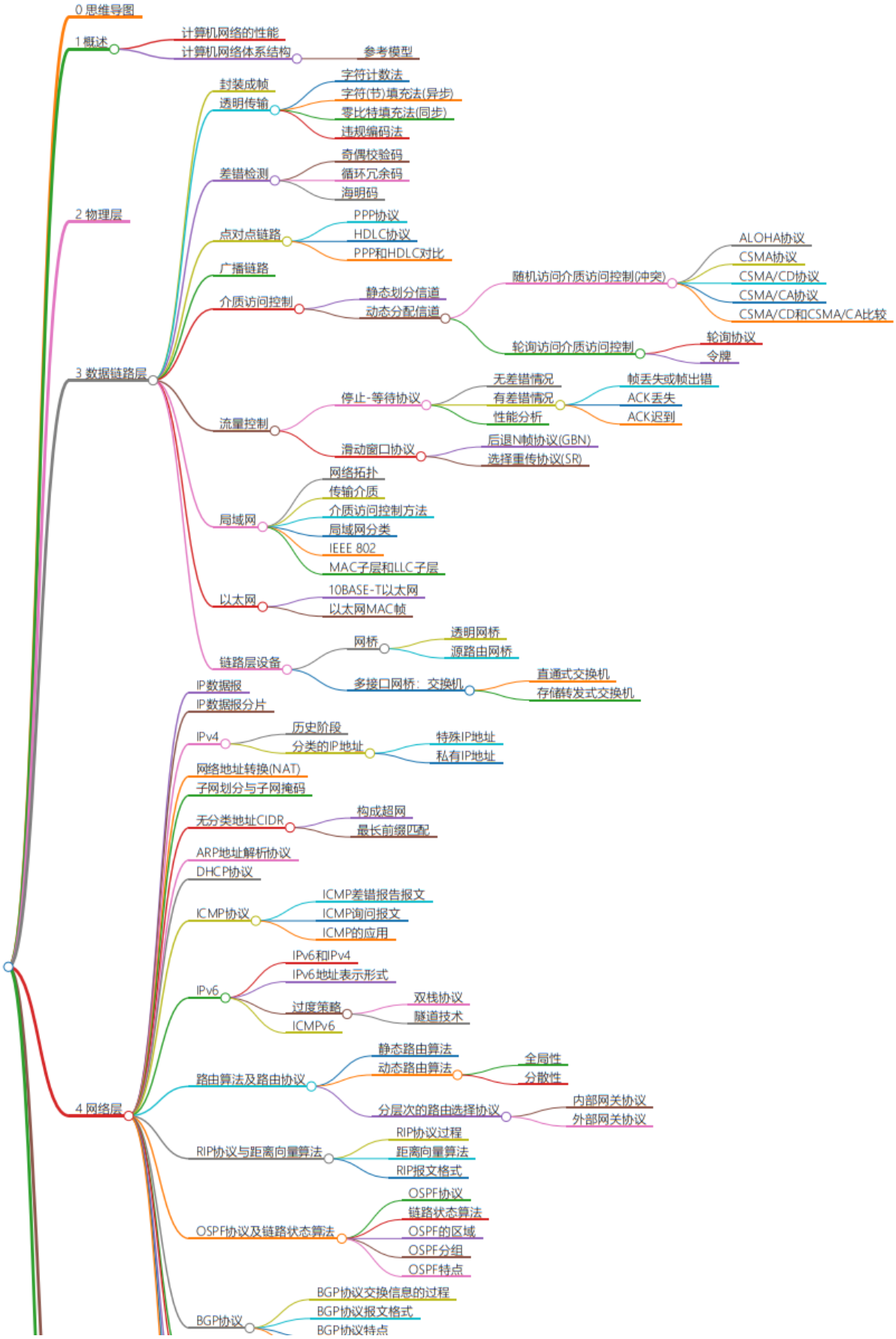
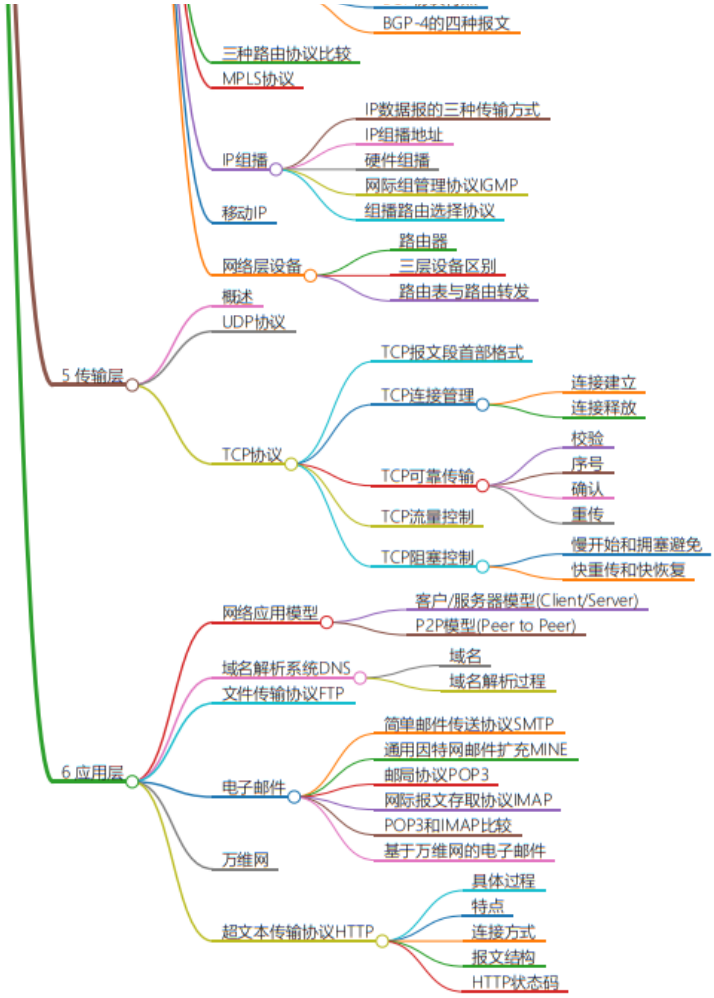
0 思维导图


0 思维导图
1 概述
计算机网络的性能
计算机网络体系结构
参考模型
2 物理层
3 数据链路层
封装成帧
透明传输
字符计数法
字符(节)填充法(异步)
零比特填充法(同步)
违规编码法
差错检测
奇偶校验码
循环冗余码
海明码
点对点链路
PPP协议
HDLC协议
PPP和HDLC对比
广播链路
介质访问控制
静态划分信道
动态分配信道
随机访问介质访问控制(冲突)
ALOHA协议
CSMA协议
CSMA/CD协议
CSMA/CA协议
CSMA/CD和CSMA/CA比较
轮询访问介质访问控制
轮询协议
令牌
流量控制
停止-等待协议
无差错情况
有差错情况
帧丢失或帧出错
ACK丢失
ACK迟到
性能分析
滑动窗口协议
后退N帧协议(GBN)
选择重传协议(SR)
局域网
网络拓扑
传输介质
介质访问控制方法
局域网分类
IEEE 802
MAC子层和LLC子层
以太网
10BASE-T以太网
以太网MAC帧
链路层设备
网桥
透明网桥
源路由网桥
多接口网桥：交换机
直通式交换机
存储转发式交换机
4 网络层
IP数据报
IP数据报分片
IPv4
历史阶段
分类的IP地址
特殊IP地址
私有IP地址
网络地址转换(NAT)
子网划分与子网掩码
无分类地址CIDR
构成超网
最长前缀匹配
ARP地址解析协议
DHCP协议
ICMP协议
ICMP差错报告报文
ICMP询问报文
ICMP的应用
IPv6
IPv6和IPv4
IPv6地址表示形式
过度策略
双栈协议
隧道技术
ICMPv6
路由算法及路由协议
静态路由算法
动态路由算法
全局性
分散性
分层次的路由选择协议
内部网关协议
外部网关协议
RIP协议与距离向量算法
RIP协议过程
距离向量算法
RIP报文格式
OSPF协议及链路状态算法
OSPF协议
链路状态算法
OSPF的区域
OSPF分组
OSPF特点
BGP协议
BGP协议交换信息的过程
BGP协议报文格式
BGP协议特点



https://file.moluuser.com/mind/cn.html

1 概述

- 互联网的基本特点：连通性和共享。
- 计算机网络由若干结点和连接这些结点的链路组成。
- 互联网基础结构发展的三个阶段：
  - 从单个网络ARPANET向互连网发展的过程。
  - 建成了三级结构的互连网：主干网，地区网，校园网(或企业网)。
  - 逐渐形成了多层次的ISP(互联网服务提供商)结构的互联网。
- 互联网交换点IXP：允许两个网络直接相连并交换分组。
- 互联网的组成：边缘部分(用来进行通信和资源共享)和核心部分(提供连通性和交换)。
- 交换方式：
  - 电路交换：计算机终端之间通信时，一方发起呼叫，独占一条物理线路。当交换机完成接续，对方收到发起端的信号，双方即可进行通信。在整个通信过程中双方一直占用该电路。它的特点是实时性强，时延小，交换设备成本较低。但同时也带来线路利用率低，电路接续时间长，通信效率低，不同类型终端用户之间不能通信等缺点。电路交换比较适用于信息量大、长报文，经常使用的固定用户之间的通信。

- 报文交换：将用户的报文存储在交换机的存储器中。当所需要的输出电路空闲时，再将该报文发向接收交换机或终端，它以“存储——转发”方式在网内传输数据。报文交换的优点是中继电路利用率高，可以多个用户同时在一条线路上传送，可实现不同速率、不同规程(规则流程)的终端间互通。但它的缺点也是显而易见的。以报文为单位进行存储转发，网络传输时延大，且占用大量的交换机内存和外存，不能满足对实时性要求高的用户。报文交换适用于传输的报文较短、实时性要求较低的网络用户之间的通信，如公用电报网。
- 分组交换：分组交换实质上是在“存储——转发”基础上发展起来的。它兼有电路交换和报文交换的优点。分组交换在线路上采用动态复用技术传送按一定长度分割为许多小段的数据——分组。每个分组标识后，在一条物理线路上采用动态复用的技术，同时传送多个数据分组。把来自用户发端的数据暂存在交换机的存储器内，接着在网内转发。到达接收端，再去掉分组头将各数据字段按顺序重新装配成完整的报文。分组交换比电路交换的电路利用率高，比报文交换的传输时延小，交互性好。
  - 数据报方式：为网络层提供无连接服务，可能发生乱序、重复和丢失。
  - 虚电路方式(结合了分组交换和电路交换的优点)：路径上的所有结点维持虚电路的建立。每个分组携带虚电路号、分组号、检验和等控制信息，无需目的地址。

	数据报服务	虚电路服务
连接的建立	不要	必须有
目的地址	每个分组都有完整的目的地址	仅在建立连接阶段使用，之后每个分组使用长度较短的虚电路号
路由选择	每个分组独立地进行路由选择和转发	属于同一条虚电路的分组按照同一路由转发
分组顺序	不保证分组的有序到达	保证分组的有序到达
可靠性	不保证可靠通信，可靠性由用户主机来保证	可靠性由网络保证
对网络故障的适应性	出故障的结点丢失分组，其他分组路径选择发生变化，可正常传输	所有经过故障结点的虚电路均不能正常工作
差错处理和流量控制	由用户主机进行流量控制，不保证数据报的可靠性	可由分组交换网负责，也可由用户主机负责

- 三网合一：电信网络、有线电视网络、计算机网络。
- 网络边缘的端系统之间的通讯方式：客户-服务器方式(C/S方式)、对等连接方式(P2P)。

计算机网络的性能

- 速率
  - 码元传输速率(波特Baud)：与进制数无关，只与码元长度有关。
  - 信息传输速率(bit/s)
- 带宽
  - 信号具有的频带宽度(单位：Hz)。
  - 网络中某通道传送数据的能力(单位：bit/s)。
- 吞吐量。
- 时延 = 发送时延 + 传播时延 + 处理时延 + 排队时延
  - 发送时延 = $\frac{\text{数据帧长度}(bit)}{\text{发送速率}(bit/s)}$
  - 传播时延 = $\frac{\text{信道长度}(m)}{\text{电磁波在信道上的传播速率}(m/s)}$
  - 处理时延

    - 排队时延
- 时延带宽积 = 传播时延 * 带宽
- 往返时间RTT
- 利用率。根据排队论的理论，当某信道的利用率增大时，该信道引起的时延也就迅速增加。
    - D_0：网络空闲时的时延。D：网络当前的时延。U：利用率。有以下关系。
    - $D = \frac{D_0}{1-U}$
- 非性能特征：费用、质量、标准化、可靠性、可扩展性和可升级性、易于管理和维护。

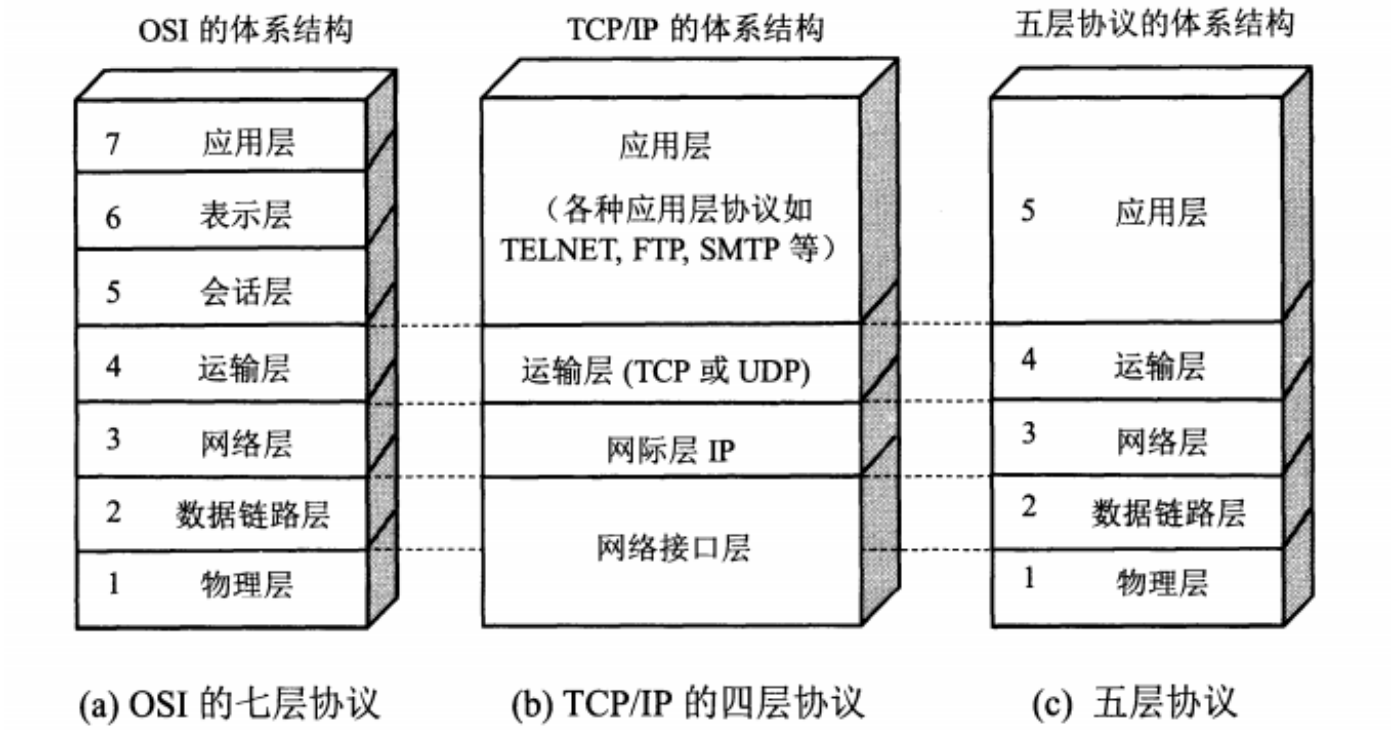
计算机网络体系结构

- 协议的要素：语法、语义、同步。
- 实体：表示任何可发送或接收信息的硬件或软件进程。
- 协议是控制两个对等实体(或多个实体)进行通信的规则的集合。

参考模型

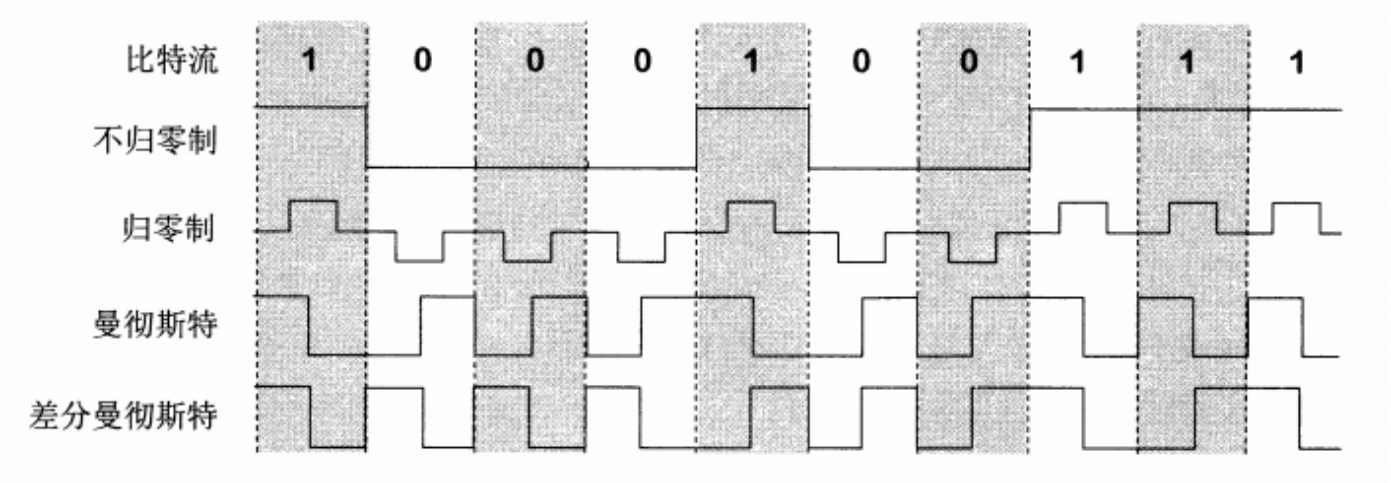
- **物理层**：建立物理连接，数据以**比特流**传输。
- **数据链路层**：IP数据报封装成**帧**，建立逻辑连接，进行硬件地址寻址，如何传输，差错校验等功能。
- **网络层**：进行逻辑地址寻址，实现不同网络之间的路径选择，分组传输，路由器的选择与转发。
- **传输层**：端到端传输数据(如TCP，UDP)，流量控制，差错校验等基本功能。
- **会话层**：建立、管理、维护、终止 应用程序进程之间会话。负责在网络中的两节点之间建立、维持和终止通信。
- **表示层**：管理数据的加密和压缩。
- **应用层**：各种应用程序，通过应用进程间的交互来完成特定的网络应用。

TCP/IP是一个四层的体系结构，包含应用层、运输层、网际层、网络接口层。



2 物理层

- 物理层特性：
    1. 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、固定和锁定装置等。
    2. 电气特性：指明在接口电缆的各条线上出现的电压范围。
    3. 功能特性：指明某条线上出现某一电平的电压意义。
    4. 过程特性：指明对于不同功能的各种可能事件的出现顺序。
- 数据通信系统：源系统(源点、发送器)、传输系统、目的系统(接收器、终点)
- 信号：模拟信号(连续信号)、数字信号(离散信号)。
- 信息交互方式：单工、半双工、全双工通信。
- 串行传输的速度比并行传输的速度要慢得多，但费用低。并行传输适用距离短，而串行传输适用远距离传输。
- 常用编码方式：不归零制、归零制、曼彻斯特编码(具有自同步能力，不需要时钟信号)、差分曼彻斯特编码。



- 基本的带通调制方法：调幅AM、调频FM、调相PM、正交振幅调制QAM。
- 在任何信道中，码元传输速率是有上限的，传输速率超过此上限，就会出现严重的码元间串扰的问题，使接收端对码元的判决(即识别)成为不可能。
- 信噪比(信号的平均功率 / 噪声) = $10\log_{10} S/N$ (dB)
- 香农公式

在 1948 年，信息论的创始人香农(Shannon)推导出了著名的**香农公式**。香农公式指出：**信道的极限信息传输速率** C 是

$$C = W \log_2(1+S/N) \quad \text{(bit/s)} \qquad (2\text{-}2)$$

式中，W 为信道的带宽（以 Hz 为单位）；S 为信道内所传信号的平均功率；N 为信道内部的高斯噪声功率。香农公式的推导可在通信原理教科书中找到。这里只给出其结果。

香农公式表明，**信道的带宽或信道中的信噪比越大，信息的极限传输速率就越高**。香农公式指出了信息传输速率的上限。香农公式的意义在于：只要信息传输速率低于信道的极限信息传输速率，就一定存在某种办法来实现无差错的传输。不过，香农没有告诉我们具体的实现方法。这要由研究通信的专家去寻找。

- 奈氏准则

奈氏准则：在理想低通（无噪声，带宽受限）条件下，为了避免码间串扰，极限码元传输速率为 2W Baud，W是信道带宽，单位是Hz。

只有在这两个公式
这带宽才用**Hz**！！

为了混淆大家，再求一步极限数据率吧~

传输速率慢：

理想低通信道下的极限数据传输率=$2W\log_2 V$ (b/s)

几种码元/码元的离散电平数目

带宽(Hz)

1.在任何信道中，码元传输的速率是有**上限**的。若传输速率超过此上限，就会出现严重的码间串扰问题，使接收端对码元的完全正确识别成为不可能。

2.信道的**频带越宽**（即能通过的信号高频分量越多），就可以用更高的速率进行码元的有效传输。

3.奈氏准则给出了码元传输速率的限制，但并没有对信息传输速率给出限制。

4.由于码元的传输速率受奈氏准则的制约，所以要提高数据的传输速率，就必须设法使每个码元能携带更多个比特的信息量，这就需要采用多元制的调制方法。

- 物理层的传输媒体：
  - 导引型：
    - 双绞线
    - 同轴电缆
    - 光缆
    - 架空明线
  - 非导引型：
    - 无线电波
    - 微波(地面微波接力通信、卫星通信)
    - 红外线、激光
- 信道复用：频分复用FDM、时分复用TDM、波分复用WDM(光的频分复用)、码分复用CMD(码分多址CDMA)、统计时分复用STDM。

> 例题：在一个CDMA移动通信系统中，假设A、B、C站分配的地址码分别为(-1-1-1+1+1-1+1+1)、(-1-1+1-1+1+1+1-1)、(-1+1-1+1+1+1-1-1)。某一时刻A发送数据位0，B发送数据位1，C未发送数据，则接收C站信息的接收者收到的信号是__。A. (0 0+2-2 0+2 0-2)
> 每一个站分配的码片序列不仅必须各不相同，并且还必须互相正交。当发送1时，计算结果为1；发送0，计算结果为-1；没有发送，计算结果为0。

- 数字传输系统：调制解调器。
- 模拟信号转为数字信号：采样、量化、编码。
- 宽带接入技术：ADSL技术、光纤同轴混合网(HFC网)、FTTx技术(Fiber To The)。
- 中继器：对信号进行再生和还原。

3 数据链路层

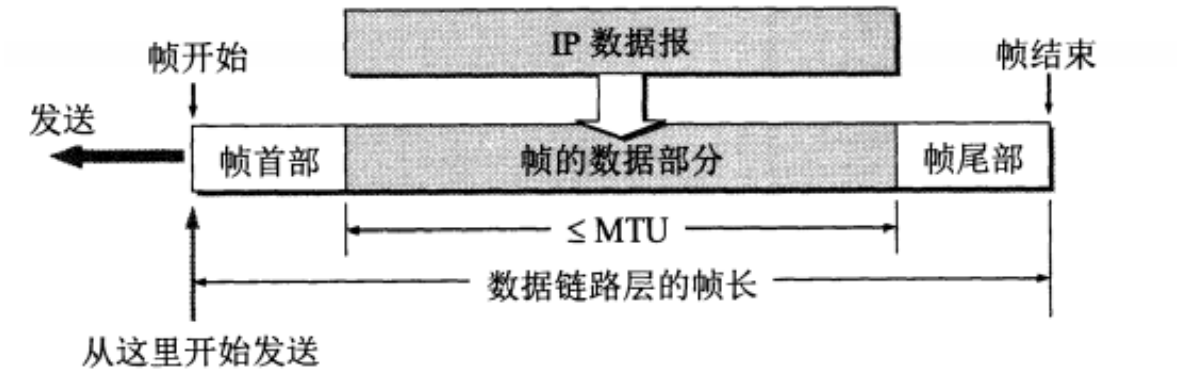
数据链路层是在物理层提供服务的基础上**向网络层提供服务**，其最基本的服务是将源自网络层来的数据可靠地传输到相邻结点的目标机网络层。其主要作用是**加强物理层传输原始比特流的功能**，将物理层可能出错的物理连接改造成逻辑上无差错的数据链路，使之对网络层表现为一条无差错的链路。

功能：

1. 为网络层提供服务。无确认无连接服务，有确认无连接服务，有确认面向连接服务。(有连接就一定要确认，

不存在无确认的面向连接的服务)
2. 链路管理。即连接的建立、维持、释放(用于面向连接的服务)。
3. 组帧。封装成帧。
4. 流量控制。限制发送方。
5. 差错控制(帧错/位错)

封装成帧



MTU，最大传输单元：数据部分长度上限。

透明传输

避免消息符号与帧定界符号相混淆。不管所传数据是什么样的比特组合，都应当能够在链路上传送。因此，链路层就看不见有什么妨碍数据传输的东西。

组帧的四种方法：

字符计数法

在帧头部使用一个计数字段来标明帧内字符数。目的结点数据链路层收到字节计数值时就知道后面跟随的字节数(计数字段也占一个字节)，从而确定帧结束的位置。
问题：如果计数字段出错，就失去帧边界划分的依据，收发双方失去同步，造成恶劣后果。

字符(节)填充法(异步)

使用特定字符来界定一帧的开始(SOH)和结束(EOT)。
当传送的帧是文本文件组成时(键盘上的字符都是ASCII码，与SOH和EOT不会冲突)，不管键盘上什么字符都可以放在帧里传过去，即透明传输。当传送的帧由非ASCII码文本文件组成时(二进制代码执行文件或图像等)，就要采用字符填充法实现透明传输，即在每个可能造成误会的字符前填充转义字符，接收端接受到时再删去转义字符回复原数据。

零比特填充法(同步)

该方法允许数据帧包含任意个数的比特。它使用01111110来标识一帧的开始与结束，为了不使比特流中的01111110被误判为首尾标识，发送方的数据链路层在数据位中遇到5个连续的1时，会自动在后插入一个0，接收方接收到时在进行恢复。该方法很容易有硬件实现，性能优于字符填充法。

违规编码法

曼彻斯特编码分别用高低和低高代表0和1.那么我们就可以用高高和低低这两个“违规”的编码来界定帧的起始和终止。

差错检测

产生的差错主要分为**位错(1、0出错)**和**帧错(帧丢失、重复、失序)**。对于通信质量好或是有线传输线路，链路层为网络层提供**无确认无连接服**务，检错纠错交给高层来解决；对于通信质量差的无线传输链路，链路层提供**有确认无连接服务**和**有确认面向连接服务**。

差错控制可分为检错编码和纠错编码。

常见检错编码：奇偶校验码、CRC循环冗余码。
常见纠错编码：海明码(可以发现双比特错，纠正单比特错)。

奇偶校验码

只能检测出奇数个错误。

奇校验：原始码流+校验位 总共有奇数个1
偶校验：原始码流+校验位 总共有偶数个1

循环冗余码

CRC是一种检错方法，FCS帧检验序列是添加在数据后面的冗余码。

> 例如，待发送的数据M = 101001(k = 6)
> 假设除数P = 1101(n = 3)(常用多项式表示).首先在末尾加r(阶数)个0，再进行模2运算(同0异1)，余数作为FCS。
> 在接收端令DATA.FCS / P，余数R = 0，则帧没有错，接收。

海明码

发现双比特错，纠正单比特错。
海明距离(码距)：两个码字的对应比特取值不同的比特数称为这两个码字的海明距离。一个有效编码集中，任意两个码字的海明距离的最小值称为该编码集的海明距离。

对于d位检验(纠正)，需要码距：

检验	纠正
d+1	2d+1

数据有m位，校验码r位(2^r种取值)。

故$2^r \geq m + r + 1$。($m + r$位错误，和没有错误的情况)①从而确定校验码的位数。

对于$D = 1100$，②确定校验码的位置，放到2^n的位置。

序号	7	6	5	4	3	2	1
值	1	1	0	x_4	0	x_2	x_1

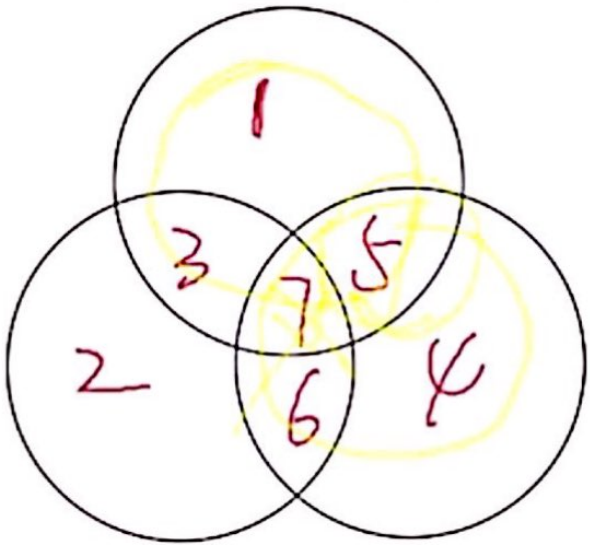
③求校验码的值。

-	-	-	-	$1**$	-	$*1*$	$**1$
二进制	111	110	101	100	011	010	001
序号	7	6	5	4	3	2	1
值	1	1	0	x_4	0	x_2	x_1

