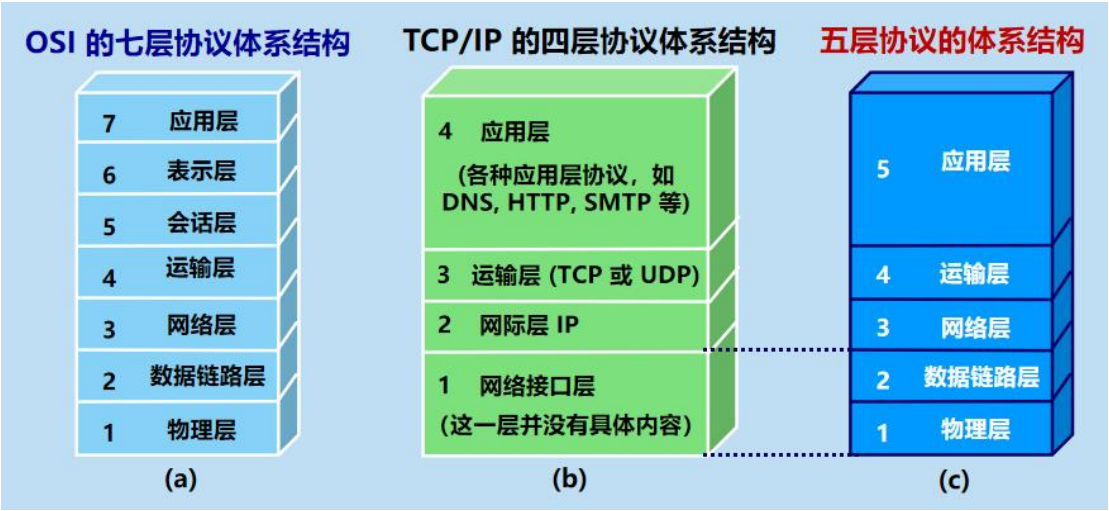
1. TCP/IP 模型与 OSI 模型各层的名称、对应关系、实现的功能、各层的主要协议及其作用，以及在主机和网络节点中的区别，分层设计的优点



