5E
- 如果数据中有转义字符7D怎么办呢？可以在转义字符7D前再加一个转义字符7D，并将数据的7D减20(16进制)，于是转义字符7D在数据中

最终会变成 7D5D

- 数据中出现的每一个 ASCII 码控制字符 【数值小于20(16进制)的字符】，则在该字符前插入一个7D，同时将该字符的编码加上20(16进制)
- 接收方只需要反变换即可恢复出原来的帧的数据部分



②比特填充法

面向比特的同步链路采用插入比特0的比特填充法

- 发送方**: 对帧的数据部分进行扫描(一般由硬件实现)。只要发现5个连续的比特1，则立即填充1个比特0
- 接收方**: 对帧的数据部分进行扫描(一般由硬件实现)。只要发现5个连续的比特1，就把其后的1个比特0删除



PPP的差错检测

接收方每收到一个 PPP 帧，就进行 CRC 检验(多项式)。若 CRC 检验正确，就收下这个帧；反之就丢弃这个帧(不可靠传输服务)。检验由尾部的 FCS 实现

用于检验的多项式为 $X^{16} + X^{12} + X^5 + 1$

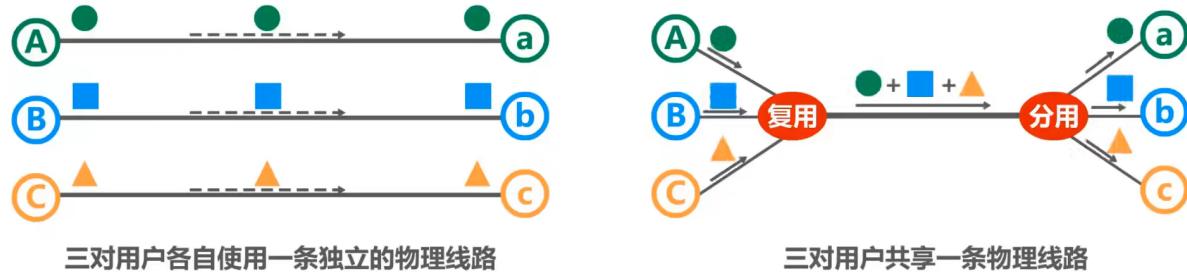


3. 媒体接入层

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收站点对一个共享传输媒体的占用，即媒体接入控制 MAC(Medium Access Control)

复用(Multiplexing)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单一信道的总通信量时，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



静态划分信道

①频分复用FDM

将传输线路的频带资源划分成多个子频带，形成多个子信道。各子信道之间留出隔离频带，以免造成子信道间干扰。当多个信号输入一个多路复用器时，这个复用器将每一个信号调制到不同频率的载波上，接收端由相应的分用器通过滤波将各路信号分隔开，将合成的复用信号恢复成原始的多路信号



频分复用的所有用户同时占用不同的频带资源并行通信。

②时分复用TDM

将时间划分为一个个时隙，将带宽资源按照时隙轮流分配给不同的用户，每对用户只在所分配时隙里使用线路传输数据。

时分复用技术将时间划分为一段段等长的时分复用帧，**每一个时分复用的用户在每一个时分复用帧中占用固定序号的时隙**。每个用户所占的时隙是周期性出现的，其周期就是时分复用帧的长度

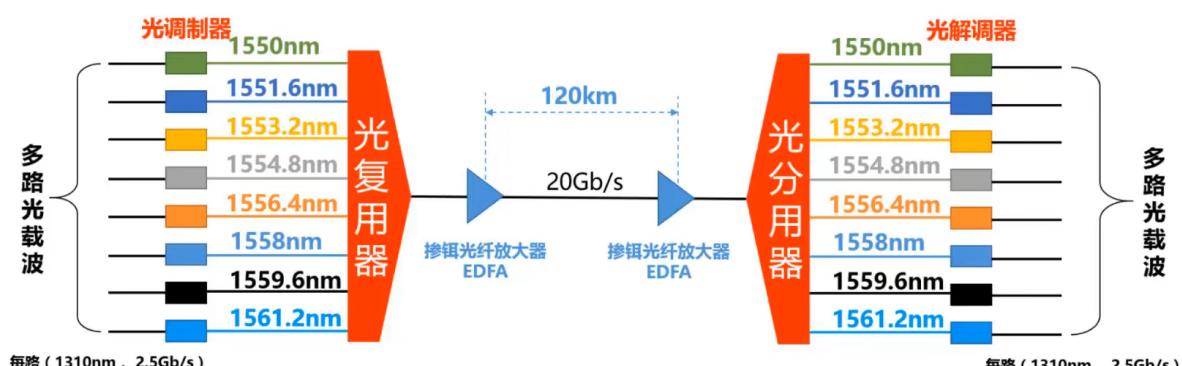


时分复用的所有用户在不同的时间占用同样的频带宽度。

③波分复用WDM

波分复用其实就是光的频分复用。经过光调制，分别将光载波变换到不同波长。这些光波经过光复用器就可以在一根光纤中传输。到达终点后用光分用器将不同波长的光进行还原得到信息

光信号传输一段距离后会衰减，对衰减的光信号必须进行放大才能继续传输



④码分复用CDM

码分复用 CDM(最初用于军事通信)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**

CDM的每一个用户可以在同样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

在CDMA中，每一个比特时间再划分为m个短的间隔，称为**码片**。通常m的值是64或128

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，则发送它自己的m bit码片序列二进制反码

【举例】

指派给CDMA系统中某个站点的码片序列为00011011

发送比特1：发送自己的码片序列00011011

发送比特0：发送自己的码片序列的二进制反码11100100

为了方便，我们按惯例将码片序列中的0写为-1，将1写为+1。

则该站点的码片序列是 (-1 -1 -1 +1 +1 -1 +1 +1)。

码片序列挑选原则：

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

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。

$S \cdot T$ (计算方式为码片序列 S 和 T 对应项相乘相加再除以长度)等于0
即规格化内积等于0，此时会有以下四个特征↓。

$S \cdot T$ 恒等于0； $S \cdot \bar{T}$ 恒等于0； $S \cdot S$ 恒等于1； $S \cdot \bar{S}$ 恒等于 -1

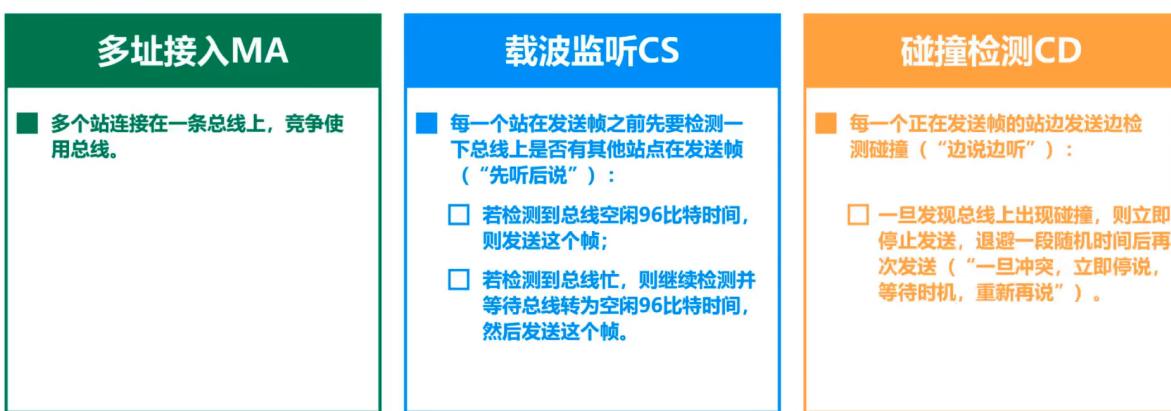
【习题1】假设给站S分配的码片序列为01011101，给站T分配的码片序列为10111000，这样的分配正确吗？

检查码片序列是否各不相同：**满足**

检查码片序列是否相互正交：**不满足**

根据题意可知，用向量 S 表示站 S 的码片序列 (-1 +1 -1 +1 +1 +1 -1 +1)，用向量 T 表示站 T 的码片序列 (+1 -1 +1 +1 +1 -1 -1 -1)

$$S \cdot T = \frac{(-1)(+1) + (+1)(-1) + (-1)(+1) + (+1)(+1) + (+1)(+1) + (+1)(-1) + (-1)(-1) + (+1)(-1)}{8} = \frac{-1-1+1+1+1-1-1-1}{8} \neq 0$$



【习题2】共有4个站进行CDMA通信，这4个站的码片序列分别为：

A: (-1 -1 -1 +1 +1 -1 +1 +1) 发送比特1

B: (-1 -1 +1 -1 +1 +1 +1 -1)

C: (-1 +1 -1 +1 +1 +1 -1 -1)

D: (-1 +1 -1 -1 -1 -1 +1 -1)

现收到码片序列(-1 +1 -3 +1 -1 -3 +1 +1)。问是哪些站发送了数据？发送的是比特1还是0？

【解析】用收到的码片序列分别与各站的码片序列进行求内积运算。若计算结果为数值1，则被判断的站发送了比特1；若计算结果为数值-1，则被判断的站发送了比特0；若计算结果为数值0，则被判断的站未发送数据。

判断A站： $(-1 -1 -1 +1 +1 -1 +1 +1) \cdot (-1 +1 -3 +1 -1 -3 +1 +1)$

$$= \frac{(-1)(-1) + (-1)(+1) + (-1)(-3) + (+1)(+1) + (+1)(-1) + (-1)(-3) + (+1)(+1) + (+1)(+1)}{8}$$

$$= \frac{-1-1+3+1-1+3+1+1}{8} = 1$$

习题2的B、C、D的情况同理↑

- 【2014年 题37】站点A、B、C通过CDMA共享链路，A、B、C的码片序列(chipping sequence)分别是(1, 1, 1, 1)、(1, -1, 1, -1)和(1, 1, -1, -1)。若C从链路上收到的序列是(2, 0, 2, 0, -2, 0, -2, 0, 2, 0, 2)，则C收到A发送的数据是 B
- A. 000 B. 101 C. 110 D. 111

