计算机考研复试面试常问问题 计算机网络篇

第一章、计算机网络体系结构

快速唤起记忆知识框架

1.计算机网络的主要功能?

> **1、硬件资源共享。**

> 可以在全网范围内提供对处理资源、存储资源、输入输出资源等昂贵设备的共享，使用户节省投资，也便于集中管理和均衡分担负荷。

> 2、**软件资源共享。**

> 允许互联网上的用户远程访问各类大弄数据库，可以得到网络文件传送服务、远地进程管理服务和远程文件访问服务，从而避免软件研制上的重复劳动以及数据资源的重复存贮，也便于集中管理。

> 3、**用户间信息交换。**

> 计算机网络为分布在各地的用户提供了强有力的通信手段。用户可以通过计算机网络传送电子邮件、发布新闻消息和进行电子商务活动。

> 4、**分布式处理**

> 当计算机网络中某个计算机系统负荷过重时，可以将其处理的某个复杂任务分配给网络中的其他计算机系统，从而利用空闲计算机资源以提高整个系统的利用率。

2.主机间的通信方式?

> **客户-服务器（C/S）**：客户是服务的请求方，服务器是服务的提供方。

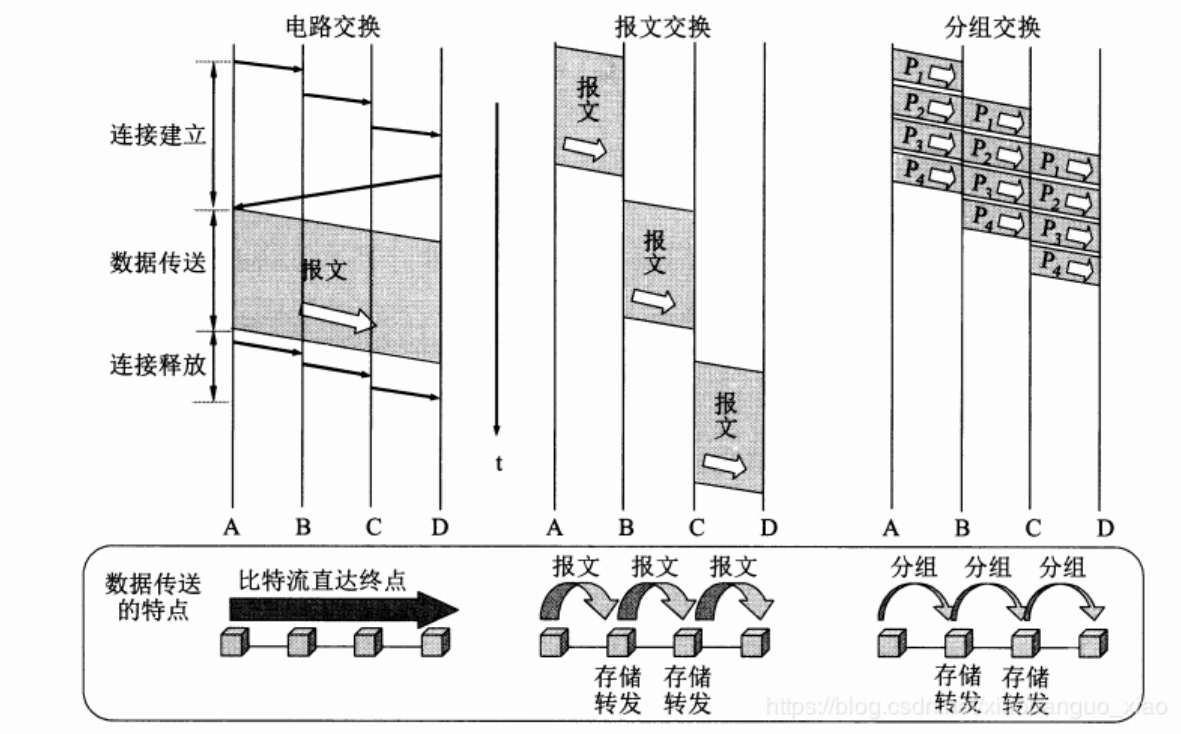
> **对等（P2P）**：不区分客户和服务器。

3.电路交换，报文交换和分组交换的区别?

> **电路交换**：整个报文的比特流从源点连续的直达终点，像在一个管道中传输。包括建立连接、传输数据和断开连接三个阶段。最典型的电路交换网络是传统电话网络。

> **报文交换**：将整个报文转发到相邻节点，全部存储下来，查找转发表，转发到下一个节点。是**存储-转发**类型的网络。

> **分组交换**：将报文分组转发到相邻节点，查找转发表，转发到下一个节点。也是存储-转发类型的网络。



4.计算机网络的主要性能指标?

> **1、带宽（Bandwidth）**

> 本来表示通信线路允许通过的信号频带范围，但在计算机网络中，带宽表示网络的通信线路所能传送数据的能力，是数字信道所能传送的“**最高数据率**”的同义词，单位是比特/秒（b/s)。

> **2、时延（Delay）**

> 总时延 = 排队时延 + 处理时延 + 传输时延 + 传播时延
>
> (1) 排队时延
>
> 分组在路由器的输入队列和输出队列中排队等待的时间，取决于网络当前的通信量。
>
> (2) 处理时延
>
> 主机或路由器收到分组时进行处理所需要的时间，例如分析首部、从分组中提取数据、进行差错检验或查找适当的路由等。
>
> (3) 传输时延(发送时延)
>
> 结点将分组所有比特推向链路所需的时间。

(4) 传播时延

电磁波在信道中传播所需要花费的时间，电磁波传播的速度接近光速。

3、时延带宽积

指发送端发送的第一个比特即将到达终点时，发送端已经发送了多少个比特，因此又称以比特为单位的链路长度，即时延带宽积 = 传播时延 * 信道带宽。

5.计算机网络提供的服务的三种分类?

1、面向连接服务与无连接服务

在面向连接服务中， 通信前双方必须先建立连接， 分配相应的资源（如缓冲区）， 以保证通信能正常进行， 传输结束后释放连接和所占用的资源。因此这种服务可以分为连接建立、数据传输和连接释放三个阶段。例如TCP就是一种面向连接服务的协议。

在无连接服务中， 通信前双方不需要先建立连接， 需要发送数据时可直接发送，把每个带有目的地址的包（报文分组） 传送到线路上， 由系统选定路线进行传输。这是一种不可靠的服务。这种服务常被描述为“尽最大努力交付" (Best-Effort-Delivery), 它并不保证通信的可靠性。例如IP、UDP就是一种无连接服务的协议。

2、可靠服务和不可靠服务

可靠服务是指网络具有纠错、检错、应答机制， 能保证数据正确、可靠地传送到目的地。
不可靠服务是指网络只是尽晕正确、可靠地传送， 而不能保证数据正确、可靠地传送到目的地， 是一种尽力而为的服务。

对于提供不可靠服务的网络， 其网络的正确性、可靠性要由应用或用户来保障。例如， 用户收到信息后要判断信息的正确性， 如果不正确， 那么用户要把出错信息报告给信息的发送者，以
便发送者采取纠正措施。通过用户的这些措施， 可以把不可靠的服务变成可靠的服务。

3、有应答服务和无应答服务

有应答服务是指接收方在收到数据后向发送方给出相应的应答，该应答由传输系统内部自动实现， 而不由用户实现。所发送的应答既可以是肯定应答， 也可以是否定应答， 通常在接收到的
数据有错误时发送否定应答。例如， 文件传输服务就是一种有应答服务。

无应答服务是指接收方收到数据后不自动给出应答。若需要应答， 则由高层实现。例如， 对于WWW服务， 客户端收到服务器发送的页面文件后不给出应答。

6.ISO/OSI参考模型和TCP/IP模型?

1、参考图片

2、五层协议

应用层： 为特定应用程序提供数据传输服务，例如 HTTP、DNS 等协议。数据单位为报文。

传输层： 为进程提供通用数据传输服务。由于应用层协议很多，定义通用的传输层协议就可以支持不断增多的应用层协议。运输层包括两种协议：传输控制协议 TCP，提供面向连接、可靠的数据传输服务，数据单位为报文段；用户数据报协议 UDP，提供无连接、尽最大努力的数据传输服务，数据单位为用户数据报。TCP 主要提供完整性服务，UDP 主要提供及时性服务。（流量控制、差错控制、服务质量、数据传输管理、端到端）

网络层： 为主机提供数据传输服务。而传输层协议是为主机中的进程提供数据传输服务。网络层把传输层传递下来的报文段或者用户数据报封装成分组。（流量控制、拥塞控制、差错控制、网际互联）

数据链路层： 网络层针对的还是主机之间的数据传输服务，而主机之间可以有很多链路，链路层协议就是为同一链路的主机提供数据传输服务。数据链路层把网络层传下来的分组封装成帧。（封装成帧、差错控制、流量控制、传输管理）

物理层： 考虑的是怎样在传输媒体上传输数据比特流，而不是指具体的传输媒体。物理层的作用是尽可能屏蔽传输媒体和通信手段的差异，使数据链路层感觉不到这些差异。

3、OSI

其中表示层和会话层用途如下：

表示层： 数据压缩、加密以及数据描述，这使得应用程序不必关心在各台主机中数据内部格式不同的问题。

会话层： 建立及管理会话。

五层协议没有表示层和会话层，而是将这些功能留给应用程序开发者处理。

4、TCP/IP

它只有四层，相当于五层协议中数据链路层和物理层合并为网络接口层。

TCP/IP 体系结构不严格遵循 OSI 分层概念，应用层可能会直接使用 IP 层或者网络接口层。

7.端到端通信和点到点通信的区别?