p31

应用层

任务：通过应用进程间的交互来完成特定网络应用。

协议：定义的是应用进程间通信和交互的规则。

把应用层交互的数据单元称为报文(message)。

例如：DNS，HTTP，SMTP

运输层

任务：负责向两台主机中进程之间的通信提供通用的数据传输服务。

具有复用和分用的功能。

主要使用两种协议：

传输控制协议 TCP

用户数据报协议 UDP

TCP (Transmission Control Protocol)：

提供面向连接的、可靠的数据传输服务。

数据传输的单位是报文段 (segment)。

UDP (User Datagram Protocol)：

提供无连接的尽最大努力 (best-effort) 的数据传输服务（不保证数据传输的可靠性）。

数据传输的单位是用户数据报。

网络层

为分组交换网上的不同主机提供通信服务。

两个具体任务：

路由选择：通过一定的算法，在互联网中的每一个路由器上，生成一个用来转发分组的转发表。

转发：每一个路由器在接收到一个分组时，要依据转发表中指明的路径把分组转发到下一个路由器。

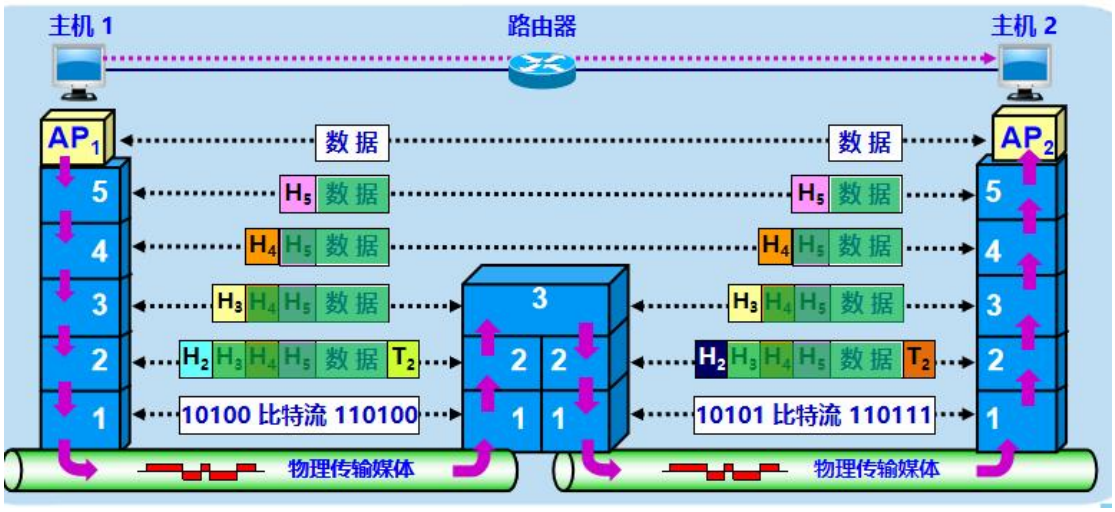
互联网使用的网络层协议是无连接的网际协议 IP (Internet Protocol) 和许多种路由选择协议，因此互联网的网络层也叫做网际层或 IP 层。IP 是用来使互连起来的许多计算机网络能

够进行通信。
IP 协议分组也叫做 IP 数据报，或简称为数据报。

数据链路层
任务：实现两个相邻节点之间的可靠通信。
在两个相邻节点间的链路上传送帧（frame）。
如发现有差错，就简单地丢弃出错帧。
如果需要改正出现的差错，就要采用可靠传输协议来纠正出现的差错。这种方法会使数据链路层协议复杂。
点对点协议 PPP 是用户计算机和 ISP 进行通信时所使用的数据链路层协议。
载波监听多点接入/碰撞检测（CSMA/CD）协议：具有广播特性的总线上实现了一对一的通信。

物理层
任务：实现比特（0 或 1）的传输。
确定连接电缆的插头应当有多少根引脚，以及各引脚应如何连接。

在主机和网络节点中的区别



分组在网络节点上升到第 3 层网络层就转发分组，然后往下传送第 2 层了。p33

分层设计的优点：p30
- 各层之间是独立的。
- 灵活性好。
- 结构上可分割开。
- 易于实现和维护
- 能促进标准化工作。

2. 滑动窗口、ARQ 的定义、原理、特点
p224
滑动窗口
发送方和接收方各自维持着发送窗口和接受窗口。发送方每收到一个确认，就把发送窗口向

前滑动一个分组的位置。接收方一般采用累计确认方式，即接收方不必对收到的分组逐个发送确认，而是可以在收到几个分组后，对按序到达的最后一个分组发送确认，这样就表示：到这个分组位置的所有分组都已经正确收到了。

ARQ，也可以是 Automatic Repeat Query 的缩写，是一种在数据传输时，使用确认（Acknoledgements，就是我们常说的 ACK，接收方发送一个消息，告诉发送方，自己是否正确接到了一个包体）和超时（Timeouts，在收到一个确认消息之前，等待的一个确定的时间段）机制，在不可靠的网络上，实现可靠的数据传输的错误控制方法。如果发送方在超时之前没有收到确认，通常会重新传输相应的包体，直到收到确认或者重试超过一定的次数。

原理
发送方每收到一个确认，就把发送窗口向前滑动一个分组的位置。接收方一般采用累计确认方式，即接收方不必对收到的分组逐个发送确认，而是可以在收到几个分组后，对按序到达的最后一个分组发送确认，这样就表示：到这个分组位置的所有分组都已经正确收到了。连续 ARQ 协议采用 Go-back-N（回退 N），表示需要再退回来重传已发送过的 N 个分组。

特点
优点：容易实现，即使确认丢失也不必重传。提高信道利用率。
缺点：不能向发送方反映出接收方已经正确收到的所有分组的信息。当通信线路质量不好时，需要等待。

3. 流量控制和拥塞控制的本质、相同点、不同点、实现的方法

p236
流量控制
让发送方的发送速率不要太快，使接收方来得及接收。
利用滑动窗口机制可以很方便地在 TCP 连接上实现对发送方的流量控制。

p238
拥塞控制
防止过多的数据注入到网络中，避免网络中的路由器或链路过载。
拥塞控制的前提：网络能够承受现有的网络负荷。

相同点
都是为了提高网络性能。

不同点
流量控制：抑制发送端发送数据的速率，以使接收端来得及接收。点对点通信量的控制，是个端到端的问题。
拥塞控制：防止过多的数据注入到网络中，避免网络中的路由器或链路过载。是一个全局性的过程，涉及到所有的主机、路由器，以及与降低网络传输性能有关的所有因素。

