



计算机通信与网络

Computer Communications & Networks

第7章 应用层

南京邮电大学通信与信息工程学院

“计算机通信与网络” 国家精品课程组

大纲要求

Requirements



01 了解 TCP/IP 体系结构应用层的概念

02 了解网络应用模式

03 掌握网络应用的基本原理和实现机制

内容纲要

Contents Page



- 01** 应用层协议与网络应用模式
- 02** 域名系统 (**DNS**)
- 03** 远程登录 (**Telnet**)
- 04** 文件传输协议 (**FTP**)
- 05** 引导程序协议与
动态主机配置协议 (**DHCP**)
- 06** 电子邮件系统和**SMTP**
- 07** 万维网与**HTTP**

内容提要

Transition Page



01 应用层协议与网络应用模式

02 域名系统 (DNS)

03 远程登录 (Telnet)

04 文件传输协议 (FTP)

**05 引导程序协议与
动态主机配置协议 (DHCP)**

06 电子邮件系统和SMTP

07 万维网与HTTP



应用层协议



网络应用模式



基本概念

7.1.1 应用层协议

应用层是计算机网络体系结构的最高层，直接为用户的应用进程提供服务。

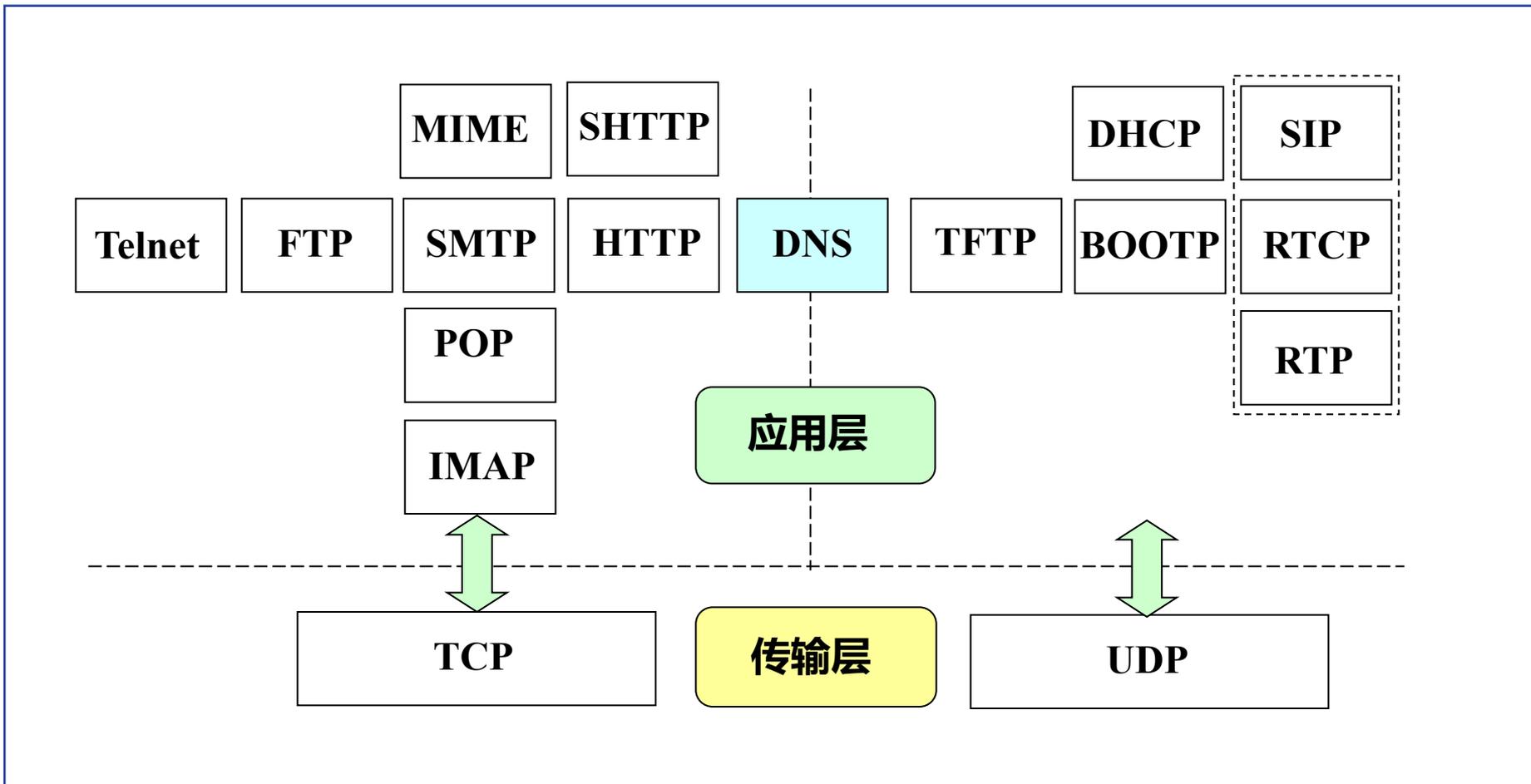
在因特网中，通过各种**应用层协议**为不同的应用进程提供服务。

