目录

第一章

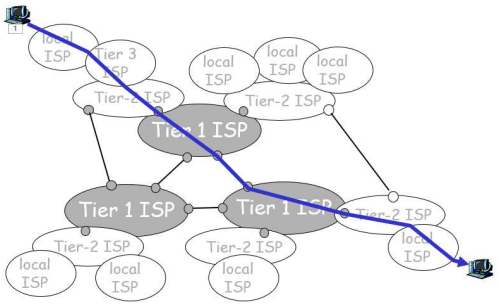
一、Internet 的基本概念

1.定义: network of networks

2.标准的制定：由 IETF 制定出标准文档 RFC

3.在计算机网中：带宽和传输速率是相同的意思

4.ISP：Internet Service Provider，分为三级，同等级之间是平等的关系，而低级需要向高级支付服务费。在一个 ISP 网络中，该 ISP 与其他 ISP 的连接点称为汇集点 POP



二、网络边缘的介绍

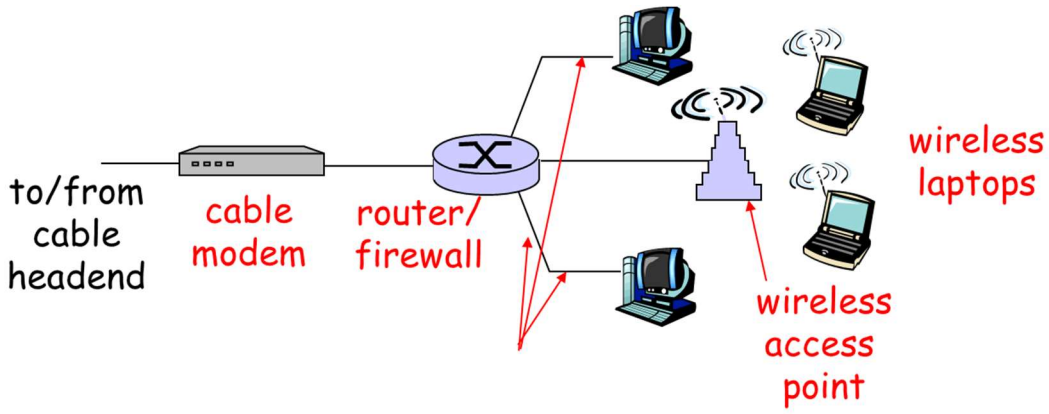
1. host（也被称为 end system)：它有两种工作模式，client/server 模式和 peer to peer 模式。对于后者所有 host 均有 client 和 server 两个程序，而前者同时只有一个。

2. access network：它是将 host 连接到网络的边缘路由器的物理链路。大致可以分为三类：Residential access networks、institutional access networks、mobile access networks。下面分别介绍三种接入网。

(1) Residential access networks

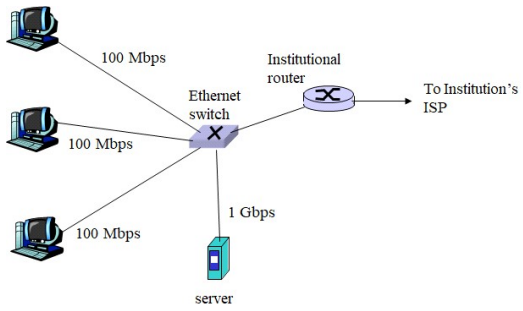
	普通电话线接入	**DSL（Digital Subscriber Line）**	**HFC（Hybrid Fiber-coaxial Cable）**
设备	Dial-up modem	DSL modem	Cable modem
网络结构	使用已有的电话网的基础设施（图 a.1）	同左（图 a.2）	使用有线电视网的基础设施（图 a.3）
速率（带宽）	$\leq 56kbps$	电话信道：0~4kHz 上行数据：$4k$~50kHz 下行数据：$50k$~1MHz	上行数据：$2Mbps$ 下行数据：$30Mbps$
带宽是否共享	打电话时不能上网，反之亦然（专用）	1.二者可同时进行 2.各用户使用专门的带宽（专用）	1.二者可同时进行 2.各用户共享带宽（则用户越多，下行速率越慢，上行越容易碰撞）

常见的家庭接入网如下图所示



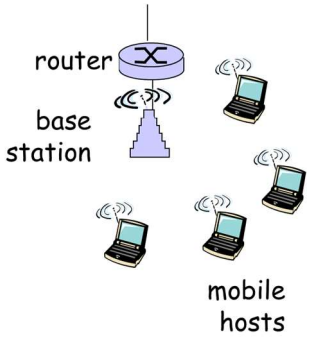
(2) institutional access networks

常用的是以太网接入（Ethernet Internet access）



(3) mobile access networks

如：wireless LANs、蜂窝网



3. Physical Media

(1) 分类：$\begin{cases} guided\ media: & fiber(光纤), coax(同轴电缆), twisted\ pair \\ unguided\ media: & radio(分为陆地无线电信道和卫星无线电信道) \end{cases}$

(2) 比较

		特点	应用
有线	双绞线	1.最便宜 2.所能达到的传输速率取决于线的厚度及双方的距离	LAN
	同轴电缆	1.可以双向传输	电缆电视、电缆Internet
	光纤	1.可以双向传输	

		2.速率快	
无线	陆地无线电信道	1.传播特性依赖于传播环境和传输信号的距离 2.其中，环境因素主要有：路径损耗、遮挡衰落、多径衰落以及干扰	蜂窝网、Wifi
	卫星无线电信道	1.时延大	分为两类：同步卫星和低地球轨道卫星

三、网络中数据传输的两种交换方式

		电路交换	分组交换
比较	复用原理	FDM、TDM	统计多路复用、存储转发传输
	链路是否需要建立专门通道	是	否
	每个用户的带宽	链路带宽的$\frac{1}{n}$	和链路带宽一样
二者的评价		电路交换：适用于实时业务 分组交换：适用于数据业务网络利用率高，相同带宽下可以容纳更多用户，但是会出现阻塞和丢包的现象 7	

1.计算机网的三个参数（分组交换）

(1) 单路由器的时延

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

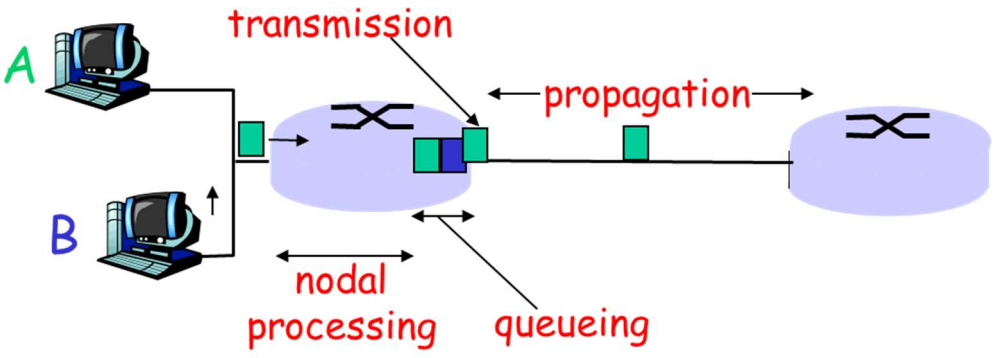
其中：

d_{proc}为处理时延（微秒级）

d_{queue}为排队时延 （取决于流量强度$\frac{La}{R}$，a 为包的平均到达率，工程中要求其小于 1）

d_{trans}为传输时延（=L/R,前者为包长 bits，后者为传输速率或带宽 bps）

d_{prop}为传播时延（= d/s 毫秒级，前者为链路的长度，后者为传播速度约 2e8 m/s，可以看做起始比特从发出到刚到目的主机所用的时间）



这里应该注意传输时延和传播时延的区别，前者是从开始发送分组的第一位到完成发送分组的最后一位总共的时间，后者是离开路由器，在链路上的传送时间。

(2)端到端的时延（源主机到目的主机）

$$d_{end-e} = N(d_{proc} + d_{trans} + d_{prop})$$

公式说明：这里假设源主机和目的主机间的路由器数目为 N-1，而不是 N，且忽略了排队时延，同时认为各节点对应的时延是相同的。

2.丢包：当队满时，交换机会不接收新到的分组，从而丢包。

3.分组交换的吞吐量（throughput）

(1) 瞬时吞吐量：某一时刻收到文件的速率（bps）

(2) 平均吞吐量：总共接收到的比特数比上时间

端到端的各链路中，传输速率最小的那条链路称为瓶颈链路，对应的传输速率即为吞吐量。

四、网络的分层体系结构

协议栈：各层所有协议总和

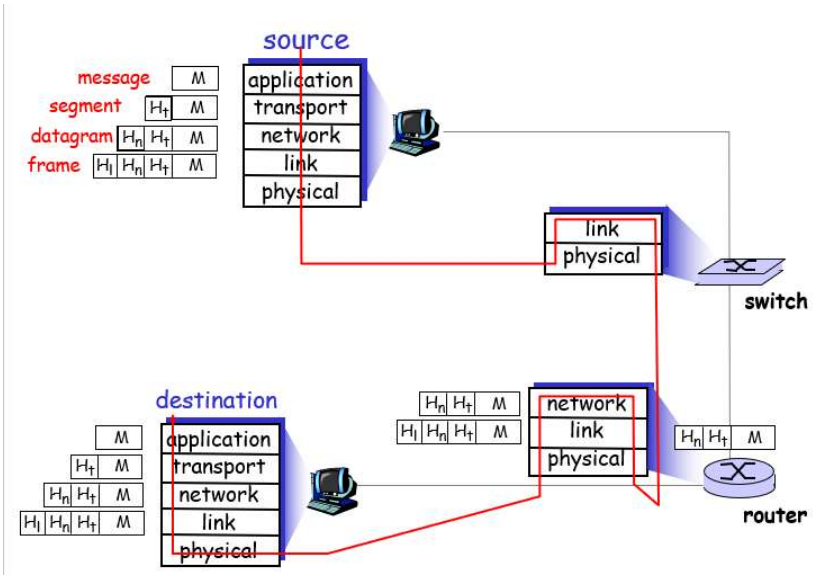
1. 五层因特网协议栈介绍

	功能	分布	基本单元	例子
应用层	应用程序及其协议	Host	报文 Message	HTTP、SMTP、 FTP
传输层	提供应用程序之间传送报文的服务	Host	报文段 Segment	TCP、UDP
网络层	将数据包从一个主机移动到另一个主机	Host 路由器	数据报 Datagram	IP
链路层		Host 路由器 链路层交换机	帧 Frame	以太网、WiFi PPP
物理层		Host 路由器 链路层交换机	Bit stream	双绞线、光纤

注意：ISO 7 层模型如今已经不用了，它在应用层和传输层之间还有表示层和会话层

另外：可以看出路由器可以实现 IP 协议而链路层交换机不能实现。

2.各层的封装



五、对网络的攻击

1.攻击个人电脑

(1) 病毒 virus：需要某种形式的用户交互来感染用户设备的恶意软件

(2) 蠕虫 worm：不需要任何明显的用户交互来感染用户设备的恶意软件

(3) 木马 trojan horse：隐藏在有用软件中的恶意软件

2.攻击服务器和网络基础设施

Dos 攻击（Denial of service），分为下面三类

(1) 弱点攻击：向目标主机上的某些易受攻击的应用程序或操作系统发送某种报文，使其停止运行。（应用层）

(2) 带宽洪泛：向目标主机发送大量分组导致链路拥塞，使合法的分组无法到达。（链路层）

(3) 连接洪泛：在目标主机中创建大量的全开或半开的 TCP 连接，从而停止合法连接（传输层）

DDos 攻击（Distributed Dos）：对于带宽洪泛攻击，当一台机器的传输速率不足以如目标链路拥塞时，会利用僵尸网络 botnet 一起对目标进行攻击。

3.嗅探分组（密码学方法防御）

4.IP 哄骗（端点鉴别防御）

5.中间人攻击：插入两主机间的通信路径，可以修改或删除报文

第二章　应用层

2.1 应用层协议原理

一、应用程序体系结构

1. Client-Server 结构

(1)定义：Client 是发起通信端的进程，Server 会话开始后等待联系的进程。

(2)Server 特征：具有固定的 IP 地址、总是打开。实际中，一台 Server 无法满足所有 Client，常用主机群集（服务器场）

(3)Client 特征：动态 IP 地址、Client 之间无法直接通信，且 Client 时断时续。

(4)实例：Web、FTP、Telnet、Email

2. P2P 结构

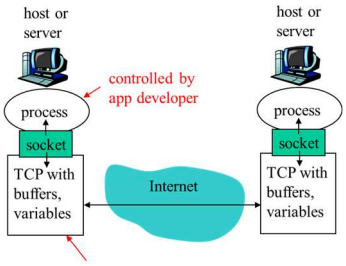
(1)特征：也是 d 动态的 IP；任意间断连接，每个端系统同时有 Client 和 Server 两个程序。“自扩展性”

(2)实例：流量密集型，如 BitTorrent、IPTV

3.混合型

(1)特征：服务器场跟踪用户的 IP 地址，报文直接在用户间发送，而不经过服务器

二、两主机间进程通信的实现----------套接字（socket）编程



1.定义：套接字 即应用程序进程和传输层协议之间的接口。

2.应注意：提供的 API 只能控制应用层，对于传输层仅限于传输层协议的选择和设定几个传输层参数。

3.进程寻址的方式：IP 地址（识别主机）+端口号（识别进程）

4.应用层协议定义了：交换的报文类型（请求、响应），语法（报文字段的分配及含义），语义（字段中各种 01 信息的含义），进程何时、如何发送报文及对报文进行响应的规则。

三、传输服务

1.传输层可以为应用层提供的四个方面的服务

(1)是否可靠数据传输（音视频业务可以容忍分组丢失，但是数据业务不可以）

(2)吞吐量（带宽敏感业务对吞吐量有下限要求，而弹性应用则没有）

(3)定时（对时延的要求）

(4)安全性（指传输过程中不让他人获得）

下表为不同的应用类型对它们的要求

Application	Data loss	Throughput	Time Sensitive
file transfer	no loss	elastic	no
e-mail	no loss	elastic	no
Web documents	no loss	elastic	no
instant messaging	no loss	elastic	yes and no

real-time audio/video	loss-tolerant	audio: 5kbps-1Mbps video:10kbps-5Mbps	yes, 100's msec
stored audio/video	loss-tolerant	same as above	yes, few secs
interactive games	loss-tolerant	few kbps up	yes, 100's msec

2.传输层提供的两个协议

	TCP（Transmission Control Protocol）	UDP（User Datagram Protocol）
提供的服务	1. 面向连接服务（连接前先进行握手，之后建立全双工的 TCP 连接，在结束通信后，拆除连接） 2.可靠数据传输服务	1. 无连接 2. 不可靠数据传输服务
	应注意：二者均只提供了上面提到的四个服务的第一个服务	
其他机制	流量控制、拥塞控制	无
适用	非实时业务	实时业务（但逐渐转向 TCP）
安全性	二者均没有提供任何加密机制。 SSL 是 TCP 的加强版（在应用层实现加密），提供了包括加密，数据完整性和端点鉴别的安全性服务	

应用	应用层协议	下面的运输层协议
电子邮件	SMTP	TCP
远程终端访问	Telnet	TCP
Web	HTTP	TCP
文件传输	FTP	TCP
远程文件服务器	NFS	通常UDP
流式多媒体	通常专用	UDP或TCP
因特网电话	通常专用	UDP或TCP
网络管理	SNMP	通常UDP
选路协议	RIP	通常UDP
名字转换	DNS	通常UDP

图3-6 流行的因特网应用及其采用的运输层协议

2.2 Web 应用和 HTTP 协议

一、概述

1.Web 基本概念

Web 页面由各种对象组成，对象即 HTML 文件、图形文件、视频文件等。

Web 浏览器是 HTTP 的客户机端；Web 服务器是服务器端，存储各种 Web 对象，所有对象通过 URL 来寻址。

2.HTTP（hypertext transfer protocol）基本概念

HTTP 采用 TCP 作为传输层协议；采用客户机/服务器的应用程序体系结构；同时还是个无状态协议，即服务器端并不储存客户机的状态信息（这样可能会导致重复发送）。