4. IPv4 和 IPv6 的过渡技术

p155
采用逐步演进的办法，同时还必须使新安装的 IPv6 系统能够向后兼容。（向后兼容：IPv6 系

统必须能够接收和转发 IPv4 分组，并且能够为 IPv4 分组选择路由。）
两种过渡策略：使用双协议栈、使用隧道技术
双协议栈：在完全过渡到 IPv6 之前，使一部分主机（或路由器）同时装有 IPv4 和 IPv6 两种协议栈。
隧道技术：在 IPv6 数据报要进入 IPv4 网络时，把 IPv6 数据报封装成为 IPv4 数据报。现在整个 IPv6 数据报变成了 IPv4 数据报的数据部分。

5. CIDR 表示法

p125
CIDR (Classless Inter-Domain Routing) ：无分类域间路由选择。
消除了传统的 A 类、B 类和 C 类地址以及划分子网的概念，可以更加有效地分配 IPv4 的地址空间，但无法解决 IP 地址枯竭的问题。
IP 地址 ::= { <网络前缀>, <主机号>}
CIDR 记法：斜线记法 (slash notation)
a.b.c.d / n：二进制 IP 地址的前 n 位是网络前缀。
CIDR 把网络前缀都相同的所有连续的 IP 地址组成一个 CIDR 地址块。

6. TCP 序号、确认号的关系，IP 分片的偏移、标志位如何设置和计算

p225
序号：占 4 字节。TCP 连接中传送的数据流中的每一个字节都有一个序号。序号字段的值则指的是本报文段所发送的数据的第一个字节的序号。
确认号：占 4 字节，是期望收到对方的下一个报文段的数据的第一个字节的序号。
若确认号 =N，则表明：到序号 N – 1 为止的所有数据都已正确收到。

IP 分片 p137
数据部分长度为 m 字节，给定 MTU（默认 1500）

- n 个分片的标志位均与原分片相同；
- 第 i 个分片的片偏移为 (i-1)*MTU/8 ；
- 第 0…n-1 个分片 MF=1，DF=0；第 n 个分片 MF=0，DF=0 。

标志(flag) ——占 3 位，目前只有前两位有意义。标志字段的最低位是 MF (More Fragment)。MF=1 表示后面还有分片，MF=0 表示最后一个分片。标志字段中间的一位是 DF (Don't Fragment) 。只有当 DF=0 时才允许分片。
片偏移——占 13 位，指出：较长的分组在分片后某片在原分组中的相对位置。片偏移以 8 个字节为偏移单位。

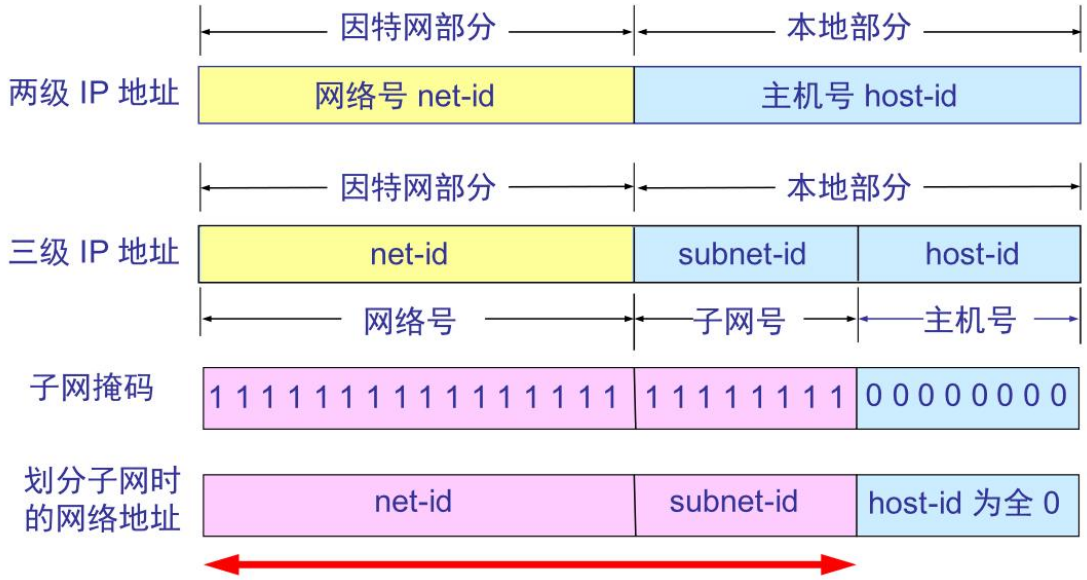
7. IP 地址的合法性、子网掩码的作用，子网号、主机号的区别，网络地址、主机地址、广播地址的区别。