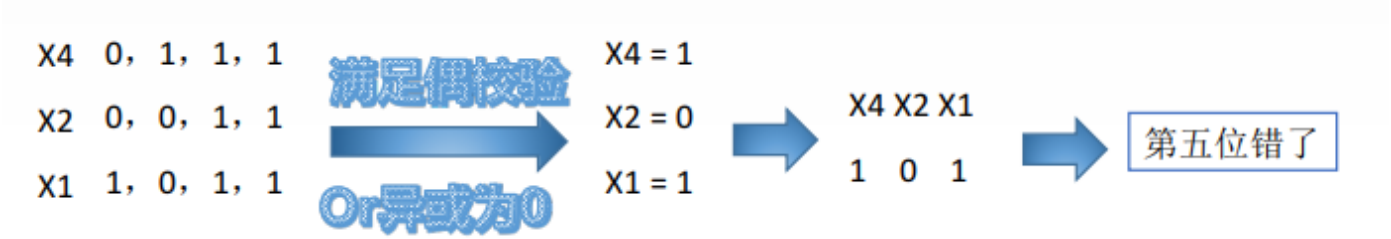
4号校验码负责4，5，6，7的校验。
2号校验码负责2，3，6，7的校验。
1号校验码负责1，3，5，7的校验。

采用偶校验：
$x_4 = 0$
$x_2 = 0$
$x_1 = 1$

④检错和纠错。
方法一：
找到不满足奇/偶校验的分组取交集，并与符合校验的分组取差集。



方法二：

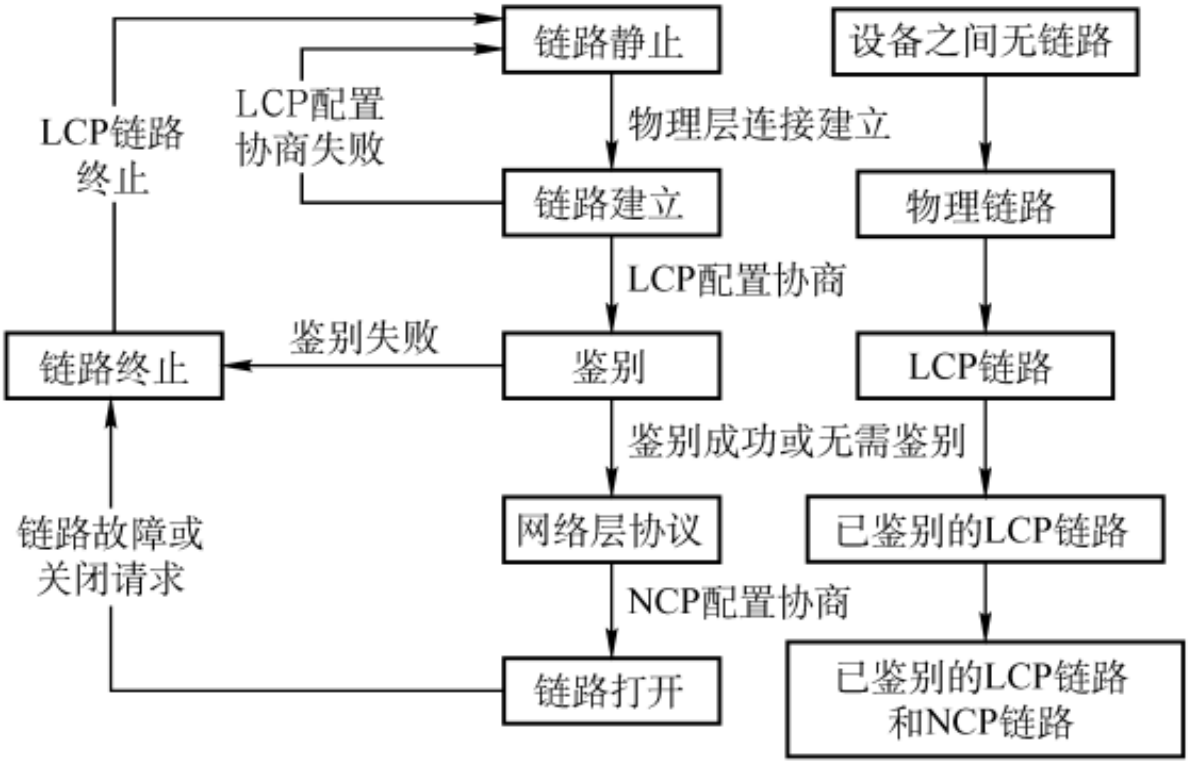


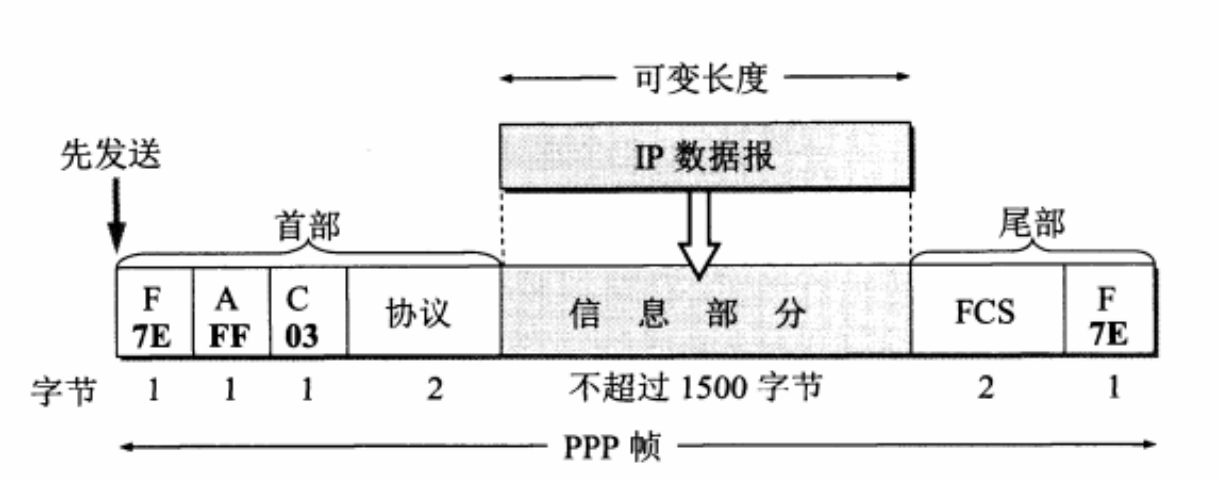
点对点链路

PPP协议

点对点协议PPP(Point-to-Point Protocol)是目前使用最广泛的数据链路层协议，用户使用拨号电话接入因特网时一般都使用PPP协议。

PPP协议必须对每一种类型的点对点链路设置最大传输单元MTU(最大接收单元)的标准值。(默认值1500字节，数据部分的最大长度)





字节填充：
在信息字段中，进行以下替换。

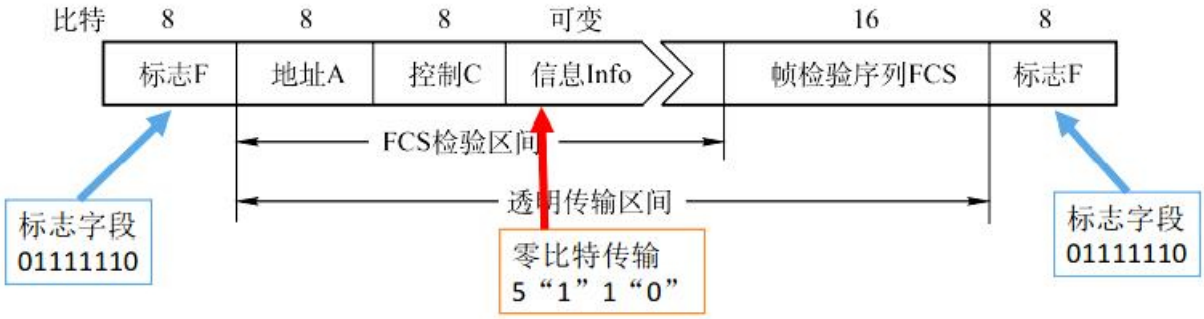
1. 7E -> 7D5E
2. 7D -> 7D5D
3. 对于ASCII码的控制字符(数值小于0x20)，例如0x03 -> 0x7D,0x23。

零比特填充(同步传输时)：
发送端，只要发现连续5个连续1，则立即填入一个0。接收端进行删除。

HDLC协议

高级数据链路控制(High-Level Data Link Control或简称HDLC)，是一个在同步网上传输数据、面向比特的数据链路层协议，它是由国际标准化组织(ISO)根据IBM公司的SDLC(SynchronousData Link Control)协议扩展开发而成的。

主站、从站、复合站。

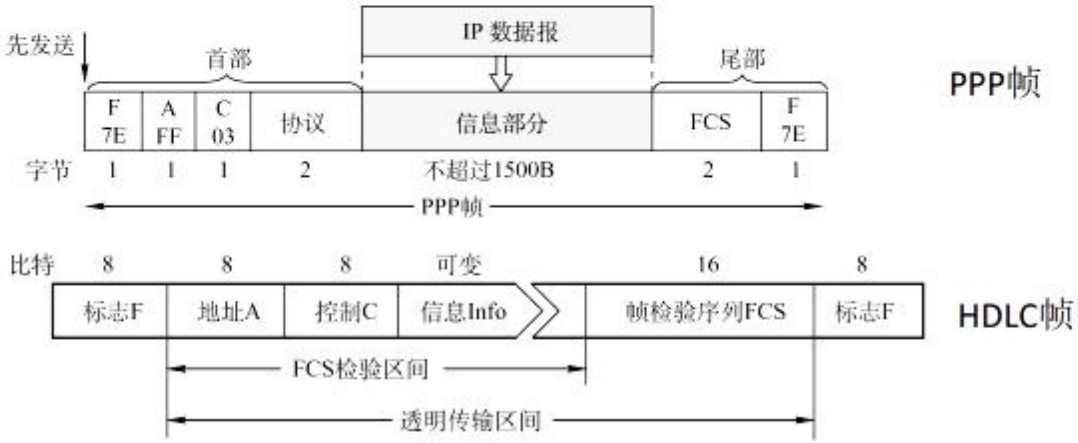


1）信息帧（I）第1位为0，用来传输数据信息，或使用捎带技术对数据进行确认；
2）监督帧（S）10，用于流量控制和差错控制，执行对信息帧的确认、请求重发和请求暂停发送等功能
3）无编号帧（U）11，用于提供对链路的建立、拆除等多种控制功能。

PPP和HDLC对比

HDLC、PPP只支持**全双工**链路。
都可以实现透明传输。
都可以实现差错检测，但不纠正差错。

PPP协议	面向字节	2B协议字段	无序号和确认机制	不可靠
HDLC协议	面向比特	没有	有编号和确认机制	可靠



广播链路

介质访问控制

采取一定的措施，使得两对节点之间的通信不会发生互相干扰的情况。

静态划分信道

信道划分介质访问控制。把时域和频域资源合理分配给网络上的设备。

多路复用技术：把多个信号组合在一条物理信道上进行传输。使得多个计算机或终端设备共享信道资源，提高信道利用率。

1. 频分多路复用FDM
2. 时分多路复用TDM
3. 统计时分复用STDM
   通过集中器(STDM下的MUX)为各个数据终端或线路动态分配时间片(大量数据要发送的数据终端占有较多的时间片，数据量小的数据终端少占用时间片，没有数据的数据终端不分配时间片)。这时，为了区分哪一个时间片是哪一个数据终端或线路的，必须在时间片的数据前加上该数据终端或线路的标识(源线路号地址)。由于一个用户的数据并不按照固定的时间间隔发送，所以称为“异步”。
4. 波分多路复用WDM
5. 码分多路复用CDM
   码分多址(CDMA)是码分复用的一种方式。
   两个向量到公共信道上，线性相加。
   数据分离：合并的数据和源站规格化内积。

动态分配信道

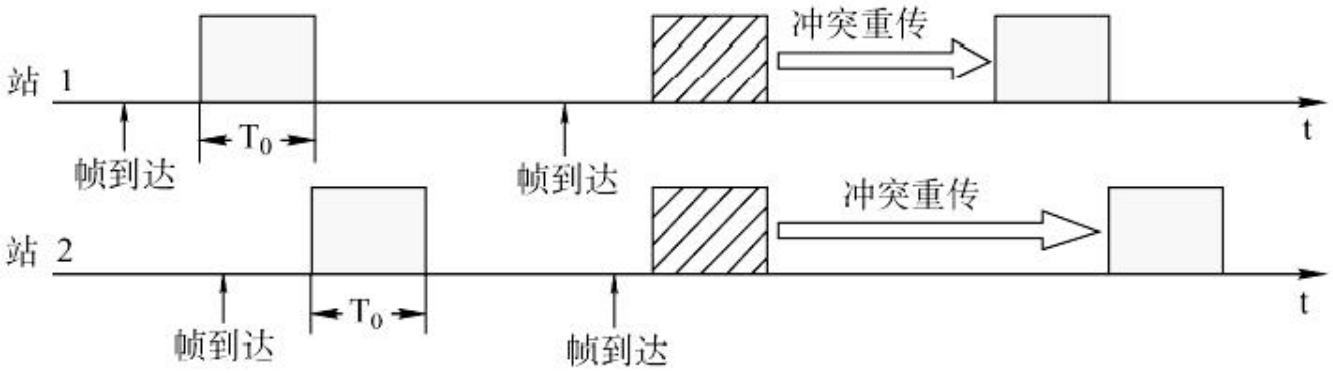
随机访问介质访问控制(冲突)

用户根据意愿随机发送信息，发送信息时可独占信道带宽。

ALOHA协议

纯ALOHA协议：不监听信道，不按时间槽发送，随机重发。
如果发送冲突，接收方检测出差错不予确认，发送方在一定时间内收不到就判断发生冲突。

时隙ALOHA协议：把时间分成若干个相同的时间片，所有用户在时间片开始时刻同步接入网络信道，若发生冲突，则必须等到下一个时间片开始时刻再发送。



1. 纯ALOHA比时隙ALOHA吞吐量更低，效率更低。
2. 纯ALOHA想发就发，时隙ALOHA只有在时间片段开始时才能发。

CSMA协议

载波监听多路访问协议CSMA(carrier sense multiple access)
CS：载波侦听/监听，每一个站在发送数据之前要检测一下总线上是否有其他计算机在发送数据。
MA：多点接入，表示许多计算机以多点接入的方式连接在一根总线上。
发送帧之前，监听信道。(当几个站同时在总线上发送数据时，总线上信号电压摆动值将增大(互相叠加))。

1-坚持CSMA
如果一个主机要发送消息，那么它先监听信道。
空闲则直接传输，不必等待。
忙则一直监听，直到空闲马上传输。
如果有冲突(一段时间内未收到肯定回复)，则等待一个随机长的时间再监听，重复上述过程。
优点：只要媒体空闲，站点就马上发送，避免了媒体利用率的损失。
缺点：假如有两个或两个以上的站点有数据要发送，冲突就不可避免。

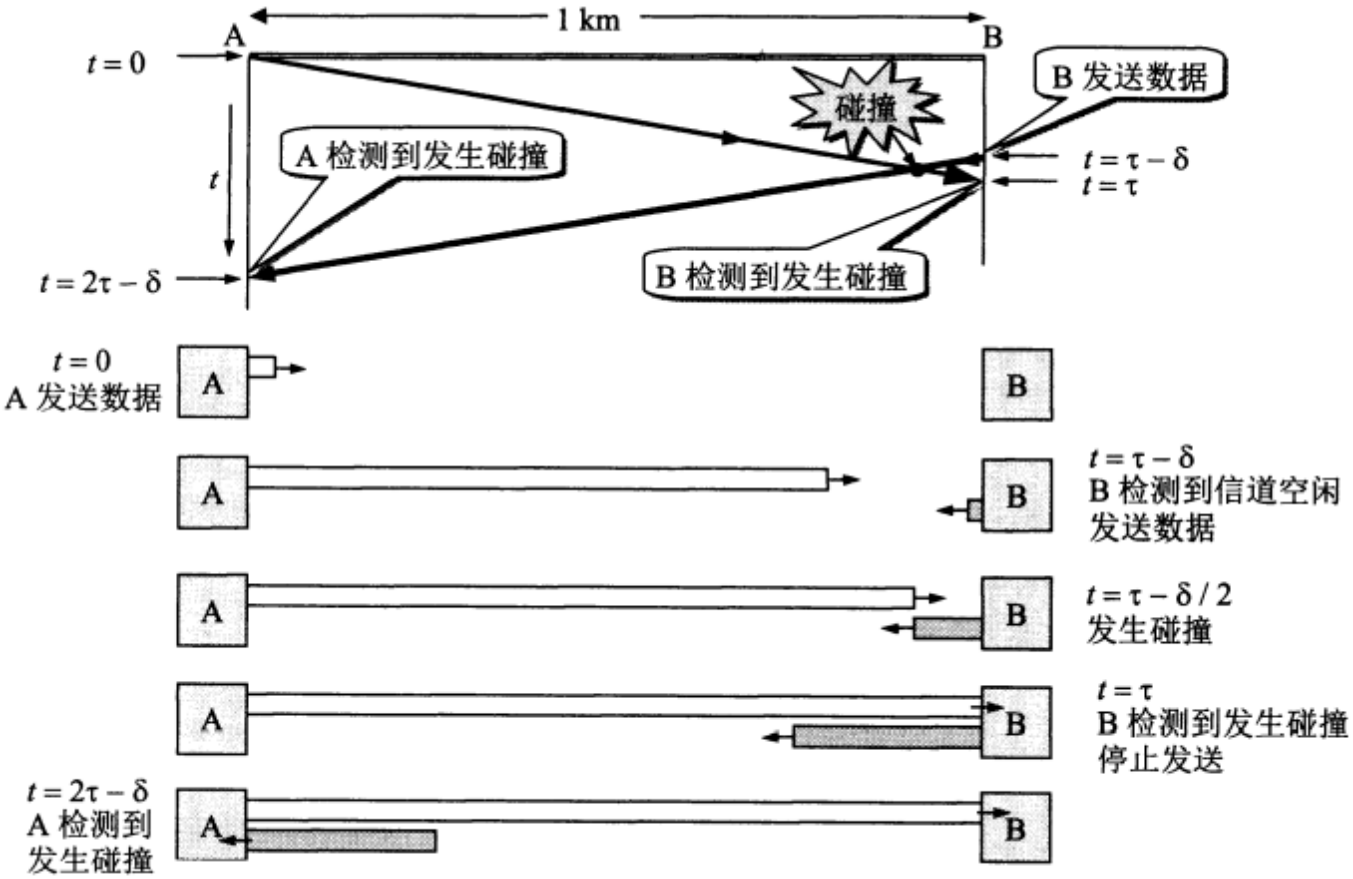
非坚持CSMA
如果一个主机要发送消息，那么它先监听信道。
空闲则直接传输，不必等待。
忙则等待一个随机的时间之后再进行监听。
优点：采用随机的重发延迟时间可以减少冲突发生的可能性。
缺点：可能存在大家都在延迟等待过程中，使得媒体仍可能处于空闲状态，媒体使用率降低。

p-坚持CSMA
如果一个主机要发送消息，那么它先监听信道。
空闲则以p概率直接传输，不必等待;概率1-p等待到下一个时间槽再传输。
忙则持续监听直到信道空闲再以p概率发送。
若冲突则等到下一个时间槽开始再监听并重复上述过程。
优点：既能像非坚持算法那样减少冲突，又能像1-坚持算法那样减少媒体空闲时间的这种方案。

CSMA/CD协议

载波监听多点接入/碰撞检测CSMA/CD (carrier sense multiple access with ollision detection)
边发送边监听。

传播时延对载波监听的影响：



单程端到端的传播时延为τ。
以太网端到端往返时间2τ称为争用期(碰撞窗口)。若经过争用期后没有检测到碰撞，才能肯定这次发送不会发生碰撞。

截断二进制指数规避

(1) 协议规定了基本退避时间为争用期 2τ，具体的**争用期时间是** 51.2 μs。对于 10 Mbit/s 以太网，在争用期内可发送 512 bit，即 64 字节。也可以说争用期是 512 **比特时间**。1 比特时间就是发送 1 比特所需的时间。所以这种时间单位与数据率密切相关。为了方便，也可以**直接使用比特作为争用期的单位**。争用期是 512 bit，即争用期是发送 512 bit 所需的时间。

(2) 从离散的整数集合[0,1,…, (2^k-1)]中随机取出一个数，记为 r。重传应推后的时间就是 r 倍的争用期。上面的参数 k 按下面的公式(3-1)计算：

$$k = \text{Min}[\text{重传次数}, 10] \tag{3-1}$$

可见当重传次数不超过 10 时，参数 k 等于重传次数；但当重传次数超过 10 时，k 就不再增大而一直等于 10。

(3) 当重传达 16 次仍不能成功时（这表明同时打算发送数据的站太多，以致连续发生冲突），则丢弃该帧，并向高层报告。

最小帧长

当发送很短的帧，发送完毕之前没有检测到碰撞，目的站将丢弃，但发送端不知道发生碰撞，因而不会重传。为了避免上述情况，规定最短帧长。

$$\frac{\text{帧长}(bit)}{\text{数据传输速率}} \geq 2\tau$$

$$\text{最小帧长} = \text{总线传输时延} \times \text{数据传输速率} \times 2$$

以太网规定的最短帧长64字节，即512bit。如果要发送的数据非常少，那么必须加入一些填充字节。
凡长度小于64字节的帧都是由于冲突而异常中止的无效帧。

强化碰撞

当发送数据的站一旦发现发生了碰撞时，除了立即停止发送数据外，还要再继续发送32比特或48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。

流程

以太网还规定了**帧间最小间隔**为 9.6 μs，相当于 96 比特时间。这样做是为了使刚刚收到数据帧的站的接收缓存来得及清理，做好接收下一帧的准备。

根据以上所讨论的，可以把 CSMA/CD 协议的要点归纳如下：

(1) 准备发送：适配器从网络层获得一个分组，加上以太网的首部和尾部（见后面的 3.4.3 节），组成以太网帧，放入适配器的缓存中。但在发送之前，必须**先检测信道**。

(2) 检测信道：若检测到信道忙，则应不停地检测，一直等待信道转为空闲。若检测到信道空闲，并在 96 比特时间内信道保持空闲（保证了帧间最小间隔），就发送这个帧。

(3) 在发送过程中仍不停地检测信道，即网络适配器要**边发送边监听**。这里只有两种可能性：

①发送成功：在争用期内一直未检测到碰撞。这个帧肯定能够发送成功。发送完毕后，其他什么也不做。然后回到(1)。

②发送失败：在争用期内检测到碰撞。这时立即停止发送数据，并按规定发送人为干扰信号。适配器接着就执行指数退避算法，等待 r 倍 512 比特时间后，返回到步骤(2)，继续检测信道。但若重传达 16 次仍不能成功，则停止重传而向上报错。

以太网每发送完一帧，一定要把已发送的帧暂时保留一下。如果在争用期内检测出发生了碰撞，那么还要在推迟一段时间后再把这个暂时保留的帧重传一次。

CSMA/CA协议

载波监听多点接入/碰撞避免CSMA/CA (carrier sense multiple access with collision avoidance)

用于无线局域网，由于无法做到360°全面检测碰撞。
隐蔽站：当A和C都检测不到信号，认为信道空闲时， 同时向终端B发送数据帧，就会导致冲突。

发送数据前，先检测信道是否空闲。
空闲则发出**RTS(request to send)**，RTS包括发射端的地址、接收端的地址、下一份数据将持续发送的时间等信息，信道忙则等待。
接收端收到RTS后，将响应**CTS(clear to send)**。
发送端收到CTS后，开始发送数据帧(同时**预约信道**：发送方告知其他站点自己要传多久数据)。
接收端收到数据帧后，将用CRC来检验数据是否正确，正确则响应**ACK帧**。
发送方收到ACK就可以进行下一个数据帧的发送，若没有则一直重传至规定重发次数为止(采用二进制指数退避算法来确定随机的推迟时间)。

1. 预约信道
2. ACK帧
3. RTS/CTS帧(可选，为解决隐蔽站问题)

CSMA/CD和CSMA/CA比较

相同点：

CSMA/CD与CSMA/CA机制都从属于CSMA的思路，其核心是**先听再说**。换言之，两个在接入信道之前都须要进行监听。当发现信道空闲后，才能进行接入。

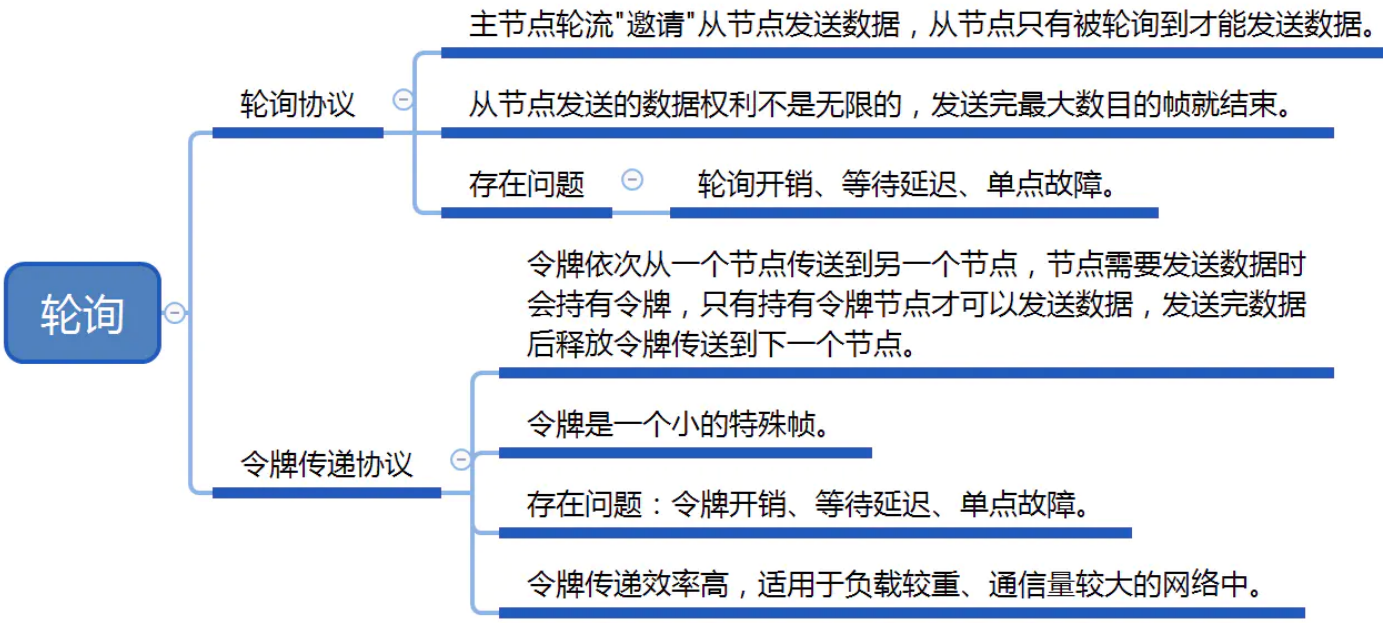
不同点：

1. **传输介质不同**：CSMA/CD 用于总线式以太网【有线】，而CSMA/CA用于无线局域网【无线】。

2.**载波检测方式不同**：因**传输介质不同**，CSMA/CD与CSMA/CA的**检测方式也不同**。CSMA/CD通过电缆中电压的变化来检测，当数据发生碰撞时，电缆中的电压就会随着发生变化；而CSMA/CA采用能量检测（ED）、载波检测（CS）和能量载波混合检测三种检测信道空闲的方式。

3.**CSMA/CD检测冲突，CSMA/CA避免冲突**，二者出现冲突后都会进行有上限的重传。

轮询访问介质访问控制



轮询协议

轮询协议要求节点中有一个被指定为主节点，其余节点是从属节点。
主节点以循环的方式轮询每一个从属节点，“邀请”从属节点发送数据(实际上是向从属节点发送一个报文，告诉从属节点可以发送帧以及可以传输帧的最大数量)，只有被主节点“邀请”的从节点可以发送数据，没有被“邀请”的节点不能发送，只能等待被轮询。

轮询协议存在的问题：轮询开销、等待延迟、单点故障。

令牌

这种协议没有主节点，令牌是一个小的特殊的帧。
令牌会依次从一个节点传送到另一个节点，当一个节点收到令牌时，如果该节点需要发送数据，它才会持有令牌，它发送了最大数目的帧数后，再把令牌释放转发给下一个节点。如果节点无数据可发，它不会持有令牌，直接转发给下一个节点即可。
令牌传输协议同样可以保证同一时刻只有一个节点独占信道。同样，每个节点也不是能无限制的持有令牌，都只能在一定的时间内获得发送数据的权利。