3.在 80 端口（？）

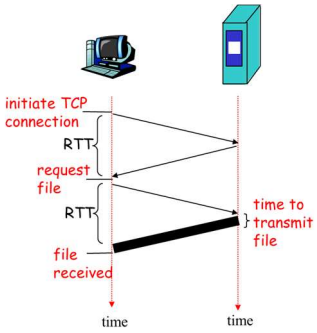
二、HTTP 的两种连接方式

1.非持久连接：每个 TCP 连接只传输一个请求报文和一个响应报文，如当有 11 个对象时，需要建立 11 次 TCP 连接，这样会给服务器带来严重的负担。

2.持久连接：一个 TCP 连接进行持久的报文传送，但是当一个连接超过一定时间都没被使用时，服务器就会关闭这个连接。

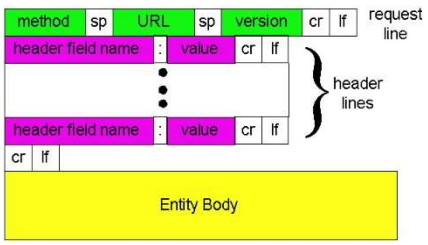
TCP 的三次握手流程：Client 向 Server 发送一个小报文段（第一次），Server 收到后再向 Client 做出确认和响应（第二次），Client 收到后再向 Server 返回确认消息，同时发送请求报文（第三次）

所以对于非持久连接，请求 1 个对象所需的时间为$2 * RTT + t_{\text{服务器传输 HTML 文件的时间}}$，而对于持久连接，只有第一次请求是上面的是上面的时间，之后的请求是$t_{\text{服务器传输 HTML 文件的时间}}$。



三、HTTP 报文格式（要会读!）

1. HTTP 请求报文



举例：

GET /somedir/page.html HTTP/1.1 要 GET 的对象的目录

Host: www.someschool.edu 目标所在主机

User-agent: Mozilla/4.0 用户代理

Connection: close 表示非持久连接

Accept-language:fr

其中：

(1)CR（Carriage return）回车；LP（line feed）换行；SP（space）空格

(2)GET: 向服务器请求数据，并返回实体部分

HEAD：向服务器请求头部（不返回实体部分，用于故障跟踪）

POST：向服务器提交数据

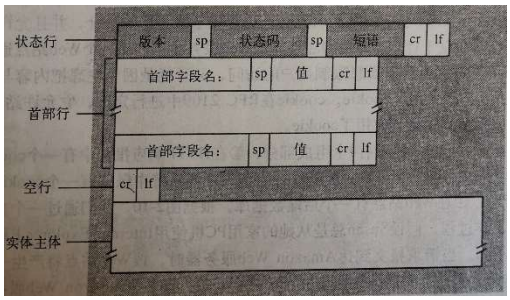
DELETE：请求服务器删除指定的页面

(3)当使用 GET 方法时主体部分为空，使用 POST 时才要写主体

(4)要 GET 的 URL 为：Host: www.someschool.edu/somedir/page.html（主机+目录）

(5)HTTP 报文无法获得 IP 信息，需要从 IP 报文中获取。

2. HTTP 响应报文



HTTP/1.1 200 OK
Connection close
Date: Thu, 06 Aug 1998 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Mon, 22 Jun 1998 ……
Content-Length: 6821
Content-Type: text/html
data data data data data ...

说明：

(1)Date 是服务器产生并发送响应的时间，不是对象创建或者最后修改的时间

(2)常用状态码：略（要记住吗）

四、无状态的改进----引入 cookie

1.功能：识别和标识用户（限制用户访问或者将内容与用户关联起来）

2.组成部分：发送和响应报文中有一个 cookie 首部行；用户端系统中保留一个 cookie 文件；服务器端有一个后端数据库。

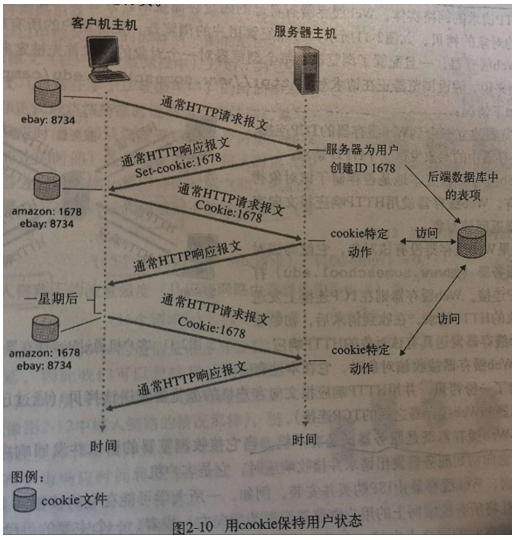


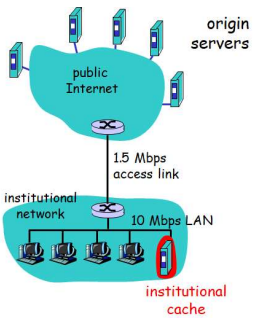
图2-10 用cookie保持用户状态

(1) 用户首次访问时，服务器端会为其生成一个 ID 存到数据库中，并发送 set-cookie 命令使用户端的 cookie 文件中写入对应的 ID

(2)以后，客户机在报文中发送 cookie，服务器通过 cookie 识别用户，在使用的过程中，会将个人信息和访问内容记录等在数据库中。

(3)这样在简化了网络服务的同时也泄露了个人的隐私信息。

五、Web 缓存（也叫代理服务器 Proxy Server）



1.实现的任务：部署在局域网中，Client 先发送请求给代理服务器，如果它存了需要的对象就直接发送给 Client；如果它没有存，则它会与存放该对象的服务器建立 TCP 连接，收到对象后，发送给 Client，并在本地存储一份 copy。

2.优点：

一是大大减少对 Client 请求的响应时间$\left(t_{响应时间}=t_{局域网时延}+t_{接入时延}+t_{因特网时延}\right)$，其中

的接入时延指的是接入链路两路由器间的时延

二是大大减少局域网与因特网接入部分的通信量以及整个因特网的 Web 流量。

3.条件 GET 的方法

(1)背景：使用 Web 缓存后，存在一个问题，即代理服务器中的 copy 不一定是最新的版本了，这时候为了就需要 Client 请求时就要用条件 Get

(2)流程：

第一步：Client 发送 Get 请求。

第二步：Proxy 收到后，向服务器发送条件 Get ，即加一句 If-modified-since: <date>

第三步：服务器返回响应报文。如果服务器对应的对象的最后一次改动时间和 date 相同，则返回状态信息 304，Proxy 直接把本地的 copy 发给 Client。

2.3 FTP：文件传输协议

一、协议特点介绍

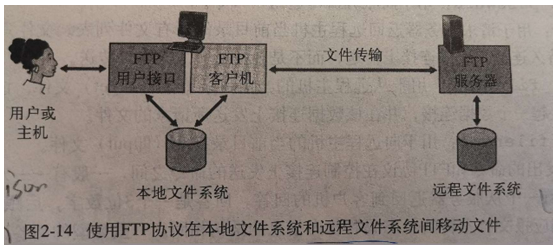


图2-14 使用FTP协议在本地文件系统和远程文件系统间移动文件

1.使用两个并行的 TCP 连接；端口 21 用于传输控制信息（持续连接），端口 20 用于传输数据（非持续性连接）

2.会话期间追踪用户的状态信息（用户在远程目录树上的当前位置）

二、整个过程

1.开始 FTP 会话前：Client 在 21 端口发起 TCP 连接，发送用户标识和口令以及改变远程目录的命令

2.Server 收到命令后：20 端口发起数据连接，发完后关闭连接

3.同一个会话期间，还需要别的文件：建立另一个数据连接（而不是在当前的连接上）

三、与 HTTP 的比较

	HTTP	FTP
相同点	都是文件传输协议	
不同点	带内发送控制信息	带外发送控制信息（与数据不同路）
	保留用户状态信息	无状态连接

2.4 电子邮件相关的协议（没讲）

2.5 DNS:因特网的目录服务

一、基本概念

1.识别主机的两种方式：主机名（如 www.baidu.com）、IP 地址。

其中：前者提供的位置信息有限，路由器处理起来困难。

2.DNS：Domain Name System 域名系统

3.DNS 协议是应用层协议，但是并不直接与用户打交道；运行在 UDP 上，使用 53 号端口。

二、DNS 提供的服务

1.主机名解析为 IP 地址

(1)流程：

第一步：Client 从 URL 中抽取出主机名，并传给正在运行的 DNS 的 Client 端程序

第二步：DNS Client 端向 DNS Server 端发送包含主机名的请求

第三步：DNS Server 端向 DNS Client 端返回对应的 IP 地址

第四步：向该 IP 对应的 HTTP Server 发起 TCP 连接。

(2)问题：带来了额外的时延

2.主机别名转为规范主机名；邮件服务器别名转为规范主机名

应注意：一个主机可以有多个别名而规范名只有一个，我们常用的都是别名。

3.负载分配：对于繁忙的站点，一个规范主机名对应的是 IP 地址集合，里面有好多服务器一起运行，DNS 负责旋转分配负载（因为 Client 总是想 IP 地址排在前面的 Server 发送请求报文）。

三、DNS 工作原理概述

1.DNS 服务器的层次结构

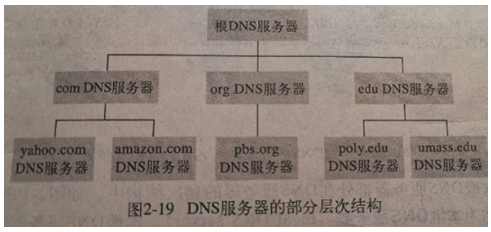


图2-19 DNS服务器的部分层次结构

(1)根 DNS 服务器：全球共 13 个，每个根服务器实际上都是冗余的服务器集群。

(2)顶级域服务器（TLD）：负责顶级域名（com、org、edu 等）和国家顶级域名（cn、uk、fr 等）

(3)权威 DNS 服务器：各组织机构自己的

另外还有一个严格来讲不属于该层级结构中的：本地 DNS 服务器。每个 ISP（如大学、公司）都有一台。

2.DNS 查询的举例

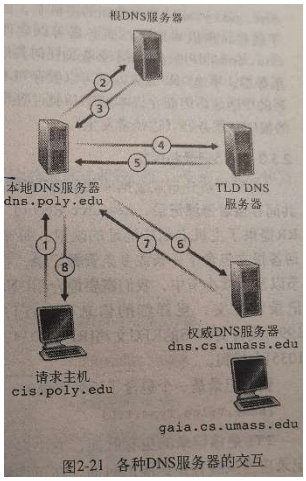


图2-21 各种DNS服务器的交互

假设 cis.poly.edu 想要知道 gaia.cs.unass.edu 的 IP 地址

第一步：向本地 DNS 发送 DNS 查询报文（报文含有目标的域名）

第二步：本地 DNS 将其转发到根 DNS。

第三步：根 DNS 注意到顶级域名为.edu，于是向本地 DNS 返回负责 edu 域名的 TLD 的地址。

第四步：本地 DNS 再将查询报文转发到对应的 TLD。

第五步：TLD 注意到 unass.edu，于是返回对应的权威 DNS 服务器的地址

第六步：本地 DNS 再将查询报文转发到对应的权威 DNS。

第七步：权威 DNS 返回对应的主机地址

第八步：本地 DNS 把地址返回给请求主机

说明：

(1)上面为了获取主机名对应的 IP，共发送了 8 份 DNS 报文，但是即使 TLD 并不一定知道权威 DNS 的 IP 地址，所以还需借助中间 DNS 来找到对应的权威 DNS，实际发送的报文数要大于 8。

(2)实际查询有两种：递归查询和迭代查询。上图中的 xx 是递归，xx 是迭代。

(3)实际中，会引入 DNS 缓存来改善时延和报文数量，但由于主机名和 IP 的映射不是永久的，所以 DNS 服务器会在一段时间后丢弃原来的缓存信息。

四、DNS 记录和报文（略）

2.6 P2P 应用（略）

2.7 TCP 和 UDP 套接字编程

一、概述

1、网络应用程序的分类

- 网络应用程序：根据 GFC 定义的标准协议实现。
- 专用的网络应用程序：不需要根据。(此时需要避开 RFC 中定义的周知端口号)

二、TCP 和 UDP 的比较

1、写代码时的区别

		TCP	UDP
有无欢迎套接字		有	无
连接与套接字的关系	关系	每次一个客户机与服务器建立一个连接，就会建立新的套接字	所有连接通过同一个套接字进入服务器
	举例：当有 N 个连接时	需要 N+1 个套接字	需要 1 个套接字（？）