【解析】由于题目所给各站的码片序列为4位，因此将站点C收到的序列分成三部分，每部分也由4位组成：

$$(2, 0, 2, 0), (0, -2, 0, -2), (0, 2, 0, 2)$$

将站点A的码片序列(1, 1, 1, 1)分别与上述三个部分进行内积运算，根据结果可判断出A发送的数据

$$(1, 1, 1, 1) \cdot (2, 0, 2, 0) = (1 \times 2 + 1 \times 0 + 1 \times 2 + 1 \times 0) \div 4 = 1 \quad \text{发送比特1}$$

$$(1, 1, 1, 1) \cdot (0, -2, 0, -2) = (1 \times 0 + 1 \times (-2) + 1 \times 0 + 1 \times (-2)) \div 4 = -1 \quad \text{发送比特0}$$

$$(1, 1, 1, 1) \cdot (0, 2, 0, 2) = (1 \times 0 + 1 \times 2 + 1 \times 0 + 1 \times 2) \div 4 = 1 \quad \text{发送比特1}$$

复用与多址的区别

复用是将单一媒体的频带资源划分成很多子信道，这些子信道之间相互独立，互不干扰。从媒体的整体频带资源上看，每个子信道只占用该媒体频带资源的一部分

多址(更确切地应该称为多点接入)处理的是动态分配信道给用户。这在用户仅仅暂时性地占用信道的应用中是必须的，而所有的移动通信系统基本上都是属于这种情况。相反，在信道永久地分配给用户的应用中，多址是不需要的(对于无线广播或电视广播站就是这样)

频分复用FDM和时分复用TDM可用于多点接入，相应名词是**频分多址FDMA**和**时分多址TDMA**。从某种程度上，**FDMA**、**TDMA**、**CDMA**可以分别看作是**FDM**、**TDM**、**CDM**的应用

动态接入控制

随机接入

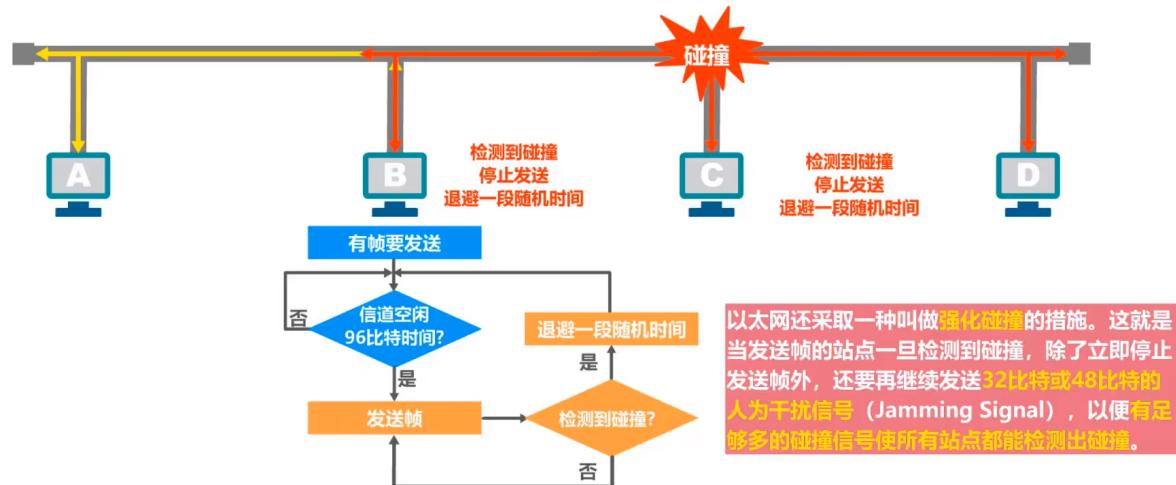
多个主机连接到一根总线上，当信息同一时间传送相遇时就会发生碰撞。

如何协调各主机的工作，使信息避免碰撞是很重要的

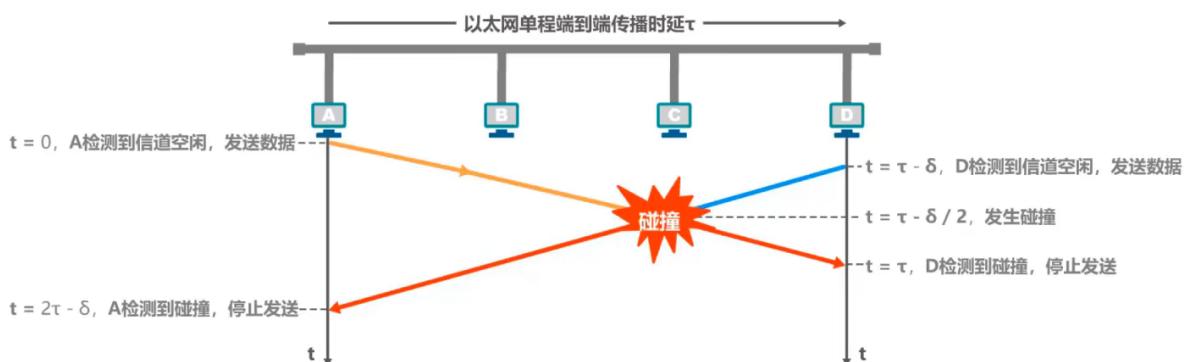
载波监听多址接入/碰撞检测(CSMA/CD)【不用于无线网络】

96比特时间是指发送96比特所需要的时间，也称为帧间最小间隔。其作用是接收方可以检测出一个帧的结束，同时也使得其他站点都有机会平等竞争信道并发送帧

多址接入MA	载波监听CS	碰撞检测CD
<ul style="list-style-type: none"> 多个站连接在一条总线上，竞争使用总线。 	<ul style="list-style-type: none"> 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”）： <ul style="list-style-type: none"> 若检测到总线空闲96比特时间，则发送这个帧； 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，然后发送这个帧。 	<ul style="list-style-type: none"> 每一个正在发送帧的站边发送边检测碰撞（“边说边听”）： 一旦发现总线上出现碰撞，则立即停止发送，退避一段随机时间后再次发送（“一旦冲突，立即停说，等待时机，重新再说”）。



争用期(碰撞窗口)



①为什么 $\tau - \frac{\delta}{2}$ 时刻发送碰撞？

当D开始发送的时，A已经走了 $\tau - \delta$ 时间，所以剩下路程所需的时间是 δ 。

又因为A和D发送速度是相同的，因此对于 δ 时间的路程，每人只需要走 $\frac{\delta}{2}$ 时间就会相遇，即碰撞

$$\text{碰撞时刻} = \tau - \delta + \frac{\delta}{2} = \tau - \frac{\delta}{2}$$

②为什么 $t = \tau$ 时，D检测到碰撞？

根据上边推论，发现D走了 $\frac{\delta}{2}$ 的时间路程后就发送了碰撞，此时它开始返回，经过同样的时间可以回到D

$$\text{检测到碰撞时间} = \frac{\delta}{2} + \frac{\delta}{2} = \delta$$

③为什么 $2\tau - \delta$ 时，A检测到碰撞？

与D检测到碰撞同理，A走了 $\tau - \frac{\delta}{2}$ 时间，因此往回走也是这么多时间

$$A \text{检测到碰撞时间} = \tau - \frac{\delta}{2} + \tau - \frac{\delta}{2} = 2\tau - \delta$$

- 主机最多经过 2τ (即 $\delta \rightarrow 0$)的时长就可以检测到本次发送是否遭受了碰撞
- 因此，以太网的端到端往返传播时延 2τ 称为**争用期或碰撞窗口期**
- 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
- 每一个主机在自己发送帧之后的一小段时间内，存在着遭遇碰撞的可能性。这一小段时间是不确定的。它取决于另一个发送帧的主机到本主机的距离，但**不会超过总线的端到端往返传播时延，即一个争用期时间**
- 显然，在以太网中发生帧的主机越多，**端到端往返传播时延越大，发生碰撞的可能性就越大**。因此，共享式以太网不能连接太多的主机，**使用的总线也不能太长**
 - $10Mb/s$ 以太网把争用期定为512bit发送时间，即 $51.2\mu s$ ，因此其总线长度不能超过 $5120m$ ，但考虑到其他一些因素，如信号衰减等，以太网规定总线长度不能超过 $2500m$

最小帧长

为什么需要规定最小帧长？

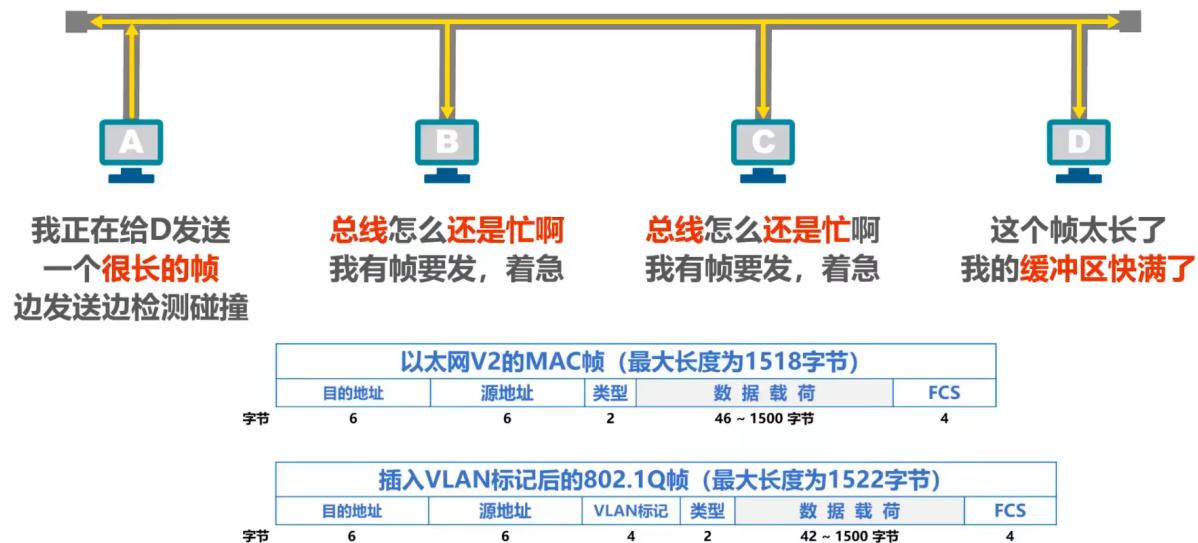
假设A向D发送帧，如果帧很短的话，在 2τ 内就会将帧发送完毕，帧发送完毕后不会进行碰撞检测，而此时依然有发生碰撞的可能。所以必须保证在 2τ 时间内帧不会被发送完，在这段时间里能够保持碰撞检测

- **以太网规定最小帧长为64字节**，即512比特(512比特即争用期)