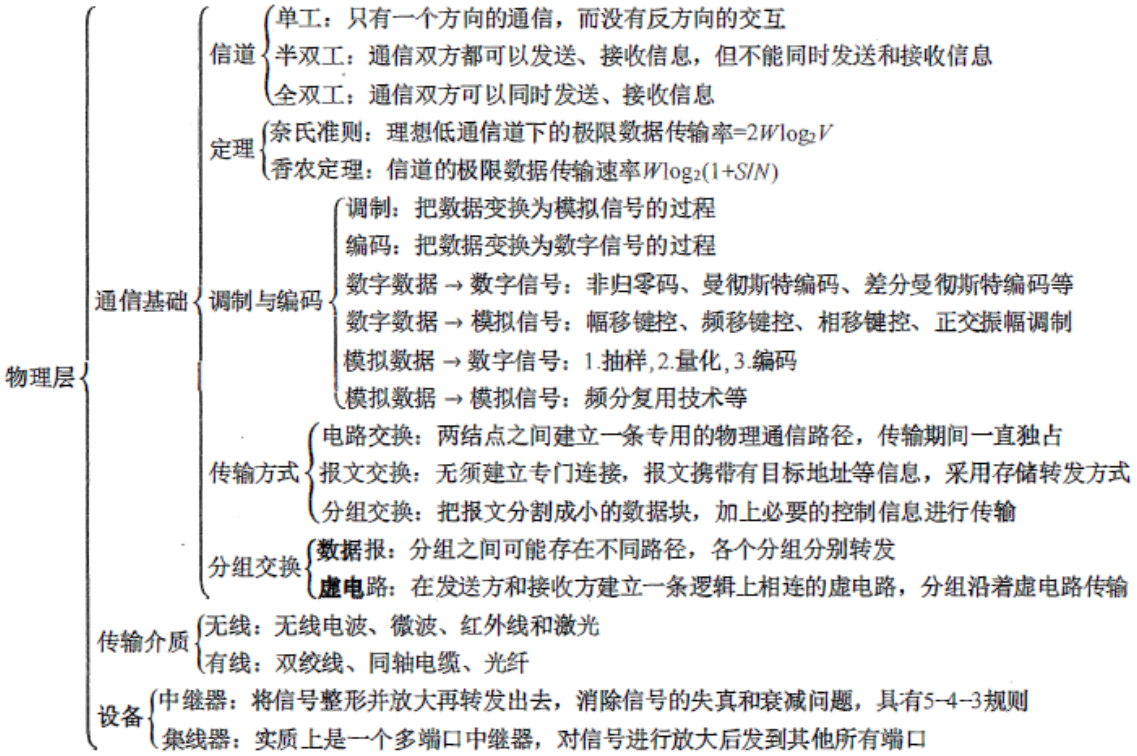
从本质上说，由物理层、数据链路层和网络层组成的通信子网为网络环境中的主机提供点到点的服务， 而传输层为网络中的主机提供端到端的通信。

直接相连的结点之间的通信称为点到点通信， 它只提供一台机器到另一台机器之间的通信，不涉及程序或进程的概念。同时，点到点通信并不能保证数据传输的可靠性，也不能说明源主机与目的主机之间是哪两个进程在通信，这些工作都是由传输层来完成的。

端到端通信建立在点到点通信的基础上，它是由一段段的点到点通信信道构成的，是比点到点通信更高一级的通信方式，以完成应用程序（进程） 之间的通信。”端” 是指用户程序的端口，端口号标识了应用层中不同的进程。

第二章、物理层

快速唤起记忆知识框架



8.如何理解同步和异步？什么是同步通信和异步通信?

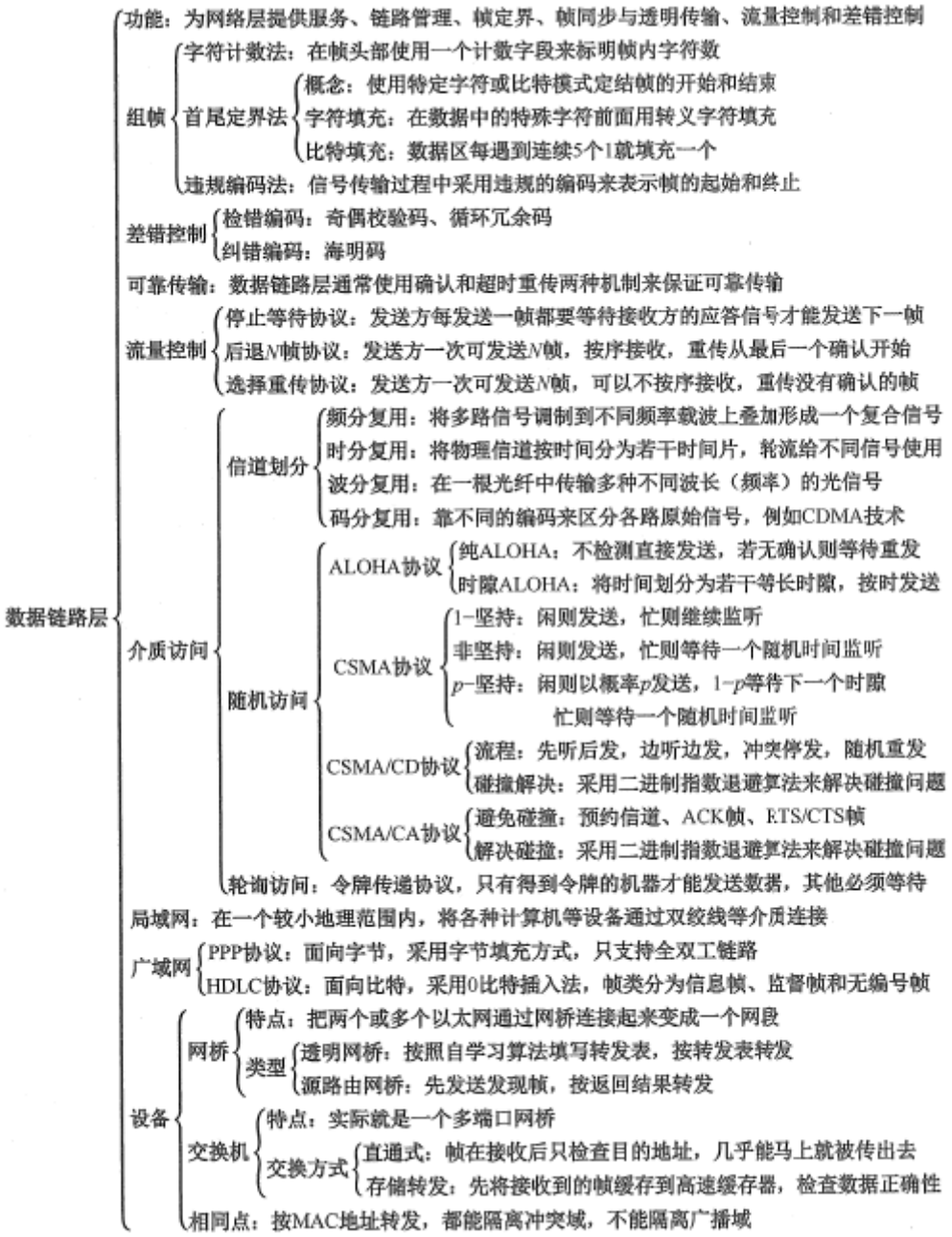
> 在计算机网络中，同步(Synchronous)的意思很广泛，没有统一的定义。例如，协议的三个要素之一就是“ 同步”。在网络编程中常提到的“同步” 则主要指某函数的执行方式， 即函数调用者需等待函数执行完后才能进入下一步。异步(Asynchronous)可简单地理解为“ 非同步”。在数据通信中，同步通信与异步通信区别较大。
>
> 同步通信的通信双方必须先建立同步， 即双方的时钟要调整到同一个频率。收发双方不停地发送和接收连续的同步比特流。主要有两种同步方式： 一种是全网同步， 即用一个非常精确的主时钟对全网所有结点上的时钟进行同步；另一种是准同步， 即各结点的时钟之间允许有微小的误差，然后采用其他措施实现同步传输。同步通信数据率较高，但实现的代价也较高。
>
> 异步通信在发送字符时，所发送的字符之间的时间间隔可以是任意的，但接收端必须时刻做好接收的准备。发送端可以在任意时刻开始发送字符， 因此必须在每个字符开始和结束的地方加上标志， 即开始位和停止位，以便使接收端能够正确地将每个字符接收下来。异步通信也可以帧作为发送的单位。这时，帧的首部和尾部必须设有一些特殊的比特组合，使得接收端能够找出一帧的开始（即帧定界）。异步通信的通信设备简单、便宜，但传输效率较低（因为标志的开销所占比例较大）。

9.频分复用 时分复用 波分复用 码分复用

> 频分复用：给每个信号分配唯一的载波频率并通过单一媒体来传输多个独立信号的方法。
>
> 时分复用：把多个信号复用到单个硬件传输信道，它允许每个信号在一个很短的时间使用信道，接着再让下一个信号使用。
>
> 波分复用：就是光的频分复用。用一根光纤同时传输多个频率很接近的光载波信号。
>
> 码分复用：码分复用是用一组包含互相正交的码字的码组携带多路信号。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，各用户之间不会造成干扰，因此这种系统发送的信号有很强的抗干扰能力。

第三章、数据链路层

快速唤起记忆知识框架



10.为什么要进行流量控制?

> 由于接收发双方各自的工作速率和缓存空间的差异，可能出现发送方的发送能力大于接收方的接收能力的现象，如若此时不适当限制发送方的发送速率（即链路上的信息流量），前而来不及接收的帧将会被后面不断发送辈的帧 " 淹没",造成帧的丢失而出错。
>
> 因此流量控制实际上就是限制发送方的数据流量，使其发送速率不超过接收方的接收能力。这个过程需要通过某种反馈机制使发送方能够知道接收方是否能跟上自己，即需要有一些规则使得发送方知道在什么情况下可以接着发送下一帧而在什么情况下必须暂停发送，以等待收到某种反馈信息后继续发送。

11.流量控制的常见方式?