令牌传递协议的问题：令牌开销、等待延迟、单点故障。

应用于令牌环网(物理星型拓扑，逻辑环形拓扑)。
常用于负载较重、通信量较大的网络。

流量控制

数据链路层的流量控制是点对点的，传输层的流量控制是端到端的。

数据链路层：接收方收不下就不回复确认。
传输层：接收端给发送端一个窗口公告。

信道利用率：发送方在一个发送周期内，有效地发送数据所需要的的时间占整个发送周期的比率。

$$信道利用率 = (L/C)/T$$

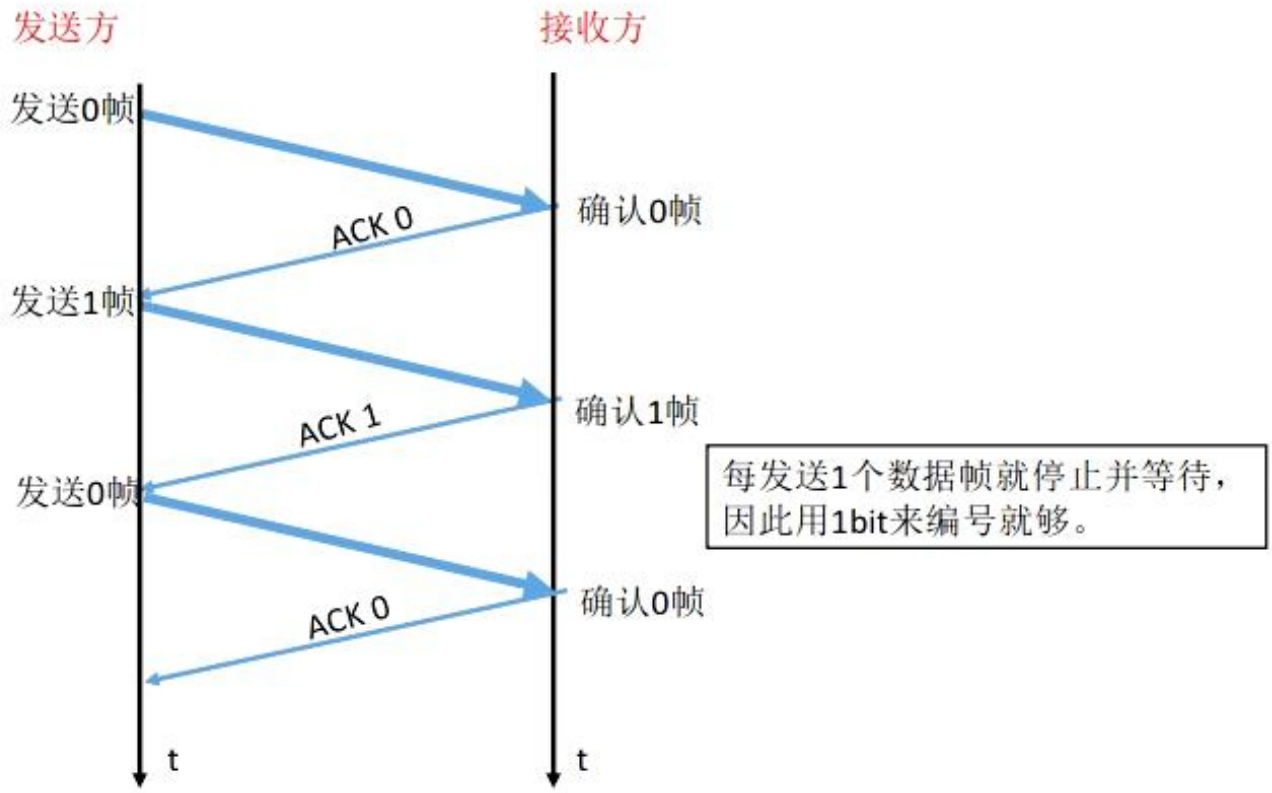
> T内发送L比特数据；C发送方数据传输率；T发送周期，从开始发送数据到收到第一个确认帧为止。

$$信道吞吐率 = 信道利用率 \times 发送发的发送速率$$

停止-等待协议

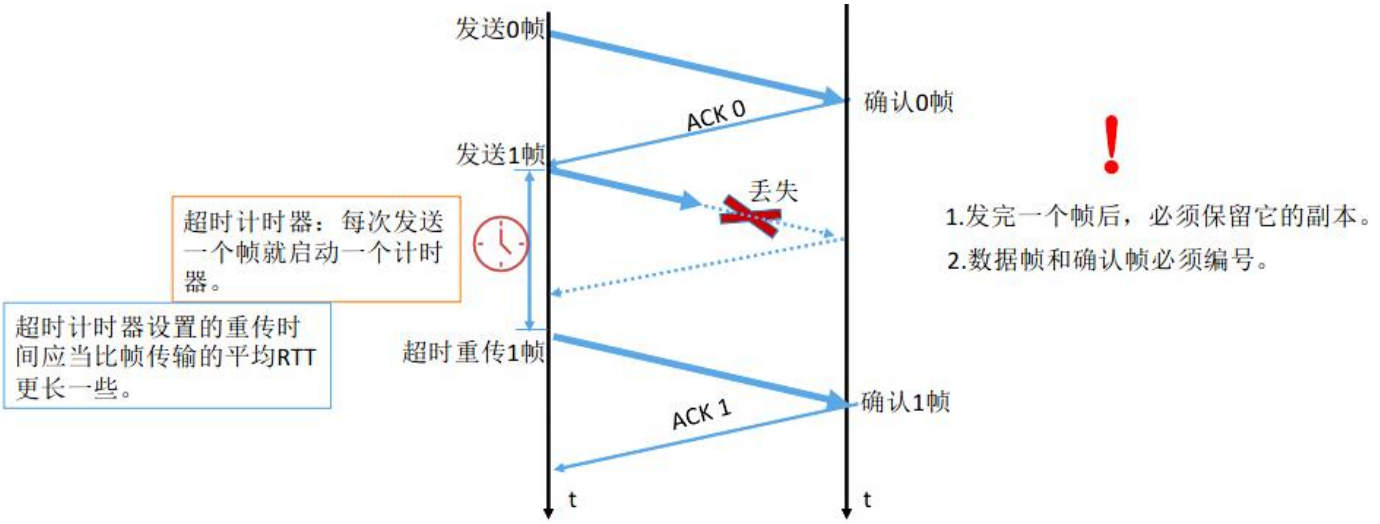
每发送完一个帧就停止发送，等待对方的确认，在收到确认后再发送下一个帧。
发送窗口大小=1，接收窗口大小=1。

无差错情况

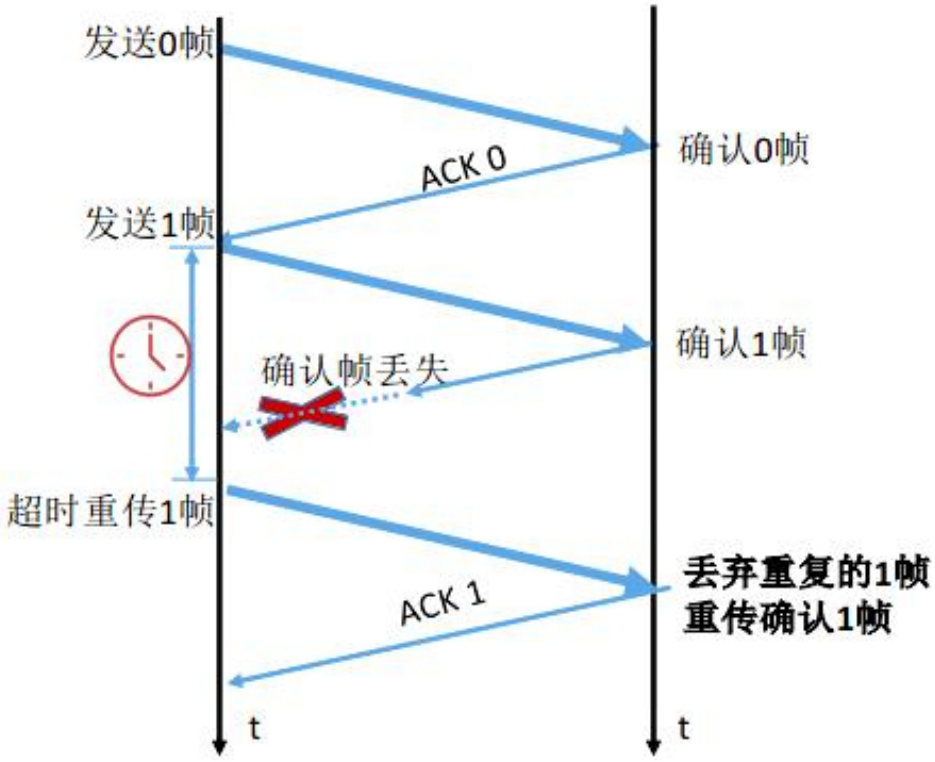


有差错情况

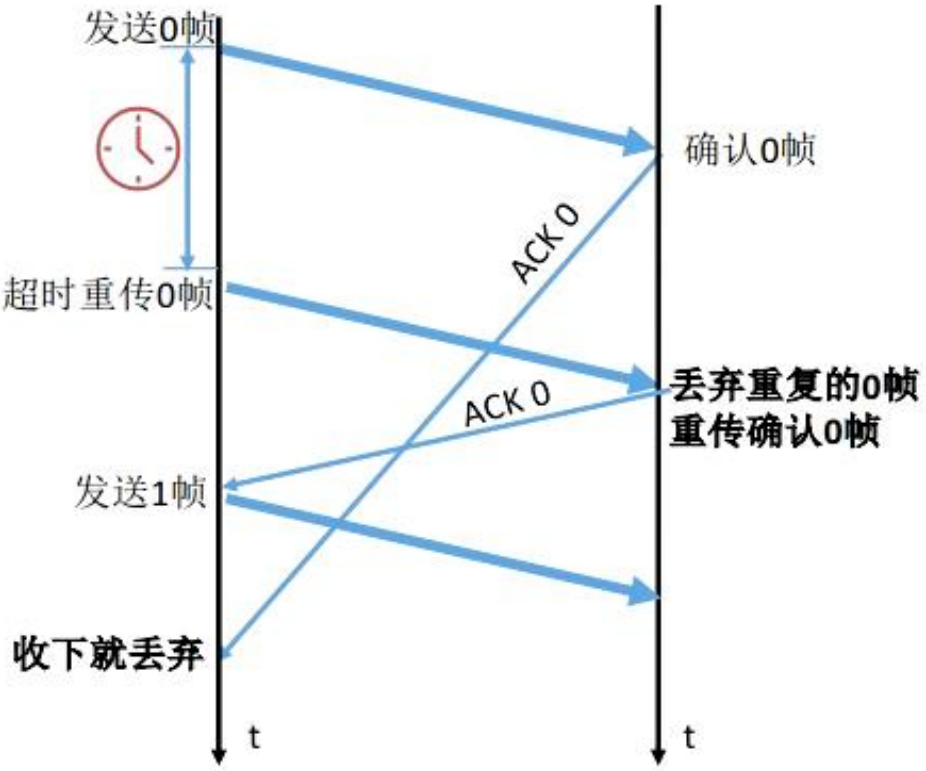
帧丢失或帧出错


发送0帧
确认0帧
ACK 0
发送1帧
丢失
超时计时器：每次发送一个帧就启动一个计时器。
超时计时器设置的重传时间应当比帧传输的平均RTT更长一些。
超时重传1帧
确认1帧
ACK 1
t
t
1.发完一个帧后，必须保留它的副本。
2.数据帧和确认帧必须编号。

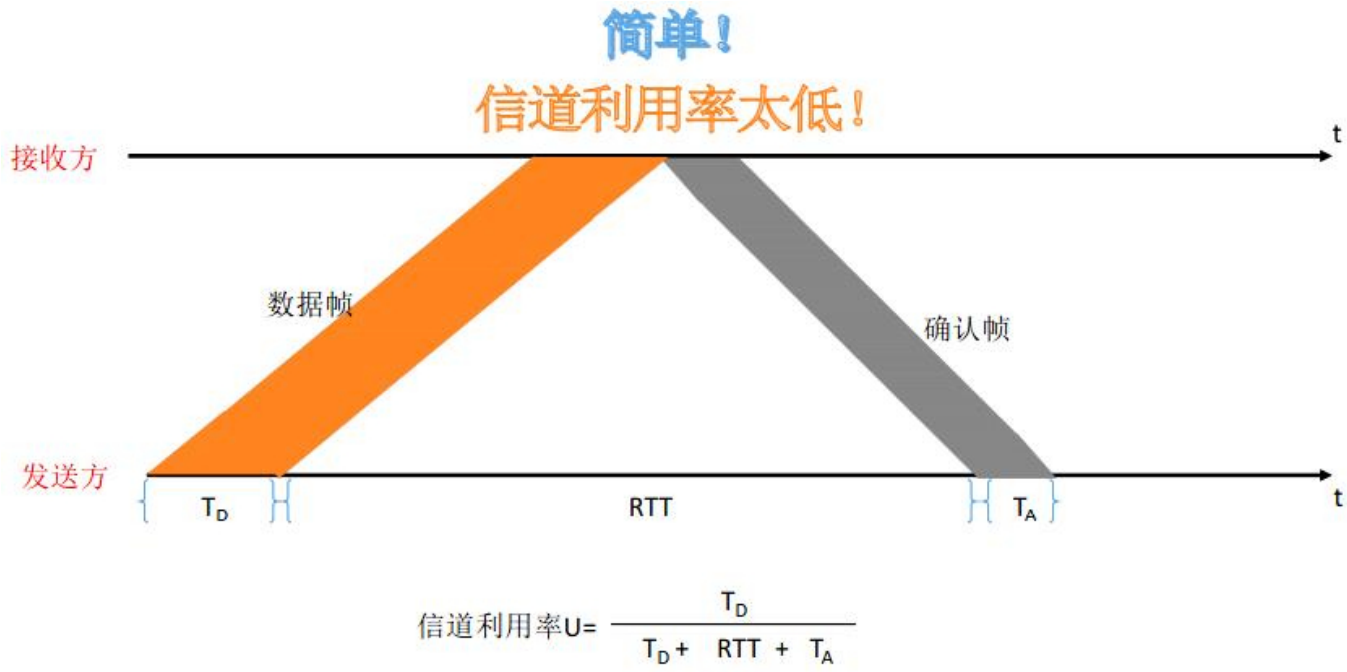
ACK丢失


发送0帧
确认0帧
ACK 0
发送1帧
确认1帧
确认帧丢失
超时重传1帧
丢弃重复的1帧
重传确认1帧
ACK 1
t
t

ACK迟到



性能分析



滑动窗口协议

若采用n个比特对帧编号，那么发送窗口的尺寸W_T应满足：$1 \leq W_T \leq (2^{n-1})$。因为发送窗口尺寸过大，就会使接收方无法区别新帧和旧帧。

后退N帧协议(GBN)

发送窗口大小>1，接收窗口大小=1。

发送(接收)窗口：发送方(接收方)维持一组连续的允许发送(接收)的帧的序号。

分为：发完确认的、已经发送但等待确认的、还能发送的、还不能发送的。

对于发送方：

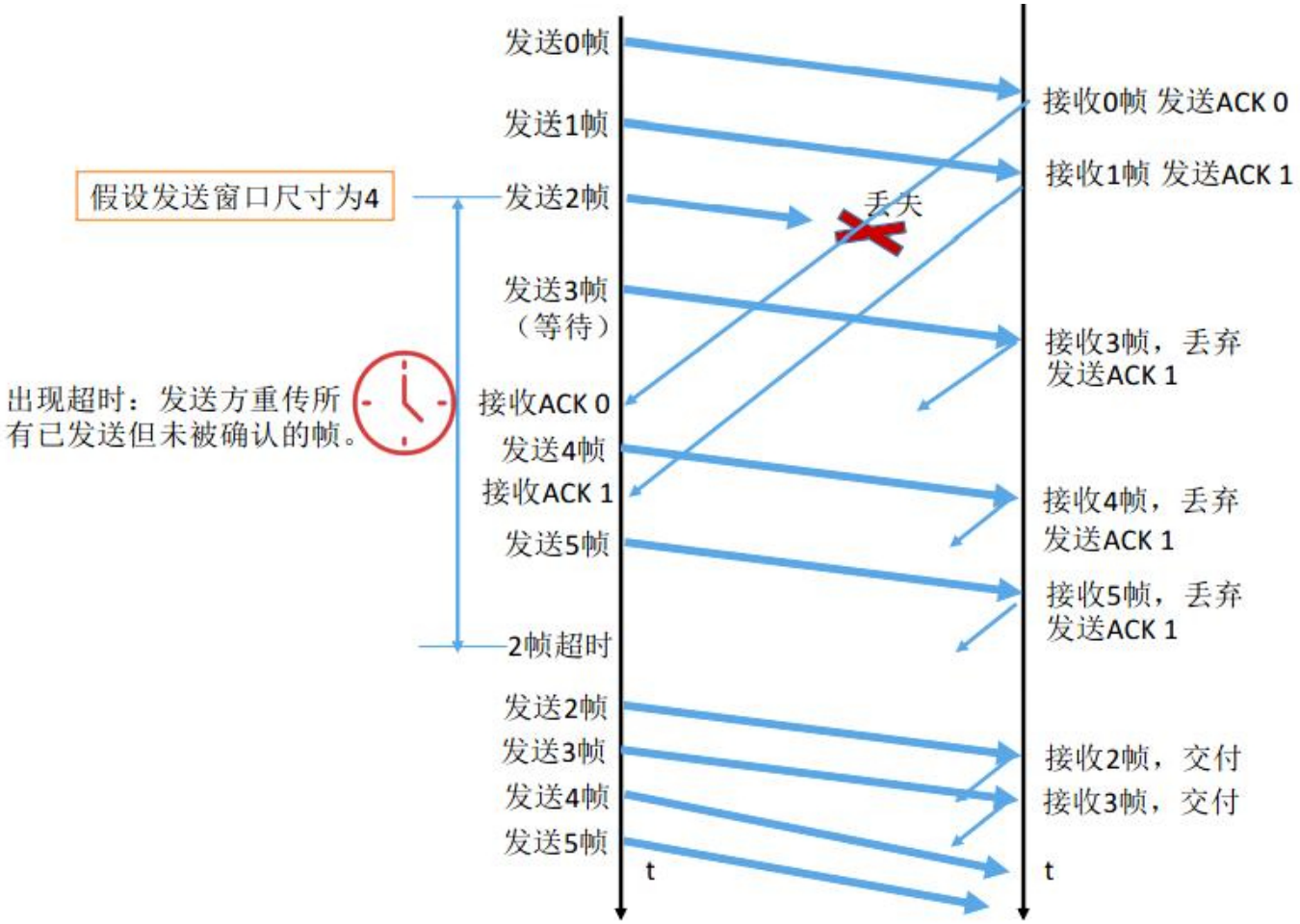
1. 上层的调用：上层要发送数据时，发送方先检查发送窗口是否已满。如果未满，则产生一个帧并将其发送；如果窗口已满，发送方只需将数据返回给上层，暗示上层窗口已满。上层等一会再发。(实际实现中，发送方可以缓存这些数据，窗口不满时再发送帧)
2. 在GBN协议中，对n号帧的确认采用**累积确认**的方式，表明接收方已经收到n号帧和它之前的全部帧。
3. 超时事件：协议的名称为后退N帧，来源于出现丢失和时延过长帧时发送方的行为。就像在停止等待协议中一样，定时器将再次用于恢复数据帧或确认帧的丢失。如果出现超时，发送方将重传所有已发送但未被确认的帧。

对于接收方：

1. 如果正确收到n号帧，并且按序，那么接收方为n帧发送一个ACK，并将该帧中的数据部分交付给上层。
2. 其余情况都丢弃帧，并为最近按序接收的帧重新发送ACK。接收方无需缓存任何失序帧，只需维护expectedseqnum(下一个按序接收的帧序号)。

接收方只按顺序接收帧，不按序的帧丢弃。

优点：连续发送数据帧而提高了信道利用率。
缺点：在重传时必须把原来已经正确传送的数据帧重传，使得传输效率降低。进一步改善为选择重传协议。



选择重传协议(SR)

发送窗口大小>1，接收窗口大小>1。

对于发送方：

1. 如果收到ACK，加入该帧序号在窗口内，则SR发送方将那个被确认的帧标记为已接收。如果该帧序号是窗口的下界(最左边第一个窗口对应的序号)，则窗口向前移动到具有最小序号的未确认帧处。如果窗口移动了并且有序号在窗口内的未发送帧，则发送这些帧。
2. 超时事件：每个帧都有自己的定时器，一个超时事件发生后只重传一个帧。

对于接收方：
SR接收方将确认一个正确接收的帧而不管其是否按序。失序的帧将被缓存，并返回给发送方一个该帧的确认帧(收谁确认谁)，直到所有帧(即序号更小的帧)皆被收到为止，这时才可以将一批帧按序交付给上层，然后向前移动滑动窗口。

发送窗口最好等于接收窗口。

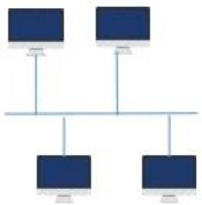
局域网

局域网是指在某一区域内由多台计算机互联成的计算机组，使用广播信道。
要素：网络拓扑、传输介质、介质访问控制方法。

网络拓扑

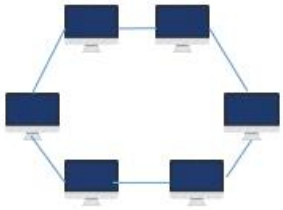
星型拓扑

中心节点是控制中心，任意两个节点间的通信最多只需**两步**，传输速度快，并且网络构形简单、建网容易、便于控制和管理。但这种网络系统，网络可靠性低，网络共享能力差，有单点故障问题。



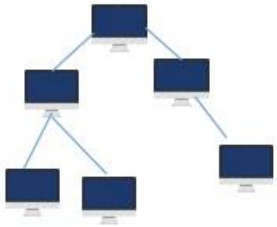
总线型拓扑

网络可靠性高、网络节点间响应速度快、共享资源能力强、设备投入量少、成本低、安装使用方便，当某个工作站节点出现故障时，对整个网络系统影响小。



环形拓扑

系统中通信设备和线路比较节省。有**单点故障**问题；由于环路是封闭的，所以不便于扩充，系统响应延时长，且信息传输效率相对较低。



树型拓扑

易于拓展，易于隔离故障，也容易有**单点故障**。

传输介质

有线(双绞线、同轴电缆、光纤)和无线(电磁波)。

介质访问控制方法

1. CSMA/CD常用于总线型局域网，也用于树型网络。
2. 令牌总线常用于总线型局域网，也用于树型网。它是把总线型或树型网络中的各个工作站按一定顺序如按接口地址大小排列形成一个逻辑环。只有令牌持有者才能控制总线，才有发送信息的权力。
3. 令牌环用于环形局域网，如令牌环网。

局域网分类

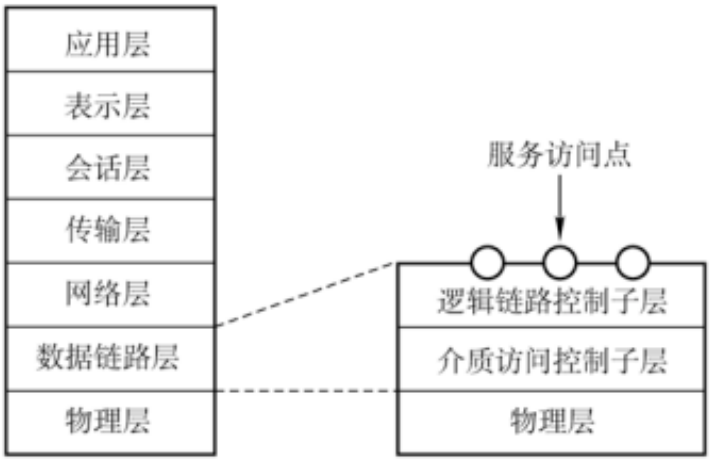
1. **以太网** 以太网是应用最为广泛的局域网，包括标准以太网(10Mbps)、快速以太网(100Mbps)、千兆以太网(1000 Mbps)和10G以太网，它们都符合IEEE802.3系列标准规范。逻辑拓扑总线型，物理拓扑是星型或拓展星型。使用CSMA/CD。
2. **令牌环网** 物理上采用了星形拓扑结构，逻辑上是环形拓扑结构。
3. **FDDI网(Flber Distriburted Data Interface)** 物理上采用了双环拓扑结构，逻辑上是环形拓扑结构。
4. **ATM网(Asynchronous Transfer Mode)** 较新型的单元交换技术，使用53字节固定长度的单元进行交换。
5. **无线局域网(Wireless Local Area Network, WLAN)** 采用IEEE 802.11标准。

IEEE 802

- IEEE 802.3： 以太网介质访问控制协议 (CSMA/CD)及物理层技术规范。
- IEEE 802.5： 令牌环网(Token-Ring)的介质访问控制协议及物理层技术规范。
- IEEE 802.8： 光纤技术咨询组，提供有关光纤联网的技术咨询。
- IEEE 802.11： 无线局域网(WLAN)的介质访问控制协议及物理层技术规范。

MAC子层和LLC子层

IEEE802委员会就把局域网的数据链路层拆成两个子层，即逻辑链路控制LLC (Logical Link Control)子层和媒体接入控制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，不管采用何种传输媒体和MAC子层的局域网对LLC子层来说都是透明的。



- LLC负责识别网络层协议，然后对它们进行封装。LLC报头告诉数据链路层一旦帧被接收到时，应当对数据包做何处理。为网络层提供服务：无确认无连接、面向连接、带确认无连接、高速传送。
- MAC子层的主要功能包括数据帧的封装/卸装，帧的寻址和识别，帧的接收与发送，链路的管理，帧的差错控制等。MAC子层的存在屏蔽了不同物理链路种类的差异性。

以太网

以太网(英语：Ethernet)是一种计算机局域网技术。

无连接：发送方和接收方之间无“握手过程”。
不可靠：不对发送方的数据帧编号，接收方不向发送方进行确认，差错帧直接丢弃，差错纠正由高层负责。

以太网只实现无差错接收，不实现可靠传输。

使用集线器的以太网在逻辑上仍是一个总线网，各站共享逻辑上的总线，使用的还是CSMA/CD协议。

以太网拓扑：逻辑上总线型，物理上星型。

在局域网中，硬件地址又称为物理地址，或MAC地址。(实际上是标识符)
MAC地址：每个适配器有一个全球唯一的48位二进制地址，前24位代表厂家(由IEEE规定)，后24位厂家自己指定。常用6个十六进制数表示，如02-60-8c-e4-b1-21。

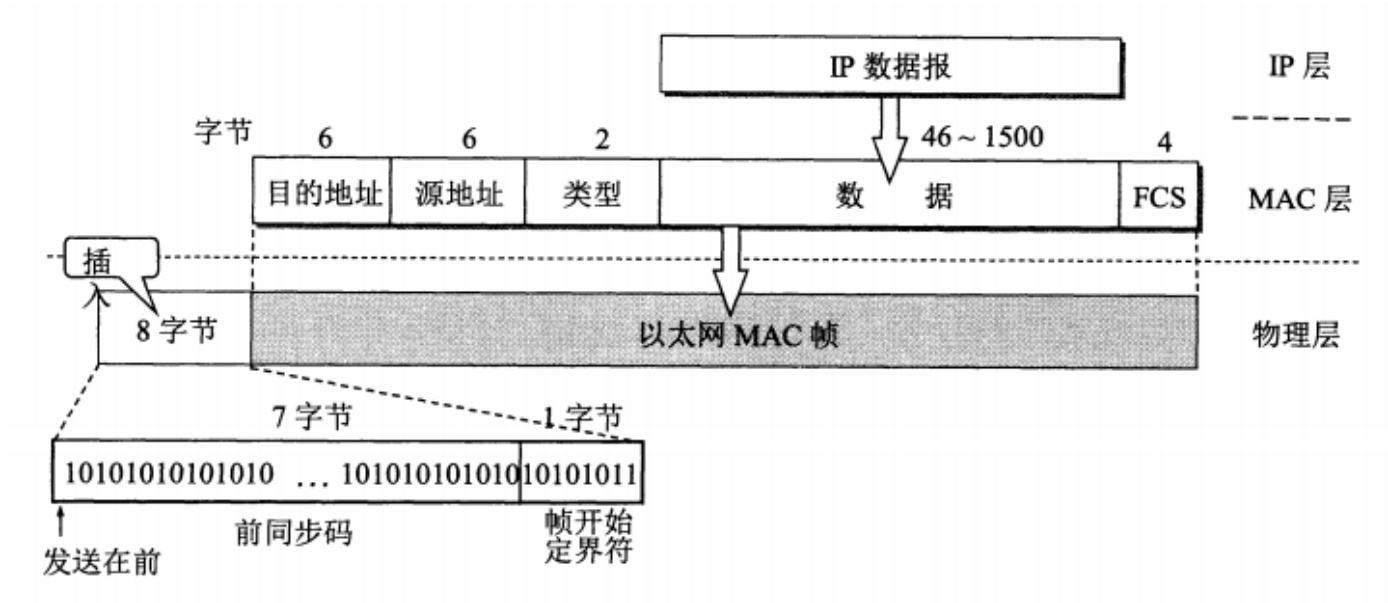
10BASE-T以太网

10BASE-T是传送基带信号的双绞线以太网，T表示采用双绞线，现10BASE-T采用的是无屏蔽双绞线(UTP)，传输速率是10Mb/s。

- 物理上采用星型拓扑，逻辑上总线型，每段双绞线最长为100m。
- 采用曼彻斯特编码。
- 采用CSMA/CD介质访问控制。

以太网MAC帧

常用的以太网MAC帧格式有两种标准，一种是DIX Ethernet V2标准(即以太网V2标准)，另一种是IEEE的802.3标准。使用得最多的以太网V2标准。



使用曼彻斯特编码，电压不变化时，即结束。

在以太网上传送数据时是以帧为单位传送的。以太网在传送帧时，各帧之间还必须有-定的间隙。因此，接收端只要找到帧开始定界符，其后面的连续到达的比特流就都属于同一个MAC帧。可见以太网不需要使用帧结束定界符，也不需要使用字节插入来保证透明传输。