- 如果要发送的数据非常少，那么必须加入一些填充字节，使帧长不小于64字节
- 以太网的最小帧长保证了主机可在帧发送完成之前就检测到该帧的发送过程中是否遭遇了碰撞
 - 如果在争用期没有检测到碰撞，那么后续发送的数据就一定不会发生碰撞
 - 如果在争用期检测到碰撞就立即中止发送，这时已经发送出去的数据一定小于64字节，因此凡长度小于64字节的帧都是由于碰撞检测而异常中止的无效帧

最大帧长

当帧过长时，其他线路会迟迟得不到资源，同时也可能导致接收方缓冲区溢出，因此帧的最大长度也有规定。



退避算法

当帧发送碰撞后会停止发送，隔一段时间后再次发送，而具体隔多少时间再发送需要根据退避算法得出

CSMA/CD协议 —— 截断二进制指数退避算法



重传次数	k	离散的整数集合{0, 1, ..., ($2^k - 1$)} 示例值	可能的退避时间
1	1	{0, 1}	0 x 2 τ , 1 x 2 τ
2	2	{0, 1, 2, 3}	0 x 2 τ , 1 x 2 τ , 2 x 2 τ , 3 x 2 τ
12	10	{0, 1, 2, 3, 4, 5, ..., 1023}	0 x 2 τ , 1 x 2 τ , 2 x 2 τ , 3 x 2 τ , 4 x 2 τ , 5 x 2 τ , ..., 1023 x 2 τ

- 若连续多次发生碰撞，就表明可能有较多的主机参与竞争信道。但使用上述退避算法可使重传需要推迟的平均时间随重传次数而增大（这也称为动态退避），因而减小发生碰撞的概率，有利于整个系统的稳定。
- 当重传达16次仍不能成功时，表明同时打算发送帧的主机太多，以至于连续发生碰撞，则丢弃该帧，并向高层报告。

极限信道利用率

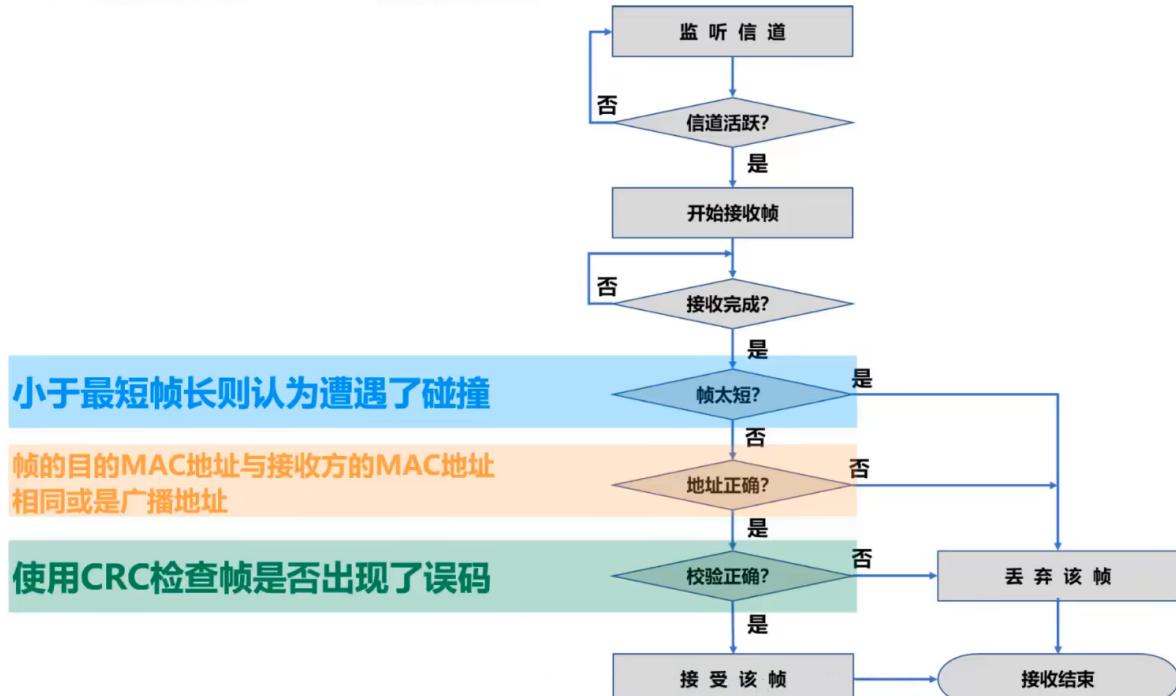
- 各主机发送帧都不会产生碰撞
- 总线一旦空闲就有某个主机立即发送帧
- 每帧的发送时延为 T_0 ，传播时延为 τ ，占用信道的时间为 $T_0 + \tau$

$$\text{极限信道利用率 } S_{max} = \frac{T_0}{T_0 + \tau} = \frac{1}{1 + \frac{\tau}{T_0}}$$

为了令 S_{max} 尽量大，所以应该让 $\frac{\tau}{T_0}$ 尽量小

即 τ 尽量小(以太网端到端距离收到限制)或 T_0 尽量大(以太网帧尽量长)

CSMA/CD协议 —— 帧接收流程

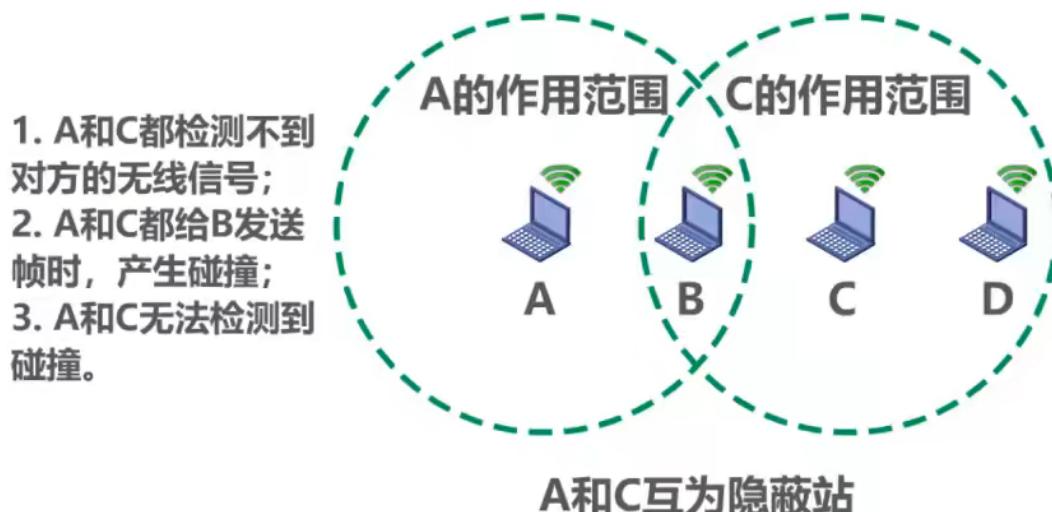


载波监听多点接入/碰撞避免(CSMA/CA)【用于无线网络】

802.11 无线局域网使用 CSMA/CA 协议，在 CSMA 的基础上增加了一个碰撞避免 CA 功能，而不再实现碰撞检测功能

由于不可能避免所有的碰撞，并且无线信道误码率较高，802.11 标准还使用了数据链路层确认机制(停止-等待协议)来保证数据被正确接收

- 在无线局域网中，仍然可以使用载波监听多址接入CSMA，即在发送帧之前先对传输媒体进行载波监听。若发现有其他站在发送帧，就推迟发送以避免碰撞
- 在无线局域网中，不能使用碰撞检测CD，原因如下：
 - 由于无线信道的传输条件特殊，其信号强度的动态范围非常大，**无线网卡上接收到的信号强度往往远小于发送信号的强度**(可能差百万倍)。如果要在无线网卡上实现碰撞检测 CD，对硬件的要求特别高。
 - 即使能够在硬件上实现无线局域网的碰撞检测功能，但由于**无线电波传播的特殊性(存在隐蔽站的问题)**，**进行碰撞检测的意义也不大(如下)**。而有线网络中信号会随着总线到达各个地方，不会出现隐蔽站



- 802.11 的 MAC 层标准定义了两种不同的媒体接入控制方式
 - 分布式协调功能 DCF。在 DCF 方式下，没有中心控制站点，每个站点使用 CSMA/CA 协议通过**争用信道来获取发送权**，这是 802.11 定义的默认方式
 - 点协调功能 PCF。PCF 方式使用集中控制的接入算法(**一般在接入点 AP 实现集中控制**)，是 802.11 定义的可选方式，在实际中较少使用

帧间间隔IFS

802.11标准规定，所有的站点必须在持续检测到信道空闲一段指定时间后才能发送帧，这段时间称为帧间间隔 IFS

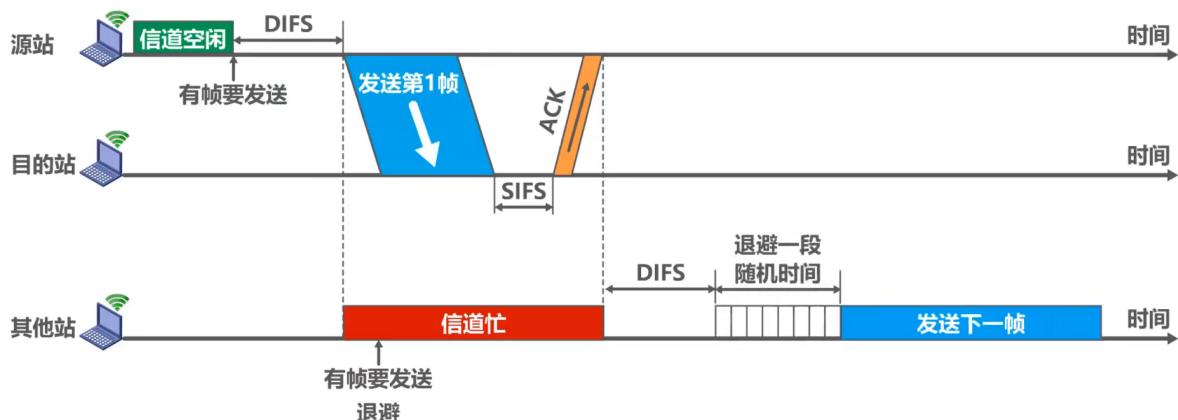
帧间间隔的长短取决于该站点要发送的帧的类型

- 高优先级帧需要等待的时间较短，因此可优先获得发送权
- 低优先级帧需要等待的时间较长。若某个站的低优先级帧还没来得及发送，而其他站的高优先级帧已发送到信道上，则信道变为忙态，因而低优先级帧就只能再推迟发送了。这样就减少了发送碰撞的机会。