> 1、**停止-等待流量控制方式基本原理(发送窗口大小=1，接受窗口大小=1)**

> 发送方每发送一帧，**都要等待接收方的应答信号，之后才能发送下一帧**；**接收方每接收一帧，都要反馈一个应答信号**，表示可接收下一帧，如果接收方不反馈应答信号，那么**发送方必须一直等待。**每次只允许发送一帧，然后就陷入等待接收方确认信息的过程中，因而传输效率很低。

> 2、**滑动窗口流量控制方式基本原理**

> 在任意时刻，发送方都维持一组连续的允许发送的帧的序号，称为**发送窗口**；同时接收方也维持一组连续的允许接收帧的序号，称为**接收窗口。**发送窗口用来对发送方进行流量控制，而发送窗口的大小代表在还未收到对方确认信息的情况下发送方最多还可以发送多少个数据帧。同理，在接收端设置接收窗口是为了控制可以接收哪些数据帧和不可以接收哪些帧。在接收方，只有收到的数据帧的序号落入接收窗口内时，才允许将该数据帧收下。若接收到的数据帧落在接收窗口之外，则一律将其丢弃。

> 3、**后退N帧协议（GBN）（发送窗口大小>1,接收窗口大小=1）**

> 在后退N 帧式ARQ 中，发送方无须在收到上一个帧的ACK 后才能开始发送下一帧，而是可以连续发送帧。当接收方检测出失序的信息帧后，要求发送方重发最后一个正确接收的信息帧之后的所有未被确认的帧；或者当发送方发送了N 个帧后，若发现该N 个帧的前一个帧在计时器超时后仍未返回其确认信息，则该帧被判为出错或丢失，此时发送方就不得不重传该出错帧及随后的N 个帧。换句话说，**接收方只允许按顺序接收帧。**（接收窗口大小=1则按序接收）
>
> 后退N 帧协议一方面因连续发送数据帧而提高了信道的利用率，另一方面在重传时又必须把原来已传送正确的数据帧进行重传（仅因这些数据帧的前面有一个数据帧出了错），这种做法又使传送效率降低。由此可见，若信道的传输质量很差导致误码率较大时，后退N 帧协议不一定优于停止 - 等待协议。

> 4、**选择重传协议（SR）（发送窗口大小>1,接收窗口大小>1）**

> 为进一K步提高信道的利用率，可设法只重传出现差错的数据帧或计时器超时的数据帧，但此时必须加大接收窗口，以便先收下发送序号不连续但仍处在接收窗口中的那些数据帧。等到所缺序号的数据帧收到后再一并送交主机。这就是选择重传ARQ 协议。
>
> 在选择重传协议中，每个发送缓冲区对应一个计时器，当计时器超时时，缓冲区的帧就会重传。另外，该协议使用了比上述其他协议更有效的差错处理策略，即一旦接收方怀疑帧出错，就会发一个否定帧NAK给发送方，要求发送方对NAK中指定的帧进行重传。

12.可靠传输机制有哪些?

> 数据链路层的可靠传输通常使用确认和超时重传两种机制来完成。

> 确认是一种无数据的控制帧，这种控制帧使得接收方可以让发送方知道哪些内容被正确接收。有些情况下为了提高传输效率，将确认捎带在一个回复帧中，称为捎带确认。超时重传是指发送方在发送某个数据帧后就开启一个计时器，在一定时间内如果没有得到发送的数据帧的确认帧，那么就重新发送该数据帧，直到发送成功为止。

> 自动重传请求(Auto Repeat reQuest， ARQ) 通过接收方请求发送方重传出错的数据帧来恢复出错的帧，是通信中用于处理信道所带来差错的方法之一。传统自动重传请求分为三种，即停止 - 等待(Stop-and-Wait) ARQ 、后退N 帧(Go-Back-N) ARQ 和选择性重传(Selective Repeat)ARQ,后两种协议是滑动窗口技术与请求重发技术的结合，由于窗口尺寸开到足够大时，帧在线路上可以连续地流动，因此又称其为连续ARQ 协议。注意，在数据链路层中流量控制机制和可靠传输机制是交织在一起的。

13.随机访问介质访问控制?

> 在随机访问协议中，不采用集中控制方式(信道划分介质访问--时分复用）解决发送信息的次序问题，所有用户能根据自己的意愿随机地发送信息，占用信道全部速率。在总线形网络中，当有两个或多个用户同时发送信息时，就会产生帧的冲突（碰撞，即前面所说的相互干扰），导致所有冲突用户的发送均以失败告终。为了解决随机接入发生的碰撞，每个用户需要按照一定的规则反复地重传它的帧，直到该帧无碰撞地通过。A/这些规则就是随机访问介质访问控制协议，常用的协议有**ALOHA 协议、CSMA协议、CSMA/CD 协议和CSMA/CA 协议等**，它们的核心思想都是：胜利者通过争用获得信道，从而获得信息的发送权。

> 1、**ALOHA协议**

> ALOHA协议的思想很简单，只要用户有数据要发送，就尽管让他们发送。当然，这样会产生冲突从而造成帧的破坏。但是，由于广播信道具有反馈性，因此发送方可以在发送数据的过程中进行冲突检测，将接收到的数据与缓冲区的数据进行比较，就可以知道数据帧是否遭到破坏。同样的道理，其他用户也是按照此过程工作。如果发送方知道数据帧遭到破坏（即检测到冲突），那么它可以等待一段随机长的时间后重发该帧。

> 2、**CSMA协议(载波侦听多路访问）(Carrier Sense Multiple Access)**

> 非持续式：
>
> 经侦听，如果介质空闲，开始发送
> 如果介质忙，则等待一个随机分布的时间，然后重复步骤1
>
> 优点：等待一个随机时间可以减少再次碰撞冲突的可能性
> 缺点：如果在这个随机时间内介质上没有数据传送，则会发生浪费
>
> 1-持续式：
>
> 经侦听，如介质空闲，开始发送
> 如介质忙，持续侦听，一旦空闲立即发送
> 如果发生冲突，等待一个随机分布的时间再重复步骤1
>
> 优点：持续式的延迟时间要少于非持续式
> 缺点：如果两个以上的站等待发送，一旦介质空闲就一定会发生冲突
>
> p-持续式：
>
> 经侦听，如介质空闲，那么以p的概率发送，以(1–p)的概率延迟一个时间单元发送
> 如介质忙，持续侦听，一旦空闲重复步骤1
> 如果发送已推迟一个时间单元，再重复步骤1

> 3、**CSMA/CD协议（Collision Detection:碰撞检测）**

> 载波侦听多路访问 / 碰撞检测(Carrier Sense Multiple Access with Collision Detection, CSMA/CD)
> 协议是CSMA 协议的改进方案。”载波帧听”就是发送前先侦听，即每个站在发送数据之前先要检测一下总线上是否有其他站点正在发送数据，若有则暂时不发送数据，等待信道变为空闲时再发送。”碰撞检测”就是边发送边侦听，即适配器边发送数据边检测信道上信号电压的变化情况，以便判断自己在发送数据时其他站点是否也在发送数据。工作流程可简单概括为“先听后发，边听边发（区别于CSMA 协议），冲突停发，随机重发”。
>
> 1) 适配器从其父结点获得一个网络层数据报，准备一个以太网帧，并把该帧放到适配器缓冲区中。
>
> 2) 如果适配器侦听到信道空闲，那么它开始传输该帧。如果适配器侦听到信道忙，那么它将等待直至侦听到没有信号能量，然后开始传输该帧。

3) 在传输过程中，适配器检测来自其他适配器的信号能量。如果这个适配器传输了整个帧，而没有检测到来自其他适配器的信号能量，那么这个适配器完成该帧的传输。否则，适配器就须停止传输它的帧，取而代之传输一个48 比特的拥塞信号。

4) 在中止（即传输拥塞信号）后，适配器采用截断二进制指数退避算法等待一段随机时间后返回到步骤2) 。

4、**CSMA/CA协议（Collision Avoidance:碰撞避免）**

CSMA/CD 协议已成功应用千使用有线连接的局域网，但在无线局域网环境下，却不能简单地搬用CSMA/CD 协议，特别是碰撞检测部分。主要有两个原因：

1) 接收信号的强度往往会远小于发送信号的强度，且在无线介质上信号强度的动态变化范围很大，因此若要实现碰撞检测，则硬件上的花费就会过大。

2) 在无线通信中，并非所有的站点都能够听见对方，即存在“隐蔽站”问题。

为此， 802.11 标准定义了广泛应用于无线局域网的CSMA/CA 协议，它对CSMA/CD 协议进行了修改，把碰撞检测改为碰撞避免(Collision Avoidance, CA) 。”碰撞避免”并不是指协议可以完全避免碰撞，而是指协议的设计要尽量降低碰撞发生的概率。

CSMA/CA 采用二进制指数退避算法。信道从忙态变为空困时，任何一个站要发送数据帧时，不仅都须等待一个时间间隔，而且还要进入争用窗口，并计算随机退避时间以便再次试图接入信道，因此降低了发生碰撞的概率。
CSMA/CA 还使用预约信道、ACK 帧、RTS/CTS 帧等三种机制来实现碰撞避免：
1) 预约信道。发送方在发送数据的同时向其他站点通知自己传输数据需要的时间长度，以便让其他站点在这段时间内不发送数据，从而避免碰撞。

2) ACK 帧。所有站点在正确接收到发给自己的数据帧（除广播帧和组播帧）后，都需要向发送方发回一个ACK 帧，如果接收失败，那么不采取任何行动。发送方在发送完一个数据帧后，在规定的时间内如果未收到ACK 帧，那么认为发送失败，此时进行该数据帧的重发，直到收到ACK 帧或达到规定重发次数为止。

3) RTS/CTS 帧。可选的碰撞避免机制，主要用于解决无线网中的“隐蔽站”问题。

14.PPP协议?