2、流程比较

(1)TCP

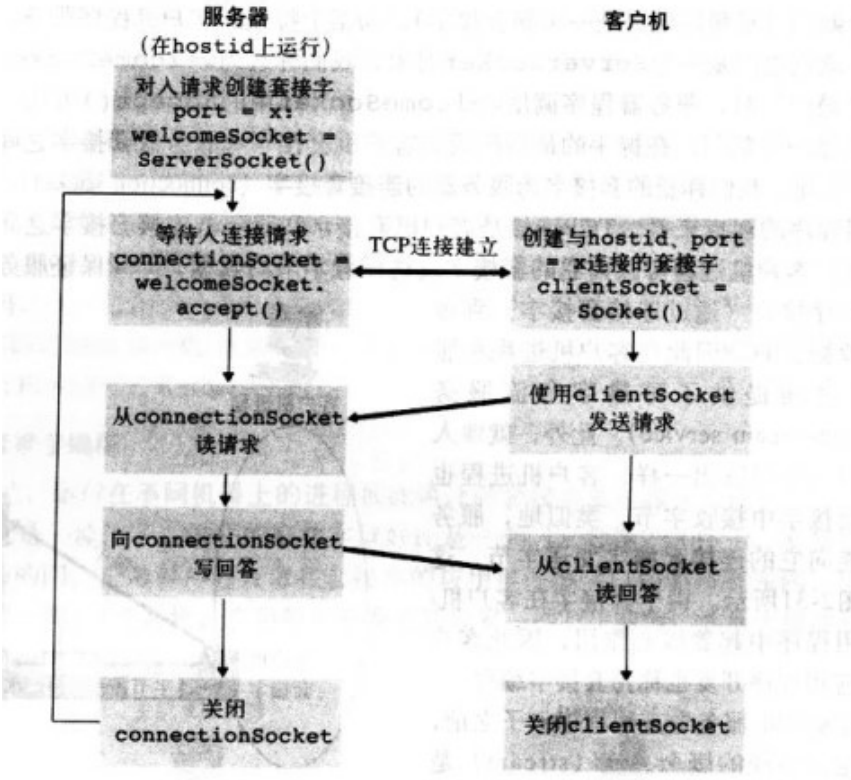


图2-32 客户机/服务器应用程序使用面向连接的运输服务

(2)UDP

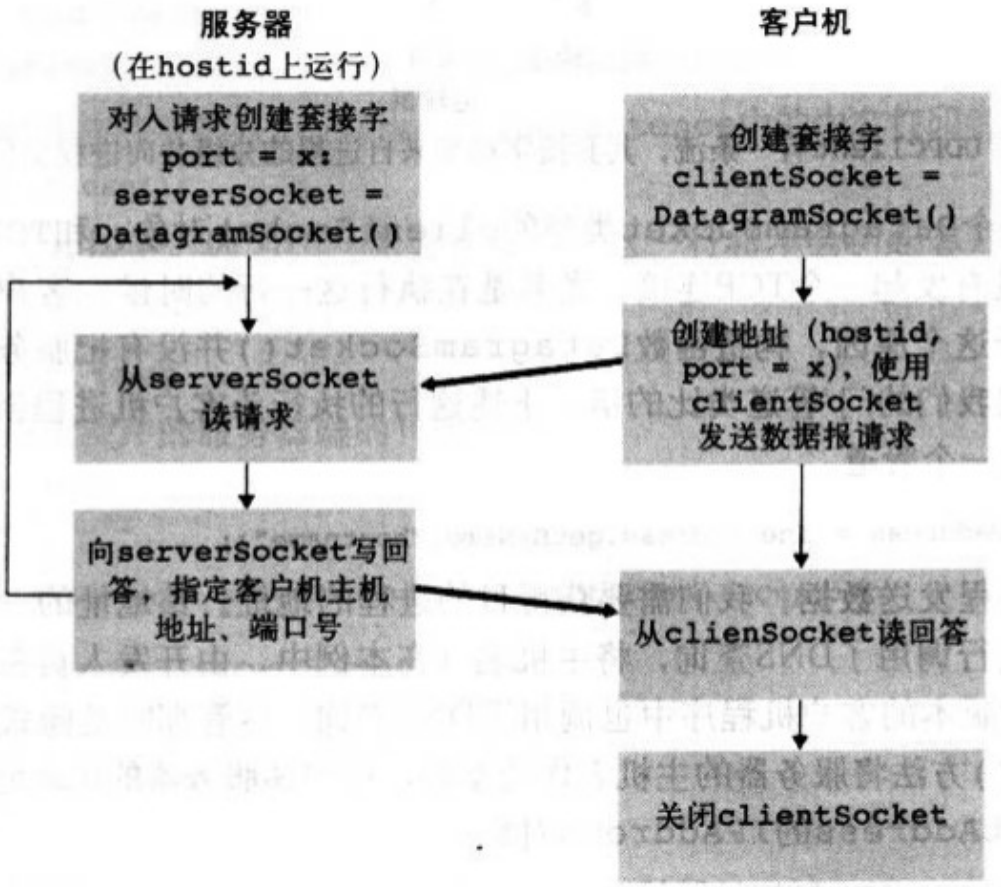


图2-34 使用无连接运输服务的客户机/服务器应用程序

第三章　传输层

3.1 传输层概述

一、传输层和网络层的关系

1.传输层协议是在端系统中而不是在网络路由器中实现的，路由器仅作用于该数据报的网络层字段。

2.传输层为运行在不同**主机上的进程**之间提供了逻辑通信，而网络层则提供了**主机**之间的逻辑通信。

3.传输层协议所能提供的服务受到了网络层协议的服务模型的限制（如时延和带宽）；但是即使网络协议在网络层不提供相应服务，传输层协议也能提供某些服务（如即使网络层可能发生丢包，传输层也能提供可靠传输服务）。

二、传输层的多路复用与多路分解

1.定义

多路复用（发端）：从源主机的不同套接字中收集数据块，并为每个数据块封装上首部信息从而生成报文段，然后传递给网络层。

多路分解（收端）：检查报文段，标识出套接字，然后将报文段定向到该套接字，然后将传输层中的数据交付到对应的套接字上，

2.套接字的识别-----端口号（英语常用#表示序号）

(1)大小：16bit，取值为 0-65535。

(2)分类：周知端口号（0-1023）：保留给诸如 HTTP、FTP 等周知应用层协议，它的使用受到严格限制；其他（未给出明确命名）

3.TCP 和 UDP 的多路复用与多路分解上的区别

	套接字的标识	对相同端口号处理的区别
TCP	二元组：源端口#、目的端口#	接收端对于同一 IP 的同一端口，当两报文具有不同的源 IP 或源端口号时， UDP：他们会被用对应于目的进程的同一个套接字定位。 TCP：他们会被用对应于目的进程的两个不同的套接字定位。
UDP	四元组：源端口#、目的端口#、源 IP、目的 IP	

补充：服务器对 TCP 的响应，并不是为每个客户机创建一个新的进程，而是创建一个线程。

三、UDP 和 TCP 的比较

UDP 的优点：见习题 R3

3.2 UDP

图3-7 UDP报文段结构

1.所实现的功能：多路复用/分解、差错检验。

2.UDP 的差错检验---校验和

(1)发送端编码-----例 P135

第一步：对报文段中前三个 16bit 字的和。求和时，遇到的任何溢出都被回卷（最高位溢出的 1 与原数的最低位相加，其余正常进位 ut）

第二步：上述结果进行反码运算

第三步：上述结果作为校验和填到第四个 16bit

PS：事实上还使用了 IP 首部的一些字段

(2)接收端检错

无错：收端前 4 个 16bit 相加，得到 16 个 1

有错：只要出现一个 0 就有错

需要注意的是：UDP 虽然提供差错检测，但是不能进行差错恢复。

(3)关于 UDP 需要差错检验的讨论（为什么不直接链路层差错检验即可）

事实上，网络层和传输层都要进行差错检验。传输层进行差错检验的原因是：首先不能保证源和目的之间所有的链路都提供差错检验；其次，即使链路层正常，报文在内存中也可能会出差错。这被称为“端到端原则”

3.较 TCP 的特点

(1)无需连接的建立：TCP 需要三次握手。

(2)无连接状态：TCP 需要在端系统中维护连接状态，包括接受和发送缓存、拥塞控制参数、序号与确认好的参数。

(3)分组首部的开销小：TCP 首部 20 字节，UDP 首部 8 字节。

4.关于 UDP 用于多媒体的争议

(1)正方：因为不需建立连接，也不做拥塞控制，所以时延小，适合于实时业务。

(2)反方：首先，UDP 没有拥塞控制，所以如果当所有用户都采用流式高比特率视频时，从而导致几乎没有分组可以到达源目的地；其次，UDP 大量进入时，TCP 会不断做流量控制，最终可能导致网络中全是 UDP。

3.3 TCP 原理之-----可靠数据传输原理

一、基本概念

1.注意点：本原理除了链路层之外实际上在网络层和链路层也都存在。

2.可靠数据传输协议的要求：首先不会有传输的数据比特受损；其次数据都是按其发送顺序进行传送；再次数据也不会因为缓冲器溢出而永远到不了目的地。应注意，可靠传输协议的下层协议不一定是可靠的（如 TCP 就建立在不可靠的 IP 上）

3.有限状态机 FSM

①初始状态：虚线指向的状态

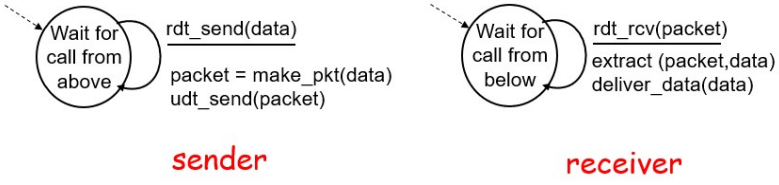
②引起状态转移的事件：写在横线上面

③状态转移后的动作：写在横线下面

④没有动作或事件：用Λ表示

二、构造可靠数据传输协议

1.完全可靠信道上的可靠信道传输：rdt1.0

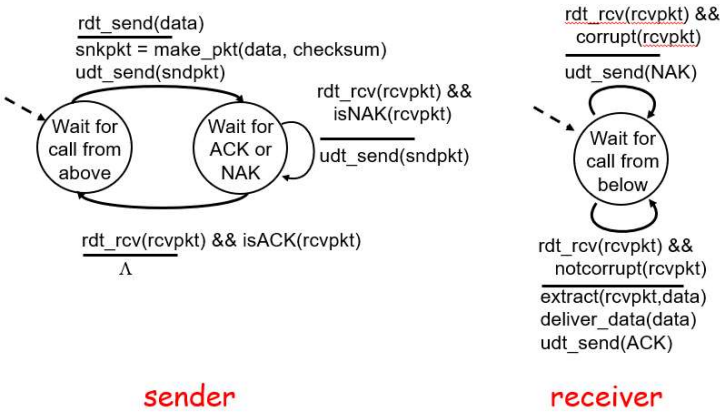


其中：deliver 表示将数据上传给较高层

2.具有比特差错信道上的可靠数据传输：rdt2.0------自动重传请求（ARQ）协议

(1)新增：ACK 和 NAK 来控制自动重传以纠错。

(2)ARQ 协议需要搭配另外三种协议来处理比特差错：差错检测、接受方反馈（即 ACK 和 NAK）、重传。



说明：①corrupt 表示包受损

②当发端在等 ACK 或 NAK 时不能从上层继续获取数据，故 2.0 又称为“停等协议”

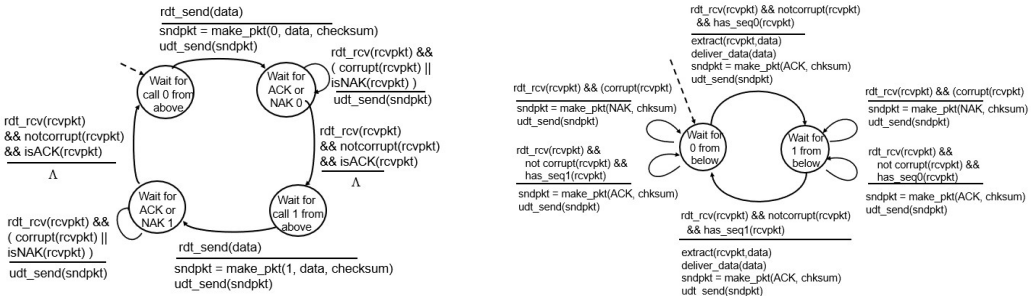
③这里存在问题：当 ACK 或 NAK 受损时，无法判断收房是否正确接收上组数据。

3.基于 rdt2.0 存在问题的改进：rdt2.1

(1)新增：当发送方收到疑似为 ACK 或 NAK 的分组时，从新发送当前数据分组。为了实现这个方法，发送方需要对数据分组进行编号（对于停等协议只需 1bit，即序号是二进制的，

所以正常情况前后两个序号一定不同，只有错了才是前后相同），这样收方才知道这是新的包还是重传。

rdt2.1: sender, handles garbled ACK/NAKs　　rdt2.1: receiver, handles garbled ACK/NAKs



说明：与原来的不同就在于多了序号 0 和 1 的区分，所以状态图为原来的两倍。

4.基于 rdt2.0 取消 NAK 的改进：rdt2.2

即收端用上次接受成功的分组的 ACK（而不是本次待接收分组）来代替 NAK（TCP 就是这样不用 NAK 了）

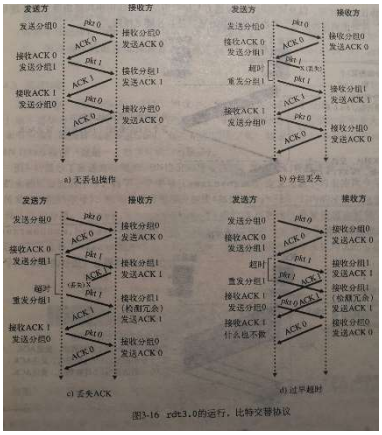
5. 具有比特差错的丢包信道上的可靠数据传输：rdt3.0 “比特交替协议”

此信道不仅有比特差错，还丢包。

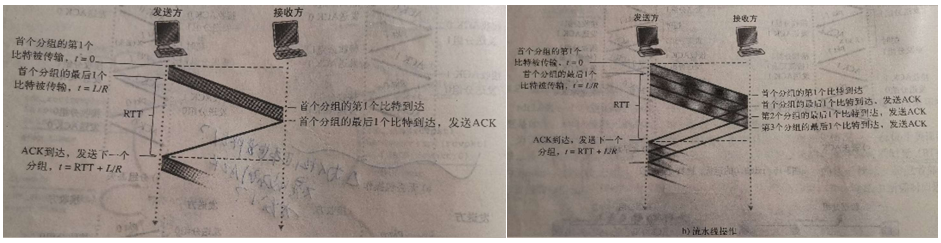
(1)新增：选择一个合适的时间，超时后仍未收到 ACK 就认为丢包，于是重传分组。

(2)问题：仍是停等协议，利用率和有效吞吐量很低（P144 例子）无法满足如今高速网络的需求。

(3)本协议对于各种情况的处理：



三、停等协议的改进-----流水线可靠数据传输协议



1.评价

(1)优点：显著地提高了利用率和有效吞吐量。

(2)需要做的改进：必须增加序号范围；收发双发都需要缓存多个分组；所需的序号范围和对缓冲的要求取决于差错恢复的方式（流水线的差错恢复有两种办法：回退 N 步和选择

重传）

2.回退 N 步（GBN）协议（也称为滑动窗口协议）

(1)内容：允许发送方发送多个分组（但是有个上限 N）而不需等待确认，但是超时后就要回退到最初那个未确认的包，重新开始发送（尽管之后的包都被发送过了）；

而在收端，之确认上次已确认的序号 N-1 的下一个序号为 N 的包，对于任何其他的包都丢弃或缓存起来（即包虽然正确，但是顺序乱了也不要，以确保数据按序交付）。

(2)GBN 协议的序号范围

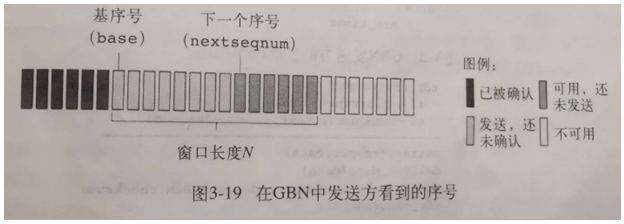


图3-19 在GBN中发送方看到的序号

若分组序号字段的比特数为 k，则序号范围是$[0, 2^k - 1]$，序号运算全用模2^k运算。

(3)扩展 FSM 描述

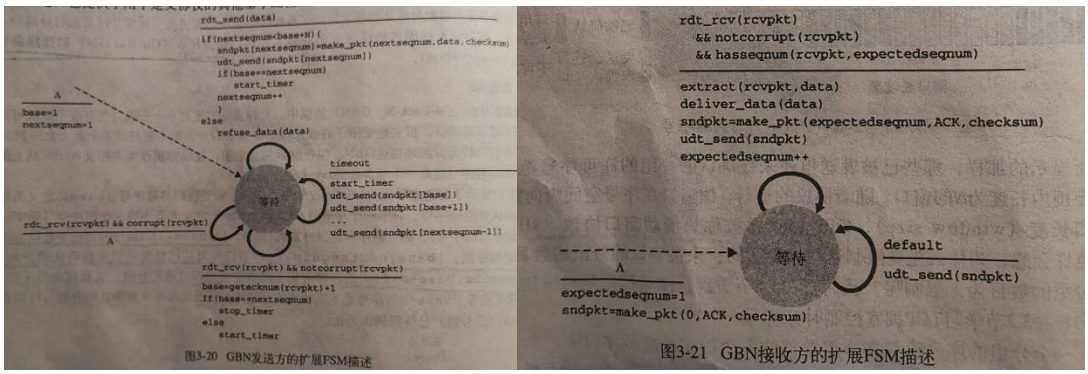
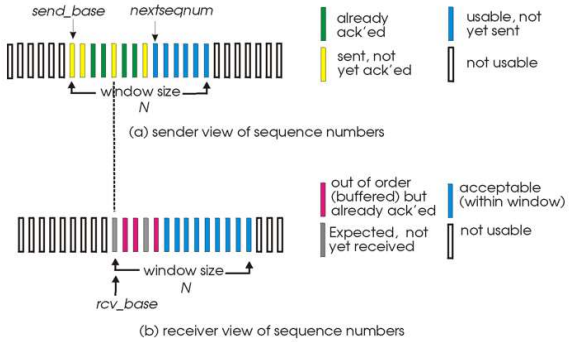


图3-20 GBN发送方的扩展FSM描述　　图3-21 GBN接收方的扩展FSM描述

3.选择重传协议

(1)内容：不再全部重传，而只重传未被确认的。



(2)发端和收端

发端

①发送数据：当前窗口内有未发送的名额，就将从上层接收到的数据打包发送，否则缓存或者返回上层。

②窗口移动：收到 ACK 后，如果是 send_base 对应的 ACK，则标记它已确认并使窗口基址一到下一个具有最小序号的分组处；如果是其他的，则只标记已确认，但不移动。

收端：

①[rcv_base, rcv_base+N-1]内的分组被接收：若是rcv_{base}被接收，则该分组及之前缓存的序号连续的分组一起交付给上层；若是其他的被接收，则被缓存起来。