常用的两种帧间间隔如下

- 短帧间间隔SIFS($28\mu m$)**。这是最短的帧间间隔，用来分隔开属于一次对话的各帧。一个站点应当能够在这段时间内从发送方式切换到接收方式。使用SIFS的帧类型由ACK帧、CTS帧、由过长的MAC帧分片后的数据帧、以及所有回答AP探询的帧和在PCF方式中接入点AP发出的任何帧
- DCF帧间间隔DIFS($128\mu s$)**。它比短帧间间隔 SIFS 要长得多，在DCF方式中用来发送数据帧和管理帧

工作原理



①为什么源站检测到信道空闲后，还需要等待DIFS时间才将帧发送呢？

因为其他站此时可能有优先级更高的帧需要发送，因此有DIFS时间进行缓冲，若这个时间内没有高优先级的帧要发送，则说明信道是真正的空闲

②为什么目的站接收到帧后还需要等到SIFS时间才返回ACK确认帧呢？

SIFS是最短的帧间间隔，用来分割一次对话的各帧，在这个时间里由接收状态转变为发送状态

③当其他站要发送数据，但是发现此时信道正忙时就会退避一段时间，等信道不忙后再进行操作，接着等待DIFS时间(与①同理)，但为什么等待了DIFS时间后还要退避一段随机时间呢？

因为可能有多个站点在信道忙时都想发送帧，因此它们都会被搁置直至信道不忙，在**DIFS**时间后他们会同时发送，而实际上多个站点同时发送数据会碰撞。因此需要一个随机时间将他们进行错峰发送。

退避算法

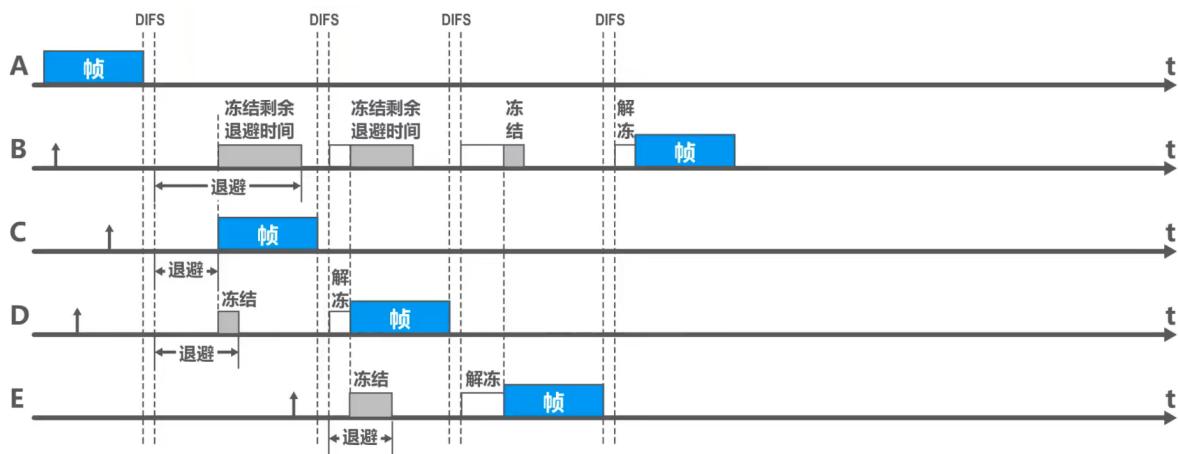
多个站点发送冲突时，各个站点需要退避一段随机时间再进行操作

以下情况必须使用退避算法

- 在发送数据帧之前检测到**信道处于忙状态**时
- 在每一次**重传一个数据帧**时
- 在每一次**成功发送后要连续发送下一个帧**时(这是为了避免一个站点长时间占用通道)

过程

- 在执行退避算法时，站点为退避计时器设置一个随机的退避时间
 - 当退避计时器的时间减小到**0**，就开始发送数据
 - 当退避计时器的时间还未减小到**0**时信道又转变为忙状态，这时就**冻结退避计时器的数值，重新等待信道变为空闲**，再经过**DIFS**后，继续启动退避计时器
- 在进行第*i*次退避时，退避时间在时隙编号 $\{0, 1, \dots, 2^{i+1} - 1\}$ 中随机选择一个，然后乘以基本退避时间(也就是一个时隙的长度)就可以得到随机的退避时间。这样做时为了使不同站点选择相同退避时间的概率减少。当时隙编号达到**255**时(对应第**6**次退避)就不再增加了

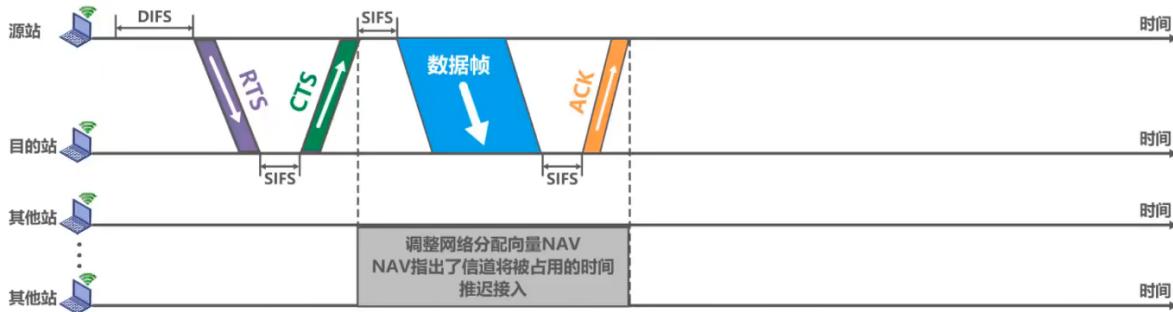


信道预约

为了尽可能减少碰撞的概率和降低碰撞的影响，802.11标准允许要发送数据的站点对信道进行预约

1. 源站在发送数据帧之前先发送一个短的控制帧，称为**请求发生RTS**，它包括源地址、目的地址以及这次通信(包括相应的确认帧)所需的持续时间
2. 若目的站正确收到源站发来的RTS帧，且媒体空闲，就发送一个响应控制帧，称为**允许发送CTS**，它也包括这次通信所需的持续时间。从RTS帧中将此持续时间复制到CTS帧中。
3. 源站收到CTS帧后，再等待一段时间SIFS后，就可发送其数据帧
 - 如果RTS帧发送碰撞，源站就收不到CTS帧，需执行退避算法重传RTS帧
 - 由于RTS帧和CTS帧很短，发生碰撞的概率、碰撞产生的开销及本身的开销都很小。而对于一般的数据帧，其发送时延往往大于传播时延(因为是局域网)，碰撞的概率很大，且一旦发生碰撞而导致数据帧重发就会浪费很多时间，因此用很小的代价对信道进行预约往往是值得的。802.11标准规定了3种情况供用户选择
 - 使用RTS帧和CTS帧
 - 不使用RTS帧和CTS帧
 - 只有当数据帧的长度超过某一数值时才使用RTS帧和CTS帧
4. 若目的站正确收到了源站发来的数据帧，在等待时间SIFS后，就向源站发送确认帧ACK

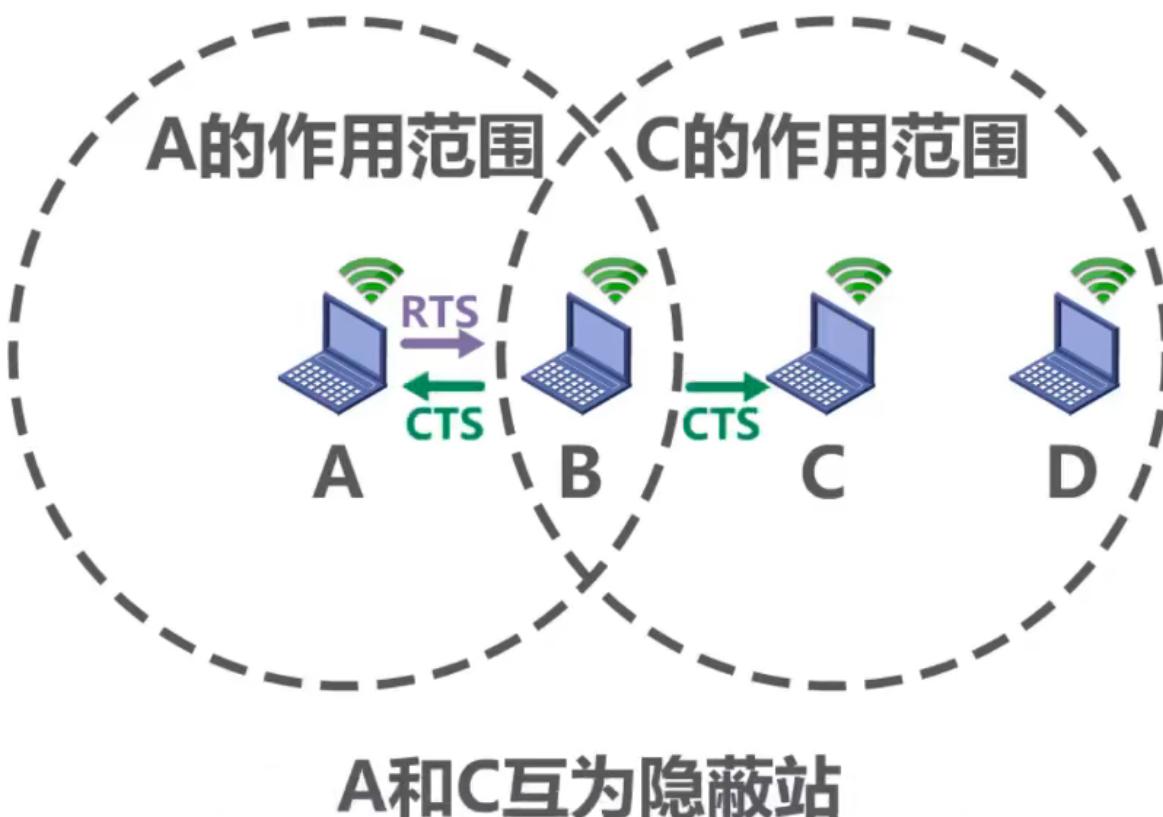
5. 除源站和目的站以外的其他各站，在收到CTS帧(或数据帧)后就推迟接入到无线局域网中。这样就保证了源站和目的站之间的通信不会收到其他站的干扰。