点到点协议（Point to Point Protocol，PPP）是为在同等单元之间传输数据包这样的简单链路设计的链路层协议。这种链路提供全双工操作，并按照顺序传递数据包。设计目的主要是用来通过拨号或专线方式建立点对点连接发送数据，使其成为各种主机、网桥和路由器之间简单连接的一种共通的解决方案。PPP具有以下功能：

（1）PPP具有动态分配IP地址的能力，允许在连接时刻协商IP地址；

（2）PPP支持多种网络协议，比如TCP/IP、NetBEUI、NWLINK等；

（3）PPP具有错误检测能力，但不具备纠错能力，所以ppp是不可靠传输协议；

（4）无重传的机制，网络开销小，速度快。

（5）PPP具有身份验证功能。

（6） PPP可以用于多种类型的物理介质上，包括串口线、电话线、移动电话和光纤（例如SDH），PPP也用于Internet接入。

15.HDLC协议?

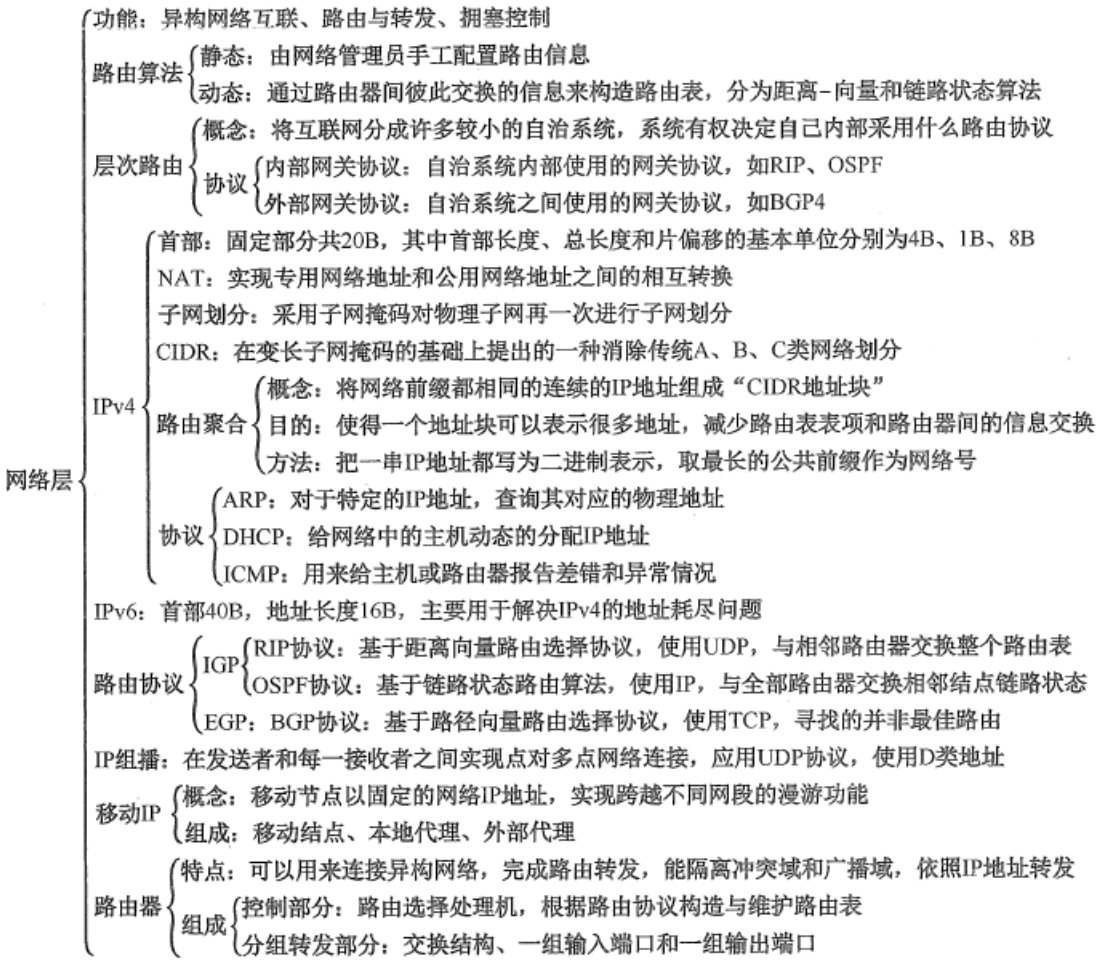
> HDLC协议使用统一的帧格式，运用方便；采用零比特插入法，易于硬件实现，且支持任意的位流传输，实现信息的透明传输；全双工通信，吞吐率高，在未收到应答帧的情况下，可连续发送信息帧，提高数据链路传输的效率；采用CRC帧校验序列，可防止漏帧，提高信息传输的可靠性。
>
> 主要有四个特点：
>
> （1）对于任何一种比特流都可透明传输。
>
> （2）较高的数据链路传输效率。
>
> （3）所有的帧都有帧校验序列（FCS），传输可靠性高。
>
> （4）用统一的帧格式来实现传输。

16.试分析中继器、集线器、网桥和交换机这四种网络互联设备的区别与联系。

> 这四种设备都是用于互联、扩展局域网的连接设备，但它们工作的层次和实现的功能不同。
>
> 中继器工作在物理层，用来连接两个速率相同且数据链路层协议也相同的网段，其功能是消除数字信号在基带传输中由于经过一长段电缆而造成的失真和衰减，使信号的波形和强度达到所需的要求；其原理是信号再生。
>
> 集线器(Hub) 也工作在物理层，相当于一个多接口的中继器，它可将多个结点连接成一个共享式的局域网，但任何时刻都只能有一个结点通过公共信道发送数据。
>
> 网桥工作在数据链路层，可以互联不同的物理层、不同的MAC 子层及不同速率的以太网。网桥具有过滤帧及存储转发帧的功能，可以隔离冲突域，但不能隔离广播域。
>
> 交换机工作在数据链路层，相当于一个多端口的网桥，是交换式局域网的核心设备。它允许端口之间建立多个并发连接，实现多个结点之间的并发传输。因此，交换机的每个端口结点所占用的带宽不会因为端口结点数目的增加而减少，且整个交换机的总带宽会随着端口结点的增加而增加。交换机一般工作在全双工方式，有的局域网交换机采用存储转发方式进行转发，也有的交换机采用直通交换方式（即在收到帧的同时立即按帧的目的MAC 地址决定该帧的转发端口，而不必先缓存再处理）。

第四章、网络层

快速唤起记忆知识框架：



17.路由器的主要功能?

> 路由器主要完成两个功能：一是路由选择（确定哪一条路径），二是分组转发（当一个分组到达时所采取的动作）。前者是根据特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。后者处理通过路由器的数据流，关键操作是转发表查询、转发及相关的队列管理和任务调度等。
>
> 1) 路由选择。指按照复杂的分布式算法，根据从各相邻路由器所得到的关于整个网络拓扑的变化情况，动态地改变所选择的路由。
>
> 2) 分组转发。指路由器根据转发表将用户的IP 数据报从合适的端口转发出去。路由表是根据路由选择算法得出的，而转发表是从路由表得出的。转发表的结构应当使查找过程最优化，路由表则需要对网络拓扑变化的计算最优化。在讨论路由选择的原理时，往往不去区分转发表和路由表，而是笼统地使用路由表一词。

18.动态路由算法?

> **1、距离-向量路由算法（例如RIP算法）**

> 在距离 - 向量路由算法中，所有结点都定期地将它们的整个路由选择表传送给所有与之直接相邻的结点。这种路由选择表包含：1.每条路径的目的地（另一结点）。2.路径的代价（也称距离）。
>
> 在这种算法中，所有结点都必须参与距离向量交换，以保证路由的有效性和一致性，也就是说，所有的结点都监听从其他结点传来的路由选择更新信息，并在下列情况下更新它们的路由选择表：
>
> 1) 被通告一条新的路由，该路由在本结点的路由表中不存在，此时本地系统加入这条新的路由。

2) 发来的路由信息中有一条到达某个目的地的路由，该路由与当前使用的路由相比，有较短的距离（较小的代价）。此种情况下，就用经过发送路由信息的结点的新路由替换路由表中到达那个目的地的现有路由。

2、**链路状态路由算法（例如OSPF算法）**

链路状态路由算法要求每个参与该算法的结点都具有完全的网络拓扑信息，它们执行下述两项任务。第一，主动测试所有邻接结点的状态。两个共享一条链接的结点是相邻结点，它们连接到同一条链路，或者连接到同一广播型物理网络。第二，定期地将链路状态传播给所有其他结点（或称路由结点）

距离 - 向量路由算法与链路状态路由算法的比较：在距离 - 向量路由算法中，每个结点仅与它的直接邻居交谈，它为它的邻居提供从自已到网络中所有其他结点的最低费用估计。在链路状态路由算法中，每个结点通过广播的方式与所有其他结点交谈，但它仅告诉它们与它直接相连的链路的费用。相较之下，距离～向量路由算法有可能遇到路由环路等问题。

3、一个自治系统内部所使用的路由选择协议称为内部网关协议(IGP), 也称域内路由选择，具体的协议有RIP 和OSPF 等。

路由信息协议(Routing Information Protocol, RIP) 是内部网关协议IGP) 中最先得到广泛应用的协议。RIP 是一种分布式的基于距离向量的路由选择协议，其最大优点就是简单。

RIP 规定：

1) 网络中的每个路由器都要维护从它自身到其他每个目的网络的距离记录（因此这是一组距离，称为距离向量）。

2) 距离也称跳数(Hop Count), 规定从一个路由器到直接连接网络的距离（跳数）为1 。而每经过一个路由器，距离（跳数）加1 。

3) RIP 认为好的路由就是它通过的路由器的数目少，即优先选择跳数少的路径。

4) RIP 允许一条路径最多只能包含15 个路由器（即最多允许15 跳）。因此距离等于16 时，它表示网络不可达。可见RIP 只适用于小型互联网。距离向量路由可能会出现环路的情况，规定路径上的最高跳数的目的是为了防止数据报不断循环在环路上，减少网络拥塞的可能性。