②[rcv_base-N, rcv_base-1]内的分组被接收：发送 ACK（即使该分组之前已经接收过）

③其他情况：忽略该分组。

(3)存在的问题

①发送方和接收方的窗口不一致（不同步）导致可能出现无分组可发的情况

②接收方在窗口太大时，可能无法区分是新的分组还是一次重传

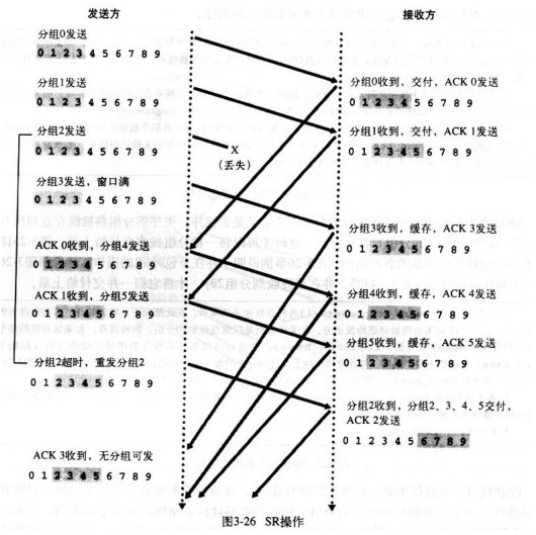


图3-26 SR操作

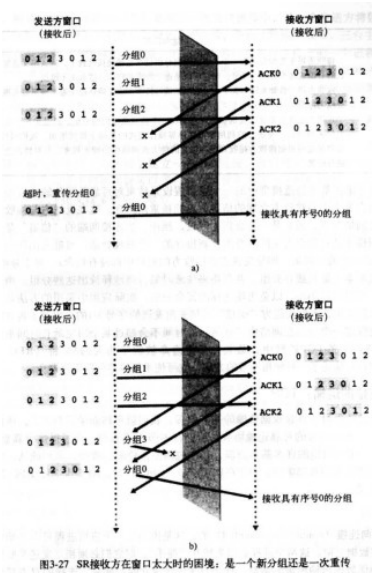


图3-27 SR接收方在窗口太大时的困境：是一个新分组还是一次重传

3.4 TCP 原理之-----拥塞控制原理

一、拥塞原因与开销（重看 173-177）

1.链路容量 R 是导致拥塞的原因。当分组到达速率不断增大时，分组传输时的排队时延也将不断增大。

2.发送方会因为超时而重传分组，可能导致网络中出现同一分组的多个 copy，转发他们占用路由器有限的带宽，导致正常的包拥塞，而当路由器的缓存溢出时，会出现丢包。

3.对于多跳传输，在重载的情况下，路由器对于跳数少的承担的载荷更多（甚至全部占有），导致跳数多的在之后的路由器上被大量丢弃，最终其端到端的吞吐量为 0。这样，网络资源就白白浪费了。

二、拥塞控制的方法

1.端到端拥塞控制：端系统通过对网络的监测（如分组丢失和时延）来判断网络是否拥塞，从而做出相应的动作来控制。(TCP)

2.网络辅助的拥塞控制：路由器向发送方提供关于网络中拥塞状态的信息，从而让端系统做出响应。(ATM ABR)

3.5 TCP 介绍

一、概述

1.TCP 中“连接”概念的辨析

既不是 CS 域中的端到端 TDM 或 FDM 电路也不是一条虚电路，其连接状态完全保留在两个端系统中。即 TCP 协议只在端系统中运行，而不在中间的网络元素（路由器和链路层交换机）中运行，所以中间要素不会维持 TCP 的连接状态（他们对连接完全不知情）

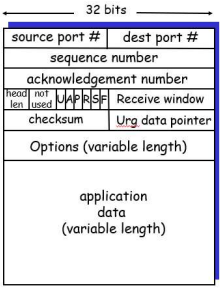
2.连接的特点：①全双工服务 ②点对点连接（无法实现多播，即一对多）

3.发送缓存和接收缓存

应用层的数据不是直接发送，而是由 TCP 引导进入该连接的发送缓存（在三次握手初期建立）中。发送时，从缓存中取出数据并放入报文段中。

注意，从缓存中取出的数据量取决于最大报文段长度 MSS（不要被名字混淆，指的是报文段中数据的最大长度，而不是整个报文段的长度）。而 MSS 通常由最大链路帧长度 MTU（源到端所有链路上能发送的最大长度）来设定。

二、TCP 报文段结构



1.接收窗口：用于流量控制，指示收方愿意接受的字节数

2.首部长度字段：指除了数据部分外的头部长度，因为选项字段，所以长度可变。

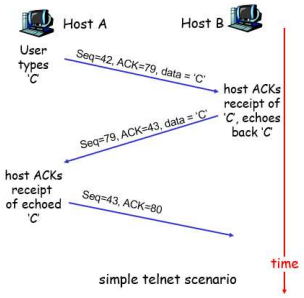
3.选项字段：用于收发双方协商 MSS 或在高速网络环境下用作窗口调节因子

4 标志字段：①ACK：指示确认字段是否有效②RST、SYN、FIN 用于连接的建立和拆除③PSH、URG：实际中并未使用。

5.序号和确认号-----实现可靠传输

(1)序号：假定数据流由一个包含 500000 字节的文件组成，MSS 为 1000 字节，数据流首字节的编号为 0，则：第一个报文段的序号为 0，第二个报文段为 1000（而不是+1），以此类推。

TCP 双方均可随机选择初始序号，这样当建立新的连接时可以减少之前连接序号的干扰。



(2)确认号：填的是期望收到的下一个字节的序号。（见图）

三、TCP 超时时间间隔的确定

1.RTT 的估计

第一步：每隔一段时间测量一个SampleRTT。

应注意的是 TCP 绝不为已重传的报文计算SampleRTT，只为传输一次的报文段测量。

第二步：求均值$EstimateRTT$和偏差$DevRTT$

$$EstimateRTT = (1-\alpha) * EstimateRTT + \alpha * \text{Sample}RTT$$

$$DevRTT = (1-\beta) * DevRTT + \beta * |\text{Sample}RTT - EstimateRTT|$$

一般取$\alpha = 0.125, \beta = 0.25$，指数加权移动平均 EWMA（越新的数据权重越大）

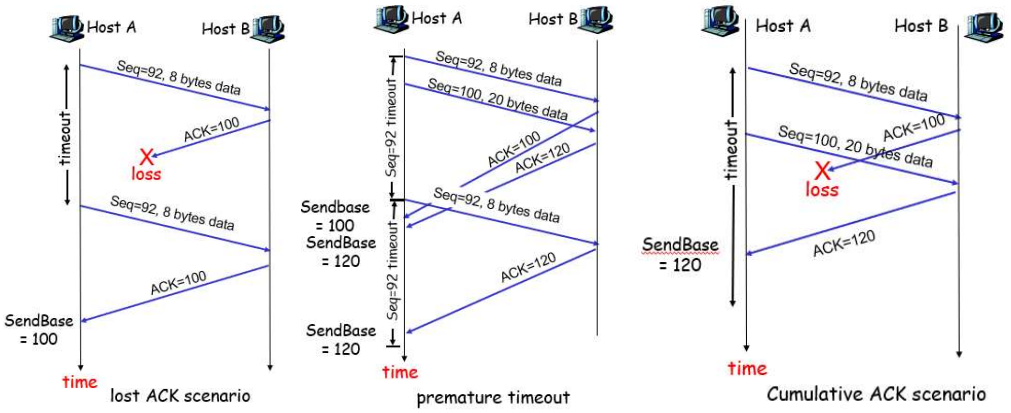
2.超时时间间隔

$$TimeoutInterval = EstimateRTT + 4 * DevRTT$$

含义：设为$EstimateRTT$加上一定的余量，当波动较小时余量也较小，反之亦然。

四、可靠数据传输的实现技术

1.TCP 对于几种错误的处理方式



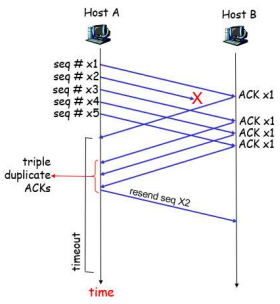
2.加倍超时间隔

当发生超时时间时，会重传并加倍超时的时间间隔（而不是由之前的时间间隔计算），如此时间间隔会呈指数的趋势增加，在某种程度上进行了拥塞控制。应注意，当收到并打包发送新的上层数据或者收到 ACK 时，超时的时间间隔由之前的公式计算。

3.快速重传

超时后重传的方式当超时周期设置得很长时会增加端到端的时延，所以通过收方发现报文丢失时（前后收到的两个报文序号不连续）发送 Duplicated ACK 来使发送方提前知道，从而不用等到超时才知道。下表是收方发送 ACK 的各种情况。（不太懂具体的意思）

事件	TCP 收方的动作
Arrival of in-order segment with expected seq #. All data up to expected seq # already ACKed	Delayed ACK. Wait up to 500ms for next segment. If no next segment,send ACK
Arrival of in-order segment with expected seq #. One other segment has ACK pending	Immediately send single cumulative ACK, ACKing both in-order segments
Arrival of out-of-order segment higher-than-expect seq. # . Gap detected	Immediately send *duplicate ACK*, indicating seq. # of next expected byte
Arrival of segment that partially or completely fills gap	Immediate send ACK, provided that segment starts at lower end of gap



情况 3 的

4.TCP 的确认方式

回退 N 步和选择重传和混合体（没懂要说什么）

五、流量控制

1.目的：控制发送方的发送速率和接收方的读取速率匹配，防止发送方速率太大导致接收方缓存溢出。（与拥塞控制相区分，后者是网络层的缓冲区满）

2.实现：通过报文段中的接收窗口字段(1)来告诉发送方，收方还有多少可用的缓存空间。然后发送方通过(2)来控制发送，RTT 内最多可以发送$RcvWindow$个字节的数据。

$$\begin{cases} RcvWindow = RcvBuffer - (LastByteRcvd - LastByteRead) \\ LastByteRcvd - LastByteRead \leq RcvBuffer \end{cases} \quad (1)$$

$$LastByteSent - LastByteAcked \leq RcvWindow \quad (2)$$

3.存在的问题及解决

假设 A 是发送方，B 是接收方。当 B 接收缓存已满时，会通知 A $RcvWindow = 0$，之后 A 不再发送，但是由于 TCP 仅在它有数据或确认要发时才会发送报文，这样当 B 开始读取数据空出接收缓存时，并不会通知 A，最终导致虽然 B 的缓存区已经空了但是 A 也不发送新的数据。

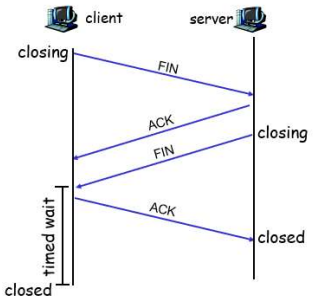
解决方法是，当 B 的接收缓存满时，A 继续发送只有一个字节的报文段。

六、TCP 连接管理

1.连接的建立-----三次握手

	内容	报文段的特点
第一步	客户机端创建 SYN 报文段并封装到 IP 数据报中发给服务器端	1.SYN 置为 1 2.客户机端选择一个起始序号 Client_isn，填到报文段的序号字段中。
第二步	服务器收到后，为该 TCP 分配缓存和变量，并向客户机发送允许连接的 SYNACK 报文段。	1.SYN 置为 1 2.确认号字段被置为 Client_isn+1（ACK） 3.服务器选择自己的起始序号 Server_isn
第三步	客户机收到后，也为该连接分配缓存和变量，并发送另一个带数据的报文段，对服务器的 SYNACK 进行确认	1.SYN 置为 0（连接建立后该位都是 0）
注意	前两个报文段不承载“有效载荷”（即不包含应用层数据），而第三个报文段可以承载。	

2.连接的拆除



七、拥塞控制

1.对于拥塞的判断（丢包的两种可能）：①出现超时 ②收到收方的 3 个冗余 ACK

2.实现：引入拥塞窗口$CongWin$，限制每个 RTT 内允许发送$CongWin$个字节数，从而控制了传输速率。

$$LastByteSent - LastByteAcked \leq CongWin$$

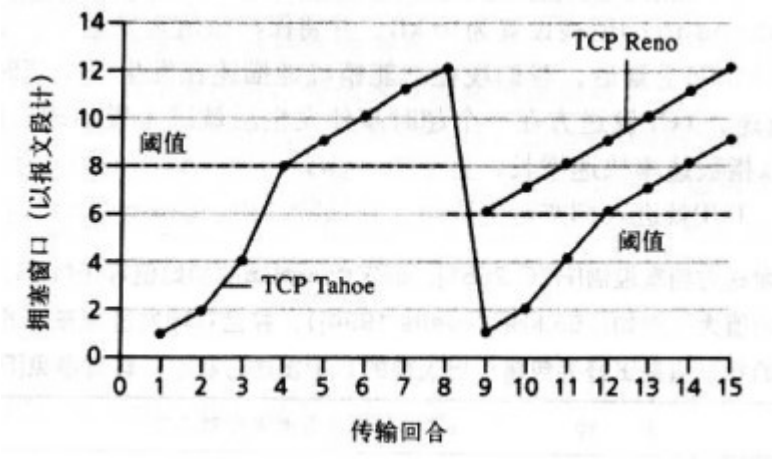
3.对拥塞做出的响应（结合作业题 Problem 34）

事件	状态	TCP 发送方拥塞控制动作
收到前面未确认的 ACK	慢启动（SS）	1.CongWin=CongWin+MSS 2.当 CongWin>=Threshold 时，进入拥塞避免状态。
	拥塞避免（CA）	1.CongWin=CongWin+MSS/CongWin
收到冗余 ACK	SS 或 CA	1.CongWin 和 Threshold 均不变 2.对收到的冗余 ACK 计数
收到三个冗余 ACK		1.Threshold= CongWin/2 2.CongWin= Threshold 进入拥塞避免状态
超时		1.Threshold= CongWin/2 2.CongWin= 1 进入慢启动状态

说明：①Threshold 用于管理时处于 SS 状态（CongWin<Threshold）还是处于 CA 状态（CongWin=Threshold），初值设为一个很大的数。

②慢启动的每个 RTT 内 CongWin 翻倍也即该 RTT 内每收到一个 ACK，CongWin+1；拥塞避免的每个 RTT 内 CongWin+1 也即该 RTT 内每收到一个 ACK，CongWin+1/N。

③TCP Reno 版本在收到三个冗余 ACK 采用了快速恢复的策略（即不进入慢启动，因为收到三个冗余 ACK 是“快速重传”，只能说明某个分组丢失，大部分分组还是能顺利通过网络，而超时就不一样了）



八、公平性问题

1.不同 TCP 连接间的公平性问题

如果所有的连接具有相同的 MSS 和 RTT 且链路中没有 UDP，则最终所有连接尽管开始分配的不一样，最终经过拥塞控制可以分到相同的吞吐量。（P186 的分析）

如果所有的连接具有不同的 MSS 和 RTT 且链路中没有 UDP，则最终具有较小 RTT 的连接能在链路中抢到更多的可用带宽，从而获得更高的吞吐量。

2.并行 TCP 连接带来的的公平性问题

如果某个应用使用多条 TCP 并行连接，这样每条连接都能公平地获得可用带宽，最终导致该应用获得的总带宽大于使用单路 TCP 连接的应用，就导致了不公平。

比如当中链路容量为 R，已有 9 个 TCP 时，新加入一个 TCP 连接会占用总的 1/10，但是如果某个新加应用开了 11 个 TCP 连接，那它就独占了一半的链路容量。（IDM 的并行下载）

3.TCP 和 UDP 间的公平问题

UDP 不会进行拥塞控制，所以可能导致 UDP 的流量压制 TCP 的流量。

第四章　网络层

4.1 概述

一、基本概念

1、网络层的功能

- 转发（forward）：将分组从路由器的一个输入链路接口转移到合适输出链路接口
- 选路（route）：决定端到端的路径（注意转发和选路的区别）
- 连接建立：在网络层数据开始流动之前，沿着从源到目的地的路径彼此握手