虚拟载波监听

除 RTS 帧和 CTS 帧会携带通信需要持续的时间，数据帧也能携带通信需要持续的时间，这称为 802.11 的虚拟载波监听机制

- 由于利用虚拟载波监听机制，**站点只要监听到RTS帧、CTS帧或数据帧中的任何一个，就能知道信道被占用的持续时间**，而不需要真正监听到信道上的信号，因此虚拟载波监听机制能减少隐蔽站带来的碰撞问题
- 如下图，**A与C虽然互相覆盖不到，但是C可收到B发出的关于A的CTS帧，从而得知A需要占用信道的时间**。在这段时间里，**C不发送数据**，从而解决隐蔽站带来的碰撞问题



4. MAC地址、IP地址和ARP协议

MAC地址

MAC地址是以太网的MAC子层所使用的地址

- 只有一条路径的信道不需要地址，因为没得选
- 当多个主机连接在同一个广播信道上，要想实现两个主机之间的通信，则每个主机都必须有一个唯一的标识，即一个数据链路层地址
- 在每个主机发送的帧中必须携带标识发送主机和接收主机的地址。由于这类地址是用于媒体接入控制MAC(Media Access Control)，因此这类地址被称为MAC地址
 - MAC地址一般被固化在网络适配器(网络适配器)的电可擦可编程只读存储器EEPROM中，因此MAC地址也被称为硬件地址
 - MAC地址有时也被称为物理地址。但是MAC地址不属于物理层而是属于数据链路层
- 一般情况下，用户主机会包含两个网络适配器：有线局域网适配器(有线网卡)和无线局域网适配器(无线网卡)。每个网络适配器都有一个全球唯一的MAC地址。而交换机和路由器往往拥有更多的网络接口，所以会拥有更多的MAC地址。综上所述，严格来说，**MAC地址是对网络上各接口的唯一标识，而不是对网络上各设备的唯一标识**

MAC地址格式

IEEE 802局域网的MAC地址格式

扩展的唯一标识符EUI
EUI-48

组织唯一标识符OUI (由IEEE的注册管理机构分配)								网络接口标识符 (由获得OUI的厂商自行随意分配)							
b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

标准表示法： XX-XX-XX-XX-XX-XX



例如： 00-0C-CF-93-8C-92

其他表示法： XX:XX:XX:XX:XX:XX

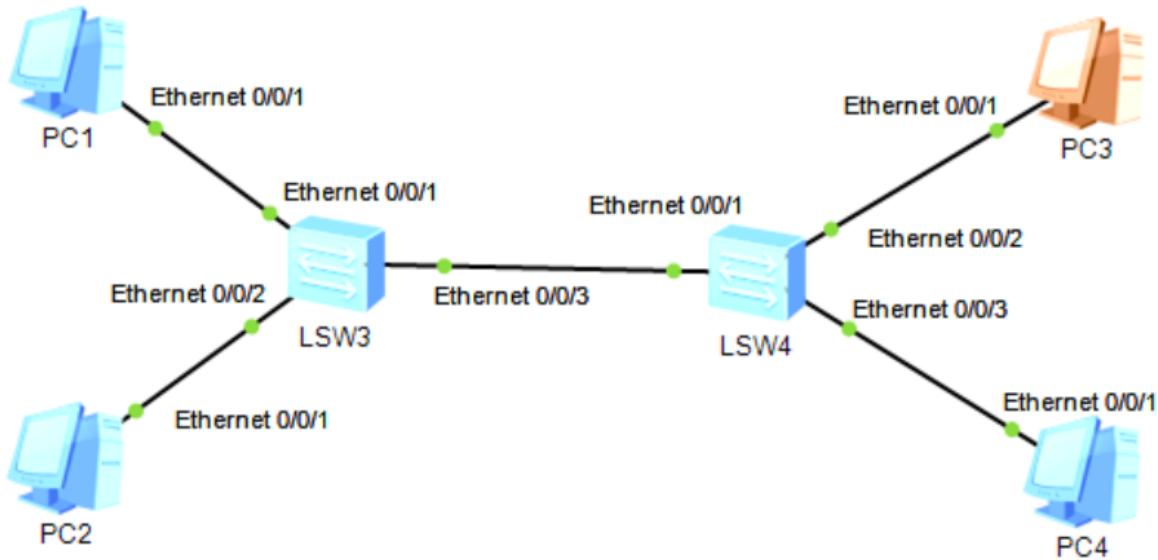


例如： 00:0C:CF:93:8C:92

XXXX.XXXX.XXXX



例如： 000C.CF93.8C92



单播地址、广播地址与多播地址

- **单播MAC地址即明确的目的MAC地址。**将此地址填入帧的目的地址栏中，接收到该帧的主机将此地址与自身MAC地址进行匹配，若相同则接收，不同则丢弃
- **广播MAC地址为FF-FF-FF-FF-FF-FF**，将此地址填入帧的目的地址栏中，接收到该帧的主机检索该地址发现是广播地址，因此接收该帧
- **MAC地址中第一字节后4比特为(1,3,5,7,9,B,D,F)时，MAC地址是多播地址。**将此地址填入帧的目的地址栏中，接收到该帧的主机**将此多播地址与自己多播组列表中的地址进行逐一配对**，如果有匹配的项，则接收，否则丢弃

IP地址

IP地址是TCP/IP体系结构的网际层所使用的地址

IP地址是Internet上的主机和路由器所使用的地址，由两部分信息构成

- **网络编号：**标识因特网上数以百万计的网络
- **主机编号：**标识同一网络上不同主机(或路由器各接口)

MAC地址不具备区分不同网络的功能，而IP地址可以通过网络号做到

如果只是一个**单独网络**，不接入因特网，则使用**MAC地址就足够了**(这不是一般用户的应用方式)

如果主机所在的网络要接入因特网，则**IP地址和MAC地址都需要使用**

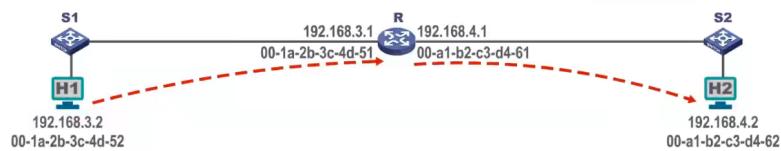
数据包转发过程中IP地址与MAC地址变换情况



- 数据报转发过程中源IP地址和目的IP地址保持不变
- 数据包转发过程中源MAC地址和目的MAC地址(由ARP协议获得)逐个链路改变
- 路由器上有路由表，记录了要到目的IP地址，先要走哪些路(即下一跳位置)。所以H1起初只带有目的IP地址，查询路由表后路由器指路，从而知道下一步该往哪里走，从而不断转发直至到达目的地。

【2018年题37】路由器R通过以太网交换机S1和S2连接两个网络，R的接口、主机H1和H2的IP地址与MAC地址如下图所示。若H1向H2发送一个IP分组P，则H1发出的封装P的以太网帧的目的MAC地址、H2收到的封装P的以太网帧的源MAC地址分别是 D

- A. 00-a1-b2-c3-d4-62 00-1a-2b-3c-4d-52
B. 00-a1-b2-c3-d4-62 00-1a-2b-3c-4d-61
C. 00-1a-2b-3c-4d-51 00-1a-2b-3c-4d-52
D. 00-1a-2b-3c-4d-51 00-a1-b2-c3-d4-61



【解析】

在数据包的转发过程中，源IP地址和目的IP地址始终保持不变；而源MAC地址和目的MAC地址逐段链路（或逐个网络）改变。

数据包 传输区间	在网络层写入IP数据报首部的IP地址		在数据链路层写入MAC帧首部的MAC地址	
	源IP地址	目的IP地址	源MAC地址	目的MAC地址
H1 ---> R	192.168.3.2	192.168.4.2	00-1a-2b-3c-4d-52	00-a1-b2-c3-d4-51
R ---> H2	192.168.3.2	192.168.4.2	00-a1-b2-c3-d4-61	00-a1-b2-c3-d4-62

ARP协议

地址解析协议 ARP 属于 TCP/IP 体系结构的网际层，其作用是已知设备所分配到的 IP 地址，使用 ARP 协议可以通过该 IP 地址获取到设备的 MAC 地址

- 数据的发送需要经过链路，仅仅知道 IP 地址是无法在数据链路层实现传输的，因此我们需要得到 IP 地址与 MAC 地址的对应关系，即地址解析。
- 每台主机都会有一个 ARP 高速缓存表，记录有 IP 地址与 MAC 地址的对应关系
- 初始 ARP 高速缓存表为空，假设此时 B 知道 C 的 IP 地址，准备发送信息，但是由于不知道 C 的 MAC 地址，因此无法封装数据帧，所以此时会广播一个 ARP 请求报文(封装在 MAC 帧中，目的地址为广播地址 FF-FF-FF-FF-FF-FF)，内容如下：

- 我的 IP 地址是: `xxx`; 我的 MAC 地址是: `xxx`; 我想知道 c 主机的 MAC 地址是多少? 这个广播帧会被该广播域的所有主机收到
 - A 主机收到后交由上层处理, 发现 B 问的不是他, 所以不予理会
 - C 收到后交由上层处理, 发现这个 IP 地址正是自己, 因此首先将 B 的 MAC 地址和 IP 地址的对应关系记录到自己的高速缓存表中, 接着返回给 ARP 响应报文(封装在 MAC 帧中, 目的地为 B 的 MAC 地址), 其中包含自己的 MAC 地址
- ARP 高速缓存条目有静态与动态两种类型
 - 动态代表是通过广播自动获取的条目, 生命周期默认为两分钟
 - 静态是手工设置的条目, 不同操作系统下的生命周期不同。
 - ARP 协议只能在一段链路或一个网络上使用