5) RIP 默认在任意两个使用RIP 的路由器之间每30 秒广播一次RIP 路由更新信息，以便自动建立并维护路由表（动态维护）。

开放最短路径优先(OSPF) 协议是使用分布式链路状态路由算法的典型代表，也是内部网关协议(IGP) 的一种。OSPF 与RIP 相比有以下4 点主要区别：

1) OSPF 向本自治系统中的所有路由器发送信息，这里使用的方法是洪泛法。而RIP 仅向自已相邻的几个路由器发送信息。

2) 发送的信息是与本路由器相邻的所有路由器的链路状态，但这只是路由器所知道的部分信息。”链路状态”说明本路由器和哪些路由器相邻及该链路的“度量”（或代价）。而在RIP 中，发送的信息是本路由器所知道的全部信息，即整个路由表。

3) 只有当链路状态发生变化时，路由器才用洪泛法向所有路由器发送此信息，并且更新过程收敛得快，不会出现RIP" 坏消息传得慢 " 的问题。而在RIP 中，不管网络拓扑是否发生变化，路由器之间都会定期交换路由表的信息。

除以上区别外， OSPF 还有以下特点：

1) OSPF 对不同的链路可根据IP 分组的不同服务类型(TOS) 而设置成不同的代价。因此，OSPF 对千不同类型的业务可计算出不同的路由，十分灵活。

2) 如果到同一个目的网络有多条相同代价的路径，那么可以将通信量分配给这几条路径。这称为多路径间的负载平衡。
3) 所有在OSPF 路由器之间交换的分组都具有鉴别功能，因而保证了仅在可信赖的路由器之间交换链路状态信息。

4、 自治系统之间所使用的路由选择协议称为外部网关协议(EGP), 也称域间路由选择，用在不同自治系统的路由器之间交换路由信息，并负责为分组在不同自治系统之间选择最优的路径。具体的协议有BGP 。

边界网关协议(Border Gateway Protocol, BGP) 是不同自治系统的路由器之间交换路由信息的协议，是一种外部网关协议。边界网关协议常用于互联网的网关之间。路由表包含已知路由器的列表、路由器能够达到的地址及到达每个路由器的路径的跳数。内部网关协议主要设法使数据报在一个AS 中尽可能有效地从源站传送到目的站。在一个AS内部不需要考虑其他方面的策略。然而BGP 使用的环境却不同，主要原因如下：

1) 因特网的规模太大，使得自治系统之间路由选择非常困难。
2) 对于自治系统之间的路由选择，要寻找最佳路由是很不现实的。
3) 自治系统之间的路由选择必须考虑有关策略。

边界网关协议(BGP) 只能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），而并非寻找一条最佳路由。BGP 采用的是路径向量路由选择协议，它与距离向量协议和链路状态协议有很大的区别。BGP 是应用层协议，它是基于TCP 的。

BGP 的工作原理如下：每个自治系统的管理员要选择至少一个路由器（可以有多个）作为该自治系统的"BGP 发言人"。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（可见BGP 报文是通过TCP 传送的，也就是说BGP 报文是TCP 报文的数据部分），然后在此连接上交换BGP 报文以建立BGP 会话，再利用BGP 会话交换路由信息。当所有BGP 发言人都相互交换网络可达性的信息后，各BGP 发言人就可找出到达各个自治系统的较好路由。

19.网络层转发分组的流程?

1) 从数据报的首部提取目的主机的IP 地址D, 得出目的网络地址N。

2) 若网络N 与此路由器直接相连，则把数据报直接交付给目的主机D, 这称为路由器的直接交付；否则是间接交付，执行步骤3) 。

3) 若路由表中有目的地址为D 的特定主机路由（对特定的目的主机指明一个特定的路由，通常是为了控制或测试网络，或出千安全考虑才采用的），则把数据报传送给路由表中所指明的下一跳路由器；否则执行步骤4)

4) 若路由表中有到达网络N 的路由，则把数据报传送给路由表指明的下一跳路由器；否则，执行步骤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，执行步骤6) 。

6) 报告转发分组出错。

注意：得到下一跳路由器的IP 地址后并不是直接将该地址填入待发送的数据报，而是将该IP 地址转换成MAC 地址（通过ARP），将其放到MAC 帧首部中，然后根据这个MAC 地址找到下一跳路由器。在不同网络中传送时， MAC 帧中的源地址和目的地址要发生变化，但是网桥在转发帧时，不改变帧的源地址，请注意区分。

20.IP地址和MAC地址?

IP 地址是网络层使用的地址，它是分层次等级的。MAC地址是数据链路层使用的地址，它是平面式的。在网络层及网络层之上使用IP 地址， IP 地址放在IP 数据报的首部，而MAC 地址放在MAC 帧的首部。通过数据封装，把IP 数据报分组封装为MAC 帧后，数据链路层看不见数据报分组中的IP地址。

由于路由器的隔离， IP 网络中无法通过广播方式依靠MAC 地址来完成跨网络的寻址，因此在IP 网络的网络层只使用IP 地址来完成寻址。寻址时，IP每个路由器依据其路由表（依靠静态路由或动态路由协议生成）选择到目标网络（即主机号全为0 的网络地址）需要转发到的下一跳（路由器的物理端口号或下一网络地址），而IP 分组通过多次路由转发到达目标网络后，改为在目标LAN 中通过数据链路层的MAC 地址以广播方式寻址。这样可以提高路由选择的效率。

注意：路由器由于互联多个网络，因此它不仅有多个IP 地址，也有多个硬件地址。

21.ARP地址解析协议?

无论网络层使用什么协议，在实际网络的链路上传送数据帧时，最终必须使用硬件地址。所以需要一种方法来完成IP 地址到MAC 地址的映射，这就是地址解析协议(Address Resolution Protocol）。每台主机都设有一个ARP 高速缓存，用来存放本局域网上各主机和路由器的IP地址到MAC 地址的映射表，称ARP 表。使用ARP 来动态维护此ARP 表。

ARP 工作在网络层，其工作原理如下：主机A 欲向本局域网上的某台主机B 发送IP 数据报时，先在其ARP 高速缓存中查看有无主机B 的IP 地址。如有，就可查出其对应的硬件地址，再将此硬件地址写入MAC 帧，然后通过局域网将该MAC 帧发往此硬件地址。如果没有，那么就通过使用目的MAC 地址为FF-FF-FF-FF-FF-FF 的帧来封装并广播ARP 请求分组，使同一个局域网里的所有主机收到ARP 请求。主机B 收到该ARP 请求后，向主机A 发出响应ARP 分组，分组中包含主机B 的IP 与MAC 地址的映射关系，主机A 在收到后将此映射写入ARP 缓存，然后按查询到的硬件地址发送MAC 帧。ARP 由于“看到了"IP 地址，所以它工作在网络层，而NAT路由器由于“看到了“端口，所以它工作在传输层。

注意： ARP 用于解决同一个局域网上的主机或路由器的IP 地址和硬件地址的映射问题。如果所要找的主机和源主机不在同一个局域网上，那么就要通过ARP 找到一个位于本局域网上的某个路由器的硬件地址，然后把分组发送给这个路由器，让这个路由器把分组转发给下一个网络。剩下的工作就由下一个网络来做，尽管ARP 请求分组是广播发送的，但ARP 响应分组是普通的单播，即从一个源地址发送到一个目的地址。

22.DHCP动态主机配置协议?

动态主机配置协议(Dynamic Host Configuration Protocol, DHCP) 常用于给主机动态地分配IP 地址，它提供了即插即用联网的机制，这种机制允许一台计算机加入新的网络和获取IP 地址而不用手工参与。DHCP 是应用层协议，它是基于UDP 的。

DHCP 的工作原理如下：使用客户 / 服务器方式。需要IP 地址的主机在启动时就向DHCP 服务器广播发送发现报文，这时该主机就成为DHCP 客户。本地网络上所有主机都能收到此广播报文，但只有DHCP 服务器才回答此广播报文。DHCP 服务器先在其数据库中查找该计算机的配置信息。若找到，则返回找到的信息。若找不到，则从服务器的IP 地址池中取一个地址分配给该计算机。DHCP 服务器的回答报文称为提供报文。
DHCP 服务器聚合DHCP 客户端的交换过程如下：

1) DHCP 客户机广播"DHCP 发现“消息，试图找到网络中的DHCP 服务器，以便从DHCP服务器获得一个IP 地址。

2) DHCP 服务器收到"DHCP 发现消息后，向网络中广播"DHCP 提供“消息，其中包括提供DHCP 客户机的IP 地址和相关配置信息。

3) DHCP 客户机收到"DHCP 提供“消息，如果接收DHCP 服务器所提供的相关参数，那么通过广播"DHCP 请求“消息向DHCP 服务器请求提供IP 地址。