2、转发表：通过**选路**算法更新路由表，存一个输入（分组首部某一字段）和一个输出（输出链路接口）。从而可以通过读取分组头部的特定字段来确定输出链路，从而完成**转发**

3、分组交换机

(1)定义：根据分组首部字段中的值，从输入链路接口到输出链路接口传送分组的通用分组交换设备。

(2)分类

- 链路层交换机：基于链路层字段中的值做转发决定
- 路由器：基于网络层字段中的值做转发决定

二、网络服务模型

1、定义：定义网络发送端系统和接收端系统之间分组的端到端传输特性。

2、网络层可以提供的服务

- 确保交付：确保分组能顺利到达目的地
- 具有时延上界的确保交付
- 有序分组交付
- 确保最小带宽：发送速率低于特定值时，分组就不会丢失且会在预定的时延内到达
- 确保最大时延抖动
- 安全性服务：机密性、数据完整性、源鉴别服务
- ……

3、Internet 和 ATM 所提供的服务

(1)Internet：尽力而为服务

实际上是最低要求的网络服务模型：分组时延得不到保证；不能保证接收分组顺序和发送时一致；传送的分组也不一定能够交付成功。

(2)ATM：多重服务模型

可以为不同连接提供不同类别的服务：恒定比特率（CBR）；可用比特率（ABR）

(3)比较

表4-1　因特网、ATM CBR和ATM ABR服务模型比较

网络体系结构	服务模型	带宽保证	无丢失保证	排　序	定　时	拥塞指示
因特网	尽力而为	无	无	任何可能的顺序	不维持	无
ATM	CBR	保证恒定速率	是	有序	维持	不出现拥塞
ATM	ABR	保证最小速率	无	有序	不维持	提供拥塞指示

4.2 虚电路和数据报网络

一、基本概念

网络层也像传输层一样可以提供无连接服务或者面向连接服务。

1、两种服务

- 无连接服务：虚电路网络（ATM、帧中继）
- 面向连接服务：数据报网络（Internet）

2、网络层和传输层提供服务的不同之处

- 网络层提供的是主机之间的服务；传输层是两主机进程间的服务
- 现有的网络体系结构（Internet、ATM、帧中继等）都不能同时提供无连接和面向连接，而传输层的则可以
- 对于面向连接的服务而言，二者是完全不同的。传输层的服务只存在于端系统中，而网络层的服务还存在于路由器中。

二、虚电路网络（VC）

1、一条虚电路的组成

- 源和目的主机间的路径（一系列链路和路由器）
- VC 号：沿着该路径的每段链路的号码
- 沿着该路径的每台路由器中的转发表项

2、路由器需要完成的功能

(1)VC 号的更换（举例说明）

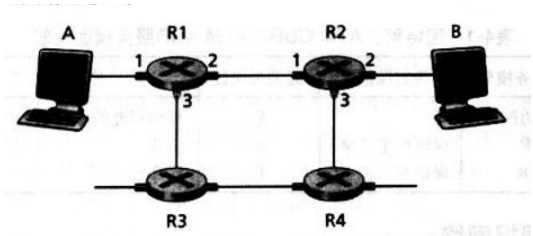


图4-3 一个简单的虚电路网络

假设选定的路径是 A-R1-R2-B，该路径上的三条链路分别分配 VC 号为 12、22、32

则 A-R1 段，分组首部的 VC 值为 12；R1-R2 段，分组首部的 VC 值为 22；R2-B 段，分组首部的 VC 值为 32。对应的路由表如下（以 R1 为例）

入接口	入VC号	出接口	出VC号
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87
...	...	...	...

为什么要一直改变 VC 号而不一直保持

- 逐链路改变 VC 号可以减少 VC 字段的长度
- 保持 VC 需要路由器交换和处理相当大量的报文来约定一个共同的未用过的 VC 号

(2)为进行中的连接维持连接状态信息（将 VC 号和输出端口联系起来）

- 每当跨越一台路由器创建一个新的连接时，就必须将一个新的连接项加到该路由器的转发表中；
- 每当释放一个链接时，就必须删除该项

注意：即使没有 VC 号的转换，也有必要维持连接状态信息。

3、虚电路的不同阶段

(1)虚电路的建立

- 发端传输层向网络层指定收方的地址，等待建立虚电路
- 网络层决定从收端到发端的路径（要通过的链路和路由器），确定一个 VC 号
- 网络层在沿着路径的每台路由器的转发表中增加一项，从而建立连接

(2)数据传送

(3)虚电路拆除

- 其中一方通知网络层想中止虚电路
- 网络层通知另一方端系统结束呼叫
- 更新路径上每台路由器中的转发表（删除一项）

说明:

①在虚电路建立期间，网络层还可以预留路径上的资源（如带宽）

②从过程中也可以看出和传输层建立连接的异同点。传输层的建立只涉及端系统双方，其中的路由器对连接并不知情；而对于网络层而言，路径上的所有路由器都知晓连接。

③端系统向网络发送指示虚电路启动与中止的报文，以及路由器间传递的用于建立虚电路的报文称为信令报文；用来交换这些报文的协议称为信令协议。

三、数据报网络

1、方式：每当一个端系统要发送分组时，它就为该分组加上目的地端系统的地址，然后将分组推进到网络中。各路由器采用最长前缀匹配原则来确定。

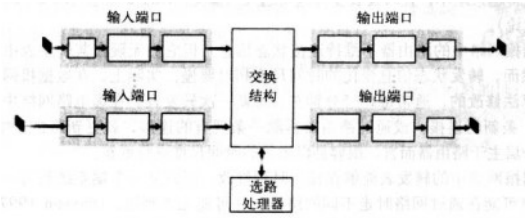
2、与虚电路的比较

- 数据报网络中的路由器不维持连接状态信息，仅维持转发状态信息，但也是通过选路算法几分钟更新一次。
- 虚电路网络不仅维持连接状态信息，还维持转发状态信息，并且每次建立和拆除连接都要更新（微妙量级）。

4.3 路由器的组成

一、路由器的组成概述

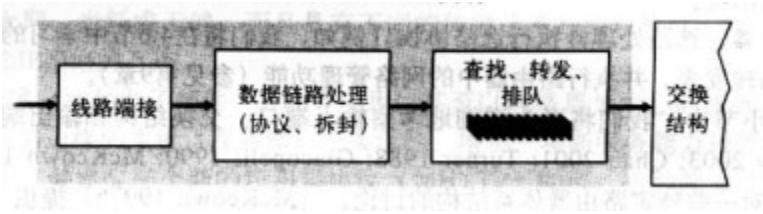
组成：输入端口、交换结构、输出端口、选路处理器



二、输入端口

1、输入端口的功能

- 物理层功能：将一条输入的物理链路端接到路由器
- 数据链路层功能：与位于入链路远端的数据链路层功能交互
- 网络层功能：完成查找与转发以便转发到路由器交换结构的分组能出现在适当的输出端口



说明：

①许多路由器的输入端口都存了一份由选路处理器计算得到的转发表的 copy 并且保持更新，这样便可以在输入端口完成转发而无需调用中央选路处理器。而当使用一台电脑作为路由器时，则无法在输入端口（对应为网卡）确定转发，而需要调用中央选路处理器（对应为 CPU）

②控制分组从输入端口转发到选路处理器

③实际上，多个端口经常被集中到路由器中的一块线路卡上

2、路由表的查找

(1)要求：希望输入端口的处理速度能够达到**线路速度**，即执行一次查找的时间短于从输入端口接收到一个分组的时间

(2)查找方法

- 线性搜索：不高效
- 二分查找树查找：仍不够快
- 内容可寻址内存 CAM：一个 32bit 的 IP 地址提交给 CAM 后，可以以基本上常数的时间返回该地址对应的转发表表项的内容
- 将最近访问的转发表表项存在高速缓存中
- ……

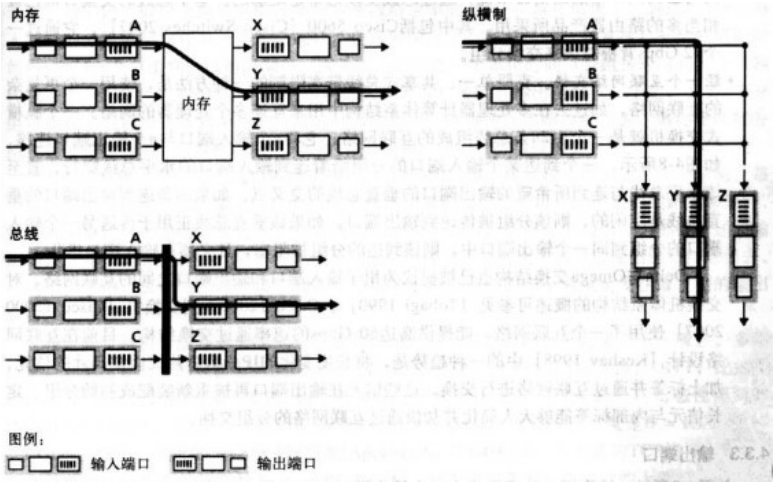
(3)分组的阻塞：分组进入交换结构时，有其他输入端口的分组正在使用，于是要排队

三、交换结构（三种）

1、经内存交换：输入端口和输出端口的交换时在 CPU 的直接控制下完成的。应注意的是，若内存带宽为每秒处理 B 个分组，则总的转发吞吐量必小于 B/2

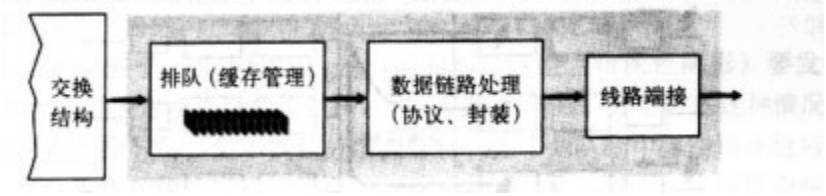
2、经一根总线交换：此时不需要选路处理器的干预，但是交换带宽受总线速率限制

3、经一个网络交换（纵横制）：带宽较前两个更大。仅当垂直总线空闲时，该分组才被传送到输出端口，否组被阻塞。



四、输出端口

1、结构与功能：和输入端口对称



五、分组的排队和丢包

1、丢包的实际位置：取决于流量负载、交换结构的相对速率和线路速率

2、输出端口的排队

(1)原因：没有足够的内存来缓存入分组（分组交付给**输入端口**的速率超过**输出链路**）

(2)解决办法：主动队列管理 AQM

以下是几种策略：

- 弃尾：队满后丢弃新到的分组
- 删除一个或多个正在排队的分组以便为新来的分组腾出空间。
- 在缓存满前便丢弃（或在首部加标记）新到的分组。

(3)AQM 的一种实现算法----随机早期检测 RED

为队列长度维护着一个加权平均值。该平均值小于最小阈值 min_th 时，接收新到的分组；该平均值大于最大阈值 max_th 时，该分组被标记或丢弃；介于之间时，以一定的概率标记或丢弃（该概率是两个阈值以及平均队列的函数）

(4)排队后的处理---分组调度程序

利用某种原则从正在排队的分组中选择一个来传送：先到先服务（FCFS）、加权公平排队（WFQ）

3、输入端口的排队

(1)原因：交换结构速率过低、多个分组发往同一个输出端口时。

- 交换结构速率：交换结构能够从输入端口到输出端口移动分组的速率。
- 假设有 n 个输入端口和 n 个输出端口。仅当速率至少为输入线路速率的 n 倍时，输入端口处才不会排队。

(2)一种特殊的排队---线路前部阻塞 HOL

原因：本来没有竞争的分组，因为前一个分组在排队，所以它也被动排队。

4.4 网际协议（IP 协议）：因特网中的转发和编址

一、概述

1、因特网的网络层组件：IP 协议、选路组件、报告数据包中的差错和对某些网络层信息请求进行响应的设施（ICMP）

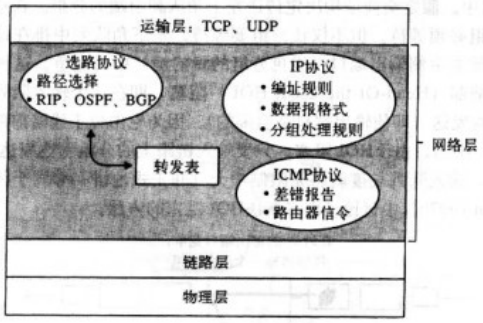
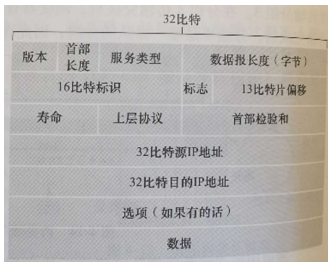


图4-12 因特网网络层的内部

二、数据报格式



1、关键字段说明

无选项时，共 20 字节。若承载的是 TCP 报文段，则每个数据报共承载 40 字节（IP 首部和 TCP 首部各 20 字节）

- 版本号：IPv4 还是 IPv6
- 首部长度：因为有可选项，所以需要它来表示数据部分从哪里开始
- 服务类型 TOS：4.1 节提到的那些服务
- 数据报长度：IP 数据报的总长度（该字段 16bit，所以数据报理论最大长度为 65535）
- 标识、标志、片漂移：与 IP 分片有关（IPv6 不允许在路由器上分片）
- 寿命 TTL：确保数据包不会永远在网络中存在，每经过一台路由器就减 1，当为 0 时就丢弃
- 协议：仅在数据报到达终点时才会使用该字段，表明交给哪个传输层协议
- 首部检验和：首部每两个字节当作一个数，用反码运算求和，再将和的反码存在这里
- 选项：会导致路由器处理时间变化大，所以 IPv6 已经不再使用
-

说明：为什么 TCP/IP 在传输层和网络层都执行差错检验

①网络层只对 IP 首部进行检验，而传输层是对整个报文段进行的

②TCP/UDP 和 IP 不一定属于同一个协议栈

2、IP 数据报分片

(1)分片原因：链路层协议规定了**最大传输单元** MTU，规定了所能承载分组的最大长度。而发送方和接收方之间的每条链路可能使用不同的链路层协议从而具有不同的 MTU，会导致某些链路的 MTU 小于 IP 数据报的长度而无法承载。

(2)数据包的分片

- 分片位置：MTU 不够处的路由器
- 举例说明：假设有 4000 字节的数据报（20 字节 IP 首部和 3980 字节数据荷载）转发到 MTU 为 1500 字节的链路上

表4-2 IP片

片	字节数	标识	偏移	标志
第1片	IP数据报的数据字段中的1480字节	identification = 777	offset = 0（表示插入的数据开始于字节0）	flag = 1（表示后面还有）
第2片	1480字节数据	identification = 777	offset = 185（表示插入的数据开始于1480字节。注意 185 · 8 = 1480）	flag = 1（表示后面还有）
第3片	1020字节（= 3980 − 1480 − 1480）数据	identification = 777	offset = 370（表示插入的数据开始于2960字节。注意 370 · 8 = 2960）	flag = 0（表示这是最后一片）