链路层设备

- 冲突域是一种物理分段，指连接到同一导线上所有工作站的集合、同一物理网段上所有节点的集合或是以太网上竞争同一带宽节点的集合。
- 在同一个冲突域中的每一个节点都能收到所有被发送的帧。简单的说就是同一时间内只能有一台设备发送信息的范围。
- 广播域：网络中能接收任一设备发出的广播帧的所有设备的集合。简单的说如果站点发出一个广播信号，所有能接收到这个信号的设备范围称为一个广播域。

-	隔离冲突域	隔离广播域
物理层设备(中继器、集线器)	×	×
链路层设备(网桥、交换机)	√	×
网络层设备(路由器)	√	√

网桥

网桥根据MAC帧的目的地址对帧进行转发和过滤。当网桥收到一个帧时，并不向所有接口转发此帧，而是先检查此帧的目的MAC地址，然后再确定将该帧转发到哪一个接口，或者是把它丢弃(即过滤)。

连接不同网段，不同冲突域。

优点：

1. 过滤通信量，增大吞吐量。
2. 扩大物理传输范围。
3. 提高可靠性。
4. 可互连不同物理层、不同MAC子层和不同速率的以太网。

透明网桥

“透明"指以太网上的站点并不知道所发送的帧将经过哪几个网桥，是一种即插即用设备(自学习)。

源路由网桥

在发送帧时，把详细的最佳路由信息(路由最少/时间最短)放在帧的首部中。
方法：源站以广播方式向欲通信的目的站发送一个发现帧。

多接口网桥：交换机

独占传输媒体带宽，交换机所连接的设备可以独占带宽。

直通式交换机

检查目的地址后直接转发。延迟小，可靠性低，无法支持具有不同速率的端口的交换。

存储转发式交换机

将帧放入高速缓存，并检查否正确，正确则转发，错误则丢弃。延迟大，可靠性高，可以支持具有不同速率的端口的交换。

VLAN

传统局域网的局限性：缺乏流量隔离；管理用户不变，必须代表物理布线；路由器成本较高。

虚拟局域网(VLAN)：是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。每个VLAN是一个单独的广播域/不同的子网。

交换机上生成的各VLAN互不相通，若想实现通信，需借助：路由器、三层交换机。

VLAN的实现：IEEE 802.1Q帧：

VLAN标记的前两个字节表明是IEEE 802.1Q帧，接下来4位没用，后面12位是VLAN标识符VID，唯一表示了该以太网帧属于哪个VLAN。

- VID的取值范围为0～4095，但0和4095都不用来表示VLAN，因此**用于表示VLAN的有效VID取值范围为1～4094**。
- IEEE 802.1Q帧是由交换机来处理的，而不是由用户主机来处理的。（即**主机和交换机之间只交换普通的以太网帧**）

4 网络层

SDN

路由器功能：

- 转发：达到路由器输入链路之一的数据报如何转发到该路由器的输出链路之一。时间短，通常硬件解决。
- 路由选择：控制数据报沿着从源主机到目的主机的端到端路径中路由器之间的路由方式。时间长，通常软件解决。

数据平面

数据平面对于数据处理过程中各种具体处理转发过程。

数据平面执行的主要功能是**根据转发表进行转发**，这是路由器的本地动作。

控制平面

控制平面用于控制和管理网络协议的运行，比如OSPF协议、RIP协议、BGP协议。

传统方法/每路由器法

路由选择算法运行在每台路由器中，并且在每台路由器中都包含转发和路由选择两种功能。

具体方法：在一台路由器中的路由选择算法与其他路由器中的路由选择算法通信（通过交换路由选择报文），计算出路由表和转发表。

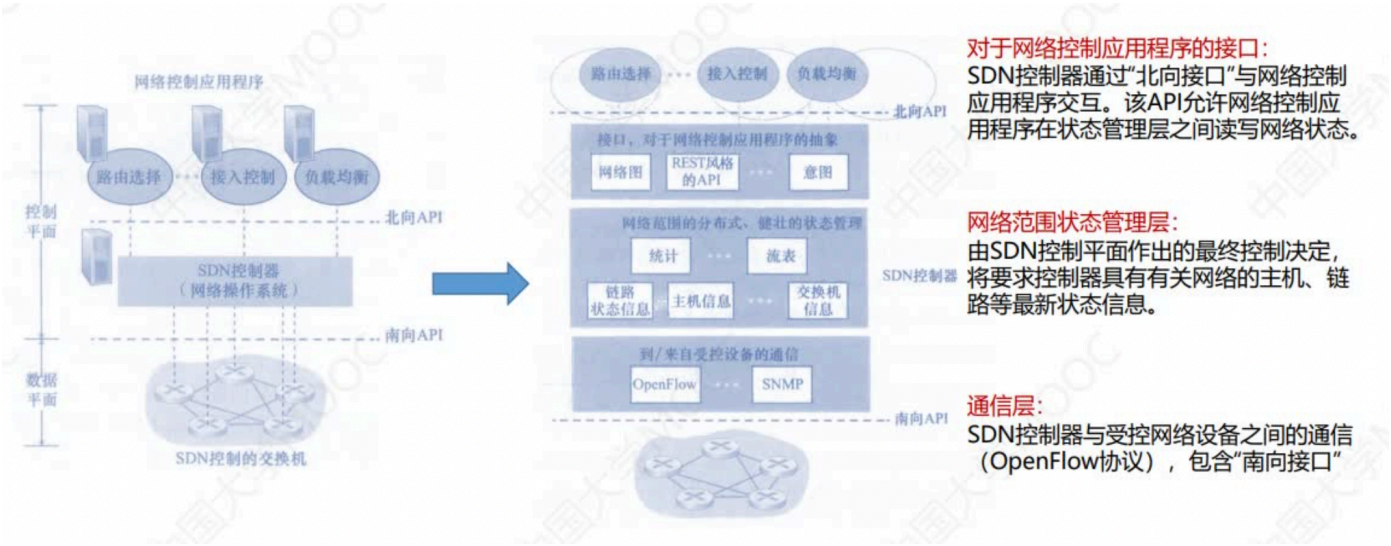
路由选择处理器执行控制平面功能。在传统的路由器中，它执行路由选择协议，维护路由选择表于关联链路状态信息，并为该路由器计算转发表。

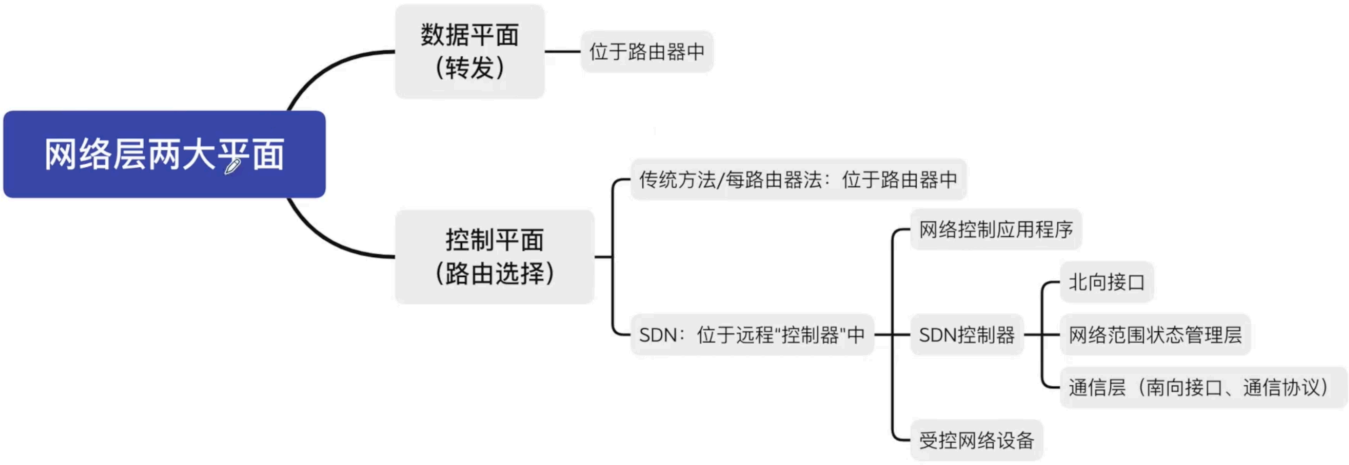
SDN方法

Software-Defined Networking
控制平面从路由器物理上分离。路由器仅实现转发，**远程控制器**计算和分发转发表以供每台路由器所使用。

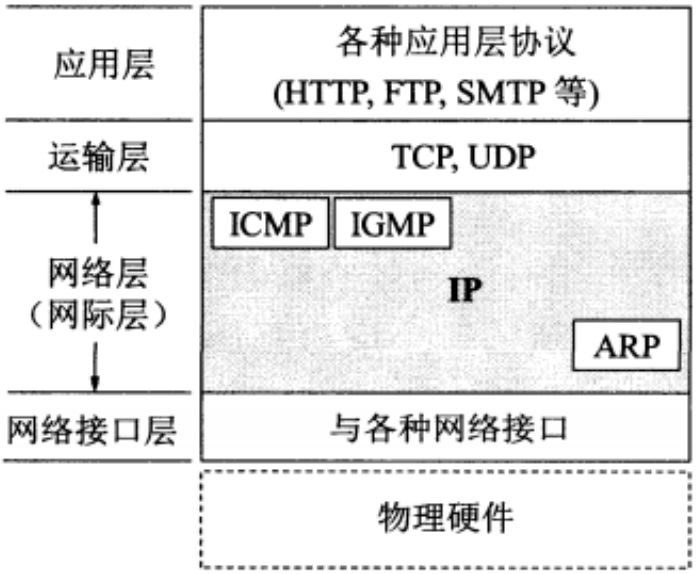
具体方法：路由器通过交换包含转发表和其他路由选择信息的报文与远程控制器通信。因为计算转发并与路由器交互的控制器是用软件实现的，所以网络是"软件定义的"。

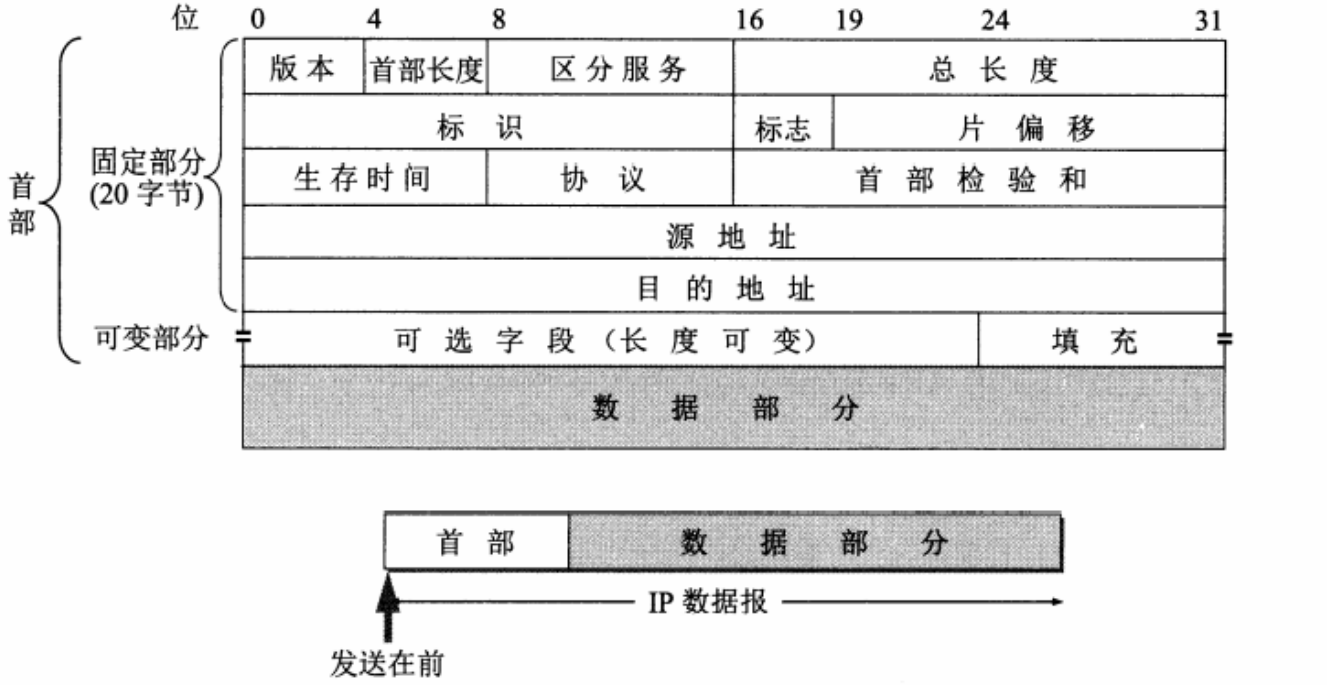
在SDN路由器中，**路由选择处理器负责于远程控制器通信**，目的是接收远程控制器计算的转发表项。




网络层两大平面
数据平面
(转发)
位于路由器中
控制平面
(路由选择)
传统方法/每路由器法：位于路由器中
SDN：位于远程"控制器"中
网络控制应用程序
SDN控制器
北向接口
网络范围状态管理层
通信层（南向接口、通信协议）
受控网络设备

IP数据报

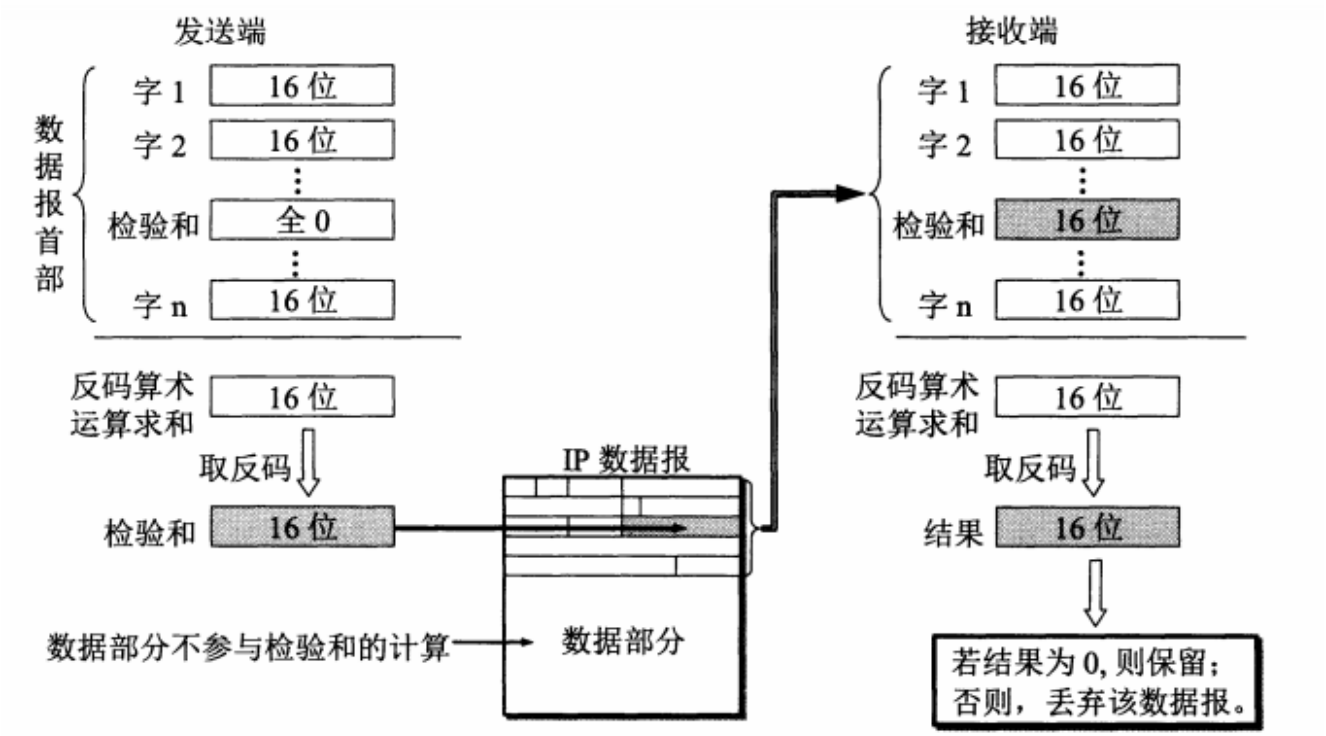

应用层
各种应用层协议
(HTTP, FTP, SMTP 等)
运输层
TCP, UDP
网络层
（网际层）
ICMP
IGMP
IP
ARP
网络接口层
与各种网络接口
物理硬件



1. 版本：IPv4和IPv6。
2. 首部长度：单位4B，最小为5。
3. 区分服务：实际上未被使用。
4. 总长度：单位1B，首部+数据。
5. ...
6. TTL(Time To Live)：跳数限制。
7. 协议：

协议名	ICMP	IGMP	IP②	TCP	EGP	IGP	UDP	IPv6	ESP	OSPF
协议字段值	1	2	4	6	8	9	17	41	50	89

8. 首部校验和：只检验首部。二进制反码求和：从低位到高位，0+0=0，0+1=1，1+1=0但需要进位，若最高位相加后产生进位，则最后结果+1.

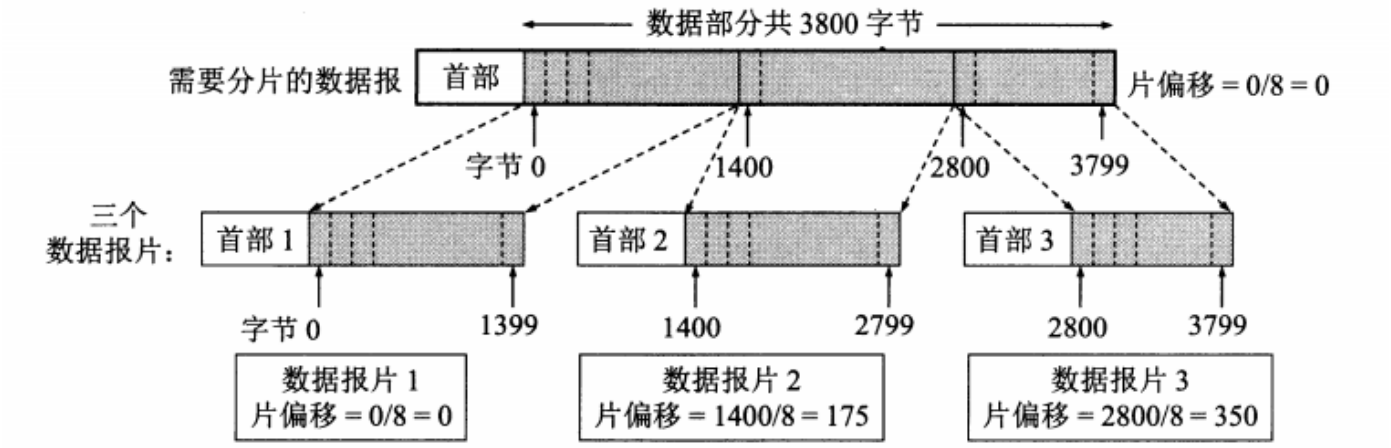


9. 源地址/目的地址：32位。
10. 可选字段：0~32B。用来支持排错、测量、安全等措施。
11. 填充：全0，把首部补成4B的整数倍。

IP数据报分片

最大传输单元MTU：链路层数据帧可封装的数据的上限。
以太网的MTU是1500字节。

1. 标识：同一数据报的分片使用同一标识。
2. 标志：前两位有意义。
    - 最低位：MF(More Fragment) 等于1时后面还有分片；等于0代表最后一片。
    - 中间位：DF(Don't Fragment) 等于0时才允许分片。
3. 片偏移：单位8B。指出较长的分组在分片后，某片在原分组中的相对位置。除了最后一个分片，每个分片的长度一定是8B的整数倍。



	总长度	标识	MF	DF	片偏移
原始数据报	3820	12345	0	0	0
数据报片 1	1420	12345	1	0	0
数据报片 2	1420	12345	1	0	175
数据报片 3	1020	12345	0	0	350

单位总结
总长度单位1B
片偏移单位8B
首部长度单位4B。

IPv4

IP地址：全世界唯一的32位/4字节标识符，标识路由器主机的接口。

IP地址 ::= < 网络号 >, < 主机号 >

点分十进制：IPv4中用四个字节表示一个IP地址，每个字节按照十进制表示为0~255。

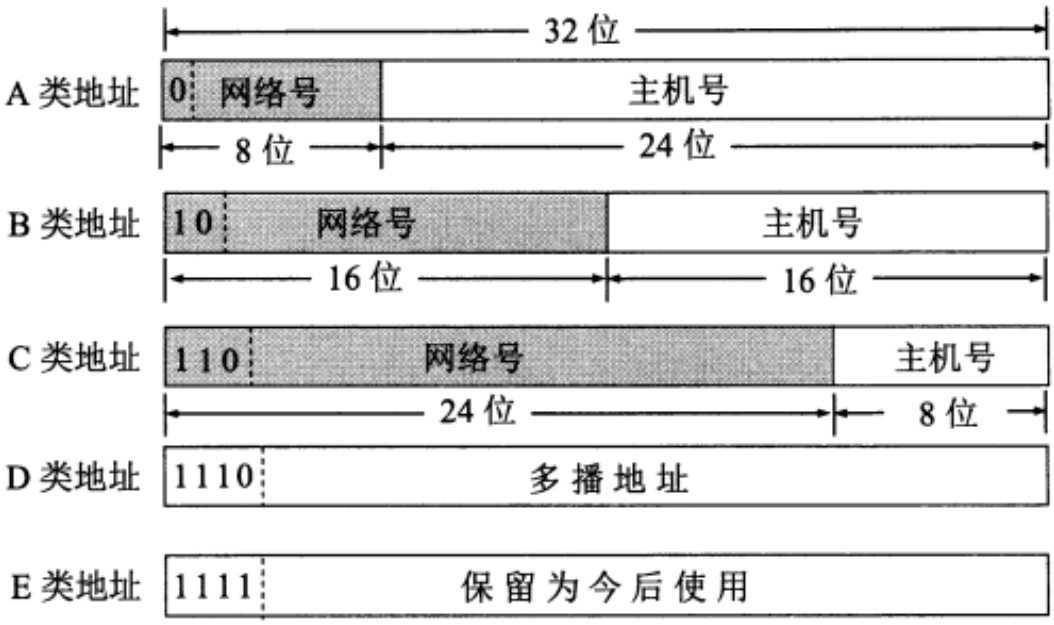
历史阶段

1. 分类的IP地址
2. 子网的划分
3. 构成超网

分类的IP地址

IP地址是标志一台主机(或路由器)和一条链路的接口。当一台主机同时连接到两个网络上时，该主机就必须同时具有两个相应的IP地址，其网络号必须是不同的。这种主机称为多归属主机(mutihomed host)。由于一个路由器至少应当连接到两个网络，因此一个路由器至少应当有两个不同的IP地址。

分类：



指派范围：

网络类别	最大可指派的网络数	第一个可指派的网络号	最后一个可指派的网络号	每个网络中的最大主机数
A	126 (2^7-2)	1	126	16777214
B	16383 ($2^{14}-1$)	128.1	191.255	65534
C	2097151 ($2^{21}-1$)	192.0.1	223.255.255	254

在A类地址中网络数要减去127(环回测试)。

特殊IP地址

NetID 网络号	HostID主机号	作为IP分组源地址	作为IP分组目的地址	用途
全0	全0	可以	不可以	本网范围内表示主机，路由表中用于表示默认路由（表示整个Internet网络）
全0	特定值	可以	不可以	表示本网内某个特定主机
全1	全1	不可以	可以	本网广播地址（路由器不转发）
特定值	全0	不可以	不可以	网络地址，表示一个网络
特定值	全1	不可以	可以	直接广播地址，对特定网络上的所有主机进行广播
127	任何数（非全0/1）	可以	可以	用于本地软件环回测试，称为环回地址

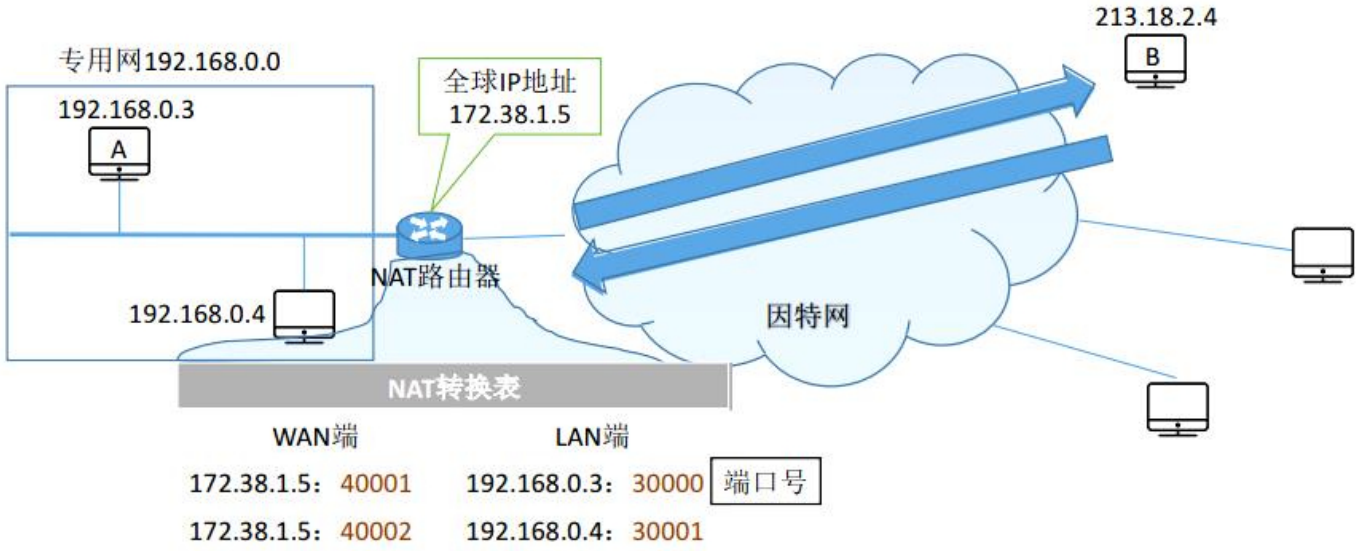
私有IP地址

地址类别	地址范围	-	网段个数
A	10.0.0.0~10.255.255.255	10.0.0.0/8	1
B	172.16.0.0~172.31.255.255	172.16.0.0/12	16
C	192.168.0.0~192.168.255.255	192.168.0.0/16	256

路由器对目的地址是私有IP地址的数据报一律不进行转发。

网络地址转换(NAT)

网络地址转换NAT(Network Address Translation)：在专用网连接到因特网的路由器上安装NAT软件，安装了NAT软件的路由器叫**NAT路由器**，它至少有一个有效的**外部全球IP地址**。



经过NAT路由器，源地址和端口号或目的地址和端口号都要发生替换。

子网划分与子网掩码

分类的IP地址缺点：

1. IP地址空间的利用率有时很低。
2. 给每一个物理网络分配一一个网络号会使路由表变得太大因而使网络性能变坏。
3. 两级IP地址不够灵活。

从两级IP地址到三级IP地址：

$$IP\text{地址} := < \text{网络号} >, < \text{子网号} >, < \text{主机号} >$$

- 由于不能全0或1，主机号至少两位。
- 子网号能否全0全1要看情况

子网掩码：
对于两级IP地址，网络号全1，主机号全0。
对于三级IP地址，网络号和子网号全1，主机号全0。

IP地址和子网掩码逐位相与，得到子网网络地址。

二进制	10000000	11000000	11100000	11110000	11111000	11111100	11111110	11111111
十进制	128	192	224	240	248	252	254	255

路由器转发分组的算法:

(1) 从收到的数据报的首部提取目的 IP 地址 D。

(2) 先判断是否为直接交付。对路由器直接相连的网络逐个进行检查：用各网络的子网掩码和 D 逐位相“与”（AND 操作），看结果是否和相应的网络地址匹配。若匹配，则把分组进行直接交付（当然还需要把 D 转换成物理地址，把数据报封装成帧发送出去），转发任务结束。否则就是间接交付，执行(3)。