IP 地址 p122
32 位二进制代码，分为每 8 位为一组，将每 8 位的二进制数转换为十进制数，采用点分十进制记法。

子网掩码 p127
位数：32 位。
目的：让机器从 IP 地址迅速算出网络地址。

由一连串 1 和接着的一连串 0 组成，而 1 的个数就是网络前缀的长度。

从主机号（前面）借用若干位作为子网号，而主机号也就相应减少了若干位。



p127
二进制的 IP 地址和子网掩码按位与运算得到网络地址
子网掩码的反码和 IP 地址按位与运算得到主机地址
广播地址=网络地址+子网掩码的反码（地址位全为 1）

8. 时延的组成及其各自的区别和计算方法
p22
（1）发送时延
是主机或路由器发送数据帧所需要的时间，也就是从发送数据帧的第一个比特算起，到该帧的最后一个比特发送完毕所需的时间。

$$\text{发送时延} = \frac{\text{数据帧长度（bit）}}{\text{发送速率（bit/s）}}$$

（2）传播时延
是电磁波在信道中传播一定的距离需要花费的时间。

$$\text{传播时延} = \frac{\text{信道长度（米）}}{\text{信号在信道上的传播速率（米/秒）}}$$

注意：发送时延与传播时延有本质上的不同。

- 发送时延发生在机器内部的发送器中，与传输信道的长度（或信号传送的距离）没有任何关系。
- 传播时延则发生在机器外部的传输信道媒体上，而与信号的发送速率无关。信号传送的距离越远，传播时延就越大。

（3）处理时延

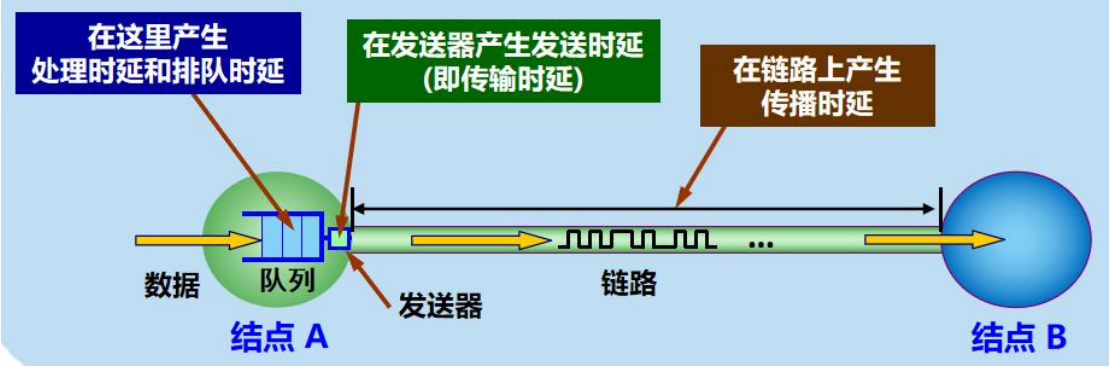
主机或路由器在收到分组时，为处理分组（例如分析首部、提取数据、差错检验或查找路由）所花费的时间。