其中：

①字节数：除了最后一片外，必须是 8 的整数倍

②标识号：用于确定各片原来属于同一分组

③偏移：开始字节数**除 8**

④标志：用于标识该片是否是结尾（结尾为 0）

(3)分片的组装

收端仅当接收当所有片时，才会组装好交给传输层，否则就丢弃。这时，传输层如果是 TCP，则会让对方重发（联系上一章的内容）。

应注意：组装只在端系统完成，而不再网络路由器中完成。

(4)分片的缺点（所以 IPv6 不用了）

- 加大开销；也使得路由器需要和端系统更复杂，前者需要将数据报分成适当大小的片，后者需要重新组装
- 分片可能会引发 DoS 攻击

三、IPv4 编址

1、基本概念

(1)接口：主机与物理链路之间的边界。IP 地址实际上是与一个接口相关联，而不是与包括该接口的主机或路由器相连。

(2)IP 的点分十进制记法：如11000001 00100000 11011000 00001001记为193.32.216.9，

(3)子网掩码：如 223.1.1.0/24，24 表示前 24 位是子网地址，后 8 位是主机地址。

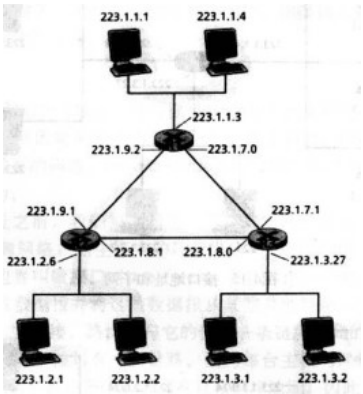


图4-17 3台路由器互连6个子网

(4)因特网的地址分配策略：

- 分类编址：即网络部分分别为 1、2、3 字节的 A、B、C 类网络
- 无类别域间选路 CIDR：a.b.c.d/x 来划分子网

(5)广播地址：255.255.255.255。全 1，报文只会在子网内传送，不会出子网（路由器识别到该地址不转发给外网）

2、获取一块地址

组织从 ISP 中获得子网掩码

为了获取一块IP地址用于一个组织的子网，网络管理员也许首先会与其ISP联系，ISP会从已分给它的更大地址块中提供一些地址。例如，某ISP也许已被分配了地址块200.23.16.0/20。该ISP可以依次将该地址块分成8个长度相等的较小地址块，为其支持的最多8个组织中的一个分配一小块，如下所示。（为了便于查看，我们已将这些地址的网络部分加了下划线。）

ISP的地址块	200.23.16.0/20	11001000 00010111 00010000 00000000
组织0	200.23.16.0/23	11001000 00010111 00010000 00000000
组织1	200.23.18.0/23	11001000 00010111 00010010 00000000
组织2	200.23.20.0/23	11001000 00010111 00010100 00000000
…	…	…
组织7	200.23.30.0/23	11001000 00010111 00011110 00000000

3、获取主机地址：动态主机配置协议 DHCP

组织为组内的主机和路由器分配**临时** IP 地址（而非永久 IP）

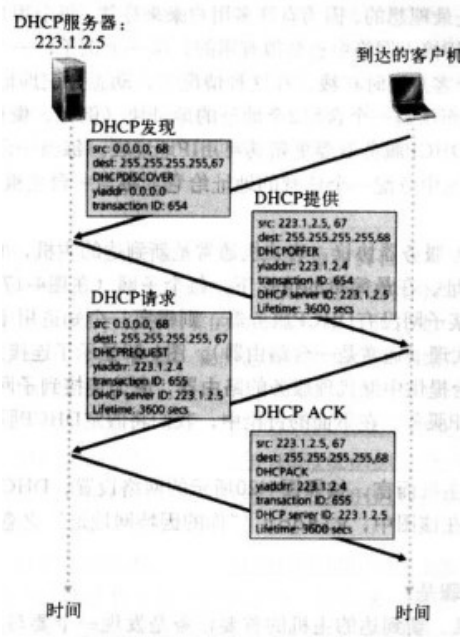
(1)适用于：住宅区因特网接入和无线局域网（有主机频繁地加入和离开网络）

(2)协议特点

- 即插即用协议
- 客户机/服务器协议：客户机通常是新加入的主机，需要获得网络的配置信息；服务器是每个子网中的 DHCP 服务器，如果某个子网没有 DHCP 服务器，则需要一个 DHCP 中继代理（通常是一台路由器）

(3)协议步骤

- DHCP 服务器发现：客户机通过 UDP 在 67 端口发送 DHCP 发现报文来发现服务器。因为不知道发给谁，所以目的地址为 255.255.255.255，源地址为 0.0.0.0
- DHCP 服务器提供：服务器收到发现报文后，用一个 DHCP 提供报文对客户机做出响应。此时仍然使用广播地址 255.255.255.255。报文包括发现报文的事务 IP、网络掩码、IP 地址租用期。
- DHCP 请求：一个子网内可能有多个服务器，所以客户机可能会收到多个报文，客户机从其中选择一个，并向其发送 DHCP 请求报文做出响应，向服务器确认配置参数。
- DHCP ACK：服务器对请求报文做出相应，证实所要求的参数。



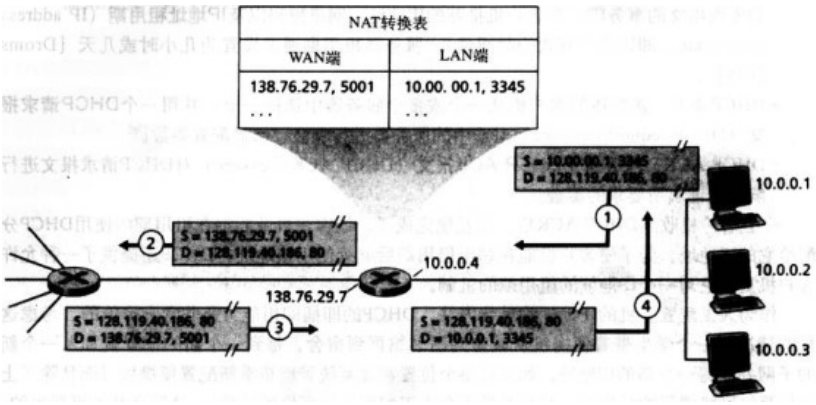
4、网络地址转换 NAT

(1)产生背景：缓解 IPv4 地址不足，用于内网和外网间通信

(2)转换过程（举例）

假设 10.0.0.1 想要请求 128.119.40.186 的 Web 服务器的 Web 页面（80 端口）

- 主机 10.0.0.1 指派了 3345 端口发送到 LAN
- NAT 路由器收到数据报后，为其生成新的端口号 5001，将 IP 地址改为广域网接口的 IP 地址 138.76.29.7，并加入 NAT 转换表中，然后向服务器发出
- 服务器收到后向 138.76.29.7 的 5001 端口发送响应数据报
- NAT 路由器再查表转换回请求主机的 IP 和端口号。



(3)缺点

- 不符合规范（见书）
- 妨碍 P2P 应用程序：若 A 向 B 发起一个 TCP 连接，而 B 在 NAT 后面，他就不能作为服务器并接受 TCP 连接。需要 NAT 穿越技术（见书）。

5、UPnP

(1)应用：NAT 穿越

(2)内容：见书

四、ICMP：互联网控制报文协议

写在 IP 数据报的数据字段处。

1、功能：用于主机和路由器彼此交互网络层信息。

2、结构：类型字段、编码字段。同时包含引起该 ICMP 报文首次生成的 IP 数据报的首部和钱 8 字节内容。

3、报文类型

ICMP类型	编　码	描　述
0	0	回显回答（对ping的回答）
3	0	目的网络不可达
3	1	目的主机不可达
3	2	目的协议不可达
3	3	目的端口不可达
3	6	目的网络未知
3	7	目的主机未知
4	0	源抑制（source quench，拥塞控制）
8	0	回显请求
9	0	路由器通告
10	0	路由器发现
11	0	TTL过期
12	0	IP首部错误

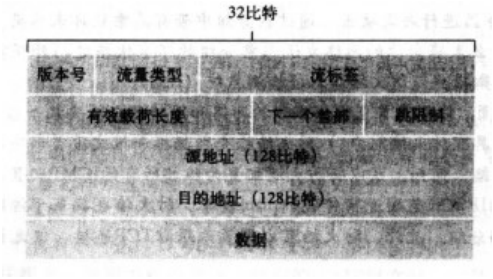
4、应用：差错报告、源抑制报文（强制主机减小其传输速率）、Traceroute 程序

五、IPv6

1、重要变化

- 地址容量扩大：IP 地址长度从 32bits 扩大到 128bits。此外除了单播和多播地址外，还引入了任播地址。
- 40 字节首部：舍弃了 IPv4 的许多首部字段，如分片/重新组装、首部校验和、选项。
- 引入流标签和优先级

2、数据报格式



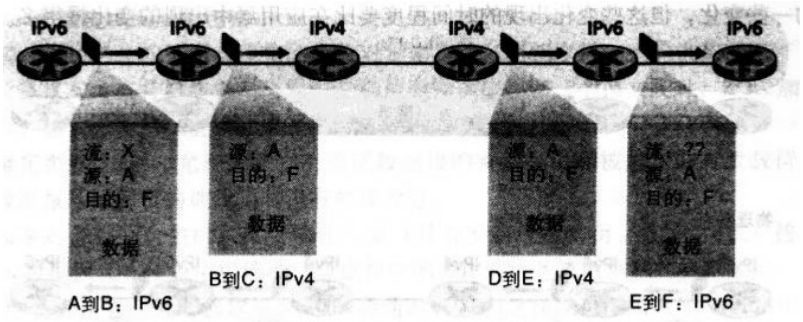
- 流量类型：与 IPv4 的 TOS 字段类似
- 有效荷载长度：数据部分的字节数
- 下一个首部：表示需要交给哪一个传输层协议
- 跳限制：和 TTL 类似

3、IPv4 向 IPv6 的迁移的方案

(1)指定一个日期（标志日），在那一天所有机器都关机并升级到 IPv6。这种方法不现实。

(2)双栈：IPv6 节点也具有 IPv4 的功能，称为 IPv6/IPv4 节点。

这种方法有一个缺陷：当中间某一个节点仅支持 IPv4 时，IPv6 报文需要转为 IPv4 的，此时会丢失 IPv6 特定的字段（如流标签）



(3)建隧道：当中间有 IPv4 节点时，整个 IPv6 报文加到 IPv4 的数据字段中。

- 隧道：两台 IPv6 路由器之间的 IPv4 路由器的集合

六、IP 安全性概述------IPsec（第八章）

4.5 选路算法（好像没讲）

掌握分类？？

4.6 因特网中的选路（好像没讲）

4.6 节讲了 RIP 的整个流程

4.7 广播和多播选路（好像没讲）

第五章、链路层和局域网

5.1 链路层：概述和服务

一、概述

1、链路层信道：

- 广播信道：许多主机被连接到同一个信道下。如以太网，需要多路访问协议
- 点对点通信链路：如拨号住宅主机。需要 PPP 协议或 HDLC 协议。

2、链路：沿着通信路径相邻节点的通信信道。节点指主机和路由器。

3、链路层协议举例：以太网、802.11 无线 LAN（WiFi）、令牌环、PPP。有时候 ATM 也被认为是链路层协议。

4、链路层的特点：数据报在同一路径的不同链路上可能由不同链路层协议承载。

二、链路层提供的服务

1、功能：将数据报从当前节点转移到链路相连的下一节点。

2、提供的服务

- 成帧：将网络层的数据报加上首部封装成帧
- 链路接入：用媒体访问控制协议（MAC）来控制帧在链路上的传输规则。当是点对点链路时可以不需要 MAC；当是广播链路时，需要 MAC 协议来协调各个节点。
- 可靠交付：保证无差错地经链路层移动每个网络层的数据报。常用于无线链路；有线链路由于不易出错，通常不提供可靠交付。
- 流量控制：防止接收缓冲区溢出导致丢失分组。
- 差错检测：比特差错通常是由信号衰减和电磁噪声导致的，链路层差错检测是用硬件实现的。
- 差错纠正
- 半双工和全双工

注意点：这里的流量控制和可靠交付与传输层的是有区别的。传输层提供的是端到端的服务；链路层提供的是相邻节点之间的。

三、链路层的实现位置

链路层是软件和硬件的结合体

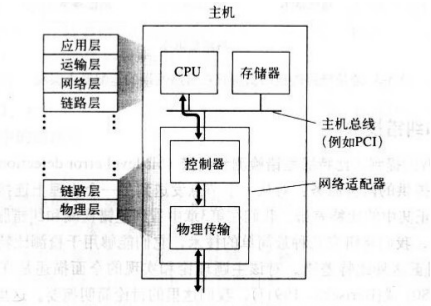


图5-2 网络适配器：它与其他主机组件及其对协议栈功能的关系

- 软件部分：运行在 CPU 上的软件完成较高层次的链路层功能，如发端的从网络层接收数据报、装配链路层寻址信息、激活控制器硬件；收端的响应控制器的中断、处理差错、数据报传给网络层
- 硬件部分：网络适配器（网络接口卡，NIC）。NIC 中的控制器实现成帧、链路接入、流量控制、差错控制等低层次链路层功能。

5.2 差错检测和纠错技术

一、奇偶校验

1、一维奇偶校验存在的问题：无法检测出偶数个错误

2、改进：二维奇偶校验

二、检验和方法

1、方法：数据每 16bits 当作一个整数对待并求和，他们和的反码作为校验和填入。收方将包括校验和的所有 16bits 相加再取反码，若全 1 则不出错。

2、评价：需要的额外分组开销较小，但是差错保护能力较弱。

三、循环冗余校验码 CRC

5.3 多路访问协议

一、概述

1、应用场景：广播链路中用于规范各个节点在共享的广播信道上的传输行为，以防止碰撞。

2、多路访问协议的分类：信道划分协议、随机接入协议、轮流协议

3、希望该协议具有的特性（假设广播信道吞吐量为 R bps）

- 当只有一个节点发送数据时，该节点具有 R bps 的吞吐量
- 当有 M 个节点要发送数据时，各节点吞吐量接近 R/M bps
- 协议是分散的：不会因为主节点故障而使整个系统崩溃
- 协议是简单的：实现代价不高

二、信道划分协议

1、TDM：有两个缺陷。首先，即使只有一个节点要发送数据，它的吞吐量也被限制在 R/N bps；即使只有一个节点要发送数据，它也必须等待自己的时隙到来才能发送数据。

2、FDM：同上，即使只有一个节点要发送数据，它的吞吐量也被限制在 R/N bps。

3、CDMA：（第六章）

三、随机接入协议

1、ALOHA 协议

(1)分类

	内容	效率
纯 ALOHA	➢ 帧一到达便立即传输进广播信道 ➢ 若发生了碰撞，则该节点立即以概率 p 重传该帧，1-p 延迟一段时间发送 ➢ 延迟完之后，以概率 p 重传，1-p 延迟另一个时间	通信网的结论
时隙 ALOHA	➢ 划分时隙，帧在时隙中传 ➢ 其他同纯 ALOHA	

(2)评价：节点从不关心信道上是否有其它节点正在传输，加大碰撞概率

2、载波侦听多路访问 CSMA

(1)分类

- CSMA/CD：碰撞检测，一般用于有线链路
- CSMA/CA：碰撞避免，一般用于无线链路

(2)CSMA/CD 机制（详见 5.5 节）

- 没有时隙的概念：适配器可以在任何时刻开始传输
- 载波侦听：适配器侦听到有其他适配器在传输时，就不传
- 碰撞检测：传输中的适配器检测到另一个适配器正式在传输时，就终止他的传输
- 重传前，适配器随即等待一个时间（该时间通常比传一帧的时间短）

(3)仍会发生碰撞的原因：端到端的传播时延。如果时延很小，CSMA/CD 效率理论可达 100%

(4)传播时空图（见书的解释）

3、评价：ALOHA 和 CSMA 都只具有之前提到的第一种特性而不具有第二种特性，下面提到的那种具有第二种

四、轮流协议

1、轮询协议

(1)内容：指定某一节点为主节点，主节点以循环的方式轮询每个节点，告诉他们能够传输的最大帧数。

(2)评价

- 消除了随机接入的碰撞和空时延问题
- 但是有两个缺点：首先是有轮询时延导致每个节点将以小于 R/N bps 的速率传输；其次如果主节点损坏，就崩溃了

2、令牌传递协议

(1)内容：没有主节点，有一个称为令牌（token）的特殊帧在节点之间按某种顺序交换。当一个节点收到令牌时，仅当他有帧要发送才持有令牌；否则将令牌转发给下一个节点。