5. 集线器与交换机

集线器

集线器 HUB 的主要功能是对接收到的信号进行再生整形放大, 以扩大网络的传输距离, 同时把所有节点集中在以它为中心的节点上。

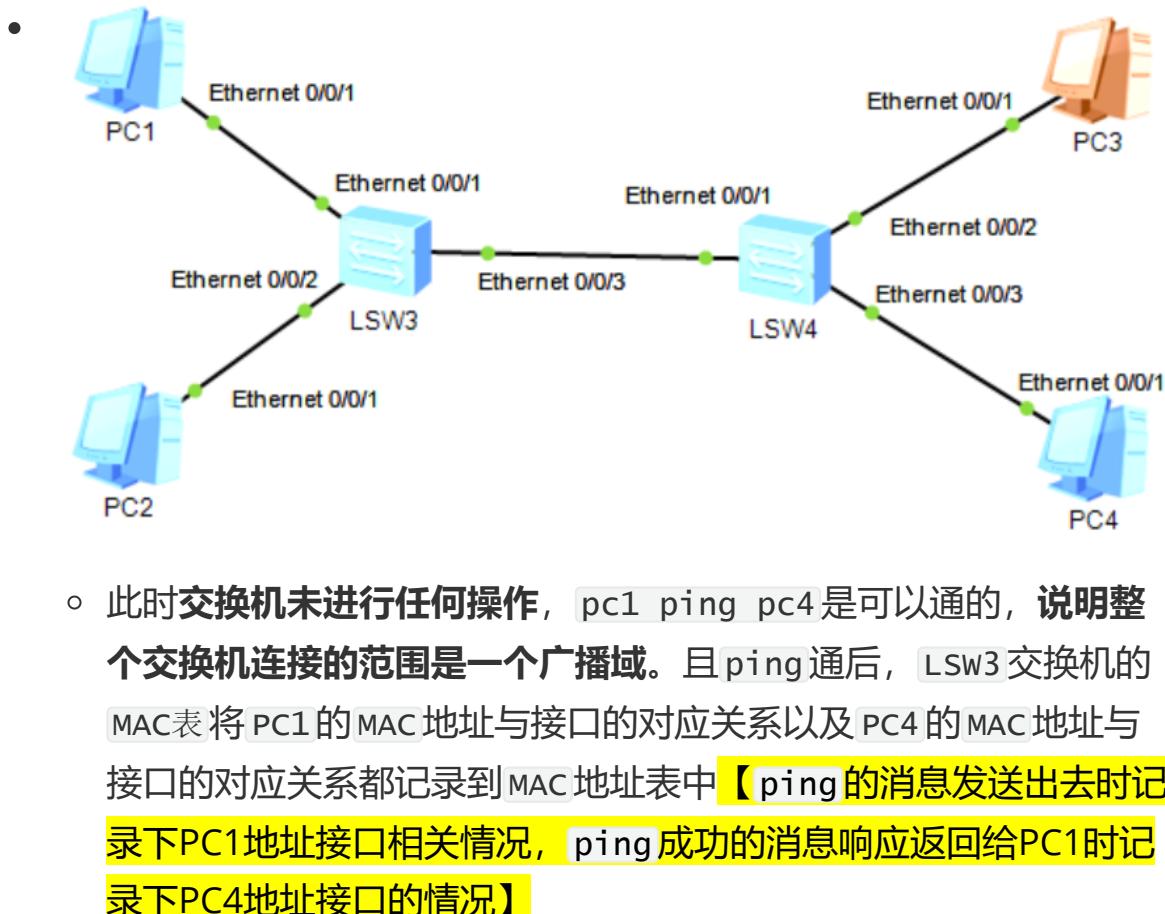
- 使用集线器的以太网在逻辑上仍是一个总线网, 各站共享总线资源, 使用的还是 CSMA/CD 协议
- 集线器只工作在物理层, 它的每个接口仅简单地转发比特, 不进行碰撞检测(由各站网卡检测)
- 集线器一般都有少量的容错能力和网络管理功能。例如, 若网络中某个网卡出了故障, 不停地发送帧。此时, 集线器可以检测到这个问题, 在内部断开与出故障网卡的连线, 使整个以太网仍然能正常工作
- 集线器是半双工模式, 收发不能同时进行, 收到帧后会广播到除本身接口外的各个接口。

交换机

交换机是一种负责转发信号的网络设备, 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

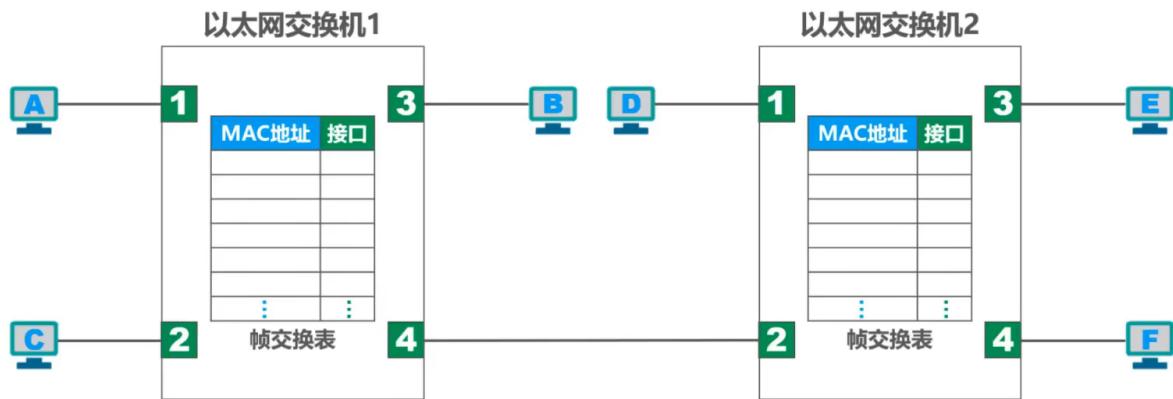
- 以太网交换机通常由多个接口, 每个接口都可以直接与一台主机或另一个以太网交换机相连。一般都工作在全双工方式

- 以太网交换机具有并行性，能同时连通多对接口，使多对主机能同时通信，无碰撞(不使用CSMA/CD协议)
- 以太网交换机一般都具有多种速率的接口
- 以太网交换机工作在数据链路层(也包括物理层)，它收到帧后，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
- 以太网交换机是一种即插即用的设备，其内部的帧交换表是通过自学习算法自动地逐渐建立起来的
- 帧的两种转发方式
 - 存储转发
 - 直通交换：采用基于硬件的交叉矩阵(交换时延非常小，但不检查是否有差错)
- **交换机每个接口是一个独立的碰撞域**



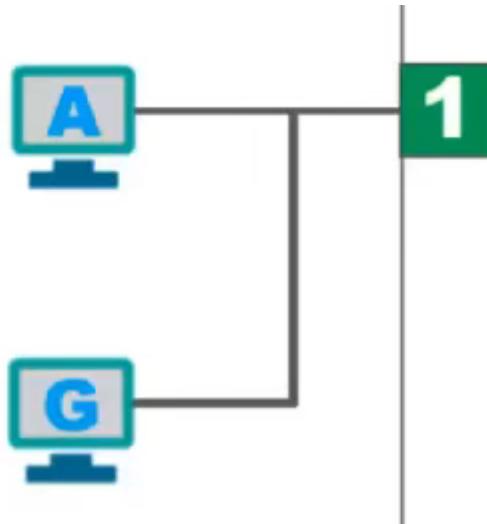
```
[L1]display mac-address
MAC address table of slot 0:
-----
MAC Address      VLAN/
VSI/SI          PEVLAN CEVLAN Port        Type      LSP/LSR-ID
                                                               MAC-Tunnel
-----
5489-9871-7e26 1           -     -       Eth0/0/1    dynamic   0/- 
5489-9824-0cd8 1           -     -       Eth0/0/3    dynamic   0/- 
-----
Total matching items on slot 0 displayed = 2
```

交换机自学习和转发帧流程



- 初始交换机 1 和 2 的 MAC 地址表都为空
- 此时有以下任务：① A-->B ② B-->A
 1. A 发送给 B 的数据通过端口 1 进入交换机，因此交换机先将 MAC 地址 A 接口 1 (表示如果要去 MAC 地址 A，可以走接口 1) 记录到 MAC 表中，同时扫描 MAC 表查看是否有 MAC 地址 B 对应的接口，发现没有。因此将这个帧从除来源外的所有端口发送出去【泛洪】。
 - 交换机 1 的端口 2 发送此帧到主机 C，主机 C 对比 MAC 地址后发现不是给自己的帧，因此丢弃
 - 交换机 1 的 3 端口发送此帧到主机 B，主机 B 对比 MAC 地址后发现是给自己的帧，因此收下该帧，交付给高层【注意：此时交换机 MAC 地址表并没有学习新的条目】
 - 交换机 1 的 4 端口发送此帧到交换机 2 的端口 2，交换机 2 首先将 MAC 地址 A 接口 2 记录到 MAC 地址表中，接着扫描 MAC 表，发现没有找到 MAC 地址 B 对应的接口，因此将这个帧从除来源外的所有端口发送出去【泛洪】。
 - 同理，接口 1, 3, 4 发送出去给主机后经过比对发现不是自己的帧，因此丢弃
 2. B-->A 时，交换机 1 的 MAC 地址表已经有了 MAC 地址 A 接口 1 这个条目
 - B 发送给 A 的帧从接口 3 进入交换机，交换机首先将 MAC 地址 B 接口 3 记录到 MAC 地址表中，接着查询 MAC 地址表是否有 MAC 地址 A 对应的接口，发现接口 1 对应的就是 MAC 地址 A，因此从接口 1 转发出去
 - 帧通过接口 1 到达主机 A，主机 A 经过比对后发现这是自己的帧，因此将其接收交付给高层

3. 此时若有一个与 A 连接在同一总线上的主机 G 向 A 发送帧，过程如何？



- G 向 A 发送帧时，由于 A 与 G 处在同一总线上，因此 G 发送的帧会直接传送到 A 处，A 进行比对发现这是发送给自己的数据，因此将数据接收
- 同时数据会跑向交换机 1 的端口 1，所以交换机 1 会将 MAC 地址 G 接口 1 记录到 MAC 表中，同时扫描 MAC 地址表看看是否有 MAC 地址 A 对应的接口，发现接口 1 就是。但是刚刚学习到的 MAC 地址 G 也是来自接口 1，因此交换机知道接口 1 是来源，没有必要将数据再从这个接口转发出去。