（4）排队时延

分组在路由器输入输出队列中排队等待处理和转发所经历的时延。

排队时延的长短往往取决于网络中当时的通信量。当网络的通信量很大时会发生队列溢出，使分组丢失，这相当于排队时延为无穷大。

四种时延产生的地方不一样



9. PPP 字符填充和比特填充的方式、特殊情况处理

p81

字节填充

当 PPP 使用异步传输时，它把转义符定义为 0x7D(即 01111101)，并使用字节填充。

把信息字段中出现的每一个 0x7E 字节转变成为 2 字节序列(0x7D,0x5E)。 -20

若信息字段中出现一个 0x7D 的字节 (即出现了和转义字符一样的比特组合)，则把 0x7D 转变成为 2 字节序列(0x7D,0x5D)。 -20

信息字段中出现 ASCII 码的控制字符(即数值小于 0x20 的字符)，则在该字符前面要加入一个 0x7D 字节，同时将该字符的编码加以改变。例如，出现 0x03(在控制字符中是“传输结束”ETX)就要把它转变为 2 字节序列(0x7D,0x23)。 +20

零比特填充

PPP 协议用在 SONET/SDH 链路时，使用同步传输(一连串的比特连续传送)而不是异步传输(逐个字符地传送)。在这种情况下，PPP 协议采用零比特填充方法来实现透明传输。

零比特填充的具体做法是: 在发送端，先扫描整个信息字段(通常用硬件实现，但也可用软件实现，只是会慢些)。只要发现有 5 个连续 1，则立即填入个 0。因此经过这种零比特填充后的数据，就可以保证在信息字段中不会出现 6 个连续 1。接收端把 5 个连 1 之后的比特 0 删除。

10. 路由聚合、子网划分

p128

路由聚合

利用较大的一个 CIDR 地址块来代替许多个较小的地址块。

将网络前缀都相同的连续的 IP 地址组成“CIDR 地址块”。 一个 CIDR 地址块可以表示很多地址。可以使得路由表中的一个项目可以表示多个原来传统分类地址的路由，有利于减少路由器之间的路由选择信息的交换，从而提高网络性能。

子网划分

划分子网纯属一个单位内部的事情。这个单位对外仍然表现为没有划分子网的网络。划分子网只是对主机号进行划分，不改变原有的网络号。

从主机号借用若干个比特作为子网号，而主机号也就相应减少了若干个比特，形成三级 IP 地址：{<网络号>, <子网号>, <主机号>}

原因：

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

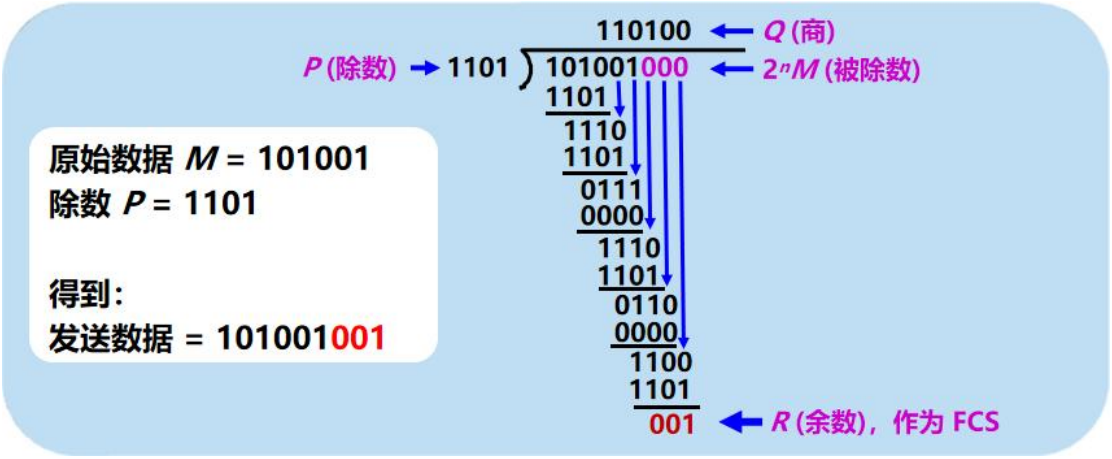
11. CRC 计算方法和过程

p76

① 用二进制的模 2 运算进行 2^n 乘 M 的运算，这相当于在 M 后面添加 n 个 0。

② 得到的 (k + n) 位的数除以事先选定好的长度为 (n + 1) 位的除数 P，得出商是 Q ，余数是 R，余数 R 比除数 P 少 1 位，即 R 是 n 位。