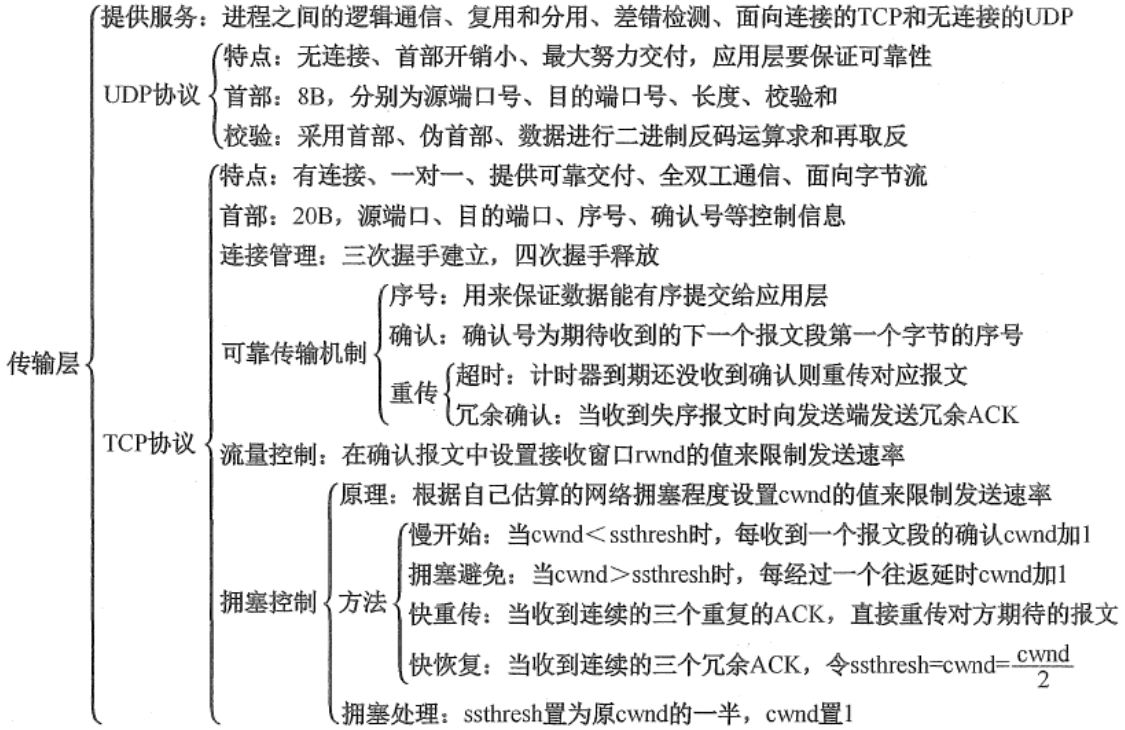
> 4) DHCP 服务器广播"DHCP 确认"消息，将IP 地址分配给DHCP 客户机。DHCP 允许网络上配置多台DHCP 服务器，当DHCP 客户机发出DHCP 请求时，有可能收到多个应答消息。这时，DHCP 客户机只会挑选其中的一个，通常挑选最先到达的。
>
> DHCP 服务器分配给DHCP 客户的IP 地址是临时的，因此DHCP 客户只能在一段有限的时间内使用这个分配到的IP 地址。DHCP 称这段时间为租用期。租用期的数值应由DHCP 服务器自己决定， DHCP 客户也可在自己发送的报文中提出对租用期的要求。

23.ICMP网际控制报文协议?

> 为了提高IP 数据报交付成功的机会，在网络层使用了网际控制报文协议(Internet Control Message Protocol, ICMP) 来让主机或路由器报告差错和异常情况。ICMP 报文作为IP 层数据报的数据，加上数据报的首部，组成IP 数据报发送出去。ICMP 是IP 层协议。ICMP 报文的种类有两种，即ICMP 差错报告报文和ICMP 询问报文。ICMP 差错报告报文用于目标主机或到目标主机路径上的路由器向源主机报告差错和异常情况。共有以下5种类型：
>
> 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。当路由器或主机由于拥塞而丢弃数据报时，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。
>
> 3) 时间超过。当路由器收到生存时间(TTL) 为零的数据报时，除丢弃该数据报外，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时，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）。

第五章、传输层

快速唤起记忆知识框架:



24.传输层的功能?

> 从通信和信息处理的角度看，传输层向它上面的应用层提供通信服务，它属千面向通信部分的最高层，同时也是用户功能中的最低层。传输层位于网络层之上，它为运行在不同主机上的进程之间提供了逻辑通信，而网络层提供主机之间的逻辑通信。显然，即使网络层协议不可靠（网络层协议使分组丢失、混乱或重复），
> 传输层同样能为应用程序提供可靠的服务。
>
> 传输层的功能如下：
>
> l) 传输层提供应用进程之间的逻辑通信（即端到端的通信）。与网络层的区别是，网络层提供的是主机之间的逻辑通信。从网络层来说，通信的双方是两台主机， IP 数据报的首部给出了这两台主机的IP 地址。但"两台主机之间的通信"实际上是两台主机中的应用进程之间的通信，应用进程之间的通信又称端到端的逻辑通信。
>
> 2) 复用和分用。复用是指发送方不同的应用进程都可使用同一个传输层协议传送数据；分用是指接收方的传输层在剥去报文的首部后能够把这些数据正确交付到目的应用进程。
>
> 3) 传输层还要对收到的报文进行差错检测（首部和数据部分）。而网络层只检查IP 数据报的首部，不检验数据部分是否出错。
>
> 4) 提供两种不同的传输协议，即面向连接的TCP 和无连接的UDP 。而网络层无法同时实现两种协议（即在网络层要么只提供面向连接的服务，如虚电路；要么只提供无连接服务，如数据报，而不可能在网络层同时存在这两种方式）。

25.UDP协议?

> RFC 768 定义的UDP 只是做了传输协议能够做的最少工作，它仅在IP 的数据报服务之上增加了两个最基本的服务：复用和分用以及差错检测。如果应用程序开发者选择UDP 而非TCP,那么应用程序几乎直接与IP 打交道。为什么应用开发人员宁愿在UDP 之上构建应用，也不选择TCP? 既然TCP 提供可靠的服务，而UDP 不提供，那么TCP 总是首选吗？答案是否定的，因为有很多应用更适合用UDP，主要是因为UDP 具有如下优点：

1) UDP 无须建立连接。因此UDP 不会引入建立连接的时延。试想如果DNS 运行在TCP 而非UDP 上，那么DNS 的速度会慢很多。HTTP 使用TCP 而非UDP, 是因为对于基于文本数据的Web网页来说可靠性是至关重要的。

2) 无连接状态。TCP 需要在端系统中维护连接状态。此连接状态包括接收和发送缓存、拥塞控制参数和序号与确认号的参数。而UDP 不维护连接状态，也不跟踪这些参数。因此，某些专用应用服务器使用UDP 时，一般都能支持更多的活动客户机
3) 分组首部开销小。TCP 有20B 的首部升销，而UDP 仅有8B 的升销。

4) 应用层能更好地控制要发送的数据和发送时间。UDP 没有拥塞控制，因此网络中的拥塞不会影响主机的发送效率。某些实时应用要求以稳定的速度发送，能容忍一些数据的丢失，但不允许有较大的时延，而UDP 正好满足这些应用的需求。UDP 常用于一次性传输较少数据的网络应用如DNS 、SNMP 等，因为对千这此应用，若采用TCP, 则将为连接创建、维护和拆除带来不小的开销。UDP 也常用于多媒体应用（如IP 电话、实时视频会议、流媒体等），显然，可靠数据传输对这些应用来说并不是最重要的，但TCP的拥塞控制会导致数据出现较大的延迟，这是它们不可容忍的。

UDP 提供尽最大努力的交付，即不保证可靠交付，但这并不意味着应用对数据的要求是不可靠的，因此所有维护传输可靠性的工作需要用户在应用层来完成。应用实体可以根据应用的需求来灵活设计自己的可靠性机制。

26.TCP协议?

TCP 是在不可靠的IP 层之上实现的可靠的数据传输协议，它主要解决传输的可靠、有序、无丢失和不重复问题。TCP 是TCP/IP 体系中非常复杂的一个协议，主要特点如下：

1) TCP 是面向连接的传输层协议。

2) 每条TCP 连接只能有两个端点，每条TCP 连接只能是点对点的（一对一）。

3) TCP 提供可靠的交付服务，保证传送的数据无差错、不丢失、不重复且有序。

4) TCP 提供全双工通信，允许通信双方的应用进程在任何时候都能发送数据，为此TCP 连接的两端都设有发送缓存和接收缓存，用来临时存放双向通信的数据。发送缓存用来暂时存放以下数据：(1)发送应用程序传送给发送方TCP 准备发送的数据；(2)TCP 已发送但尚未收到确认的数据。接收缓存用来暂时存放以下数据：(1)按序到达但尚未被接收应用程序收取的数据；（2）不按序到达的数据。

TCP连接的建立

在TCP 连接建立的过程中，要解决以下三个问题：

1) 要使每一方都能够确知对方的存在。

2) 要允许双方协商一些参数（如最大窗口值、是否使用窗口扩大选项、时间戳选项及服务质量等）。

3) 能够对运输实体资源（如缓存大小、连接表中的项目等）进行分配。

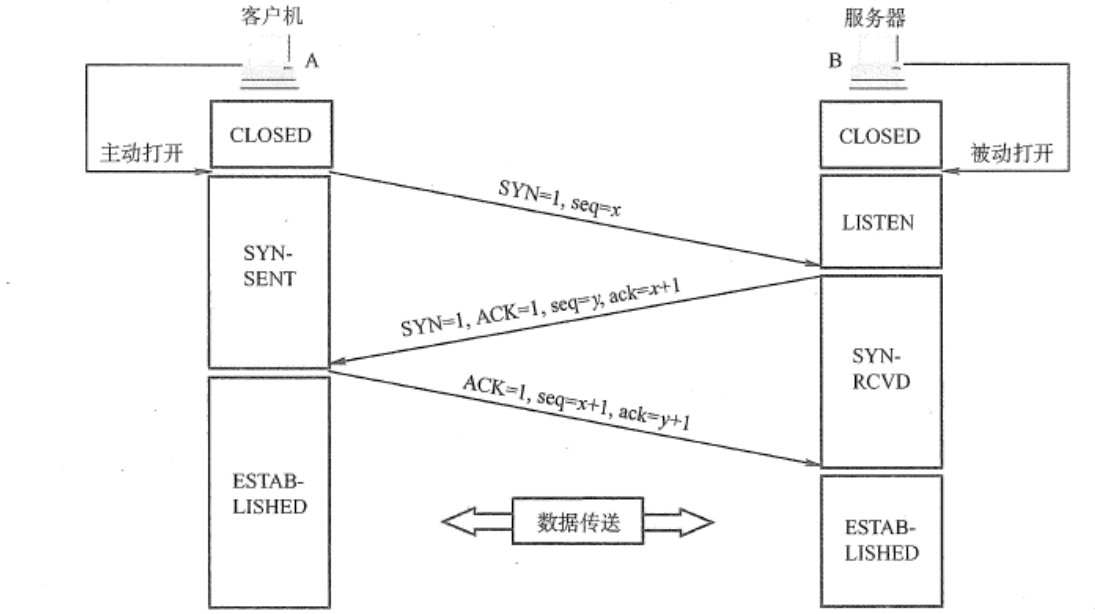
三次握手建立连接

第一步：客户机的TCP 首先向服务器的TCP 发送一个连接请求报文段。这个特殊的报文段中不含应用层数据，其首部中的SYN标志位被置为1 。另外，客户机会随机选择一个起始序号seq = x（连接请求报文不携带数据，但要消耗一个序号）。