(3) 若路由表中有目的地址为 D 的特定主机路由，则把数据报传送给路由表中所指明的下一跳路由器；否则，执行(4)。

(4) 对路由表中的每一行（目的网络地址，子网掩码，下一跳地址），用其中的子网掩码和 D 逐位相“与”（AND 操作），其结果为 N。若 N 与该行的目的网络地址匹配，则把数据报传送给该行指明的下一跳路由器；否则，执行(5)。

(5) 若路由表中有一个默认路由，则把数据报传送给路由表中所指明的默认路由器；否则，执行(6)。

(6) 报告转发分组出错。

默认路由：0.0.0.0/0

无分类地址CIDR

IP地址 $:=<$ 网络前缀 $>,<$ 主机号 $>$

- CIDR还使用“斜线记法”(slash notation)， 或称为CIDR记法，即在IP地址后面加上斜线“/”，然后写上网络前缀所占的位数。
- CIDR把网络前缀都相同的连续的IP地址组成一个“CIDR地址块”。
- CIDR使用32位的地址掩码(address mask)。地址掩码由一串1和一串0组成，而1的个数就是网络前缀的长度。斜线记法中，斜线后面的数字就是地址掩码中1的个数。

构成超网

将多个子网聚合成一个较大的子网，叫做构成超网，或路由聚合。
方法：将网络前缀缩短(所有网络地址取相同的部分作为新的网络地址，其余为主机地址)。

划分子网是 少 -> 多
构成超网是 多 -> 少

最长前缀匹配

在使用CIDR时，由于采用了网络前缀这种记法，IP地址由网络前缀和主机号这两个部分组成，因此在路由表中的项目也要有相应的改变。这时，每个项目由“网络前缀”和“下一跳地址”组成。但是在查找路由表时可能会得到不止一-个匹配结果。这样就带来一个问题：我们应当从这些匹配结果中选择哪一条路由呢?
正确的答案是：应当从匹配结果中选择具有最长网络前缀的路由。这叫做最长前缀匹配(longest-prefix matching)，这是因为网络前缀越长，其地址块就越小，因而路由就越具体(morespecific)。最长前缀匹配又称为

最长匹配或最佳匹配。

ARP地址解析协议

由于在实际网络的链路上传送数据帧时，最终必须使用MAC地址。

ARP协议：完成主机或路由器IP地址到MAC地址的映射。(解决下一跳走哪的问题)

每一台主机都设有一个ARP高速缓存(ARP cache)，里面有本局域网上的各主机和路由器的IP地址到硬件地址的映射表，这些都是该主机目前知道的一些地址。

使用过程
检查ARP高速缓存，有对应表项则写入MAC帧，没有则用目的MAC地址为FF-FF-FF-FF-FF的帧封装并广播ARP请求分组，同一局域网中所有主机都能收到该请求。目的主机收到请求后就会向源主机单播一个ARP响应分组，源主机收到后将此映射写入ARP缓存(10-20min更新一次)。

4种情况

1. 主机A发给本网络上的主机B：用ARP找到主机B的硬件地址。
2. 主机A发给另一网络上的主机B：用ARP找到本网络上一个路由器(网关)的硬件地址。
3. 路由器发给本网络的主机A：用ARP找到主机A的硬件地址。
4. 路由器发给另一网络的主机B：用ARP找到本网络上的一个路由器的硬件地址。

DHCP协议

IP地址的静态配置：IP地址、子网掩码、默认网关。

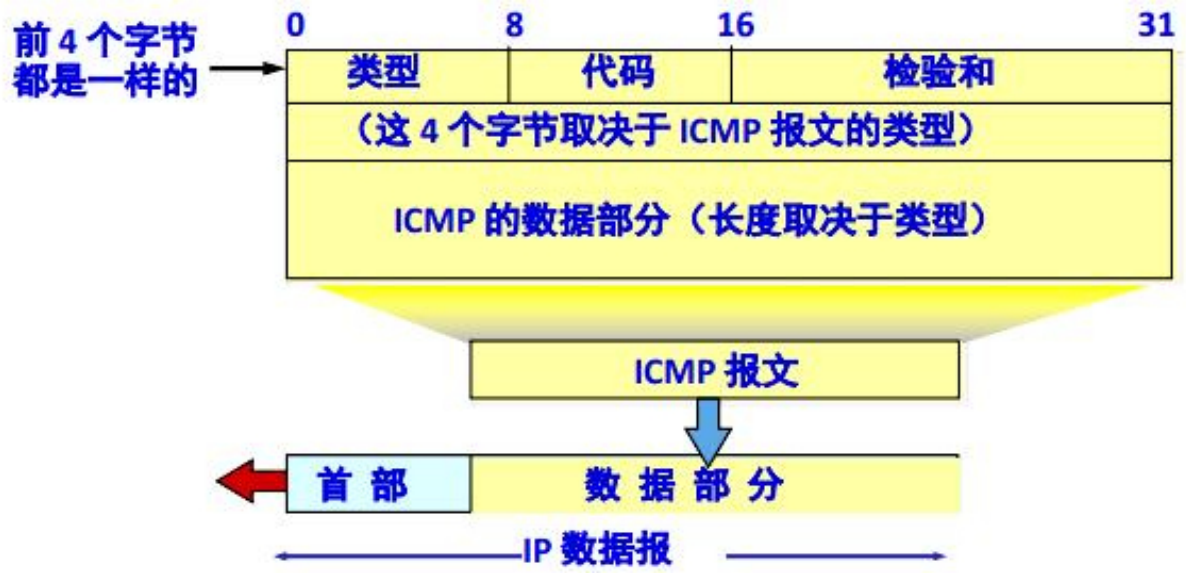
动态主机配置协议DHCP是**应用层**协议，使用**客户/服务器**方式，客户端和服务端通过**广播**方式进行交互，基于**UDP**。

DHCP提供即插即用联网的机制，主机可以从服务器动态获取IP地址、子网掩码、默认网关、DNS服务器名称与IP地址，允许地址重用，支持移动用户加入网络，支持在用地址续租。

1. 主机广播**DHCP发现**报文
2. DHCP服务器广播**DHCP提供**报文
3. 主机广播**DHCP请求**报文
4. DHCP服务器广播**DHCP确认**报文

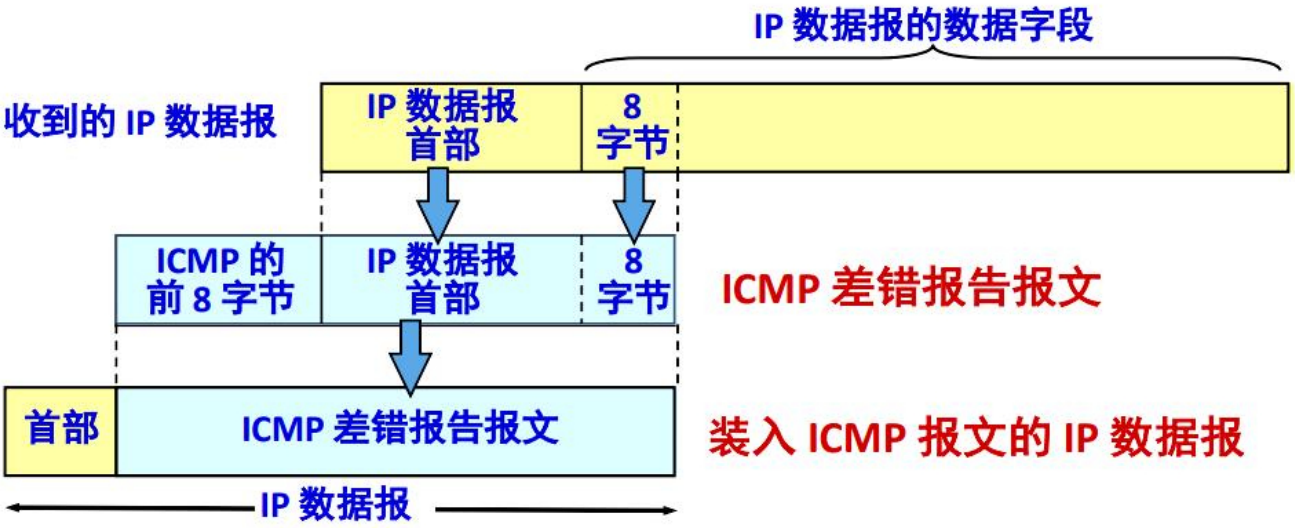
ICMP协议

ICMP协议是一种面向无连接的协议，用于传输出错报告控制信息。它属于网络层协议，主要用于在主机与路由器之间传递控制信息，包括报告错误、交换受限控制和状态信息等。



ICMP差错报告报文

1. 终点不可达：当路由器或主机不能交付数据报时就向源点发送终点不可达报文。
2. 源点抑制(该方法已经被抛弃)：当路由器或主机由于拥塞而丢弃数据时，就向源点发送源点抑制报文，使源点知道应当把数据报的发送速率放慢。(拥塞丢数据)
3. 时间超过：当路由器收到生存时间TTL=0的数据报时，除丢弃数据报外，还要向源点发送时间超过报文。当终点在预先规定的时间内不能收到一个数据报的全部数据报片时，就把已收到的数据报片都丢弃，并向源点发送时间超过报文。
4. 参数问题：当路由器或目的主机收到的数据报的首部中有的字段的值不正确，就丢弃该数据报，并向源点发送参数问题报文。
5. 改变路由(重定向)：路由器把改变路由报文发送给主机，让主机知道下次应将数据报发送给另外的路由器(可通过更好的路由)。值得更好的路由



不发送ICMP差错报文的情况：

1. 对ICMP差错报告报文不再发送ICMP差错报告报文。
2. 对第一个分片的数据片的所有后续数据报片都不发送ICMP差错报告报文。
3. 对具有组播地址的数据报都不发送ICMP差错报告报文。。
4. 对具有特殊地址(如127.0.0.0或0.0.0.0)的数据不发送ICMP差错报告报文。

ICMP询问报文

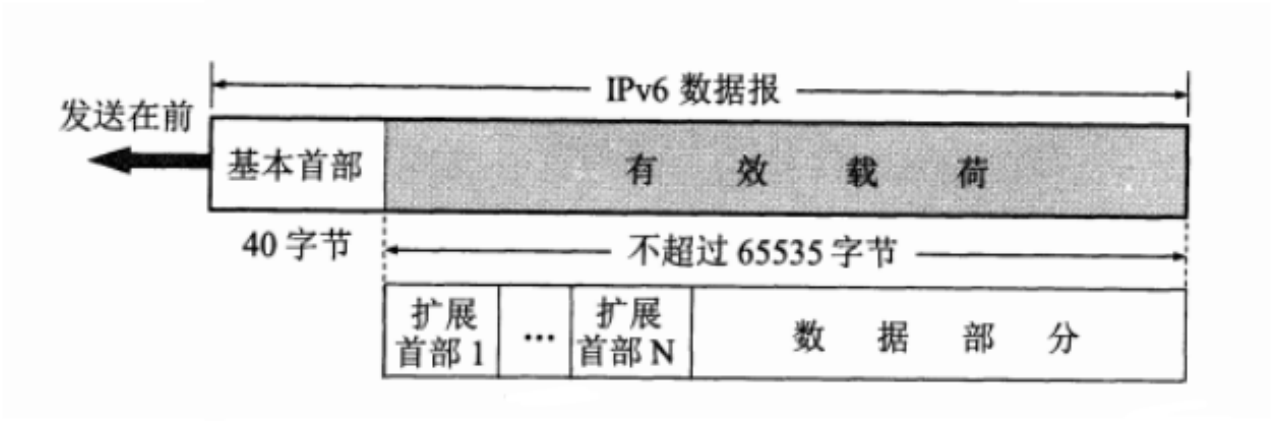
1. 回送请求和回答报文：主机或路由器特定目的主机发出的询问，收到此报文的主机必须给源主机或路由器发送ICMP回送回答报文。**测试目的站是否可达以及了解其相关状态(Ping)**。
2. 时间戳请求和回答报文：请某个主机或路由器回答当前的日期和时间，**用来进行时钟同步和测量时间**。
3. 掩码地址请求和回答报文(已不再使用)
4. 路由器询问和通告报文(已不再使用)

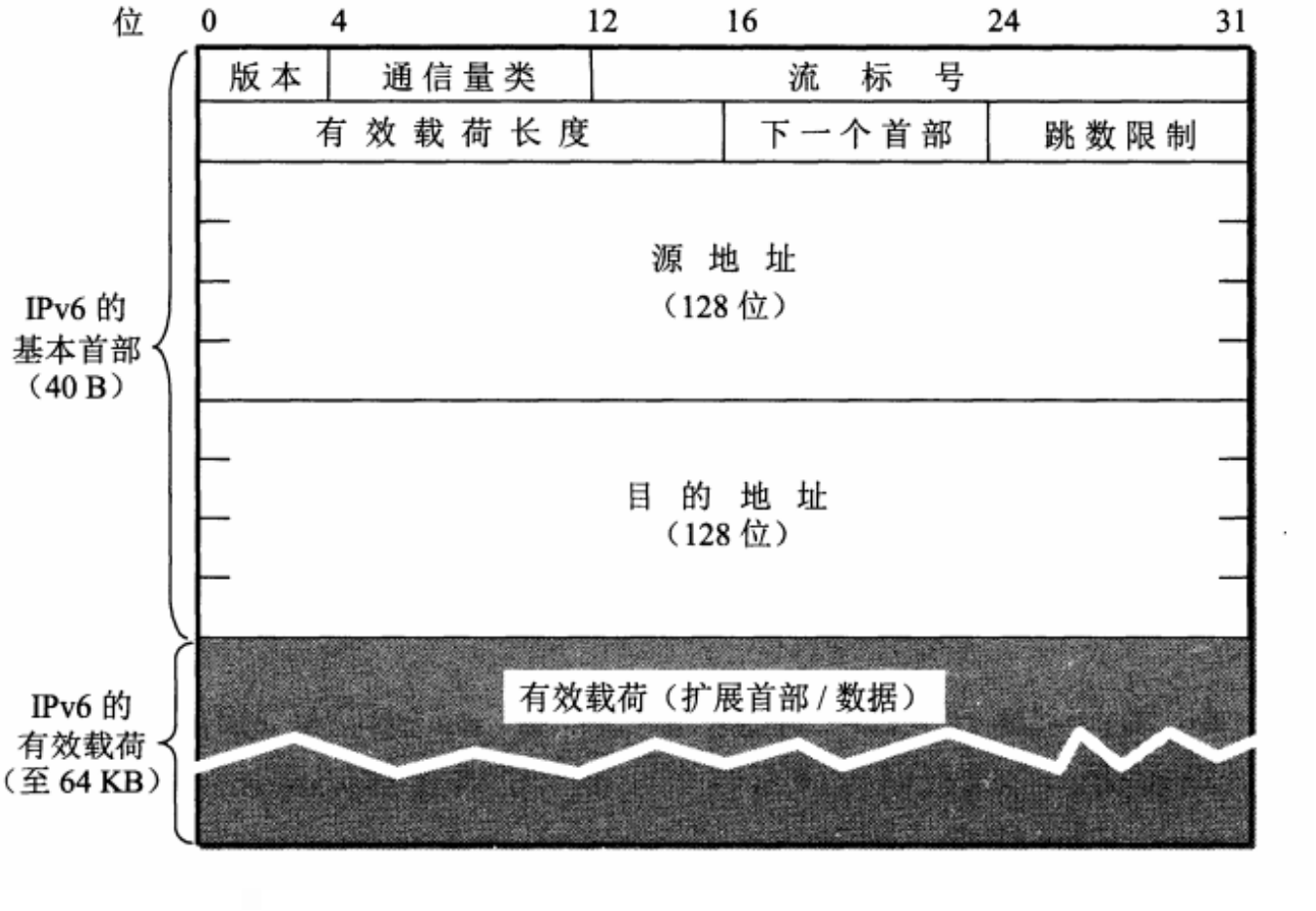
ICMP的应用

- Ping：测试连个主机之间的连通性，使用了ICMP回送请求和回答报文。
- Traceroute：跟踪一个分组从源点到终点的路径，使用了ICMP时间超过差错报告报文。

IPv6

IPv4地址分配殆尽，CIDR、NAT治标不治本。





单播、多播、任播。

IPv6和IPv4

1. IPv6将地址从32位(4B)扩大到**128位(16B)**，更大的地址空间。
2. IPv6将IPv4的校验和字段彻底移除，以减少每跳的处理时间。
3. IPv6将IPv4的可选字段移出首都，变成**扩展首部**，成为灵活的首部格式，路由器通常不对扩展首部进行检查，大大提高了路由器的处理效率。
4. IPv6支持**即插即用**(即自动装置)，不需要DHCP协议。
5. IPv6首部长度必须是**8B的整数倍**，IPv4首部是4B的整数倍。
6. IPv6**只能在主机处分片**，IPv4可以在路由器和主机处分片
7. ICMPv6：附加报文类型“分组过大”。
8. IPv6支持资源的预分配，支持实时视像等要求，保证一定的带宽和时延的应用。

9. IPv6取消了协议字段，改成下一个首部字段。
10. IPv6取消了总长度字段，改用有效载荷长度字段。
11. IPv6取消了服务类型字段。

IPv6地址表示形式

冒号十六进制记法、零压缩：一连串连续的0可以被一对冒号取代，仅可出现一次。

过度策略

双栈协议

双协议栈技术就是指在一台设备上同时启用IPv4协议栈和IPv6协议栈。这样的话，这台设备既能和IPv4网络通信，又能和IPv6网络通信。如果这台设备是一个路由器，那么这台路由器的不同接口上，分别配置了IPv4地址和IPv6地址，并很可能分别连接了IPv4网络和IPv6网络。如果这台设备是一个计算机，那么它将同时拥有IPv4地址和IPv6地址，并具备同时处理这两个协议地址的功能。

隧道技术

通过使用互联网络的基础设施在网络之间传递数据的方式。使用隧道传递的数据(或负载)可以是不同协议的数据帧或包。隧道协议将其它协议的数据帧或包重新封装然后通过隧道发送。

ICMPv6

和IPv4一样，IPv6也不保证数据报的可靠交付，因为互联网中的路由器可能会丢弃数据报。因此IPv6也需要使用ICMP来反馈些差错信息。新的版本称为ICMPv6，它比ICMPv4要复杂得多。地址解析协议ARP和网际组管理协议IGMP的功能都已被合并到ICMPv6中。
ICMPv6是面向报文的协议，它利用报文来报告差错，获取信息，探测邻站或管理多播通信。

路由算法及路由协议

静态路由算法

非自适应路由算法

管理员手工配置路由信息。

优点：简便、可靠，在负荷稳定、拓扑变化不大的网络中运行效果很好，广泛用于高度安全性的军事网络和较小的商业网络。
缺点：路由更新慢，不适用大型网络。

动态路由算法

自适应路由算法

路由器间彼此交换信息，按照路由算法优化出路由表项。

优点：路由更新快，适用大型网络，及时响应链路费用或网络拓扑变化。
缺点：算法复杂，增加网络负担。

全局性

链路状态路由算法 OSPF
所有路由器掌握完整的网络拓扑和链路费用信息。

分散性

距离向量路由算法 RIP
路由器只掌握物理相连的邻居及链路费用。

分层次的路由选择协议

自治系统AS：在单一的技术管理下的一组路由器，而这些路由器使用一种AS内部的路由选择协议和共同的度量以确定分组在该AS内的路由，同时还使用一种AS之间的路由协议以确定在AS之间的路由。
一个AS内的所有网络都属于一个行政单位来管辖，一个自治系统的所有路由器在本自治系统内都必须连通。

内部网关协议

内部网关协议IGP (Interior Gateway Protocol)
即在一个自治系统内部使用的路由选择协议，而这与在互联网中的其他自治系统选用什么路由选择协议无关。目前这类路由选择协议使用得最多，如RIP和OSPF协议。

外部网关协议

外部网关协议EGP (External Gateway Protocol)
若源主机和目的主机处在不同的自治系统中(这两个自治系统可能使用不同的内部网关协议)，当数据报传到一个自治系统的边界时，就需要使用一种协议将路由选择信息传递到另一个自治系统中。这样的协议就是外部网关协议EGP。目前使用最多的外部网关协议是BGP的版本4(BGP-4)。

RIP协议与距离向量算法

RIP是一种分布式的基于**距离向量**的路由选择协议，是因特网的协议标准，最大优点是简单。
RIP协议要求网络中每一个路由器都维护**从它自己到其他每一个目的网络的唯一最佳距离记录**(即一组距离)。

距离：通常为“跳数”，即从源端口到目的端口所经过的路由器个数，经过一个路由器跳数+1。特别的，从一路由器到直接连接的网络距离为1。RIP允许一条路由最多只能包含15个路由器，因此距离为**16表示网络不可达**。

RIP协议只适用于小互联网。

RIP协议过程

1. 仅和**相邻路由器**交换信息。
2. 路由器交换的信息是**自己的路由表**。
3. **每30秒**交换一次路由信息，然后路由器根据新信息更新路由表。若超过180s没收到邻居路由器的通告，则判定邻居没了，并更新自己路由表。

路由器刚开始工作时，只知道直接连接的网络的距离(距离为1)，接着每一个路由器也只和数目非常有限的相邻路由器交换并更新路由信息。

经过若干次更新后，所有路由器最终都会知道到达本自治系统任何一个网络的**最短距离**和**下一跳路由器的地址**，即“收敛”。

距离向量算法

对**每一个相邻路由器**发送过来的 RIP 报文，进行以下步骤：

(1) 对地址为 X 的相邻路由器发来的 RIP 报文，先修改此报文中的**所有项目**：把“下一跳”字段中的地址都改为 X，并把所有的“距离”字段的值加 1（见后面的解释 1）。每一个项目都有三个关键数据，即：到目的网络 N，距离是 d，下一跳路由器是 X。

(2) 对修改后的 RIP 报文中的每一个项目，进行以下步骤：

若原来的路由表中没有目的网络 N，则把该项目添加到路由表中（见解释 2）。

否则（即在路由表中有目的网络 N，这时就再查看下一跳路由器地址）

若下一跳路由器地址是 X，则把收到的项目替换原路由表中的项目（见解释 3）。

否则（即这个项目是：到目的网络 N，但下一跳路由器不是 X）

若收到的项目中的距离 d 小于路由表中的距离，则进行更新（见解释 4），

否则什么也不做。（见解释 5）

(3) 若 3 分钟还没有收到相邻路由器的更新路由表，则把此相邻路由器记为不可达的路由器，即把距离置为 16（距离为 16 表示不可达）。

(4) 返回。

上面给出的距离向量算法的基础就是 Bellman-Ford 算法（或 Ford-Fulkerson 算法）。这种算法的要点是这样的：

设 X 是结点 A 到 B 的最短路径上的一个结点。若把路径 A→B 拆成两段路径 A→X 和 X→B，则每一段路径 A→X 和 X→B 也都分别是结点 A 到 X 和结点 X 到 B 的最短路径。

下面是对上述距离向量算法的五点解释。

解释 1：这样做是为了便于进行本路由表的更新。假设从位于地址 X 的**相邻**路由器发来的 RIP 报文的某一个项目是：“Net2, 3, Y”，意思是“我经过路由器 Y 到网络 Net2 的距离是 3”，那么本路由器就可推断出：“我经过 X 到网络 Net2 的距离应为 3 + 1 = 4”。于是，本路由器就把收到的 RIP 报文的这一个项目修改为“Net2, 4, X”，作为下一步和路由表中原有项目进行比较时使用（只有比较后才能知道是否需要更新）。读者可注意到，收到的项目中的 Y 对本路由器是没有用的，因为 Y 不是本路由器的下一跳路由器地址。

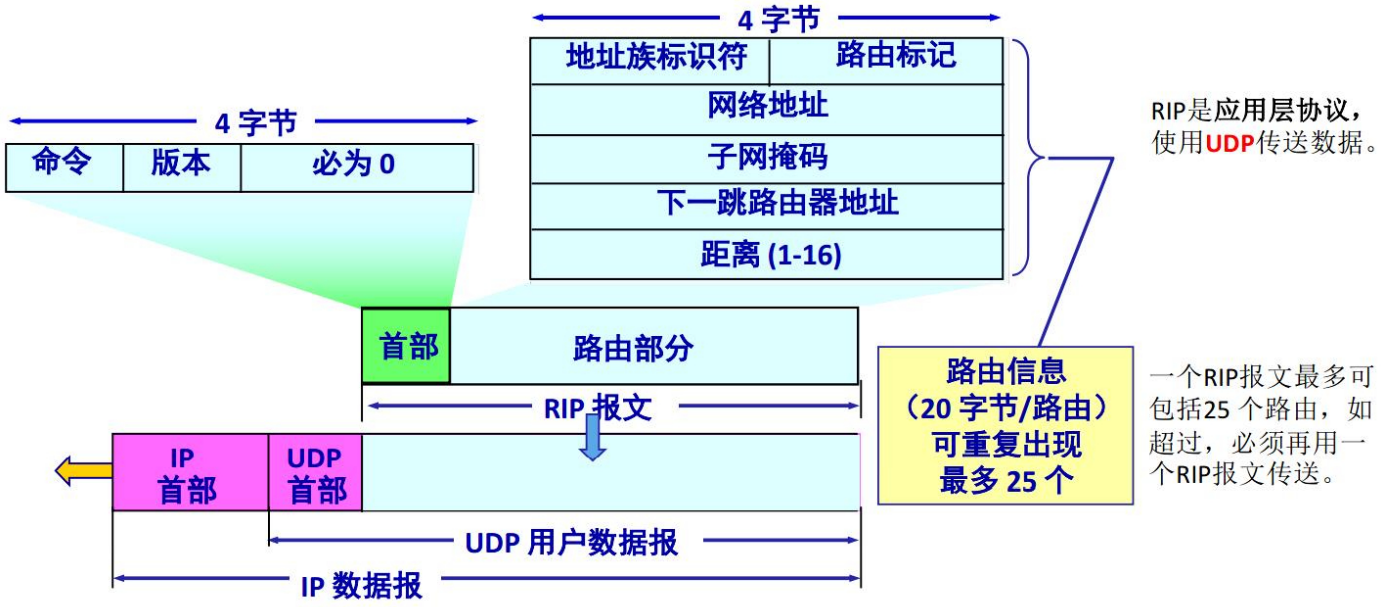
解释 2：表明这是新的目的网络，应当加入到路由表中。例如，本路由表中没有到目的网络 Net2 的路由，那么在路由表中就要加入新的项目“Net2, 4, X”。

解释 3：为什么要替换呢？因为这是最新的消息，要以最新的消息为准。到目的网络的距离有可能增大或减小，但也可能没有改变。例如，不管原来路由表中的项目是“Net2, 3, X”还是“Net2, 5, X”，都要更新为现在的“Net2, 4, X”。

解释 4：例如，若路由表中已有项目“Net2, 5, P”，就要更新为“Net2, 4, X”。因为到网络 Net2 的距离原来是 5，现在减到 4，更短了。

解释 5：若距离更大了，显然不应更新。若距离不变，更新后得不到好处，因此也不更新。

RIP报文格式



RIP存在的一个问题是当网络出现故障时，要经过比较长的时间才能将此信息传送到所有的路由器。

RIP协议好消息传得快，坏消息传得慢。

OSPF协议及链路状态算法

OSPF协议

开放最短路径优先OSPF协议：“开放”标明OSPF协议不是受某一家厂商控制，而是公开发表的；“最短路径优先”是因为使用了Dijkstra提出的最短路径算法SPF。

OSPF最主要的特征就是使用分布式的**链路状态协议**。

1. 和谁交换：使用洪泛法向自治系统内**所有路由器**发送信息，即路由器通过输出端口向所有相邻的路由器发送信息，而每一个相邻路由器又再次将此信息发往其所有的相邻路由器。(最终整个区域内所有路由器都得到了这个信息的一个副本。)
2. 交换什么：发送的信息就是与本路由器**相邻的所有路由器的链路状态**(本路由器和哪些路由器相邻，以及该链路的度量/代价——费用、距离、时延、带宽等)。
3. 多久交换：只有当**链路状态发生变化时**，路由器才向所有路由器洪泛发送此信息。