③ 将余数 R 作为冗余码拼接在数据 M 后面，一起发送出去。



12. internet 与 Internet 的区别

p5

Internet，因特网，专有名词，全球最大的、开放的、由众多网络相互连接而成的特定计算机网络，它采用 TCP/IP 协议族作为通信的规则，且其前身是美国的 ARPANET

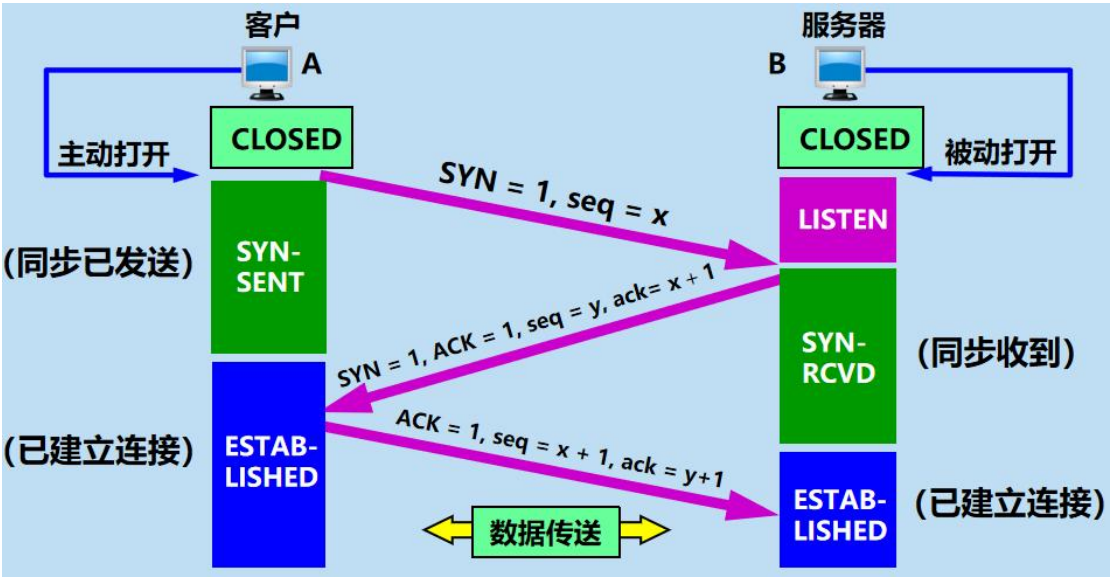
internet，互联网，通用名词，它泛指由多个计算机网络互连而成的网络。

13. TCP 连接的建立和拆除过程，及其简化、拆分、修改方法

建立 p247

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 (client)。
被动等待连接建立的应用进程叫做服务器 (server)。
TCP 建立连接的过程叫做握手。
采用三报文握手：在客户和服务器之间交换三个 TCP 报文段，以防止已失效的连接请求报文段突然又传送到了，因而产生 TCP 连接建立错误。

- B 的 TCP 服务器进程先创建传输控制块 TCB，准备接受客户进程的连接请求。
- A 的 TCP 服务器进程也是首先创建传输控制块 TCB。然后，在打算建立 TCP 连接时，向 B 主动发出连接请求报文段，其首部中的同步位 SYN = 1，并选择一个初始序号 seq = x，表明传送数据时的第一个数据字节的序号是 x。（注意：TCP 规定，SYN 报文段（即 SYN = 1 的报文段）不能携带数据，但要消耗掉一个序号。）
- B 的 TCP 收到连接请求报文段后，如同意，则发回确认。在确认报文段中 SYN = 1，ACK = 1，其确认号 ack = x + 1，也为自己选择序号 seq = y。（这个报文段也不能携带数据，但同样要消耗掉一个序号。）
- A 收到此报文段后向 B 给出确认，其 ACK = 1，确认号 ack = y + 1，自己的序号 seq = x + 1。A 进入已建立连接状态。A 的 TCP 通知上层应用进程，连接已经建立。（TCP 标准规定：ACK 报文段可以携带数据。但如果不携带数据，则不消耗序号。下一个数据报文段的序号仍是 seq = x + 1。）
- B 的 TCP 收到主机 A 的确认后，进入已建立连接状态，也通知其上层应用进程：TCP 连接已经建立。双方可以开始数据传送。