第二步：服务器的TCP 收到连接请求报文段后，如同意建立连接，就向客户机发回确认，并为该TCP 连接分配TCP 缓存和变量。在确认报文段中， SYN 和ACK 位都被置为1, 确认号字段的值为x+ 1, 并且服务器随机产生起始序号seq= y(确认报文不携带数据，但也要消耗一个序号）。确认报文段同样不包含应用层数据。

第三步：当客户机收到确认报文段后，还要向服务器给出确认，并且也要给该连接分配缓存和变量。这个报文段的ACK 标志位被置1, 序号字段为x+ 1, 确认号字段ack=y+ 1 。该报文段可以携带数据，若不携带数据则不消耗序号。

成功进行以上三步后，就建立了TCP 连接，接下来就可以传送应用层数据。TCP 提供的是全双工通信，因此通信双方的应用进程在任何时候都能发送数据。另外，值得注意的是，服务器端的资源是在完成第二次握手时分配的，而客户端的资源是在完成第三次握手时分配的，这就使得服务器易于受到SYN 洪泛攻击。



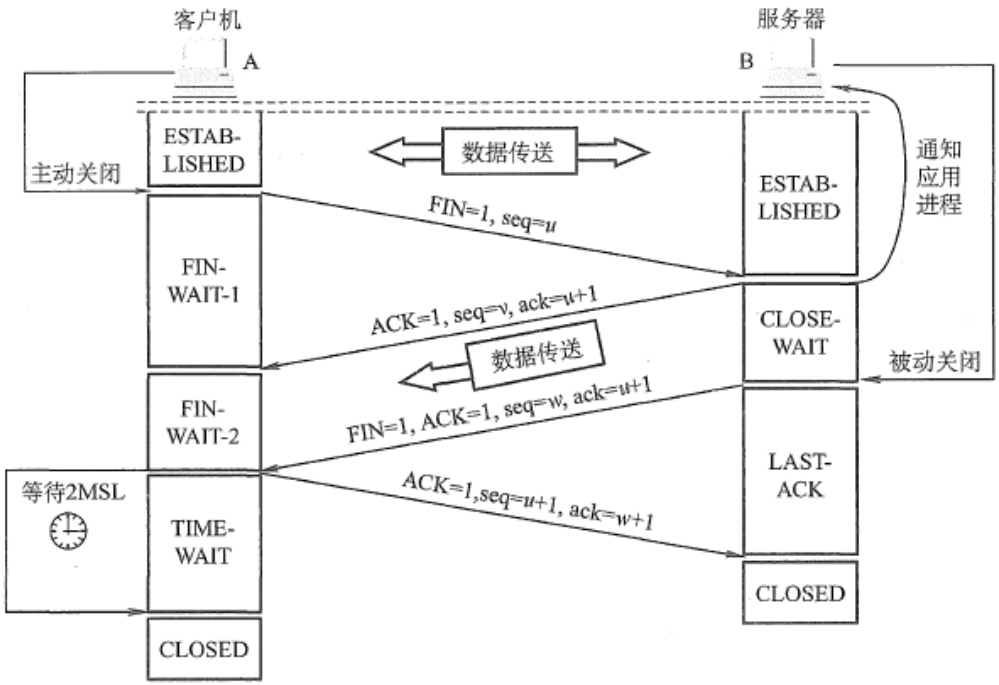
四次握手释放连接

第一步：客户机打算关闭连接时，向其TCP 发送一个连接释放报文段，并停止发送数据，主动关闭TCP 连接，该报文段的FIN 标志位被置1, seq= u, 它等于前面已传送过的数据的最后一个字节的序号加1 (FIN 报文段即使不携带数据，也要消耗一个序号）。TCP 是全双工的，即可以想象为一条TCP 连接上有两条数据通路。发送FIN 报文时，发送FIN 的一端不能再发送数据，即关闭了其中一条数据通路，但对方还可以发送数据。

第二步：服务器收到连接释放报文段后即发出确认，确认号是ack = u + 1, 而这个报文段自己的序号是V, 等于它前面已传送过的数据的最后一个字节的序号加1 。此时，从客户机到服务器这个方向的连接就释放了， TCP 连接处于半关闭状态。但服务器若发送数据，客户机仍要接收，即从服务器到客户机这个方向的连接并未关闭。

第三步：若服务器已经没有要向客户机发送的数据，就通知TCP 释放连接，此时其发出FIN= 1 的连接释放报文段。

第四步：客户机收到连接释放报文段后，必须发出确认。在确认报文段中， ACK 字段被置为1, 确认号ack= w + 1, 序号seq= u + 1 。此时TCP 连接还未释放，必须经过时间等待计时器设置的时间2MSL 后， A 才进入连接关闭状态。



对上述TCP 连接建立和释放的总结如下：

1) 连接建立。分为3 步：

1、SYN = 1, seq = x 。

2、SYN= 1, ACK= 1, seq= y, ack = x + 1 。

3、ACK= 1, seq= x + 1, ack = y + l 。

2) 释放连接。分为4 步：

1、FIN= 1, seq =u 。

2、ACK = 1, seq = v, ack = u + 1 。

3、FIN= 1, ACK= 1, seq= w, ack = u + 1 。

4、ACK= 1, seq= u + 1, ack = w + 1。

27.拥塞控制的四种算法?

所谓拥塞控制，是指防止过多的数据注入网络，保证网络中的路由器或链路小致过载。出现拥塞时，端点并不了解到拥塞发生的细节，对通信连接的端点来说，拥塞往往表现为通信时延的增加。当然，拥塞控制和流量控制也有相似的地方，即它们都通过控制发送方发送数据的速率来达到控制效果。

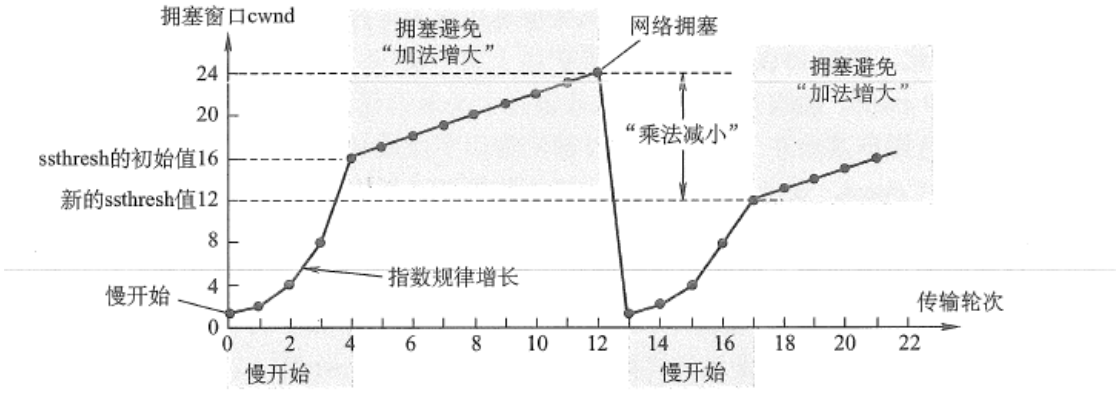
拥塞控制与流量控制的区别：拥塞控制是让网络能够承受现有的网络负荷，是一个全局性的过程，涉及所有的主机、所有的路由器，以及与降低网络传输性能有关的所有因素。相反，流量控制往往是指点对点的通信量的控制，即接收端控制发送端，它所要做的是抑制发送端发送数据的速率，以便使接收端来得及接收。

1、**慢开始算法（接收窗口rwnd，拥塞窗口cwnd）**

在TCP 刚刚连接好并开始发送TCP 报文段时，先令拥塞窗口cwnd = 1, 即一个最大报文段长度MSS 。每收到一个对新报文段的确认后，将cwnd 加1, 即增大一个MSS 。用这样的方法逐步增大发送方的拥塞窗口cwnd, 可使分组注入网络的速率更加合理。使用慢开始算法后，每经过一个传输轮次（即往返时延RTT), 拥塞窗口cwnd 就会加倍，即cwnd 的大小指数式增长。这样，慢开始一直把拥塞窗口cwnd 增大到一个规定的慢开始门限ssthresh（阔值），然后改用拥塞避免算法。

2、拥塞避免算法

拥寒避免算法的做法如下：发送端的拥塞窗口cwnd 每经过一个往返时延RTT 就增加一个MSS的大小，而不是加倍，使cwnd 按线性规律缓慢增长（即加法增大），而当出现一次超时（网络拥塞）时，令慢开始门限ssthresh 等于当前cwnd 的一半（即乘法减小）。



3、快重传

快重传技术使用了冗余ACK 来检测丢包的发生。同样，冗余ACK 也用千网络拥塞的检测（丢了包当然意味着网络可能出现了拥塞）。快重传并非取消重传计时器，而是在某些情况下可更早地重传丢失的报文段。当发送方连续收到三个重复的ACK 报文时，直接重传对方尚未收到的报文段，而不必等待那个报文段设置的重传计时器超时。

4、快恢复

快恢复算法的原理如下：发送端收到连续三个冗余ACK (即重复确认）时，执行“乘法减小”算法，把慢开始门限ssthresh 设置为出现拥塞时发送方cwnd 的一半。与慢开始（慢开始算法将拥塞窗口cwnd 设置为1) 的不同之处是，它把cwnd 的值设置为慢开始门限ssthresh 改变后的数值，然后开始执行拥塞避免算法("加法增大")'使拥塞窗口缓慢地线性增大。由于跳过了cwnd 从1 起始的慢开始过程，所以被称为快恢复。

28.为何不采用“三次握手“释放连接，且发送最后一次握手报文后要等待2MSL 的时间呢?