最后，所有路由器都能建立一个**链路状态数据库**，即**全网拓扑结构图**。

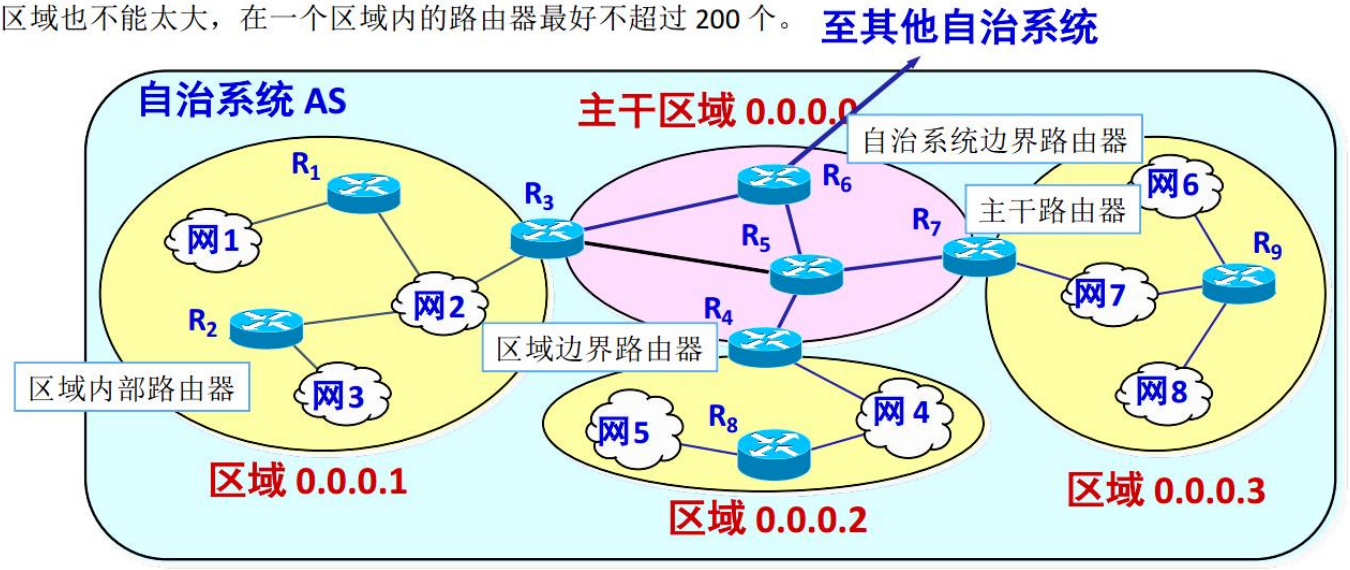
链路状态算法

1. 每个路由器发现它的邻居结点【HELLO问候分组】，并了解邻居节点的网络地址。
2. 设置到它的每个邻居的成本度量metric。
3. 构造【DD数据库描述分组】，向邻站给出自己的链路状态数据库中的所有链路状态项目的摘要信息。
4. 如果DD分组中的摘要自己都有，则邻站不做处理；如果有没有的或者是更新的，则发送【LSR链路状态请求分组】
   请求自己没有的和比自己更新的信息。
5. 收到邻站的LSR分组后，发送【LSU链路状态更新分组】进行更新。
6. 更新完毕后，邻站返回一个【LSAck链路状态确认分组】进行确认。
   只要一个路由器的链路状态发生变化：

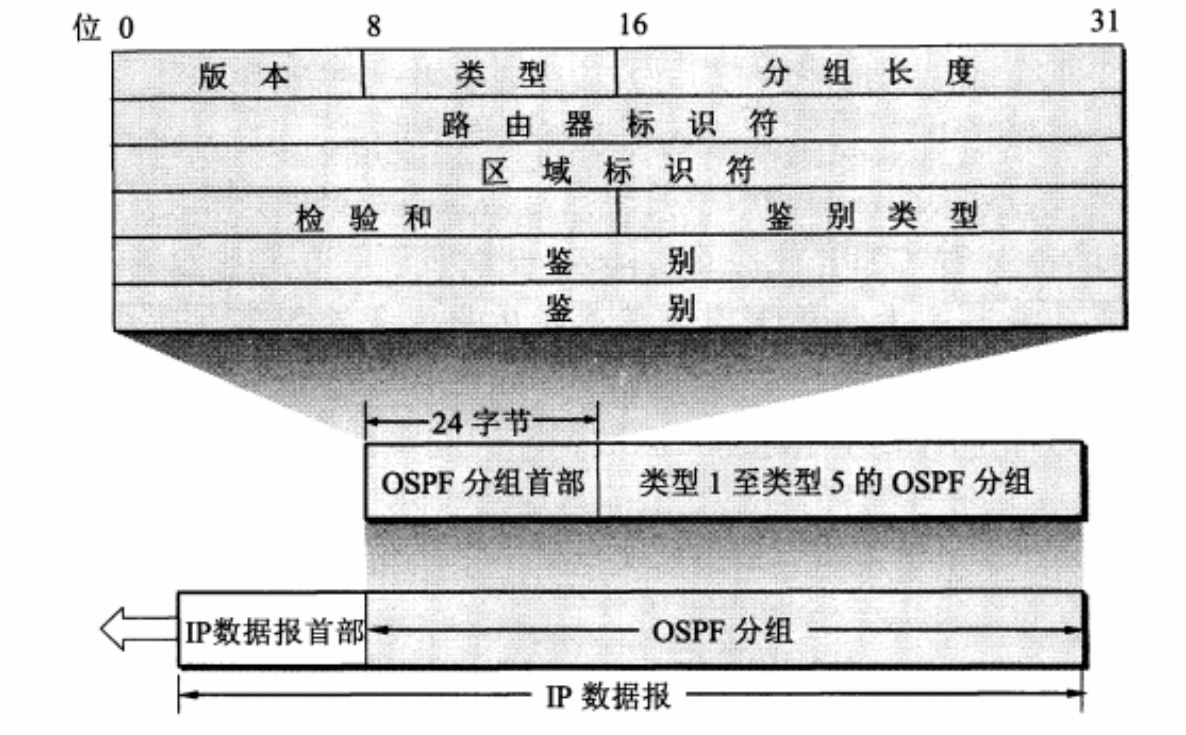
7. 泛洪发送【LSU链路状态更新分组】进行更新。
8. 更新完毕后，其他站返回一个【LSAck链路状态确认分组】进行确认。
9. 使用Dijkstra根据自己的链路状态数据库构造到其他节点间的最短路径。

OSPF的区域

为了使 OSPF 能够用于规模很大的网络，OSPF 将一个自治系统再划分为若干个更小的范围，叫做区域。
每一个区域都有一个 32 位的区域标识符（用点分十进制表示）。
区域也不能太大，在一个区域内的路由器最好不超过 200 个。



OSPF分组



OSPF特点

1. 适应范围：OSPF支持各种规模的网络，最多可支持几百台路由器。
2. 最佳路径：OSPF是基于带宽来选择路径。
3. 快速收敛：如果网络的拓扑结构发生变化，OSPF立即发送更新报文，使这一变化在自治系统中同步。
4. 无自环：由于OSPF通过收集到的链路状态用最短路径树算法计算路由，故从算法本身保证了不会生成自环路由。
5. 子网拖码：由于OSPF在描述路由时携带网段的掩码信息，所以OSPF协议不受自然掩码的限制，对VLSM和CIDR提供很好的支持。
6. 区域划分：OSPF协议允许自治系统的网络被划分成区域来管理，区域间传送的路由信息被进一步抽象，从而减少了占用网络的带宽。
7. 等值路由：OSPF支持到同一目的地址的多条等值路由。
8. 路由分级：OSPF使用4类不同的路由，按优先顺序来说分别是：区域内路由、区域间路由、第一类外部路由、第二类外部路由。
9. 支持验证：它支持基于接口的报文验证以保证路由计算的安全性。

BGP协议

边界网关协议BGP
边界网关协议BGP只能是力求寻找一条能够到达目的网络且比较好的路由(不能兜圈子)，而并非要寻找一条最佳路由。BGP采用了路径向量(path vector)路由选择协议，它与距离向量协议(如RIP)和链路状态协议(如OSPF)都有很大的区别。

在配置BGP时，每一个自治系统的管理员要选择至少一个路由器作为该自治系统的“BGP发言人”。一般说来，两个BGP发言人都是通过一个共享网络连接在一起的，而BGP发言人往往就是BGP边界路由器，但也可以不是BGP边界路由器。

BGP协议交换信息的过程

图 4-40 给出了一个 BGP 发言人交换路径向量的例子。自治系统 AS_2 的 BGP 发言人通知主干网的 BGP 发言人：“要到达网络 N_1，N_2，N_3 和 N_4 可经过 AS_2。”主干网在收到这个通知后，就发出通知：“要到达网络 N_1，N_2，N_3 和 N_4 可沿路径（AS_1，AS_2）。”同理，主干网还可发出通知：“要到达网络 N_5，N_6 和 N_7 可沿路径（AS_1，AS_3）。”

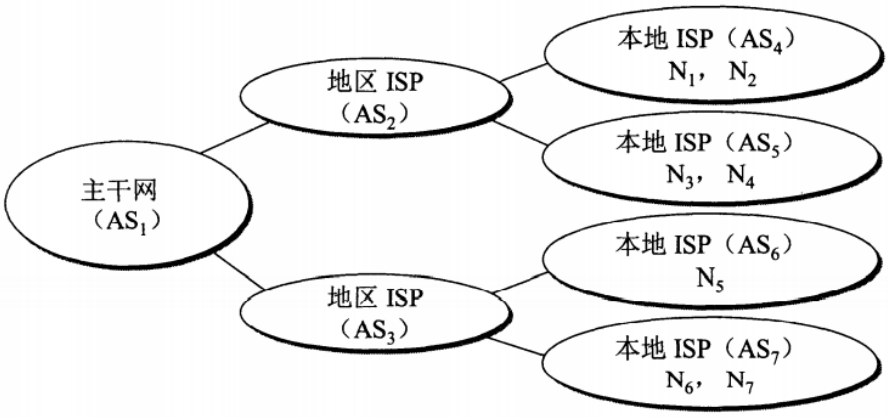
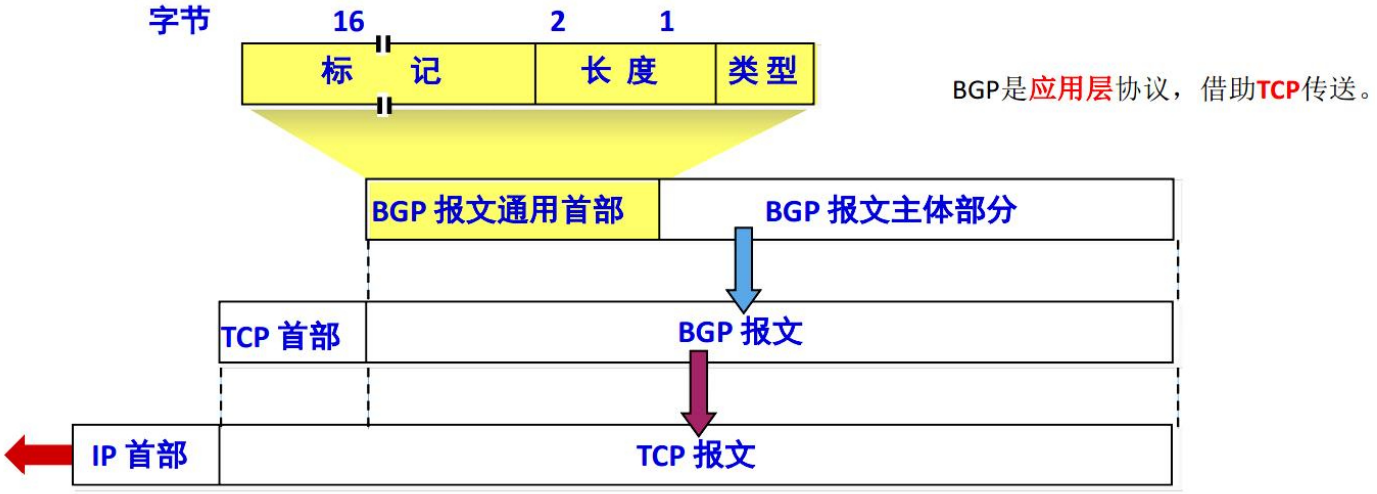


图 4-40　BGP 发言人交换路径向量的例子

BGP协议报文格式

一个BGP发言人与其他自治系统中的BGP发言人要交换路由信息，就要先**建立TCP 连接**，即通过TCP传送，然后在此连接上交换BGP报文以建立BGP会话(session)，利用BGP会话交换路由信息。



BGP协议特点

- BGP支持CIDR，因此 BGP的路由表也就应当包括目的网络前缀、下一跳路由器，以及到达该目的网络所要经过的各个自治系统序列。
- 在BGP刚刚运行时，BGP的邻站是交换整个的BGP路由表。但以后只需要在发生变化时更新有变化的部分。这样做对节省网络带宽和减少路由器的处理开销都有好处。

BGP-4的四种报文

1. **OPEN(打开)报文**：用来与相邻的另一个BGP发言人建立关系，并认证发送方。
2. **UPDATE(更新)报文**：通告新路径或撤销原路径。
3. **KEEPALIVE(保活)报文**：在无UPDATE时，周期性证实邻站的连通性；也作为OPEN的确认。
4. **NOTIFICATION(通知)报文**：报告先前报文的差错；也被用于关闭连接。

三种路由协议比较

1. **RIP**是一种分布式的基于距离向量的内部网关路由选择协议，通过广播**UDP**报文来交换路由信息。
2. **OSPF**是一个内部网关协议，要交换的信息量较大，应使报文的长度尽量短，所以不使用传输层协议(如UDP或TCP)，而是直接采用**IP**。
3. **BGP**是一个外部网关协议，在不同的自治系统之间交换路由信息，由于网络环境复杂，需要保证可靠传输，所以采用**TCP**。

协议	RIP	OSPF	BGP	
类型	内部	内部	外部	
路由算法	距离-向量	链路状态	路径-向量	
传递协议	UDP	IP	TCP	
路径选择	跳数最少	代价最低	较好，非最佳	
交换结点	和本结点相邻的路由器	网络中的所有路由器	和本结点相邻的路由器	
交换内容	当前本路由器知道的全部信息，即自己的路由表	与本路由器相邻的所有路由器的链路状态	首次	整个路由表
			非首次	有变化的部分

MPLS协议

MPLS位于TCP/IP协议栈中的链路层和网络层之间，用于向IP层提供连接服务，同时又从链路层得到服务。MPLS以标签交换替代IP转发，标签是一个短而定长的、只具有本地意义的连接标识符，与ATM的VPI/VCI以及Frame Relay的DLCI类似。

MPLS的特点: (1) 支持面向连接的服务质量; (2) 支持流量工程，平衡网络负载; (3) 有效地支持虚拟专用网VPN。

MPLS在入口结点给每一个IP数据报打上固定长度的“标记”，然后根据标记在第二层(链路层)用硬件进行转发(在标记交换路由器中进行标记对换)，因而转发速率大大加快。

IP组播

IP数据报的三种传输方式

- 单播：单播用于发送数据包到单个目的地，且每发送一份单播报文都使用一个单播IP地址作为目的地址。是一种**点对点**传输方式。
- 广播：广播是指发送数据包到同一广播域或子网内的所有设备的一种数据传输方式，是一种**点对多点**传输方式。
- 组播(多播)：当网络中的某些用户需要特定数据时，组播数据发送者仅发送一次数据，借助组播路由协议为组播数据包建立组播分发树，被传递的数据到达距离用户端尽可能近的节点后才开始复制和分发，是一种**点对多点**传输方式。

组播提高了数据传送效率。减少了主干网出现拥塞的可能性。组播组中的主机可以是在同一个物理网络，也可以来自不同的物理网络(如果有组播路由器的支持)。

IP组播地址

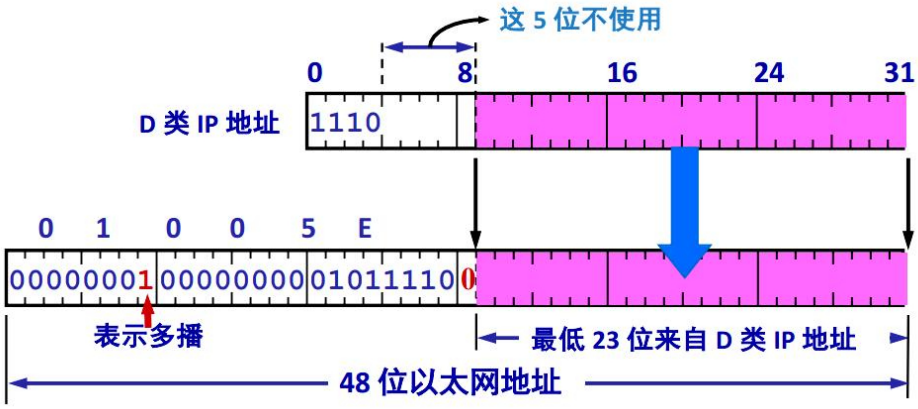
IP组播地址让源设备能够将分组发送给一组设备。属于多播组的设备将被分配**一个组播组IP地址**(一群共同需求主机的相同标识)。

组播地址范围为224.0.0.0~239.255.255.255(D类地址)，一个D类地址表示一个组播组。只能用作分组的**目标地址**。源地址总是为**单播地址**。

1. 组播数据报也是“尽最大努力交付”，不提供可靠交付，应用于UDP。
2. 对组播数据报不产生ICMP差错报文。
3. 并非所有D类地址都可以作为组播地址。

硬件组播

同单播地址一样，组播IP地址也需要相应的组播MAC地址在本地网络中实际传送帧。组播MAC地址以十六进制值01-00-5E打头，余下的6个十六进制位是根据IP组播组地址的最后23位转换得到的。
TCP/IP协议使用的以太网多播地址的范围是从01-00-5E-00-00-00到01-00-5E-7F-FF-FF。



收到多播数据报的主机，还要在IP 层利用软件进行过滤，把不是本主机要接收的数据报丢弃。

软件解决，IP地址中5位不使用的位不同，而后面23位一样的问题。

网际组管理协议IGMP

IGMP协议让路由器知道本局域网上是否有主机(的进程)参加或退出了某个组播组。

ICMP和IGMP都使用IP数据报传递报文。

过程：

1. 某主机要加入组播组时，该主机向组播组的组播地址发送一个IGMP报文，声明自己要称为该组的成员。本地组播路由器收到IGMP报文后，要利用组播路由选择协议把这组成员关系发给因特网上的其他组播路由器。
2. 本地组播路由器周期性探询本地局域网上的主机，以便知道这些主机是否还是组播组的成员。

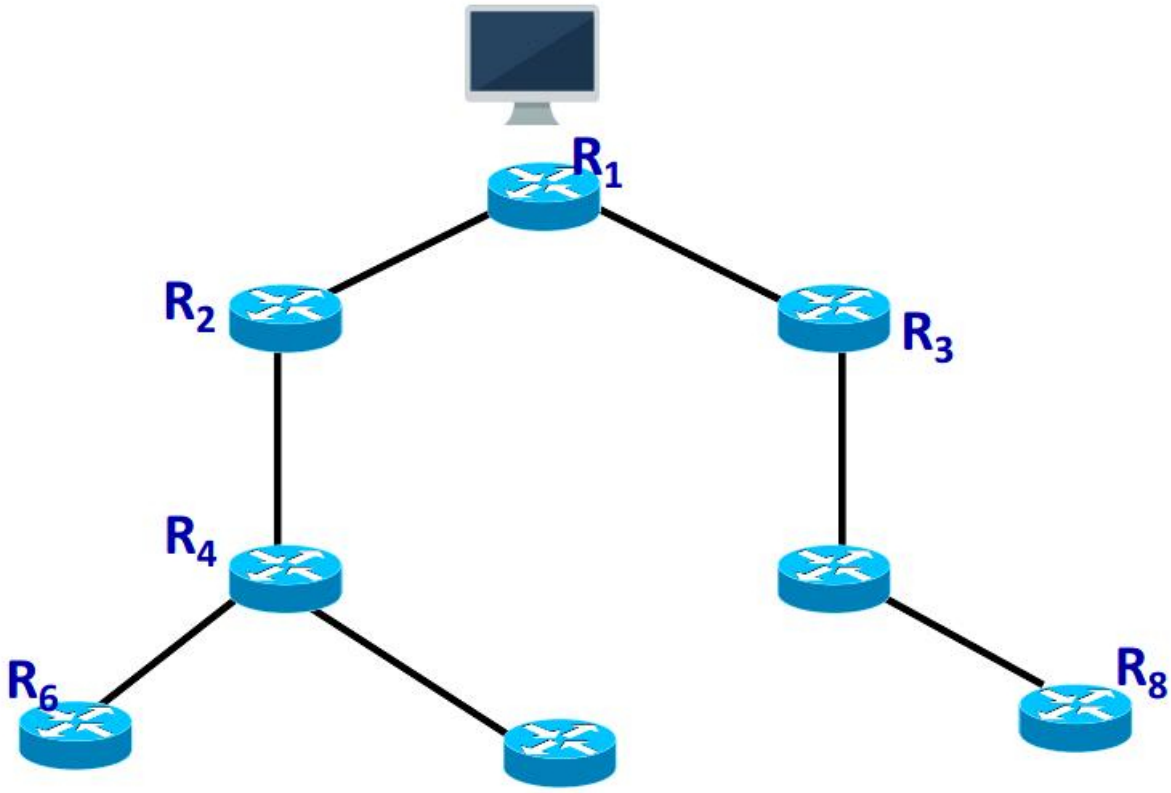
只要有一个主机对某个组响应，那么组播路由器就认为这个组是活跃的；如果经过几次探询后没有一个主机响应，组播路由器就认为本网络上的没有此组播组的主机，因此就不再把这组的成员关系发给其他的组播路由器。

组播路由器知道的成员关系只是所连接的局域网中有无组播组的成员。

组播路由选择协议

组播路由选择协议目的是找出以源主机为根节点的组播转发树。

对不同的多播组对应于不同的多播转发树；同一个多播组，对不同的源点也会有不同的多播转发树。



组播路由选择协议常使用的三种算法：

1. 基于链路状态的路由选择
2. 基于距离向量的路由选择
3. 协议无关的组播(稀疏/密集)

移动IP

移动IP技术是移动结点(计算机/服务器等)以固定的网络IP地址，实现跨越不同网段的漫游功能，并保证了基于网络IP的网络权限在漫游过程中不发生任何改变。

- 移动结点：具有永久IP地址的移动设备。
- 归属代理(本地代理)：一个移动结点的永久“居所”称为归属网络，在归属网络中代表移动节点执行移动管理功能的实体叫做归属代理。
- 永久地址(归属地址/主地址)：移动站点在归属网络中的原始地址。
- 外部代理(外地代理)：在外部网络中帮助移动节点完成移动管理功能的实体称为外部代理。
- 转交地址(辅地址)：可以是外部代理的地址或动态配置的一个地址。

通信过程：

A刚进入外部网络：

1. 获得外部代理的转交地址(外部代理广播报文)。
2.移动节点通过外部代理发送注册报文给归属代理（包含永久地址&转交地址）。
3.归属代理接收请求，并将移动节点的永久地址和转交地址绑定（以后到达该归属代理的数据报且要发往移动节点的数据报将被封装并以隧道方式发给转交地址），并返回一注册响应报文。
4.外部代理接收注册响应，并转发给移动节点。

A移动到了下一个网络：

1.在新外部代理登记注册一个转交地址。
2.新外部代理给本地代理发送新的转交地址（覆盖旧的）。
3.通信

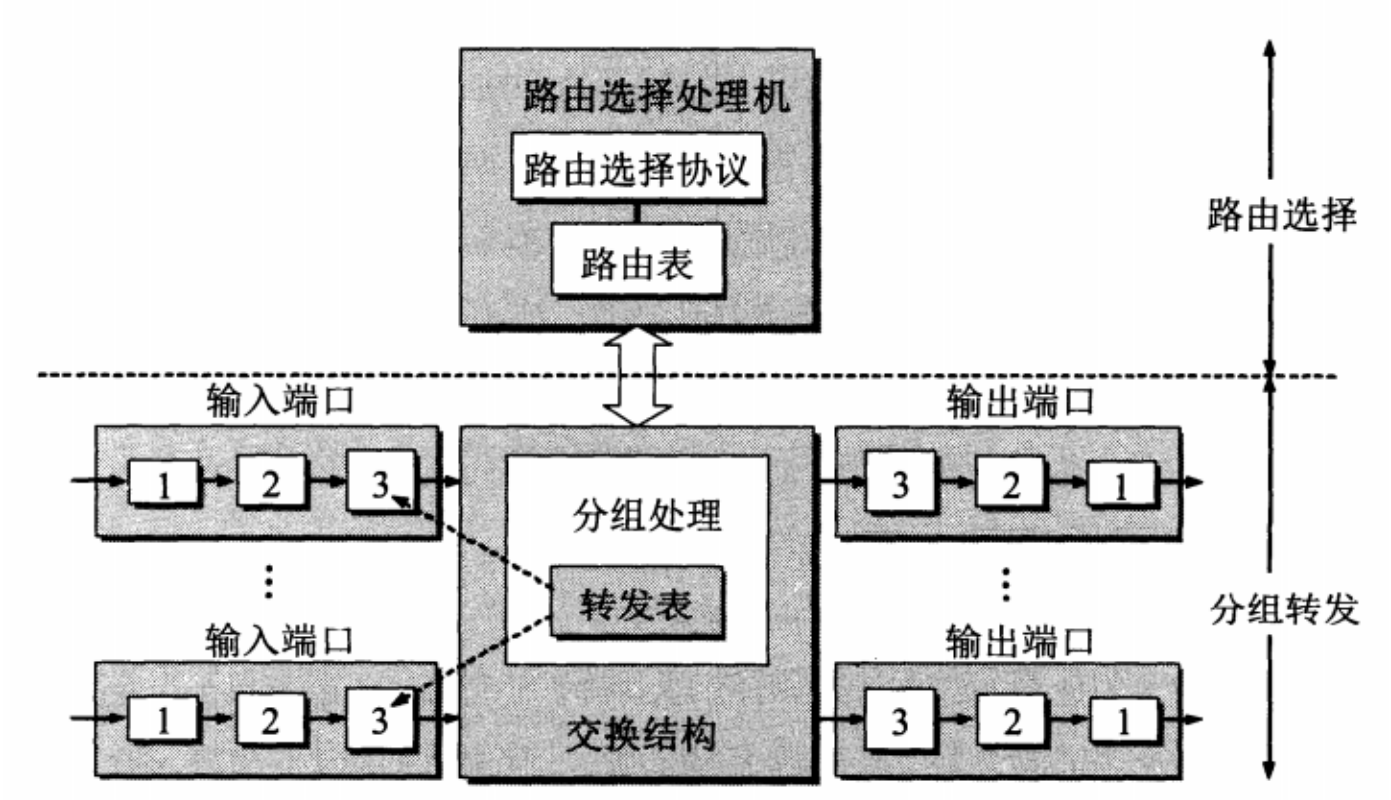
A回到了归属网络：

1.A向本地代理注销转交地址。
2.按原始方式通信。

网络层设备

路由器

路由器是一种具有多个输入端口和多个输出端口的专用计算机，其任务是转发分组。



图中数字表示相应层次的构件：3网络层；2数据链路层；1网络层。

可划分为：路由选择部分和分组转发部分。

路由选择：路由选择部分也叫做控制部分，其核心构件是路由选择处理机。路由选择处理机的任务是根据所选定的路由选择协议构造出路由表，同时经常或定期地和相邻路由器交换路由信息而不断地更新和维护路由表。

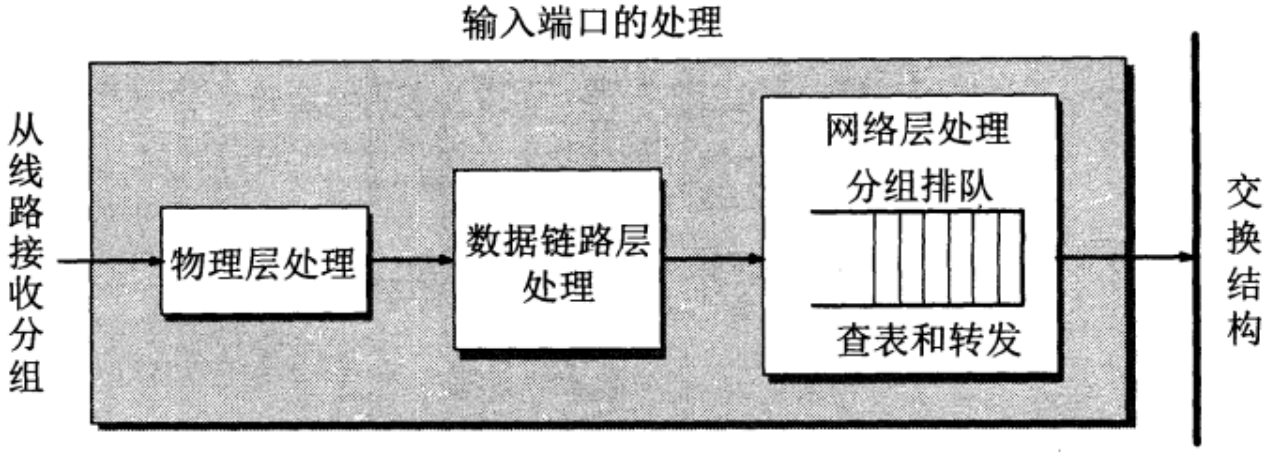
分组转发：若收到RIP/OSPF分组等，则把分组送往路由选择处理机；若收到数据分组，则查找转发表并输出。