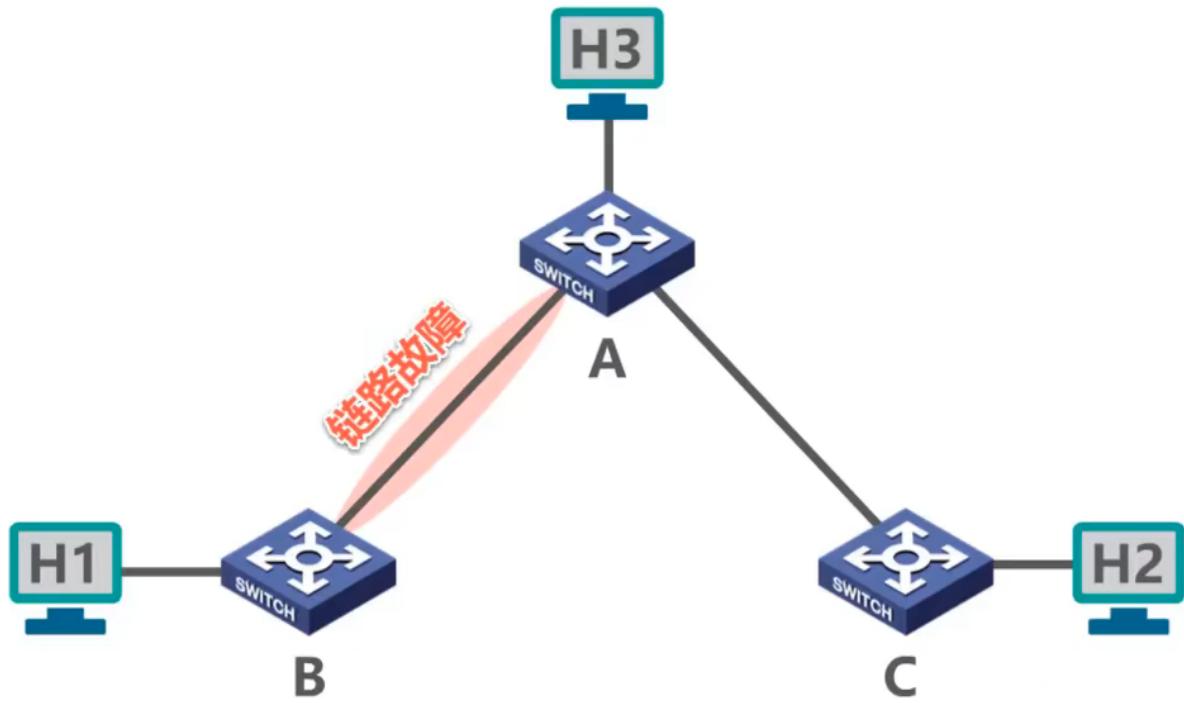
MAC地址表中的每条记录都有自己的有效时间，到期自动删除。这是因为 MAC地址与交换机接口的对应关系并不是永久性的

STP生成树协议

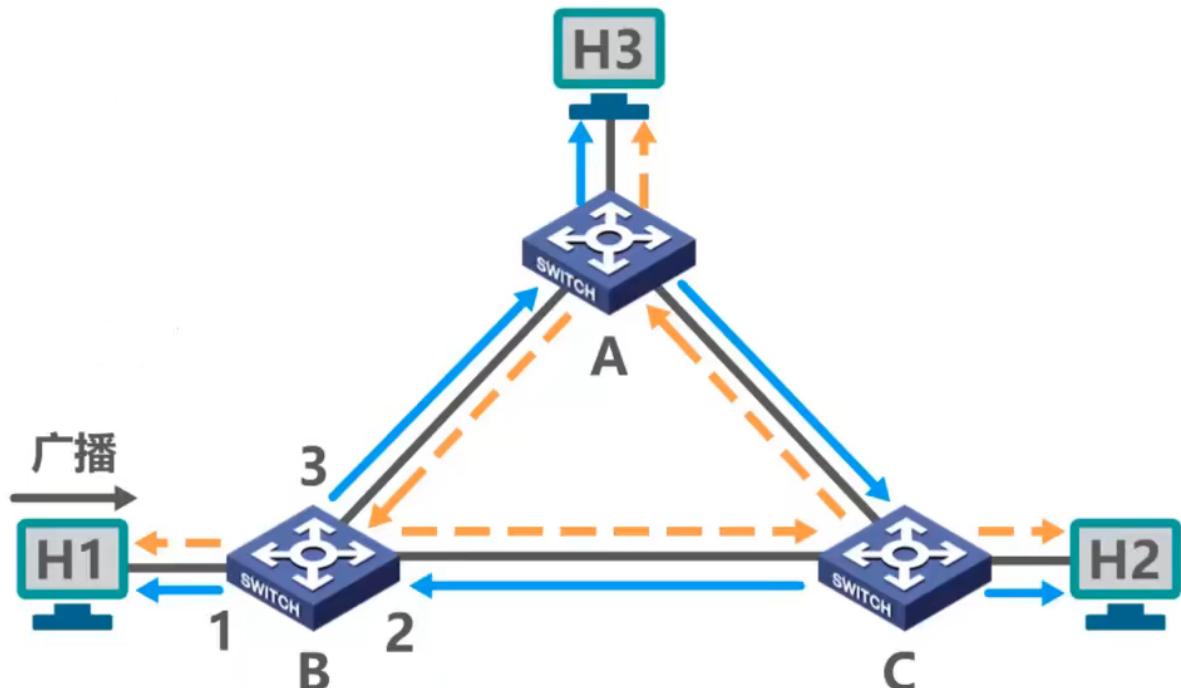
STP 可以在增加冗余链路来提高网络可靠性的同时又避免网络环路带来的各种问题

当交换机之间链路较少时，某一条链路发送故障，会导致其中有些主机无法通信，即链路不可靠。

如下图，A 与 B 之间的链路发送故障后，H1 便无法与 H2 和 H3 进行通信



很容易想到的一个方法是在B与C之间多拉一条链路，这样即使A与B之间的链路发送故障，H1、H2、H3之间还是能够相互通信，但是这会带来一些问题。如下，当H1发送一个广播帧时，我们针对交换机B进行过程分析



MAC地址	接口
H1	1
H1	2
H1	3
⋮	⋮

- 首先 H1 发送的帧进入交换机，交换机将 MAC 地址 H1 接口 1 记录到 MAC 表中，接着检索 MAC 表，发现没有 MAC 地址 H2 的接口条目，因此进行泛洪，交换机 A 和 C 都接收到此帧
- 交换机 A/C 首先将 MAC 地址 H1 接口 1 记录到自身的 MAC 表中，接着检索 MAC 表，发现没有广播帧的接口条目，因此进行泛洪。A 泛洪的帧会被 H3、交换机 B、C 收到；C 泛洪的帧会被 H2、交换机 A、B 收到。H2 与 H3 接收帧后发现是一个广播帧，于是接收并交上层处理
- 我们针对交换机 B 进行分析
 - 此刻它收到了来自交换机 C 的泛洪，帧内信息依然是 MAC 地址 H1 【源 MAC 地址】，首先它会将此条目添加进 MAC 表中，发现原先有 MAC 地址 H1 接口 1 的记录，此时它会认为这条记录已经出现错误，因此更新为 MAC 地址 H1 接口 2。由于该条目的来源就是接口 2，所以交换机不会再从接口 2 转发出去，而是在接口 1、3 处进行泛洪……
 - 同时 B 也收到来自交换机 A 的泛洪，情况与交换机 C 泛洪类似，于是又将 MAC 地址 H1 接口 2 更新为 MAC 地址 H1 接口 3 ……
- 因此帧会在交换机中不停转发，究其原因是网络形成了环路，**网络环路会带来以下问题**
 - 广播风暴**
 - 大量消耗网络资源，使得网络无法正常转发其他数据帧
 - 主机收到重复的广播帧**
 - 大量消耗主机资源
 - 交换机的帧交换表震荡(内容不断更新)**

生成树原理

- 不论交换机之间采用怎样的物理连接，**交换机都能够自动计算并构建一个逻辑上没有环路的网络，其逻辑拓扑结构必须是树型的(无逻辑环路)**
- 最终生成树的逻辑拓扑要确保连通整个网络**
- 当首次连接交换机或网络物理拓扑发生变化时(有可能是人为改变或故障)，交换机都将进行生成树重新计算**

VLAN虚拟局域网

一种将局域网内的设备划分成与物理位置无关的逻辑组的技术，这些逻辑组具有某些共同的需求，每个VLAN就是一个独立的广播域

为什么需要VLAN？

随着交换式以太网规模的扩大，广播域相应扩大，而巨大的广播域会带来很多弊端

1. 广播风暴

如数台交换机连接了数台主机，当主机A要向主机B发送数据帧，此时各交换机的MAC表均为空，因此帧每到一个交换机就会进行泛洪，由于网络巨大，因此泛洪的范围也巨大

2. 难以管理和维护

3. 潜在的安全风险

交换机端口类型

缺省VLAN ID

华为交换机上叫PVID，每个端口有且只有一个PVID。默认情况下端口的PVID都为1(即端口属于VLAN 1)

端口上接收时总希望能够打上标签，发送出去时候总希望能去除标签

① Access端口

- Access端口一般用于终端设备与交换机之间

注意：交换机与路由器连接的接口也需要使用access接口。这是因为路由器中的消息也不带VLAN标签，就像终端一样，保证路由器的数据能够进入交换机领域，数据由路由器进入交换机是会被打上默认标签，接着按照交换机间VLAN的规则行走。简单理解就是把路由器当作终端

- 交换机初始端口类型是Access
- Access端口只能属于一个VLAN
- Access端口的PVID值与端口所属VLAN的ID相同(默认为1)

- Access 端口接收方法
 - 一般只接收"未打标签"的普通以太网MAC帧。根据接收帧的端口 PVID 值给帧"打标签"，即插入4字节的 VLAN 标记字段，字段中的 VID 取值与端口 PVID 取值相等
- Access 端口发送处理方法
 - 若帧中的 VID 与端口的 PVID 相等，则"去标签"并转发该帧；否则不转发

②Trunk端口

- Trunk 端口一般用于交换机之间或交换机与路由器之间的互连
- Trunk 端口可以属于多个 VLAN
- 用户可以设置 Trunk 端口的 PVID 值，默认情况下，Trunk 端口 PVID 值为 1
- Trunk 端口发送处理方法
 - 对 VID 等于 PVID 的帧，"去标签，再转发"
 - 对 VID 不等于 PVID 的帧，直接转发
- Trunk 端口接收处理方法
 - 接收"未打标签"的帧。根据接收帧的端口的 PVID 给帧"打标签"，即插入4字节的 VLAN 标记字段，字段中的 VID 取值与端口的 PVID 取值相等
 - 直接接收"已打标签的帧"