五、局域网

1、以太网 LAN：基于随机接入的；IEEE802.3

2、令牌环 LAN：IEEE802.5 和 FDDI

5.4 链路层编址

一、链路层地址-----MAC 地址（也叫 LAN 地址、物理地址）

1、基本概念

(1)长度：6 字节，共 2^48 个可能的地址。

(2)表示方法：十六进制表示法，如 1A-23-F9-CD-06-9B

(3)归属：不是节点（主机或路由器）具有 MAC 地址，而是节点的适配器具有。

(4)MAC 广播地址：FF-FF-FF-FF-FF-FF

2、特点及与 IP 的比较

MAC 地址	IP 地址
扁平结构：即不能划分子网	层次结构
不会因为位置的改变而改变（身份证号）	位置移动后就改变（家庭住址）

3、为什么除了 IP 地址外还需要 MAC 地址

-------已有的三类地址：应用层主机名、网络层 IP 地址、链路层 MAC 地址-------

- LAN 是为任意网络层协议而设计的，不止服务于 IP 和因特网。如果适配器被指派 IP 地址而非 MAC 地址，则不能支持其他网络层协议。
- 适配器用 MAC 地址不许频繁更改，并且重启后不需重新配置。

二、地址解析协议 ARP

1、用途：网络层地址和链路层地址的转换。发送节点为了知道下一跳的 MAC 地址，根据目的节点的 IP 地址查询 ARP 表来确定。

2、相关概念

- ARP 表：每个节点（主机或路由器）的 ARP 模块都在它的 RAM 中维护着一个 ARP 表（如下）。其中 TTL 是表项的过期时间，通常是 20 分钟。

IP地址	MAC地址	TTL
222.222.222.221	88-B2-2F-54-1A-0F	13:45:00
222.222.222.223	5C-66-AB-90-75-B1	13:52:00

图5-18 在节点222.222.222.220中的一张可能的ARP表

- ARP 分组：包括发送节点和接收节点的 IP 地址和 MAC 地址等字段。

3、子网内传送分组步骤

利用目的节点的 IP 地址查找自身的 ARP 表。如果有对应的表项，则填上 MAC 地址后发送；如果没有，更新完 ARP 表后再向目标发送。

ARP 表更新的过程：节点广播一个 ARP 查询分组，子网中的节点接收并向上传给自己的 ARP 模块，如果该节点的 IP 地址和 ARP 分组中的相匹配，则加上自身的 MAC 地址后向原节点发送 ARP 响应分组，利用该分组中的 MAC 字段来更新自身的 ARP 表。

4、数据报发送到子网外节点

- 源和目的 IP 地址一直是最终节点的地址
- 而目的 MAC 地址则是子网内下一跳节点的 MAC 地址，源 MAC 同样，都是每一跳更新一次。

5、ARP 的注意点：

- 查询 ARP 报文实在广播帧中发送的，而响应 ARP 报文实在标准帧中发送的
- ARP 是即插即用的。ARP 表会自动更新而无需人工配置
- 网络中有两种类型的节点：主机和路由器。每台主机只有一个 IP 地址和一个适配器（所以只有一个 MAC 地址）；而路由器则是每个接口都有一个 IP 地址和一个适配器和一个 ARP 模块。

5.5 以太网

一、概述

1、LAN 技术：以太网、令牌环网、FDDI、ATM 等。而以太网最终几乎完全占领了现有的有线局域网市场。

2、早期以太网：基于集线器的星型拓扑。

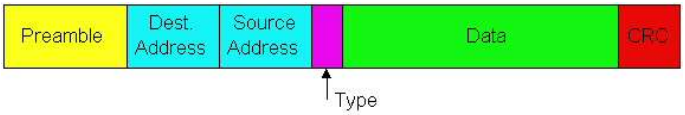
- 集线器是物理层设备，作用于各个比特而不是帧。当比特到达一个接口时，集线器只是重新生成这个比特，将其能量强度放大，并将其从**所有**接口传输出去。
- 如果某集线器同时从两个不同的接口接收到帧，则会碰撞，各节点必须重传。
- 由于集线器的上述特点，这种结构是一个广播 LAN。

3、改进版以太网：用交换机替代中心的集线器，但仍使用星型结构

- 交换机运行在倒数两层上（路由器是倒数三层）

二、以太网的 xxx

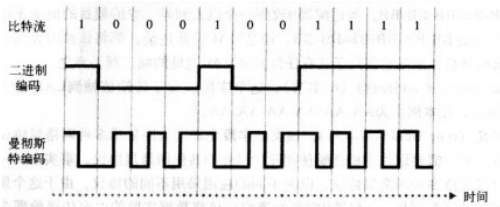
1、以太网的帧结构及各字段说明



- 前同步码（8 字节）：前 7 个字节都是 10101010，最后一个字节是 10101011。前 7 个字节用于唤醒接收适配器，并将他们的时钟和发送方的同步；第 8 个字节是告诉适配器重要内容来了。
- 源地址、目的地址（各 6 字节）：指的是 MAC 地址。当收到目的地址不是自己的帧时，会直接丢弃。
- 类型字段（2 字节）：要穿输给网络层的哪个协议。其他链路层协议和 ARP 协议也有自己的编号。
- 数据字段（46-1500 字节）：如果超过 1500 字节需要给数据报分段；如果小于 46 字节需要填充到 46 字节，这时，收端需要利用首部长度字段来去除填充。
- 循环冗余校验

2、以太网的编码（物理层）

- 使用基带传输；采用曼彻斯特编码：1 表示下降沿，0 表示上升沿。
- 采用这种编码的原因：发送适配器和接收适配器的时钟没有精确同步，所以在每个比特的**中间**包含一次跳变（？）。



3、以太网提供的服务----不可靠的无连接服务

(1)无连接服务：当要发送帧时，没有与接收适配器实现握手就直接发送到 LAN 上

(2)不可靠服务：接收适配器虽然做了 CRC 校验，但是 CRC 通过了不会向发送适配器发送确认帧，没通过直接丢包也不发送否定帧。这样发送方无法知道它传输的帧是否通过了 CRC 校验。

这样数据报流上可能会出现 gap，而至于应用层会不会察觉到 gap 要取决于传输层输

用 TCP 还是 UDP（是否重传）。

二、CSMA/CD：以太网的多路访问协议

1、场景：使用集线器的以太网（为广播 LAN）

2、CSMA/CD 的机制

(1)机制

- 适配器从网络层的到数据报，封装成帧并放到适配器的缓存中。
- 如果适配器侦听到信道空闲(96 **比特时间**内,没有信号能量从信道进入到适配器)，则开始传输；如果侦听到忙，则等待 96 比特时间再开始侦听和传输。
- 传输过程中监视信道，如果整个过程没有检测到来自其他适配器的信号，则完成；否则，马上停止传输帧并传输一个 48 bits 的阻塞信号，然后进入指数退避阶段。

(2)说明

- 阻塞信号：目的是为了确保所有其他适配器都知道发生了碰撞。比如，A 先于 B 传输帧，并且恰好当 A 的帧到达 B 之前，B 开始传输。此时，B 刚传了几个比特就检测到碰撞并终止传输，这种情况下 B 的这几个比特到达 A 的能量可能不足够让 A 发现，所以需要发阻塞信号。
- 比特时间：传输 1 bit 需要的时间，即传输速率的倒数。
- 指数退避：传输给定帧时，若该帧遇到了第 n 次碰撞，则适配器从$\{0,1,2,\cdots,2^m-1\}$中为 K 选择一个值，其中$m=\min(n,10)$，然后适配器等待 K*512 个比特时间。

3、以太网效率

(1)定义：当有大量的活跃节点，且每个节点有大量的帧要发送时，帧在信道中无碰撞地传输地那部分时间站长期运行时的份额。

(2)近似表达式

$$\eta=\frac{1}{1+5d_{prop}/d_{trans}}$$

从上式可以看出，传播时延越小效率越高；传输时延越大效率也越高。

三、以太网技术

1、协议标准的解释

(1)举例：10BASE-T、100BASE-2、100BASE-T、1000BASE-LX、10GBASE-T

(2)解释：第一个数字表示速率，默认为 Mbps，有 G 的表示 Gbps；BASE 表示基带以太网；最后一部分表示物理媒体，如 T 表示双绞铜线。

2、以太网的延伸----转发器

- 是一种物理层设备。在输入端接收信号并在输出端再生信号。

四、以太网的发展

在早期的总线拓扑和基于集线器的星型拓扑时代，以太网是一种广播链路。

如今使用的是基于交换机的星型拓扑，采用存储转发分组交换。交换机协调传输过程，不会发生碰撞，实际上没有必要使用 MAC 协议。

5.6 链路层交换机

一、概述

1、交换机对节点的透明性：某节点向另一节点寻址一个帧（而不是向交换机寻址该帧），顺利地将该帧发送进 LAN 而不知道交换机将会接受该帧并将其转发给另一节点。

2、网络结构举例

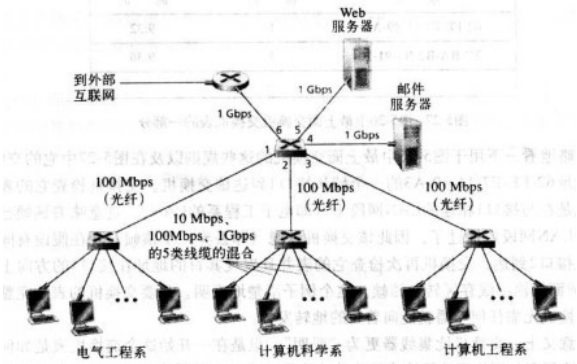


图5-26 使用集线器、以太网交换机和路由器的某机构网络

3、转发和过滤功能

- 过滤（filtering）：交换机决定一个帧应该转发到某个接口还是应当丢弃的功能
- 转发（forwarding）：决定一个帧应该被导向哪个接口并移动到那个接口。

上述两项功能的实现都借助于交换机表完成。

4、交换机的特点

- 即插即用（见自学习）
- 双工：这样在任何与交换机连接的链路上，节点和交换机能够同时传输而无碰撞

二、交换机表（switch table）

1、表项：节点的 MAC 地址、该节点对应的交换机接口、节点表项加入表的时间

2、过滤和转发功能的实现

目的 MAC 地址为 DD-DD-DD-DD-DD-DD-DD 的帧从交换接口 x 到达时的几种情况

- 表中没有 DD-DD-DD-DD-DD-DD-DD 的表项：交换机向除了接口 x 外的所有其他接口广播该帧。
- 表中有 DD-DD-DD-DD-DD-DD-DD 的表项，且与 x 接口联系起来：直接丢弃该帧，执行过滤功能。
- 表中有 DD-DD-DD-DD-DD-DD-DD 的表项，且与不为 x 的另一个接口联系起来：执行转发功能，将该帧放到 y 接口的输出缓存区中。

3、自学习

- 交换机表初始为空
- 对于在某接口接收到的每一入帧，都会在转发表中更新对应的三个表项
- 如果一段时间后（称为老化期），交换机没有收到某一 MAC 地址来的帧，就会在表中删除那个地址对应的表项。

三、评价与比较

1、使用链路层交换机相对于总线或基于集线器的星型广播链路的优点

- 消除碰撞：碰撞会浪费带宽，交换机会缓存帧并且绝不会在网段上同时传输多于一个的帧
- 异质的链路：交换机的不同接口可以以不同速率运行并且能够在不同的物理媒质上运行
- 管理：提供强化的安全性；易于进行网络管理

2、组网时选择路由器还是交换机

(1)交换机的优缺点

- 优点
  - 即插即用
  - 具有较高的分组过滤和转发速率
- 缺点
  - 为了防止广播帧循环，交换网络的活跃拓扑结构限制为一棵生成树
  - 大型交换网络要求节点有大的 ARP 表，这将产生很大的 ARP 流量和处理量
  - 交换机对于广播风暴不提供任何保护措施。即如果某主机无休止地传输广播帧流，交换机将转发所有这些帧，导致网络崩溃。

(2)路由器的优缺点

- 优点
  - 网络结构是分层次的，即使网络中存在冗余路径，分组也通常不会通过路由器循环。所以可以用各种结构来组网。
  - 对第二层的广播风暴提供了防火墙保护
- 缺点
  - 不是即插即用，需要人工配置
  - 对每个分组处理的时间更长(因为交换机只处理下两层,路由器要处理下三层)

(3)总结

- 几百台主机组成的小网络：交换机
- 更多主机组成的大网络：交换机+路由器

表5-1 流行的互联设备的典型特色的比较

	集线器	路由器	交换机
流量隔离	无	有	有
即插即用	有	无	有
优化选路	无	有	无
直通交换	有	无	有

5.7 PPP：点对点协议

一、概述

1、应用场景：家庭用户与 ISP 通过拨号建立连接所选用的协议

2、PPP 的设计目标

- 分组成帧：发送方能将网络层分组封装成帧
- 透明性：本协议不能对出现在网络层中的数据做任何限制
- 多种网络层协议：能够支持同时运行在相同物理链路上的多种网络层协议
- ……..见书，如果有题再整理

二、PPP 数据成帧

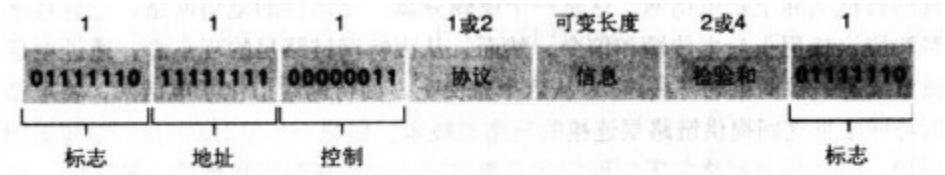


图5-30 PPP数据帧格式

1、各字段的说明

- 标志字段：都是用 01111110 作为开始和结束的
- 地址字段、控制字段：唯一可能值分别是是 11111111 和 00000011，两个字段都没有用，是保留以后定义备用的
- 协议：告诉 PPP 接收方该帧中的数据要传给哪一个上层协议
- 检验和：使用 CRC

2、字节填充

(1)场景：信息字段包含 01111110 时，应该怎么办。

(2)解决办法：提供一个转义字符 01111101，提示下一字节是数据而非原来的意思

- 01111101 01111110：表示数据 01111110（而非标志字段）
- 01111101 01111101：表示数据 01111101（而非转义字符）

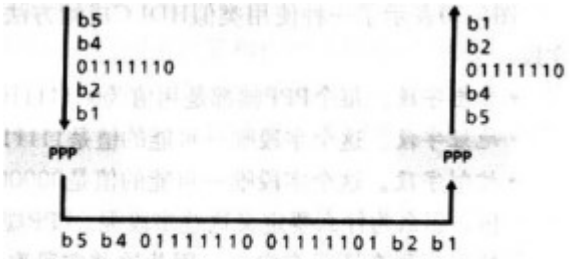


图5-31 字节填充

另一种思路是禁止上层协议发送包含 01111110 的数据，但是这违背了透明性的准则。

第六章、无线网络和移动网络

6.1 概述

一、基本概念

1、无线网络的基本元素

- 无线主机：主机本身可能移动也可能不移动
- 无线链路
- 基站：在有线网络中没有明确的对应设备。负责协调与之相关联的多个无线主机的传输以及作为中继与更大的网络互联。典型的例子是蜂窝网络中的蜂窝塔和无线 LAN 中的接入点。
- 网络基础设施：无线主机希望与之进行通信的更大网络

2、网络的模式

(1)基础设施模式（infrastructure）：所有主机都通过接入点来获取网络服务。存在因为主机移动进另一个基站的覆盖范围而产生的切换问题。

(2)自组织网络（Ad Hoc）：没有接入点，主机之间直接互联

3、无线网络的分类

分类	特点	实例
单跳，基于基础设施	➢ 具有与较大的有线网络(如因特网）连接的基站 ➢ 该基站与所有无线主机间的通信都只经过一个无线跳	802.11、802.16（WiMAX）
单跳，无基础设施	➢ 存在与无线网相连的基站 ➢ 其中的节点也可以协调其他节点的传输	蓝牙网络、具有自组织模式的 802.11
多跳，基于基础设施	➢ 以有线的方式与较大网络相连 ➢ 个别节点需要经多跳才能与基站通信	传感网络、无线网状网络
多跳，无基础设施	➢ 没有基站 ➢ 节点之间可能多跳通信	移动自组织网络（MANET）、车载自组织网络（VANET）