分组转发由三部分组成：交换结构、一组输入端口和一组输出端口(请注意:这里的端口就是硬件接口)。

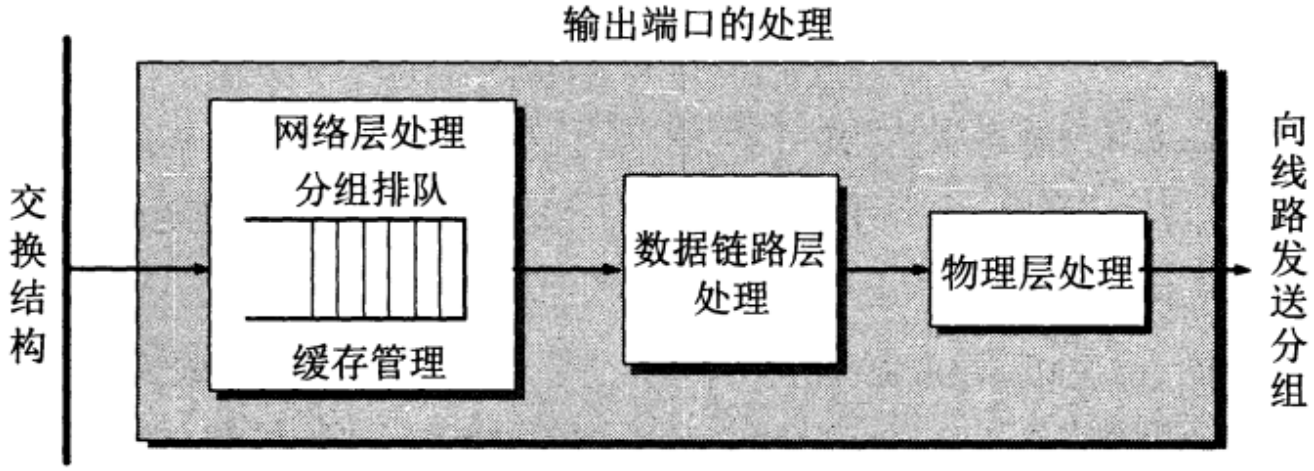
交换结构：根据转发表(路由表得来)对分组进行转发。

路由和转发：“转发”和“路由选择”是有区别的。在互联网中，“转发”就是路由器根据转发表把收到的IP数据报从路由器合适的端口转发出去。“转发"仅仅涉及到一个路由器。但路由选择”则涉及到很多路由器，路由表则是许多路由器协同工作的结果。

输入端口：
输入端口中的查找和转发功能在路由器的交换功能中是最重要的。



输出端口：
路由器中的输入或输出队列产生溢出是造成分组丢失的重要原因。



三层设备区别

- 路由器：可以互联两个不同网络层协议的网段。
- 网桥：可以互联两个物理层和链路层不同的网段。
- 集线器：不能互联两个物理层不同的网段。

路由表与路由转发

路由表根据**路由选择算法**得出的，主要用途是路由选择，总用软件来实现。

路由表：

目的网络IP地址	子网掩码	下一跳IP地址	接口
-	-	-	-

转发表由路由表得来，可以用软件实现，也可以用特殊的硬件来实现。转发表必须包含完成转发功能所必需的信息，在转发表的每一行必须包含从要到达的目的网络到输出端口和某些MAC地址信息的映射。

5 传输层

概述

只有主机才有的层次。

传输层的功能：

1. 传输层提供进程和进程之间的逻辑通信。(网络层提供主机之间的逻辑通信)
2. 复用和分用。
3. 传输层对收到的报文进行差错检测。
4. TCP和UDP两种协议。

UDP协议
无连接的用户数据报协议，传输单元：用户数据报。

传送数据之前不需要建立连接，收到UDP报文后也不需要给出任何确认。

不可靠，无连接，时延小，适用于小文件。

TCP协议
面向连接的传输控制协议，传输单元：报文段。

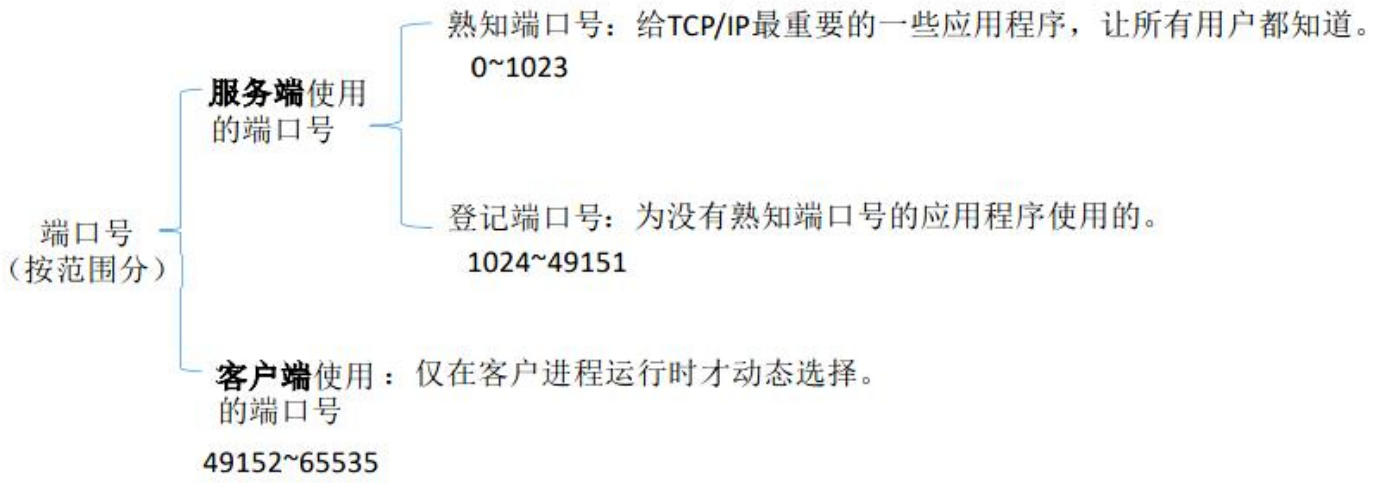
TCP在传送数据之前必须先建立连接，数据传送结束后要释放连接。
TCP不提供广播或多播服务，由于TCP要提供可靠的、面向连接的运输服务，因此不可避免地增加了许多的开销，如确认、流量控制、计时器以及连接管理等。

可靠，面向连接，时延大，适用于大文件。

传输层的寻址与端口
复用：应用层所有的应用进程都可以通过传输层再传输到网络层。
分用：传输层从网络层收到数据后交付指明的应用进程。

逻辑端口/软件端口是传输层的SAP，标识主机中的应用进程。

端口号只有本地意义，在因特网中不同计算机的相同端口是没有联系的。
端口号长度为16bit，能表示65536个不同的端口号。



应用程序	FTP	TELNET	SMTP	DNS	TFTP	HTTP	SNMP	SNMP(trap)	HTTPS
端口号	21	23	25	53	69	80	161	162	443

在网络中采用发送方和接收方的套接字组合来识别端点，**套接字**唯一标识了网络中的一个主机和它上面的一个进程。

套接字Socket =(主机IP地址: 端口号)

UDP协议

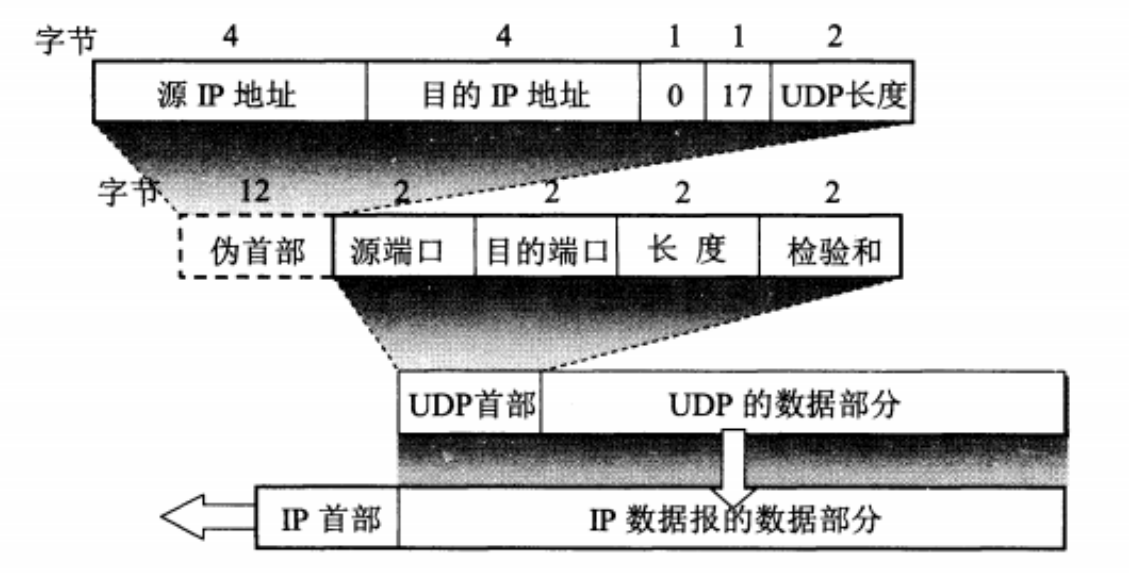
UDP只在IP数据报服务之上增加了很少功能，即复用分用和差错检测功能。

特点
1. UDP是**无连接**的，减少开销和发送数据之前的时延。

2. UDP使用最大努力交付，即**不保证可靠交付**。
3. UDP是**面向报文**的，适合一次性传输少量数据的网络应用。
4. UDP无拥塞控制，适合很多实时应用。
5. UDP首部开销小，8B。TCP 20B。

UDP首部格式

分用时，找不到对应的目的端口号，就丢弃报文，并给发送方发送ICMP“端口不可达”差错报告报文。



UDP用户数据报首部中检验和的计算方法有些特殊。在计算检验和时，要在UDP用户数据报之前增加12个字节的**伪首部**。所谓“伪首部”是因为这种伪首部并不是UDP用户数据报真正的首部。只是在计算检验和时，临时添加在UDP用户数据报前面，得到一个临时的UDP用户数据报。检验和就是按照这个临时的UDP用户数据报来计算的。伪首部既不向下传送也不向上递交，而仅仅是为了计算检验和。

17：封装UDP报文的IP数据报首部协议字段是17。
UDP长度：UDP首部8B+数据部分长度(不包括伪首部)。

在发送端

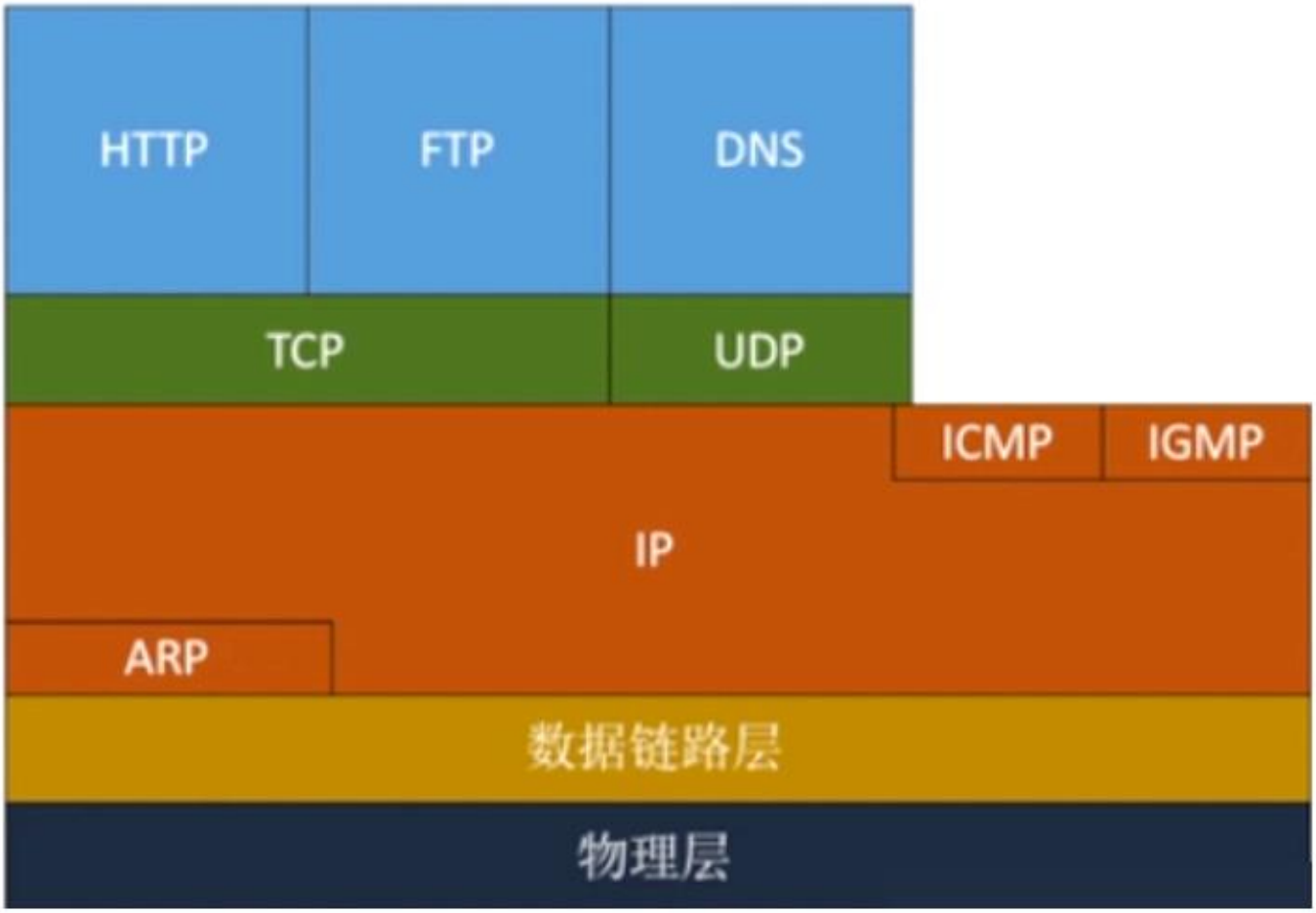
1. 填上伪首部
2. 全0填充检验和字段
3. 全0填充数据部分(UDP数据报要看成许多4B的字串接起来)
4. 伪首部+首部+数据部分采用二进制反码求和
5. 把和求反码填入检验和字段
6. 去掉伪首部，发送

在接收端

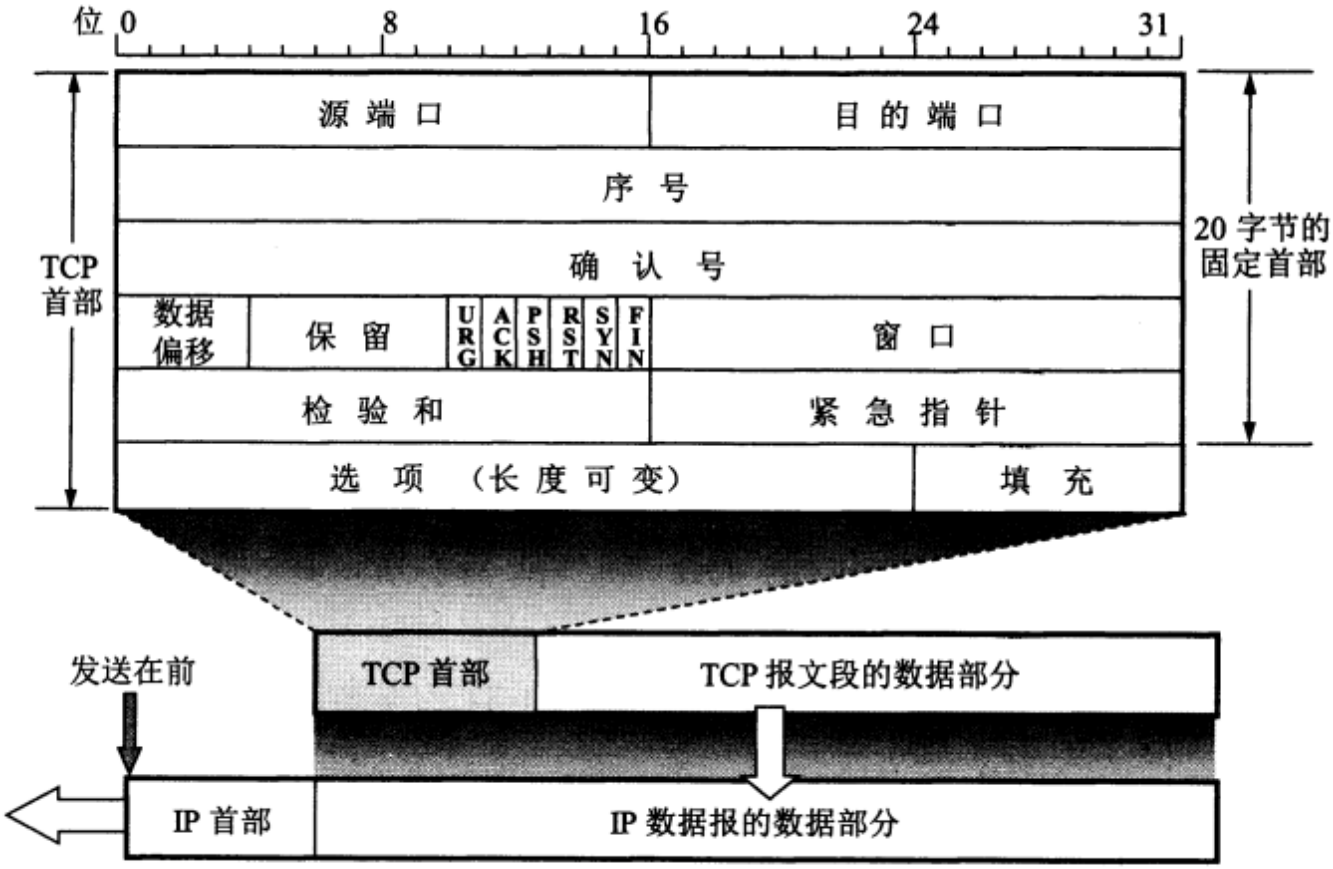
1. 填上伪首部
2. 伪首部+首部+数据部分采用二进制反码求和
3. 结果全为1则无差错，否则丢弃数据报/交给应用层附上出差错的警告

TCP协议

1. TCP是面向连接的传输层协议。
2. 每一条TCP连接(虚连接)只能有两个端点，每一条TCP连接只能是点对点的。无法用于广播和多播。
3. TCP提供可靠交付的服务，无差错、不丢失、不重复、按序到达。
4. TCP提供全双工通信。发送缓存(准备发送的数据&已发送但尚未收到确认的数据)和接收缓存(按序到达但尚未被接受应用程序读取的数据&不按序到达的数据)。
5. TCP面向字节流：TCP把应用程序交下来的数据看成仅仅是一连串的无结构的字节流。



TCP报文段首部格式



- 序号：在一个TCP连接中传送的字节流中的每一个字节都按顺序编号，本字段表示本报文段所发送数据的第一个字节的序号。
- 确认号：期望收到对方下一个报文段的第一个数据字节的序号。若确认号为N，则证明到序号N-1为止的所有数据都已正确收到。
- 数据偏移(首部长度)：TCP报文段的数据起始处距离TCP报文段的起始处有多远，以4B位单位，即1个数值是4B。

6个控制位：

- **紧急位URG**：URG=1时，标明此报文段中有紧急数据，是高优先级的数据，应尽快传送，不用在缓存里排队，配合紧急指针字段使用。
- **确认位ACK**：ACK=1时，确认号有效，在连接建立后所有传送的报文段都必须把ACK置为1。
- 推送位PSH：PSH=1时，接收方尽快交付接收应用进程，不再等到缓存填满再向上交付。
- 复位RST：RST=1时，表明TCP连接中出现严重差错，必须释放连接，然后再重新建立传输连接。
- **同步位SYN**：SYN=1时，表明是一个连接请求/连接接受报文。
- **终止位FIN**：FIN=1时，表明此报文段发送方数据已发完，要求释放连接。

- 窗口：指的是发送本报文段的一方的接收窗口，即现在允许对方发送的数据量。
- 检验和：检验首部+数据，检验时要加上12B伪首部，第四个字段为6(协议字段)。
- 紧急指针：URG=1时才有意义，指出本报文段中紧急数据的字节数。
- 选项：最大报文段长度MSS、窗口扩大、时间戳、选择确认...
- 填充：使得TCP首部单位为4B。

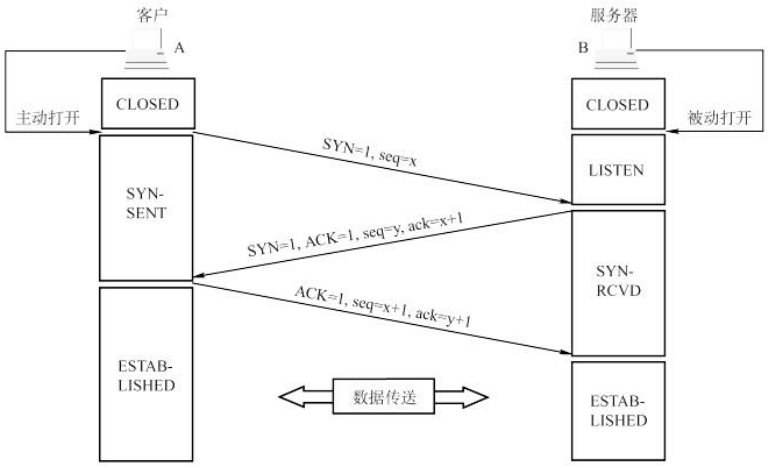
TCP连接管理

阶段：连接建立 -> 数据传送 -> 连接释放

TCP连接的建立采用**客户服务器方式**，主动发起连接建立的应用进程叫做客户，而被动等待连接建立的应用进程叫服务器。

连接建立

假设运行在一台主机(客户)上的一个进程想与另一台主机(服务器)上的一个进程建立一条连接，客户应用进程首先通知客户TCP，他想建立一个与服务器上某个进程之间的连接，客户中的TCP会用以下步骤与服务器中的TCP建立一条TCP连接：



ROUND 1:

客户端发送**连接请求报文段**，无应用层数据。

SYN=1，seq=x(随机)

ROUND 2:

服务器端为该TCP连接**分配缓存和变量**，并向客户端返回**确认报文段**，允许连接，无应用层数据。

SYN=1，ACK=1，seq=y(随机)，ack=x+1

ROUND 3:

客户端为该TCP连接**分配缓存和变量**，并向服务器端返回确认的确认，可以携带数据。

SYN=0，ACK=1，seq=x+1，ack=y+1

SYN同步位、seq序号位、ACK确认位(Acknowledgement)、ack确认号(Acknowledgement Number)

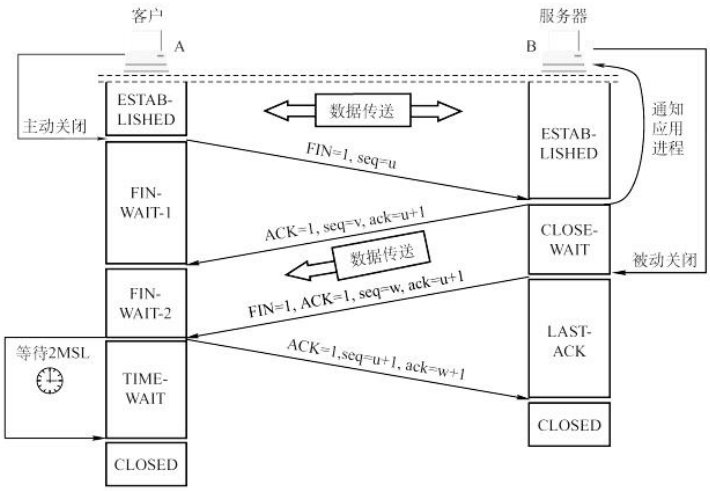
SYN洪泛攻击

SYN洪泛攻击发生在OSI第四层，这种方式利用TCP协议的特性，就是三次握手。攻击者发送TCP SYN，SYN是TCP三次握手中的第一个数据包，而当服务器返回ACK后，该攻击者就不对其进行再确认，那这个TCP连接就处于挂起状态，也就是所谓的半连接状态，服务器收不到再确认的话，还会重复发送ACK给攻击者。这样更加会浪费服务器的资源。攻击者就对服务器发送非常大量的这种TCP连接，由于每一个都没法完成三次握手，所以在服务器上，这些TCP连接会因为挂起状态而消耗CPU和内存，最后服务器可能死机，就无法为正常用户提供服务了。

设置SYN cookie解决。

连接释放

参与一条TCP连接的两个进程中的任何一个都能终止该连接，连接结束后，主机中的“资源”(缓存和变量)将被释放。



ROUND 1:

客户端发送**连接释放报文段**，停止发送数据，主动关闭TCP连接。

FIN=1，seq=u

ROUND 2:

服务器端回送一个确认报文段，客户到服务器这个方向的连接就释放了——半关闭状态。

ACK=1，seq=v，ack=u+1

ROUND 3:

服务器端发完数据，就发出连接释放报文段，主动关闭TCP连接。

FIN=1，ACK=1，seq=w，ack=u+1

ROUND 4:

客户端回送一个确认报文段，再等到时间等待计时器设置的2MSL（最长报文段寿命）后，连接彻底关闭。

ACK=1，seq=u+1，ack=w+1

TCP可靠传输

传输层：使用TCP实现可靠传输。
网络层：提供尽最大努力交付，不可靠传输。

可靠：保证接收方进程从缓存区读出的字节流与发送方发出的字节流是完全一样的。

TCP实现可靠传输的机制：校验、序号、确认、重传。

校验

与UDP校验一样，增加伪首部。

序号

一个字节占一个序号。
序号字段指的是一个报文段第一个字节的序号。

确认

TCP默认使用累计确认。

重传

超时重传：TCP的发送方在规定的时间(重传时间)内没有收到确认就要重传已发送的报文段。

TCP采用自适应算法，动态改变重传时间RTTs(加权平均往返时间)。

冗余ACK(冗余确认)
每当比期望序号大的失序报文段到达时，发送一个**冗余ACK**，指明下一个期待字节的序号。
发送方已发送1，2，3，4，5报文段：

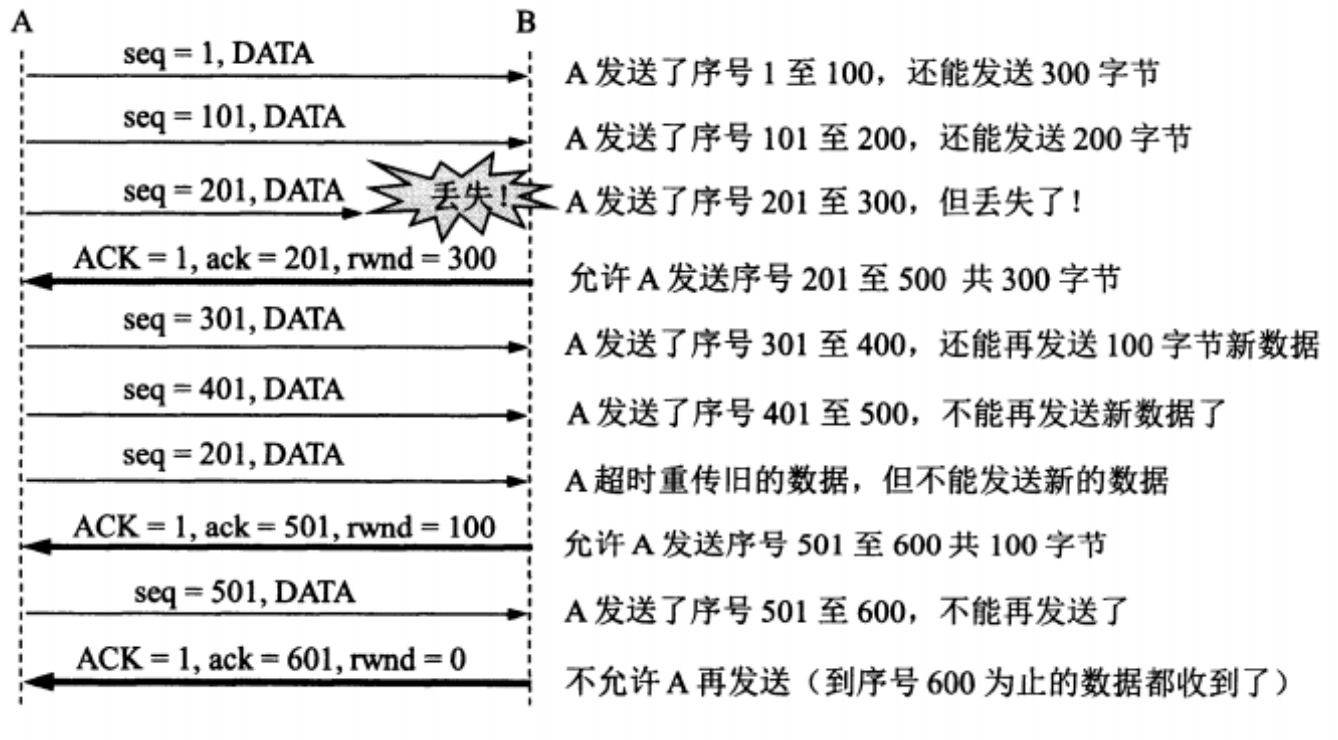
接收方收到1，返回给1的确认(确认号为2的第一个字节)
接收方收到3，仍返回给1的确认(确认号为2的第一个字节)
接收方收到4，仍返回给1的确认(确认号为2的第一个字节)
接收方收到5，仍返回给1的确认(确认号为2的第一个字节)
发送方收到3个对于报文段1的冗余ACK，认为2报文段丢失，重传2号报文段(快速重传)。

TCP流量控制

流量控制：让发送方慢点，要让接收方来得及接收。

TCP利用**滑动窗口**机制实现流量控制。

在通信过程中，接收方根据自己**接收缓存的大小**，**动态地**调整发送方的发送窗口大小，即接收窗口rwnd(接收方设置确认报文段的**窗口字段**来将rwnd通知给发送方)，发送方的发送窗口取**接收窗口rwnd和拥塞窗口cwnd的最小值**。



TCP为每一个连接设有一个持续计时器，只要TCP连接的一方收到对方的零窗口通知，就启动持续计时器。
若持续计时器设置的时间到期，就发送一个零窗口探测报文段。接收方收到探测报文段时给出现在的窗口值。
若窗口仍然是0，那么发送方就重新设置持续计时器。