原因有两个：

1) 保证A 发送的最后一个确认报文段能够到达B 。如果A 不等待2MSL, 若A 返回的最后确认报文段丢失，则B 不能进入正常关闭状态，而A 此时已经关闭，也不可能再重传。

2) 防止出现“已失效的连接请求报文段“。A 在发送最后一个确认报文段后，再经过2MSL可保证本连接持续的时间内所产生的所有报文段从网络中消失。

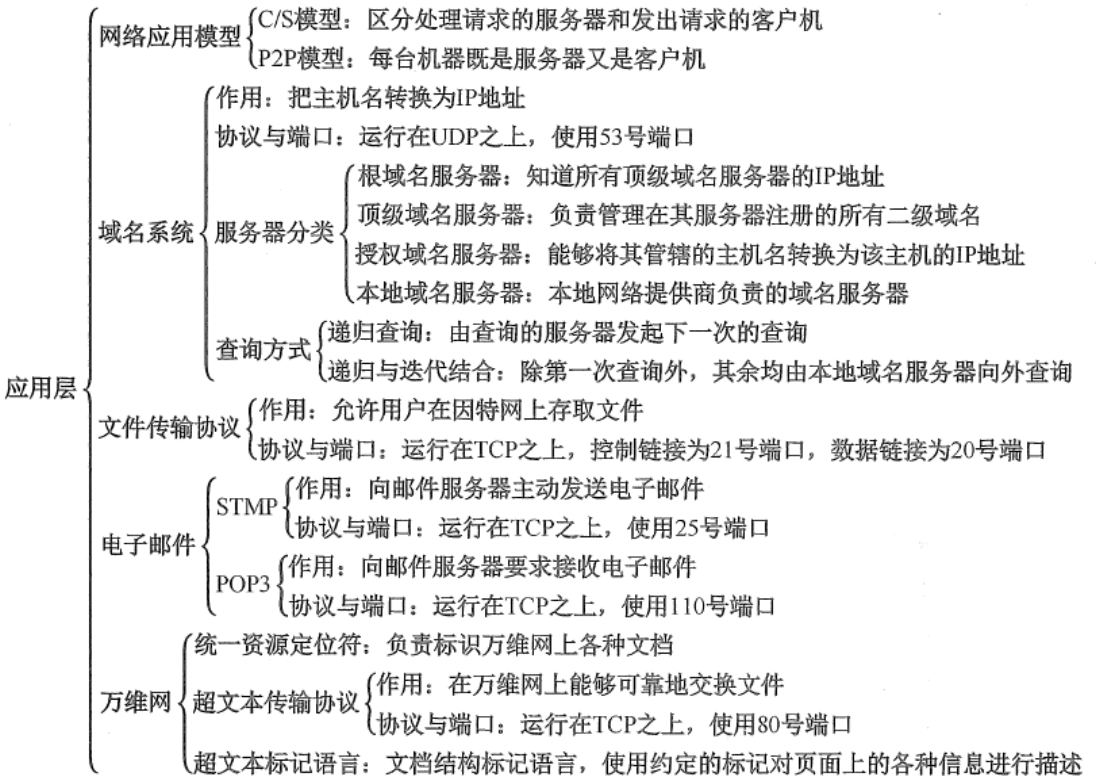
服务器结束TCP 连接的时间要比客户机早一些，因为客户机最后要等待2MSL 后才可进入CLOSED 状态。

29.为什么不采用“两次握手”建立连接呢?

这主要是为了防止两次握手情况下已失效的连接请求报文段突然又传送到服务器而产生错误。考虑下面这种情况。客户A 向服务器B 发出TCP 连接请求，第一个连接请求报文在网络的某个结点长时间滞留， A 超时后认为报文丢失，千是再重传一次连接请求， B 收到后建立连接。数据传输完毕后双方断开连接。而此时，前一个滞留在网络中的连接请求到达服务器B, 而B 认为A 又发来连接请求，此时若使用“三次握手”，则B 向A 返回确认报文段，由于是一个失效的请求，因此A 不予理睬，建立连接失败。若采用的是“两次握手”，则这种情况下B 认为传输连接已经建立，并一直等待A 传输数据，而A 此时并无连接请求，因此不予理睬，这样就造成了B的资源白白浪费。

第六章、应用层

快速唤起记忆知识框架:



30.DNS域名解析协议?

> 域名解析是指把域名映射成为IP 地址或把IP 地址映射成域名的过程。前者称为正向解析，后者称为反向解析。当客户端需要域名解析时，通过本机的DNS 客户端构造一个DNS 请求报文，以UDP数据报方式发往本地域名服务器。域名解析有两种方式：递归查询和递归与迭代相结合的查询。

31.FTP文件传输协议?

> 文件传输协议（file Transfer Protocol, FTP) 是因特网上使用得最广泛的文件传输协议。FTP提供交互式的访问，允许客户指明文件的类型与格式，并允许文件具有存取权限。它屏蔽了各计算机系统的细节，因而适合于在异构网络中的任意计算机之间传送文件。FTP 提供以下功能：
> （1）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。
> （2）以用户权限管理的方式提供用户对远程FTP 服务器上的文件管理能力。
> （3）以匿名FTP 的方式提供公用文件共享的能力。
>
> FTP 采用客户 / 服务器的工作方式，它使用TCP 可靠的传输服务。一个FTP 服务器进程可同时为多个客户进程提供服务。FTP 的服务器进程由两大部分组成：一个主进程，负责接收新的请求；另外有若干从属进程，负责处理单个请求。其工作步骤如下：
> （1）打开熟知端口21 (控制端口），使客户进程能够连接上。
> （2）等待客户进程发连接请求。
> （3）启动从属进程来处理客户进程发来的请求。主进程与从属进程并发执行，从属进程对客户进程的请求处理完毕后即终止。
> （4）回到等待状态，继续接收其他客户进程的请求。

32.SMTP简单邮件传输协议?

> 简单邮件传输协议(Simple Mail Transfer Protocol, SMTP) 是一种提供可靠且有效的电子邮件传输的协议，它控制两个相互通信的SMTP 进程交换信息。由于SMTP 使用客户 / 服务器方式，因此负责发送邮件的SMTP 进程就是SMTP 客户，而负责接收邮件的SMTP 进程就是SMTP 服务器。SMTP 用的是TCP 连接，端口号为25 。SMTP 通信有以下三个阶段：(1) 连接建立(2) 邮件传送(3) 连接释放。

33.POP3

> 邮局协议(Post Office Protocol, POP) 是一个非常简单但功能有限的邮件读取协议，现在使用的是它的第3 个版本POP3 。POP3 采用的是“拉" (Pull) 的通信方式，当用户读取邮件时，用户代理向邮件服务器发出请求，“拉”取用户邮箱中的邮件。POP 也使用客户 / 服务器的工作方式，在传输层使用TCP, 端口号为110 。接收方的用户代理上必须运行POP 客户程序，而接收方的邮件服务器上则运行POP 服务器程序。POP 有两种工作方式：“下载并保留”和“下载并删除 " 。在“下载并保留”方式下，用户从邮件服务器上读取邮件后，邮件依然会保存在邮件服务器上，用户可再次从服务器上读取该邮件；而使用“下载并删除“方式时，邮件一旦被读取，就被从邮件服务器上删除，用户不能再次从服务器上读取。
>
> 随着万维网的流行，目前出现了很多基于万维网的电子邮件，如Hotmail 、Gmail 等。这种电子邮件的特点是，用户浏览器与Hotmail 或Gmail 的邮件服务器之间的邮件发送或接收使用的是HTTP, 而仅在不同邮件服务器之间传送邮件时才使用SMTP

34.HTTP超文本传输协议?

> HTTP 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。从层次的角度看， HTTP 是面向事务的(Transaction-oriented) 应用层协议，它规定了在浏览器和服务器之间的请求和响应的格式与规则，是万维网上能够可靠地交换文件（包括文本、声音、图像等各种多媒体文件）的重要基础。

> **用户单击鼠标后所发生的事件按顺序如下**（以访问清华大学的网站为例）：
> 1) 浏览器分析链接指向页面的URL (http://www.tsinghua.edu.cn /chn/index.htm) 。
> 2) 浏览器向DNS 请求解析www.tsinghuaaedu.CR 的IP 地址。 -
> 3) 域名系统统DNS 胪祈出清毕大学服务器的IP 地址。
> 4) 浏览器与该服务器建立TCP 连接（默认端口号为80) 。
> 5) 浏览器发出HTTP 请求： GET /chn/index.htm 。
> 6) 服务器通过HTTP 响应把文件index.htm 发送给浏览器。
> 7) TCP 连接释放。
> 8) 浏览器解释文件index.htm, 并将Web 页显示给用户。

表 6.2　常见应用层协议小结

应用程序	FTP 数据链接	FTP 控制链接	TELNET	SMTP	DNS	TFTP	HTTP	POP3	SNMP
使用协议	TCP	TCP	TCP	TCP	UDP	UDP	TCP	TCP	UDP
熟知端口号	20	21	23	25	53	69	80	110	161