6.2 无线链路和网络特征

一、基本概念

1、无线链路较有限链路的区别

(1)较高且时变的误比特率

主要因素如下

- 路径损耗
- 同频（段）干扰：来自其他源或者电磁噪声
- 多径传播:反射导致的多径

可以看出无线链路比有线链路更容易出错，因此无线链路协议不仅采用了 CRC 校验，还使用 ARQ‘协议来重传错误帧。

(2)隐藏终端和衰减问题

有两种情况，碰撞双方不知道已经发生碰撞了

- 隐藏终端：二者之间有物理阻挡导致双方各自的信号不能到对方那里（左图）
- 衰减：信号传到对方那里时已经衰减到发现不了了（右图）

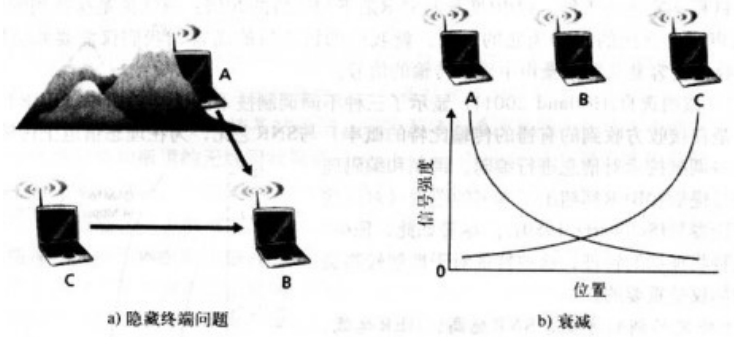


图6-4 隐藏终端问题和衰减

2、接收方比特差错率（BER）和信噪比（SNR）的关系

- 对于给定的调制方案，SNR 越高，BER 越低
- 对于给定的 SNR，采用传输速率越高的调制技术，BER 越高
- 物理层自适应调制和编码以应对移动或环境改变。

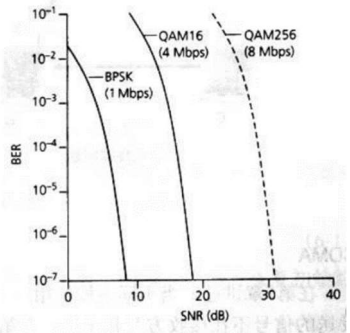


图6-3 比特差错率、传输率和SNR

二、码分多址 CDMA（略）

6.3 WiFi：802.11 无线 LAN

一、基本概念

1、WiFi：Wireless Fidelity 无线保真

2、WiFi 丛林：该处有多个 WiFi 信号覆盖

3、体系结构举例

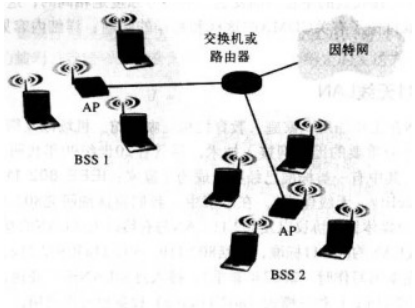


图6-7 IEEE 802.11 LAN体系结构

二、802.11 体系结构

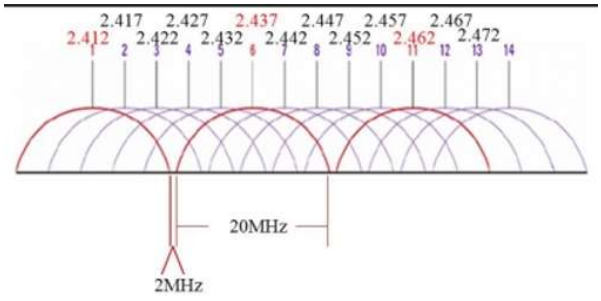
1、基本服务集 BSS，包括：

- 几个无线站点：理解为用户终端，每个站点都有一个 6 字节的 MAC 地址保存在 802.11 网卡中。
- 一个接入点 AP（基站）：是链路层设备

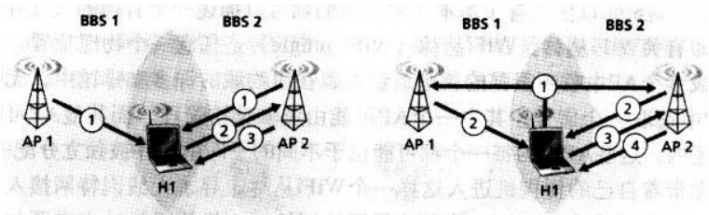
三、网络的初始和新站点的加入

1、AP 的安装

- 配置服务集标识 SSID
- 配置信道号： 802.11 运行在 2.4G-2.485GHZ 频段上，这 85MHZ 上定义了 11 个部分重叠的信道，当且仅当互不重叠时才可用，所以一共就三种组合：1，6，11 或者 2，7，12，或者 3，8，13。从而可知同一个物理网络中最多只能安装 3 个 AP



2、无线站点与 AP 关联



a) 被动扫描
① 自AP发送信标帧
② H1向选择的AP发送关联请求帧
③ 选择的AP向H1发送关联响应帧

b) 主动扫描
① 自H1广播探测请求帧
② 自AP发送探测响应
③ H1项选择的AP发送关联请求帧
④ 选择的AP向H1发送关联的响应帧

图6-9 对接入点的主动扫描和被动扫描

- 信标帧包括 AP 的 SSID 和 MAC 地址
- 802.11 没有规定关联算法，这一部分留给固件和无线主机的软件设计者
- 应注意的是，信号强度不是唯一评判指标。例如，有可能某一 AP 信号强度很强但是已经被其他主机过载了，这时候就不应该选它关联。

3、AP 鉴别无线站点

- 规定一个站点的 MAC 地址只允许接入一个无线网络
- 引入用户名和口令

三、802.11 MAC 协议（关联完成之后开始通信了）

1、基本概念---802.11 链路层确认机制

(1)引入原因：无线 LAN 中站点发送一个帧时，该帧可能会因为各种原因不能无损地到达目的站点，为了处理这种故障而引入的。

(2)具体内容：目的站点收到一个通过 CRC 校验的帧后，等待一个短帧间间隔 SIFS 的时间后，再发回一个确认帧。

2、多路访问协议：CSMA/CA（碰撞避免）。

(1)不再使用碰撞检测原因

- 检测碰撞要求站点具有同时发送和接收的能力。在无线链路上，接收信号的强度远小于发射信号的强度，且不像以太网一样经由两条线路传输上下行而是混在一起。
- 存在隐藏终端和衰减的问题无法检测到所有的碰撞。

(2)实现

- 站点监听到信道空闲后，不会马上发送，经过 DIFS（分布式时间间隔）时间后再发送
- 否则，随机选择一个值退避，当监听到信道空闲时该值减 1，忙时不变，减到 0 后重新进入上面的状态
- 发送后，等待确认帧。如果收到了且它要继续发送，则继续发送；如果没收到，将重新进入退避阶段（并在更大的范围内选值），之后重传，多次没收到就放弃发送然后丢弃该帧。

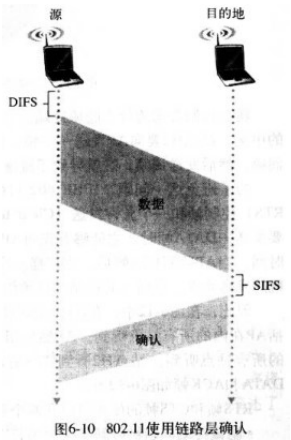


图6-10 802.11使用链路层确认

(3)与 CSMA/CD 的不同

- CD 只要检测到信道空闲就会发送。

- CA 检测到信道空闲不会立即发送；同时如果它处在计数阶段，即使信道空闲它也不会发送（因为它不能检测碰撞和放弃发送，所以只能尽可能避免）

(4)评价：仍可能出现碰撞问题。比如隐藏终端或者是选择了非常接近的退避值。

3、基于 CSMA/CA 的改进：RTS 和 CTS（处理了隐藏终端问题）

(1)概念

- RTS 帧：请求发送控制帧（Request to Send）。发送方要发送数据帧时，首先广播一个短的 RTS 帧，该帧指出数据帧和确认帧需要的总时间。该帧可以被圈内包括 AP 的其他站点收到。
- CTS 帧：允许发送控制帧（Clear to Send）。AP 收到 RTS 帧后，广播 CTS 作为响应。该响应有两个目的，给发送方发送允许；指示其他站点在预约期内不要发送

(2)评价

优点：可以在两方面提高性能

- 解决了隐藏终端问题（长的数据帧只有预约成功后才能发送）
- RTS 和 CTS 帧较短，涉及他们的碰撞仅持续很短的时间。而一旦 RTS 和 CTS 被正确传输，后续的数据帧和 ACK 就不会发生碰撞了

缺点：引入了时延，增加了信道资源的消耗。因此实际中，设置了 RTS 门限，仅当帧长超过门限时才会使用 RTS/CTS 帧。

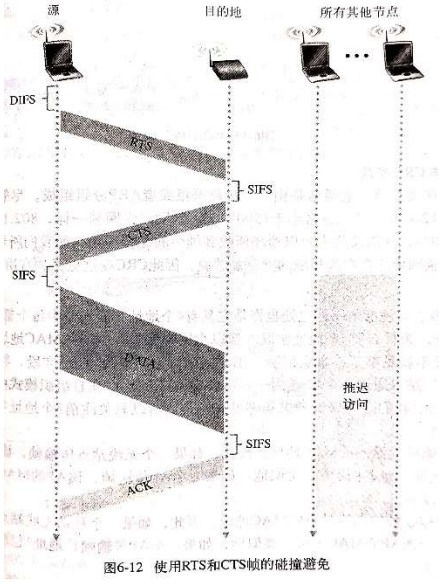


图6-12 使用RTS和CTS帧的碰撞避免

3、使用 802.11 作为一个点对点链路：两节点需要使用定向天线

四、802.11 帧

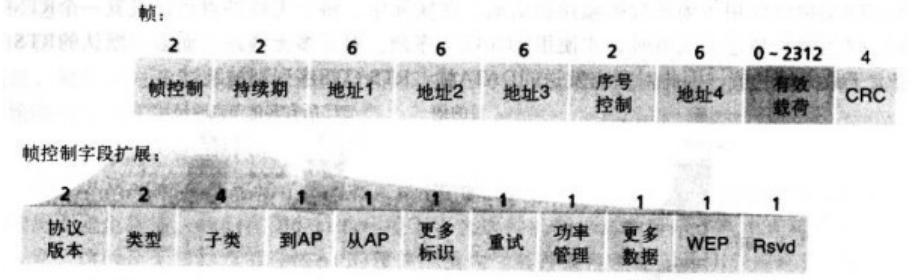


图6-13 802.11帧

其中：上面一排的数字代表字节数，下面一排代表比特数

1、有效荷载：通常是 IP 数据报或者 ARP 分组

2、地址字段：（有 4 个!）

- 地址 1：接收方无线站点的 MAC 地址
- 地址 2：发送方无线站点的 MAC 地址
- 地址 3：当前子网中路由器接口的 MAC 地址，用于与有线局域网的互联（见本节开头的图），具体过程见下页的图。
- 地址 4：？？？？？

- 路由器知道H1的IP地址（从数据报的目的地址中得到），使用ARP决定H1的MAC地址，这与在普通的以太网LAN中相同。获取H1的MAC地址后，路由器接口R1将该数据报封装在一个以太网帧中。该帧的源地址字段包含了R1的MAC地址，目的地址字段包含H1的MAC地址。
- 当该以太网帧到达AP后，该AP在将其传输到无线信道前，先将该802.3以太网帧转换为一个802.11帧。如前所述，AP将地址1和地址2分别填上H1的MAC地址和其自身的MAC地址。对于地址3，AP插入R1的MAC地址。通过这一方式，H1可以确定（从地址3）将数据报发送到子网中的路由器接口的MAC地址。

现在考虑当从无线站点H1移动一个数据报到R1时H1进行响应时发生的情况。

- H1生成一个802.11帧，分别用AP的MAC地址和H1的MAC地址填上地址1和地址2字段，如上所述。对于地址3，H1插入R1的MAC地址。
- 当AP接收该802.11帧后，它将其转换为以太网帧。该帧的源地址字段是H1的MAC地址，目的地址字段是R1的MAC地址。因此，地址3允许AP在构建以太网帧时能够确定目的MAC地址。

3、序号、持续期和帧控制字段

- 序号：rdt2.1 中的作用完全一致。由于 ACK 可能丢失，发送方会发送同一帧的多个 copy，序号便用于区分是新传输来的还是重复发送的。
- 持续期：预约信道的时间
- 帧控制字段：略

五、在相同的 IP 子网中的移动性

1、移动性问题：常常在同一个 IP 子网中部署多个 BSS 以增加无线 LAN 的物理范围

- 在同一个子网内切换 BSS 时，保持 TCP 连接是一个问题
- 切换到不同的子网时，需要复杂的移动性管理协议和移动 IP 协议来控制保持 TCP 连接。

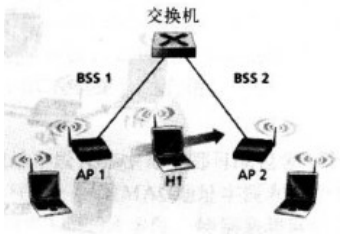


图6-15 同一子网中的移动性

- 这里用交换机是属于子网内切换的情况。如果用路由器则属于子网间切换，会直接断开 TCP 连接。

2、子网内切换的流程

- 移动过程中，AP2 的信号逐渐增强，AP1 的信号逐渐减弱。H1 扫描信标帧，根据关联算法最终解除与 AP1 的关联，然后于 AP2 建立关联。（过程中维持 TCP 连接）
- 交换机的自学习机制可以直接处理移动的问题。也可以通过其他方式来处理

六、802.11 中的高级特色

1、802.11 速率适应：自适应地根据当前和近期信道特点选择物理层调制方法。

2、功率管理：节点可以通过设置首部地功率管理比特为来告诉接入点自己是否要睡眠，从而在没有帧要发送和接收时进入睡眠状态来降低能耗。

七、802.11 以外的标准：蓝牙和 WiMAX

1、WiFi：大功率、中等范围（100m）、高速率接入

2、蓝牙

(1)特点

- 802.15 协议
- 小范围、低功率、低速率通信
- 网络结构是自组织网络，组成 picnet。常用于无线个人区域网路 WPAN。

(2)技术

- 以 TDM 的方式工作与无需许可证的 2.4GHz 无线电波段，每个时隙 625us
- 跳频扩展频谱 FHSS：每个时隙内，发送方利用 79 个信道中的一个进行传输，同时从时隙到时隙以一个为随机的方式变更信道。

(3)picnet 简介

- 主设备：只有一个，控制 picnet，可以在每个奇数时隙中发送。它的时钟确定了 picnet 中的时间
- 从设备：仅当主设备在前一时隙与其通信后才可发送，并且只能发给主设备
- 寄放设备（parked）：最多可以有 255 个，仅当其状态被主节点从“寄放”改为“活动”之后才可进行通信

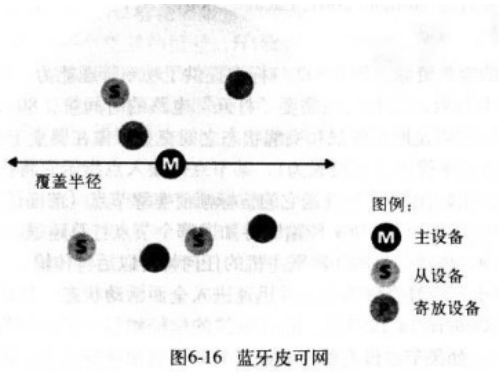


图6-16 蓝牙皮可网

3、WiMAX（全球微波接入互操作）

(1)特点

- 802.16 协议，该标准的目标是通过广阔的区域以可以与电缆调制解调器和 ADSL 网络相比的速率，向大量用户交付无线数据
- 与 Wifi 基础设施架构方式类似

(2)技术（略）