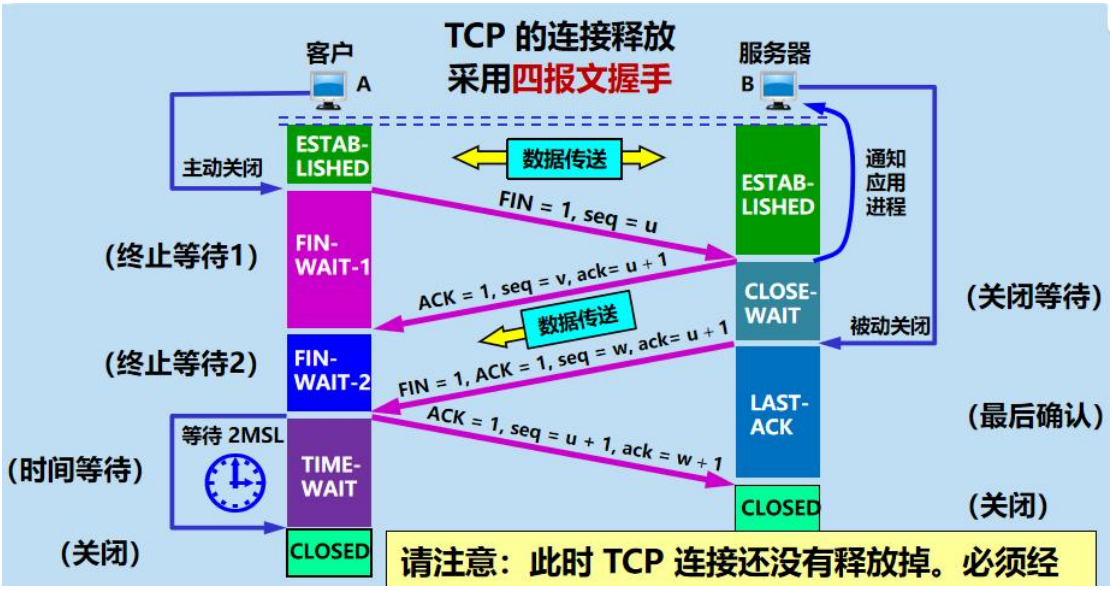
释放 p248
TCP 连接释放过程比较复杂。
数据传输结束后，通信的双方都可释放连接。
TCP 连接释放过程是四报文握手。
A 的应用进程先向其 TCP 发出连接释放报文段，并停止再发送数据，主动关闭 TCP 连接。A 把连接释放报文段首部的 FIN = 1，其序号 seq = u（它等于前面已传送过的数据的最后一个字节的序号加 1），等待 B 的确认。（TCP 规定：FIN 报文段即使不携带数据，也消耗掉一个序号。）
B 发出确认，ACK=1，确认号 ack = u+1，这个报文段的序号 seq = v（它等于 B 前面已传

送过的数据的最后一个字节的序号加 1）。TCP 服务器进程通知高层应用进程。从 A 到 B 这个方向的连接就释放了，TCP 连接处于半关闭 (half-close) 状态。B 若发送数据，A 仍要接收。

若 B 已经没有要向 A 发送的数据，其应用进程就通知 TCP 释放连接。 FIN=1，ACK=1，确认号 ack = u+1 ，假定 B 的序号 seq = w。

A 收到连接释放报文段后，必须发出确认。ACK=1，确认号 ack=w+1，自己的序号 seq = u + 1。

请注意：此时 TCP 连接还没有释放掉。必须经过时间等待计时器 (TIME-WAIT timer) 设置的时间 2MSL 后，A 才释放 TCP 连接。



14. 习题 1-29，1-30，1-31，3-09，3-10，3-20，4-22，4-41，4-48