③Hybrid端口

- Hybrid 端口既可以用于交换机之间或交换机与路由器之间的互连(同 Trunk 端口)，也可用于交换机与用户计算机之间的互连(同 Access 端口)
- Hybrid 端口可以属于多个 VLAN (同 Trunk 端口)
- 用户可以设置 Hybrid 端口的 PVID 值。默认情况下，Hybrid 端口的 PVID 值为 1
- Hybrid 端口发送处理方法↓
 - 查看帧的 VID 是否在端口的"去标签"列表中
 - 若存在，则"去标签"后转发

- 若不存在，则直接转发
- Hybrid 端口接收处理方法(同 Trunk)
 - 接收“未打标签”的帧，根据接收帧的端口的 PVID 给帧“打标签”，即插入 4 字节 VLAN 标记字段，字段中的 VID 取值与端口的 PVID 取值相等
 - 接收“已打标签的帧”

[华为交换机 Hybrid 端口应用例子](#)

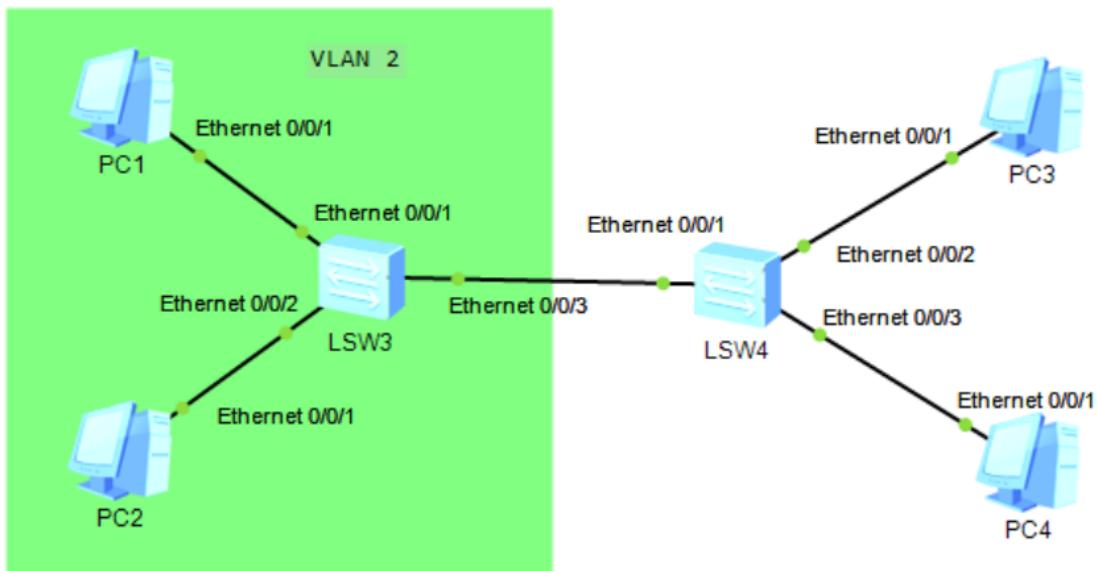
如何分割广播域？

1. 路由器

由于路由器属于网络层设备，默认情况下不对广播数据包进行转发，因此自然达到隔离的效果。但是成本较高，局域网内部全靠路由器分割广播域是不现实的

2. VLAN 虚拟局域网技术

默认情况下，交换机下的主机会被划入 VLAN 1，可以通过设置相关语句为交换机设置 VLAN，如下是将 PC1 与 PC2 划入 VLAN2 的方法



```

1 LSW3:
2 valan batch 2 //在LSW3上创建VLAN2
3 interface Ethernet 0/0/1 //进入交换机接口1
4 port link-type access //将此接口类型设置为access
5 port default vlan 2 //信息经过此接口时若没有VLAN标签则打上
VLAN2标签；若有VLAN标签则检查是
否为VLAN2，若为VLAN2则去除标签并转发；若非VLAN2，则不转发。
6 //对接口2的设置同理

```

设置完毕后，发现PC1不能ping通PC4，因为此时他们不属于一个同一个广播域，但是可以ping通PC2，因为PC1与PC2属于同一个广播域VLAN 2

```

PC1

基础配置 命令行 组播 UDP发包工具 串口
0.00% packet loss
round-trip min/avg/max = 62/65/78 ms

PC>ping 192.168.25.4

Ping 192.168.25.4: 32 data bytes, Press Ctrl_C to break
From 192.168.25.1: Destination host unreachable

--- 192.168.25.4 ping statistics ---
5 packet(s) transmitted
0 packet(s) received
100.00% packet loss

PC>ping 192.168.25.2

Ping 192.168.25.2: 32 data bytes, Press Ctrl_C to break
From 192.168.25.2: bytes=32 seq=1 ttl=128 time=47 ms
From 192.168.25.2: bytes=32 seq=2 ttl=128 time=31 ms
From 192.168.25.2: bytes=32 seq=3 ttl=128 time=16 ms
From 192.168.25.2: bytes=32 seq=4 ttl=128 time=46 ms
From 192.168.25.2: bytes=32 seq=5 ttl=128 time=47 ms

```

[eNSP路由与交换技术笔记](#)

VLAN实现机制

插入VLAN标记后的802.1Q帧 (最大长度1522字节)	6字节 目的MAC地址	6字节 源MAC地址	4字节 VLAN标记	2字节 类型	46 ~ 1500字节 数据载荷	4字节 FCS
-----------------------------------	----------------	---------------	---------------	-----------	---------------------	------------

- VLAN 标记的最后12比特称为VLAN标识符VID，它唯一地标志了以太网帧属于哪一个 VLAN
 - VID 的取值范围是 0~4095
 - 0 和 4095 都不用来表示 VLAN，因此用于表示 VLAN 的 VID 的有效范围是 1~4094
- 802.1Q 帧时由交换机来处理的，而不是用户主机来处理的
 - 当交换机收到普通的以太网帧时，会将其插入 4 字节的 VLAN 标记转变为 802.1Q，简称“打标签”
 - 当交换机转发 802.1Q 帧时，可能会删除其 4 字节 VLAN 标记转变为普通以太网帧，简称“去标签”

第4章 网络层

| 主要任务是实现网络互连，进而实现数据包在各网络之间的传输

1. 面向连接的虚电路服务

- 可靠的通信由网络来保证
- 必须建立网络层的连接----虚电路VC(virtual circuit)
- 通信双方沿着已建立的虚电路发送分组
- 目的主机的地址仅在连接建立阶段使用，之后每个分组的首部只需携带一条虚电路的编号(构成虚电路的每一段链路都有一个虚电路编号)。
- 这种通信方式如果再使用可靠传输的网络协议，就可使所发送的分组最终正确到达接收方(无差错按序到达、不丢失、不重复)。
- 通信结束后，需要释放之前所建立的虚电路

2. 无连接的数据报服务

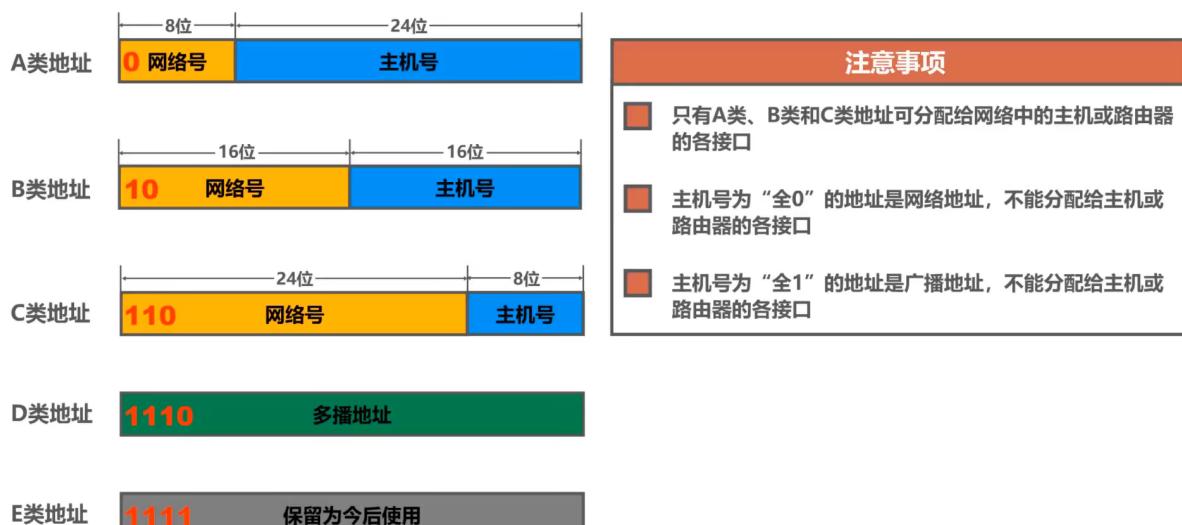
- 可靠通信应当由用户主机来保证
- 不需要建立网络层连接
- 每个分组可走不同路径
- 每个分组的首部必须携带目的主机的完整地址
- 这种通信方式所传送的分组可能误码、丢失、重复和失序
- 由于网络本身不提供端到端的可靠传输服务，这就使网络中的路由器可以做得比较简单，而且价格低廉

- 因特网采用了这种设计思想，也就是**将复杂的网络处理功能置于因特网的边缘(用户主机和其内部的运输层)**，而将相对简单的尽最大努力的分组交付功能置于因特网核心。

3. IPv4地址

IPv4 地址就是因特网上的**每一台主机(或路由器)的每一个接口分配一个在全世界范围内是唯一的32比特的标识符**

分类编址



- A类地址**网络号第1位固定为**0**，网络号后面部分不能全**0**，也不能全**1**，所以**网络号范围是1~126**
- B类地址**网络号前2位固定为**10**，网络号后面部分可以全取**0**或**1**，所以**网络号范围是128.0~191.255**
- C类地址**网络号前3位固定为**110**，网络后后面部分可以全取**0**或**1**，所以**网络号范围是192.0.0~223.255.255**
- D类地址**为多播地址，**IP地址为224.0.0.0~239.255.255.255**

地址**0.0.0.0**是一个特殊的IPv4地址，只能作为源地址使用，表示“**在本网络上的本主机**”。封装有DHCP Discovery报文的IP分组的源地址使用**0.0.0.0**；

以**127**开头且后面三个字节非“全0”或“全1”的IP地址是一类特殊的IPv4地址，既可以作为源地址使用，也可以作为目的地址使用，用于本地软件环回测试，例如常用的环回测试地址**127.0.0.1**；

地址**255.255.255.255**是一个特殊的IPv4地址，只能作为目的地址使用，表示“**只在本网络上进行广播（各路由器均不转发）**”。