TCP阻塞控制

为什么出现：对资源需求的总和 > 可用资源

网络中有许多资源同时呈现供应不足 -> 网络性能变坏 -> 网络吞吐量将随输入负荷增大而下降

目的：防止过多的数据注入到网络中。(全局性)

流量控制是点到点。(发送过快，来不及接收)

假定：

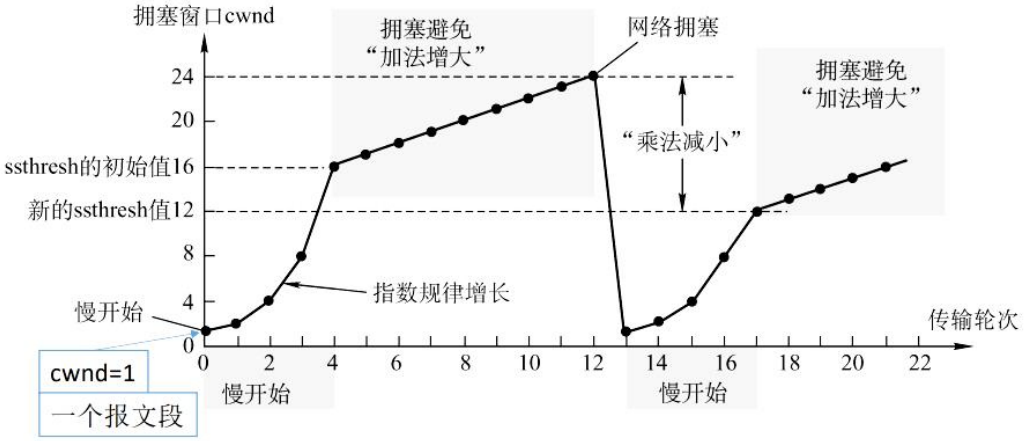
1.数据单方向传送，而另一个方向只传送确认

2.接收方总是有足够大的缓存空间，因而发送窗口大小取决于拥塞程度

发送窗口=Min{接收窗口rwnd，拥塞窗口cwnd}

接收窗口	**接收方**根据接受缓存设置的值，并告知给发送方，反映接收方容量。
拥塞窗口	**发送方**根据自己估算的网络拥塞程度而设置的窗口值，反映网络当前容量。

慢开始和拥塞避免



一个最大报文段长度MSS

一个**传输轮次**：

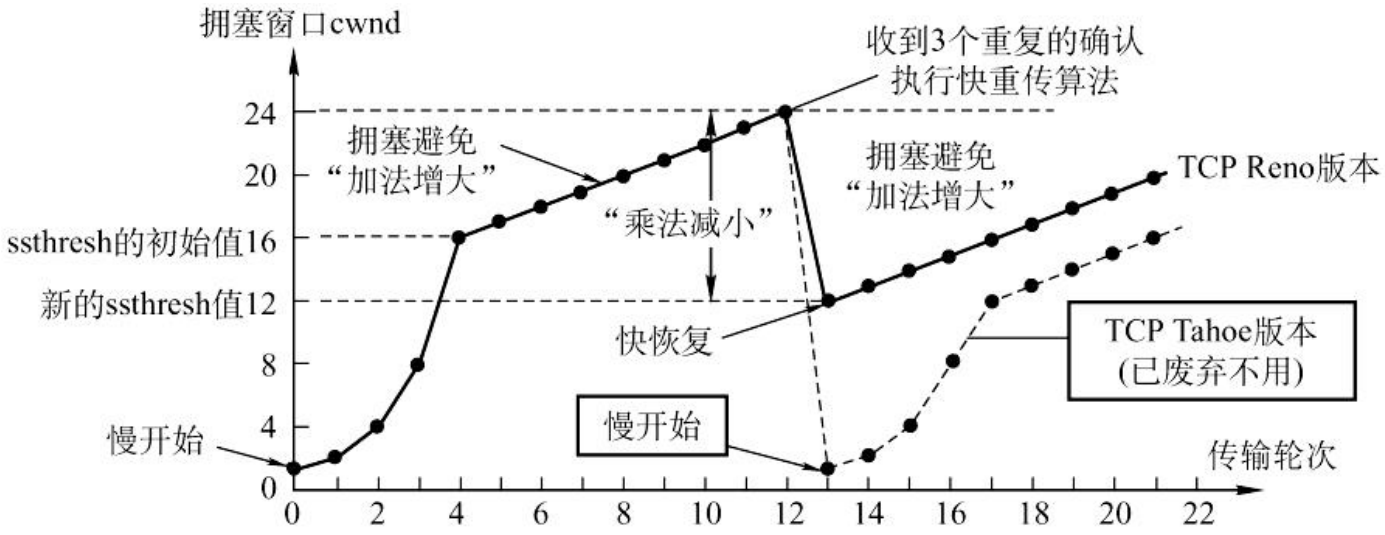
发送了一批报文段并收到它们的确认的时间。

一个往返时延RTT。

开始发送一批拥塞窗口内的报文段到开始发送下一批拥塞窗口内的报文段的时间。

新的ssthresh值是上一次网络拥塞的窗口大小除以2。

快重传和快恢复



收到3个冗余ACK快重传。

6 应用层

应用层对应用程序的通信提供服务。

应用层协议定义：

- 应用进程交换的报文类型，请求还是响应?
- 各种报文类型的语法，如报文中的各个字段及其详细描述。
- 字段的语义，即包含在字段中的信息的含义。
- 进程何时、如何发送报文，以及对报文进行响应的规则。

网络应用模型

客户/服务器模型(Client/Server)

服务器：提供计算服务的设备。

1. 永久提供服务
2. 永久性访问地址/域名

客户机：请求计算服务的主机。

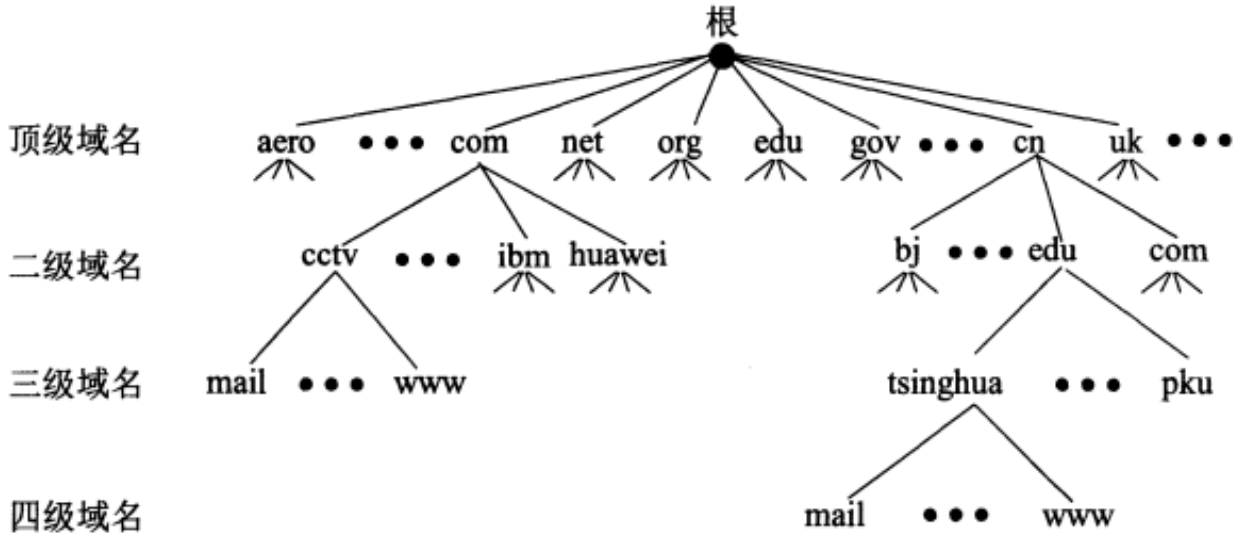
1. 与服务器通信，使用服务器提供的服务
2. 间歇性接入网络
3. 可能使用动态IP地址
4. 不与其他客户机直接通信

P2P模型(Peer to Peer)

- 不存在永远在线的服务器
- 每个主机既可以**提供服务**，也可以**请求服务**
- 任意端系统/节点之间可以**直接通讯**
- 节点间歇性接入网络
- 节点可能改变IP地址
- 可扩展性好
- 网络健壮性强

域名解析系统DNS

域名

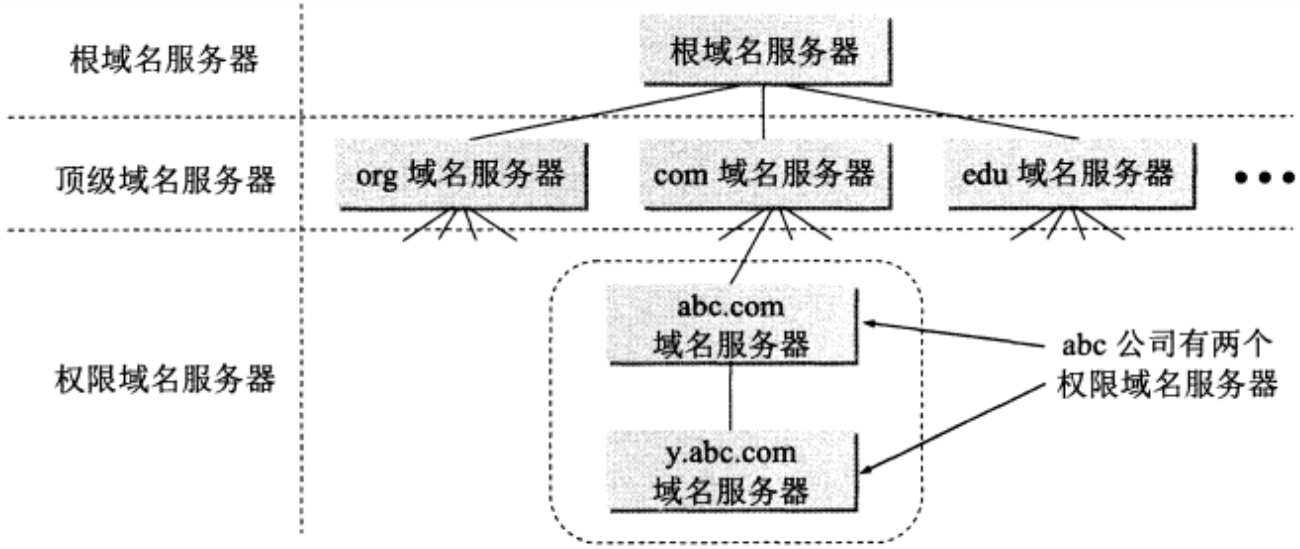


本地域名服务器：当一个主机发出DNS查询请求时，这个查询请求报文就发给本地域名服务器。

根域名服务器：是互联网域名解析系统(DNS)中最高级别的域名服务器，负责返回顶级域名的权威域名服务器的地址。截至2014年10月，全球有504台根服务器，被编号为A到M共13个标号。

顶级域名服务器：管理该顶级域名服务器注册的所有二级域名。

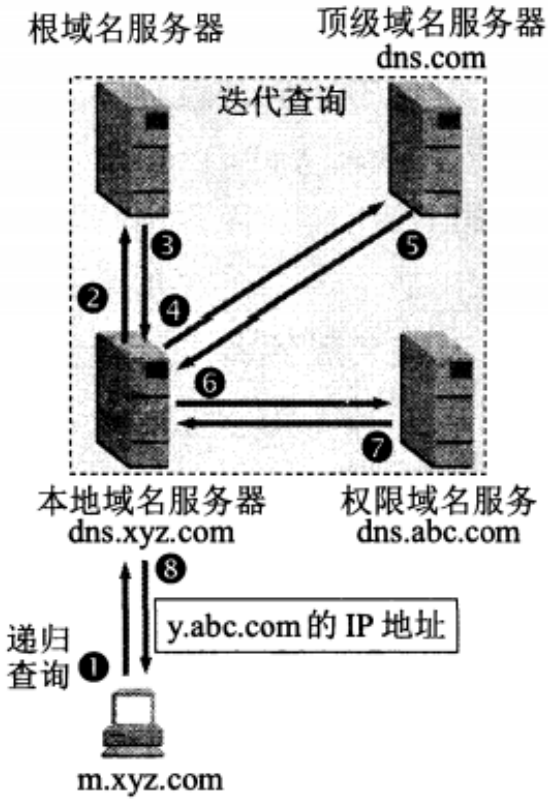
权限域名服务器：负责一个区的域名服务器。如果查询结果为空，则通知发起请求的NDS用户应到哪个权限域名服务器进一步查询。



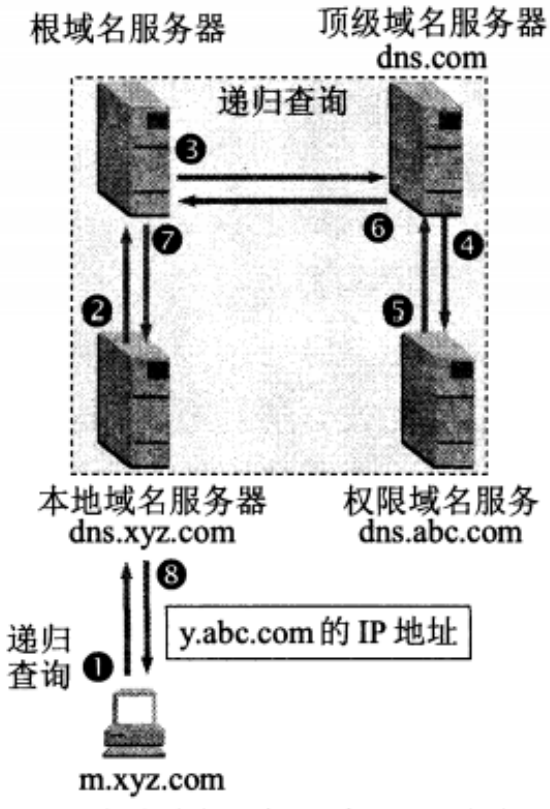
域名解析过程

第一，主机向本地域名服务器的查询一般都是采用**递归查询**(recursive query)。所谓递归查询就是：如果主机所询问的本地域名服务器不知道被查询域名的 IP 地址，那么本地域名服务器就以 DNS 客户的身份，向其他根域名服务器继续发出查询请求报文（即替该主机继续查询），而不是让该主机自己进行下一步的查询。因此，递归查询返回的查询结果或者是所要查询的 IP 地址，或者是报错，表示无法查询到所需的 IP 地址。

第二，本地域名服务器向根域名服务器的查询通常是采用**迭代查询**(iterative query)。迭代查询的特点是这样的：当根域名服务器收到本地域名服务器发出的迭代查询请求报文时，要么给出所要查询的 IP 地址，要么告诉本地域名服务器：“你下一步应当向哪一个域名服务器进行查询”。然后让本地域名服务器进行后续的查询（而不是替本地域名服务器进行后续的查询）。根域名服务器通常是把自己知道的顶级域名服务器的 IP 地址告诉本地域名服务器，让本地域名服务器再向顶级域名服务器查询。顶级域名服务器在收到本地域名服务器的查询请求后，要么给出所要查询的 IP 地址，要么告诉本地域名服务器下一步应当向哪一个权限域名服务器进行查询，本地域名服务器就这样进行迭代查询。最后，知道了所要解析的域名的 IP 地址，然后把这个结果返回给发起查询的主机。当然，本地域名服务器也可以采用递归查询，这取决于最初的查询请求报文的设置是要求使用哪一种查询方式。



(a) 本地域名服务器采用迭代查询



(b)本地域名服务器采用递归查询

文件传输协议FTP

文件传送协议FTP(File Transfer Protocol)：提供不同种类主机系统(硬、软件体系等都可以不同)之间的文件传输能力。

简单文件传送协议TFTP(Trivial File Transfer Protocol)

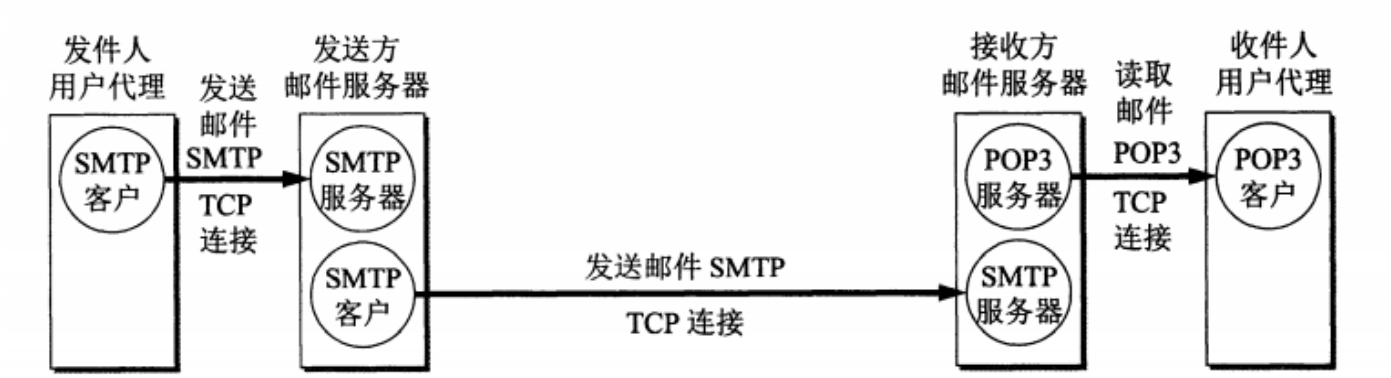
- FTP是基于客户/服务器(C/S)的协议。
- 用户通过一个客户机程序连接至在远程计算机上运行的服务器程序。
- 依照FTP协议提供服务，进行文件传送的计算机就是**FTP服务器**。
- 连接FTP服务器，遵循FTP协议与服务器传送文件的电脑就是**FTP客户端**。
- FTP使用TCP实现可靠传输。

FTP传输模式

- 文本模式：ASCII模式，以文本序列传输数据。
- 二进制模式：Binary模式，以二进制序列传输数据。

FTP有两种传输模式：主动(FTP Port)模式和被动(FTP Passive)模式。由于主动模式存在着安全问题，最近几年，大部分的TFP客户端开始默认使用被动模式。

电子邮件



简单邮件传送协议SMTP

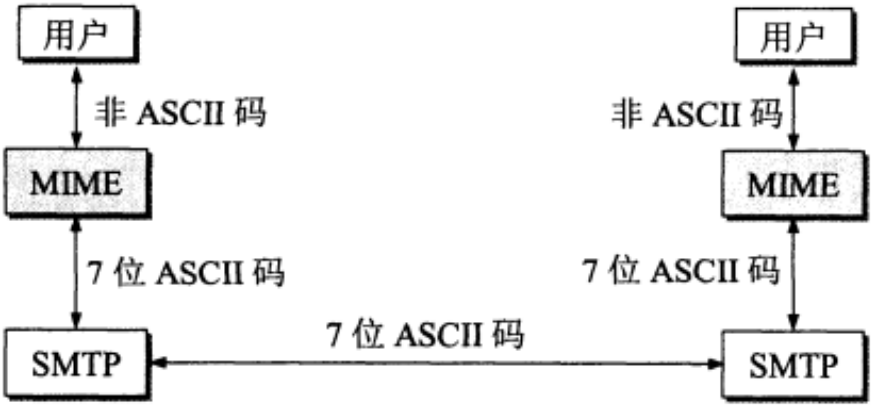
- SMTP规定了在两个相互通信的**SMTP进程**之间应如何交换信息。
- 负责发送邮件的SMTP进程就是**SMTP客户**，负责接收邮件的进程就是SMTP服务器。
- SMTP规定了14条命令(几个字母)和21种应答信息(三位数字代码+简单文字说明)。
- TCP连接 端口号25 C/S
- SMTP通信阶段：连接建立 邮件传送 连接释放

命令	参数	状态	描述
HELO	<domin>	连接/开始	客户端发送此命令与SMTP服务器建立连接，将发送者邮件地址发送给SMTP服务器
AUTH LOGIN		认证	
MAIL	FROM:<reverse-path>	处理	客户端将邮件发送者的名称传送给SMTP服务器
RCPT	TO:<forward-path>	处理	客户端将邮件接收者的名称传送给SMTP服务器
DATA		处理	客户端将邮件报文内容传送给SMTP服务器
SEND	FROM:<reverse-path>	处理	用于向指定用户传送邮件
SAML/SOML	FROM:<reverse-path>	处理	用于发送邮件
RSET		处理	取消客户端与SMTP服务器间的当前事务，释放与当前事务相关的内存
EXPN	<string>	处理	标识邮件接收者列表
QUIT		更新/结束	终止客户端与SMTP服务器间的连接

通用因特网邮件扩充MINE

STMP缺点：

1. SMTP不能传送可执行文件或者其他二进制对象。
2. SMTP仅限于传送7位ASCII码，不能传送其他非英语国家的文字。
3. SMTP服务器会拒绝超过一定长度的邮件。



使电子邮件系统可以支持声音、图像、视频、多种国家语言等。

邮局协议POP3

POP3是第一个离线协议标准，用户登录客户端设置POP3，可以将邮件从服务器存储到本地计算机，同时删除服务器上的邮件。POP3服务器是遵循POP3协议的接受邮件服务器，用于接收电子邮件。

POP3工作方式：下载并保留(在服务器)，下载并删除。

TCP连接 端口号110 C/S

网际报文存取协议IMAP

IMAP协议比POP协议复杂。当用户PC上的IMAP客户程序打开IMAP服务器的邮箱时，用户可以看到邮箱的首部，若用户需要打开某个邮件，该邮件才上传到用户的计算机上。

IMAP可以让用户在不同的地方使用不同的计算机随时上网阅读处理邮件，还允许只读取邮件中的某一个部分(先看正文，有WiFi的时候再下载附件)。

POP3和IMAP比较

操作位置	操作内容	IMAP	POP3
收件箱	阅读、标记、移动、删除邮件等	客户端与邮箱更新同步	仅在客户端内
发件箱	保存到已发送	客户端与邮箱更新同步	仅在客户端内
创建文件夹	新建自定义的文件夹	客户端与邮箱更新同步	仅在客户端内
草稿	保存草稿	客户端与邮箱更新同步	仅在客户端内
垃圾文件夹	接收并移入垃圾文件夹的邮件	支持	不支持
广告邮件	接收并移入广告邮件夹的邮件	支持	不支持

基于万维网的电子邮件

用户在浏览器中浏览各种信息时需要使用HTTP协议。因此，在浏览器和互联网上的邮件服务器之间传送邮件时，仍然使用HTTP协议。但是在各邮件服务器之间传送邮件时，则仍然使用SMTP协议。

万维网

万维网WWW(World Wide Web)是一个大规模的、联机式的信息储藏所/资料空间，是无数个网络站点和网页的集合。

统一资源定位符URL一般形式：

< 协议 >: // < 主机 >:< 端口 > / < 路径 >

URL不区分大小写。

超文本传输协议HTTP

HTTP协议定义了浏览器(万维网客户进程)怎样向万维网服务器请求万维网文档，以及服务器怎样把文档传送给浏览器。

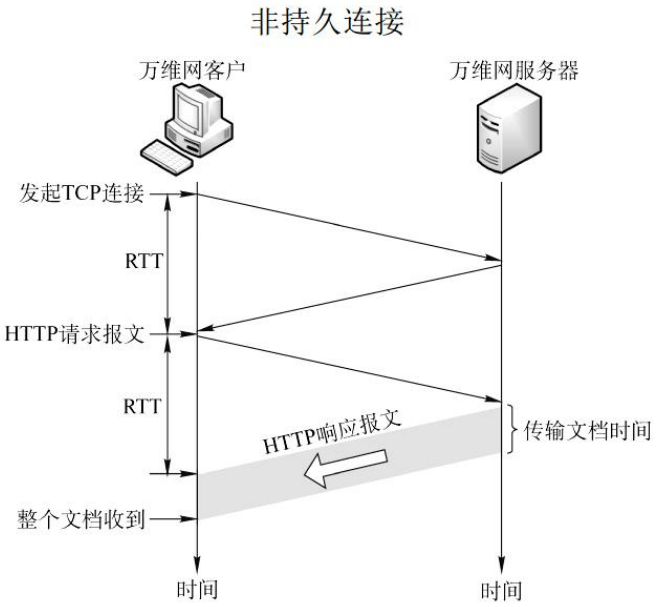
具体过程

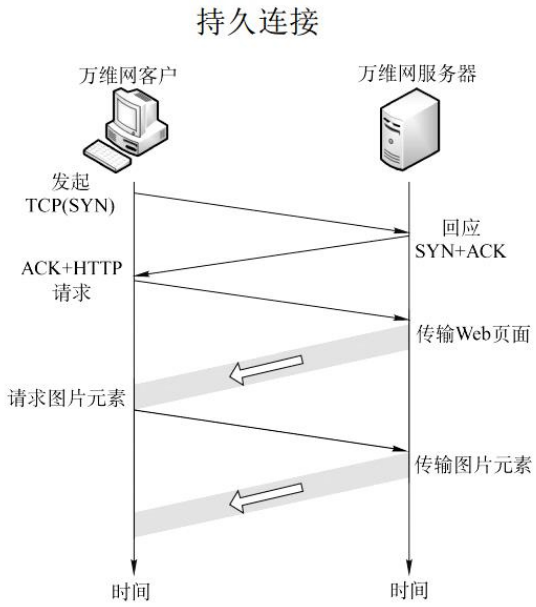
1. 浏览器分析URL
2. 浏览器向DNS请求解析IP地址
3. DNS解析出IP地址
4. 浏览器与服务器建立TCP连接
5. 浏览器发出取文件命令
6. 服务器响应
7. 释放TCP连接
8. 浏览器显示

特点

- HTTP协议是无状态的。
- Cookie是存储在用户主机中的文本文件，记录一段时间内某用户的访问记录。
- HTTP采用TCP作为运输层协议，但**HTTP协议本身是无连接**的(通信双方在交换HTTP报文之前不需要先建立HTTP连接)。

连接方式



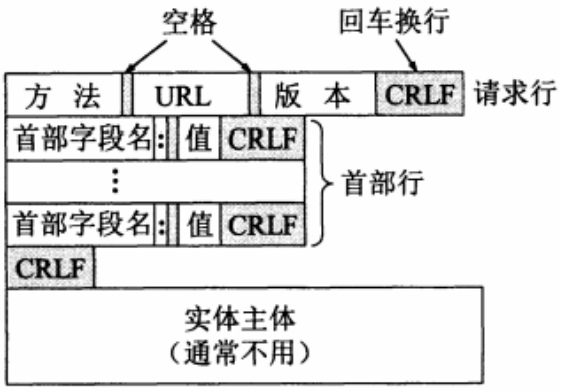


报文结构

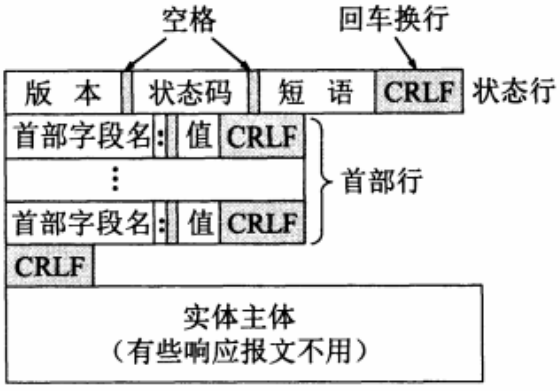
HTTP 有两类报文：

(1) 请求报文——从客户向服务器发送请求报文，见图 6-12(a)。

(2) 响应报文——从服务器到客户的回答，见图 6-12(b)。



(a) 请求报文



(b) 响应报文

下面是 HTTP 的请求报文的开始行（即请求行）的格式。请注意，在 GET 后面有一个空格，接着是某个完整的 URL，其后面又有一个空格，最后是 HTTP/1.1。

```
GET http://www.xyz.edu.cn/dir/index.htm HTTP/1.1
```

下面是一个完整的 HTTP 请求报文的例子：

```
GET /dir/index.htm HTTP/1.1                    {请求行使用了相对 URL}
Host: www.xyz.edu.cn                {此行是首部行的开始。这行给出主机的域名}
Connection: close                   {告诉服务器发送完请求的文档后就可释放连接}
User-Agent: Mozilla/5.0             {表明用户代理是使用火狐浏览器 Firefox}
Accept-Language: cn                 {表示用户希望优先得到中文版本的文档}
                                    {请求报文的最后还有一个空行}
```

HTTP状态码

当浏览者访问一个网页时，浏览者的浏览器会向网页所在服务器发出请求。当浏览器接收并显示网页前，此网页所在的服务器会返回一个包含HTTP状态码的信息头(server header)用以响应浏览器的请求。

HTTP状态码的英文为HTTP Status Code。

下面是常见的HTTP状态码：

200 - 请求成功
301 - 资源(网页等)被永久转移到其它URL
404 - 请求的资源(网页等)不存在
500 - 内部服务器错误

分类	分类描述
1**	信息，服务器收到请求，需要请求者继续执行操作
2**	成功，操作被成功接收并处理
3**	重定向，需要进一步的操作以完成请求
4**	客户端错误，请求包含语法错误或无法完成请求
5**	服务器错误，服务器在处理请求的过程中发生了错误

参考：

- https://leetcode-cn.com/circle/discuss/8Oi6Ma/
- https://www.jianshu.com/p/51f90f5046ca
- https://www.runoob.com/http/http-status-codes.html