应用层协议则是应用进程间在通信时所必须遵循的规定。



应用层与传输层协议对应关系

7.1.1 应用层协议





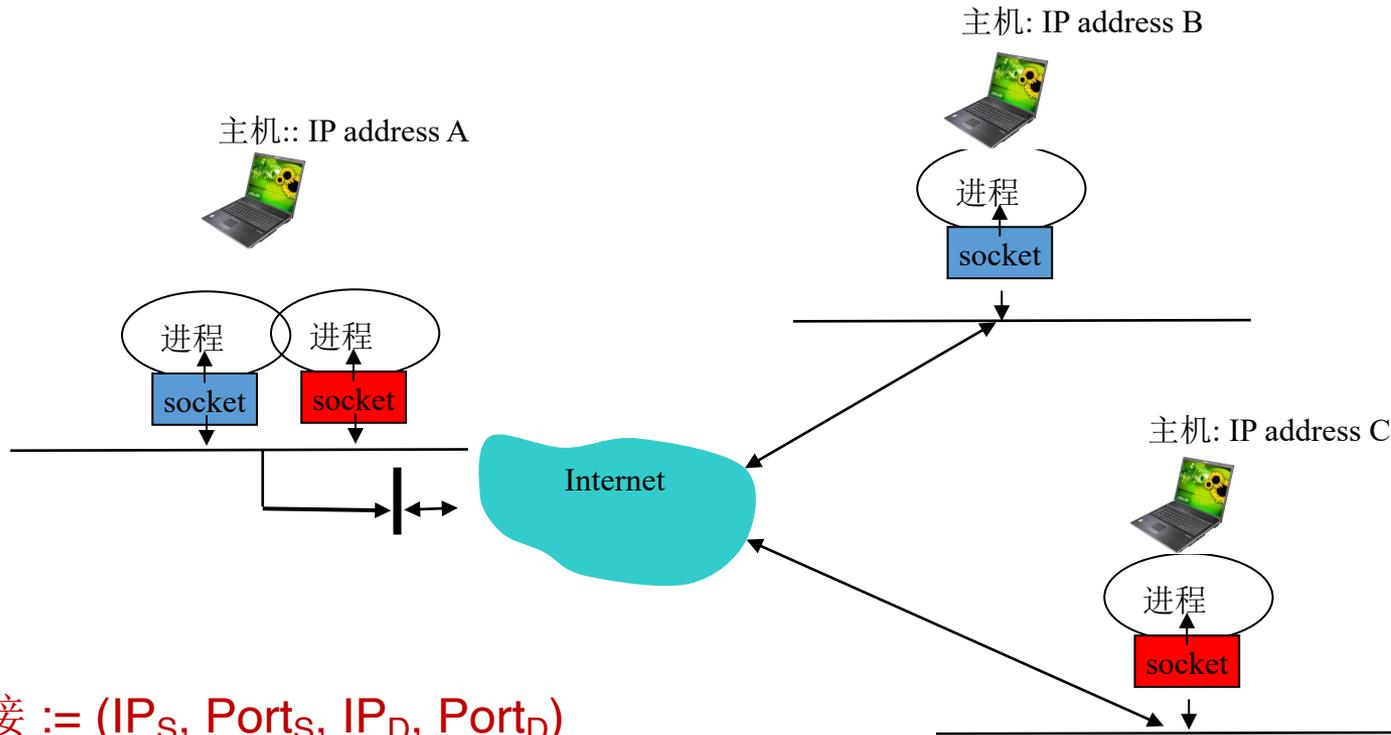
应用层协议的特点

7.1.1 应用层协议

每个**应用层协议**都是为了解决某一类应用问题，而问题的解决又往往是通过位于不同主机中的多个应用进程之间的通信和协同工作来完成的。

应用层的具体内容就是规定应用进程在通信时所遵循的协议。

应用层连接是一个四元组



连接 := $(IP_S, Port_S, IP_D, Port_D)$
(S = source, D = destination)



7.1.2 网络应用模式

1. 以大型机为中心的应用模式
2. 以服务器为中心的应用模式
3. 客户机 - 服务器模式 (Client / Server , C/S)
4. 浏览器 - 服务器模式 (Browser / Server , B/S)
5. 对等网络服务模式 (Peer-to-Peer, P2P)



7.1.2 网络应用模式

1. 以大型机为中心的应用模式

该应用模式又称为分时共享模式，也就是面向终端的多用户计算机系统（主从结构）

2. 以服务器为中心的应用模式

该应用模式又称为资源模共享模式，向单个用户站点提供灵活的服务，但管理控制和系统维护工具的功能较弱。

7.1.2 网络应用模式

3. 客户机 - 服务器模式 (Client/Server, C/S)

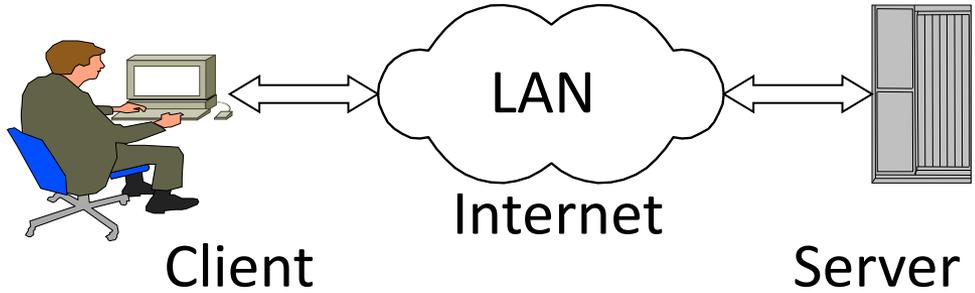
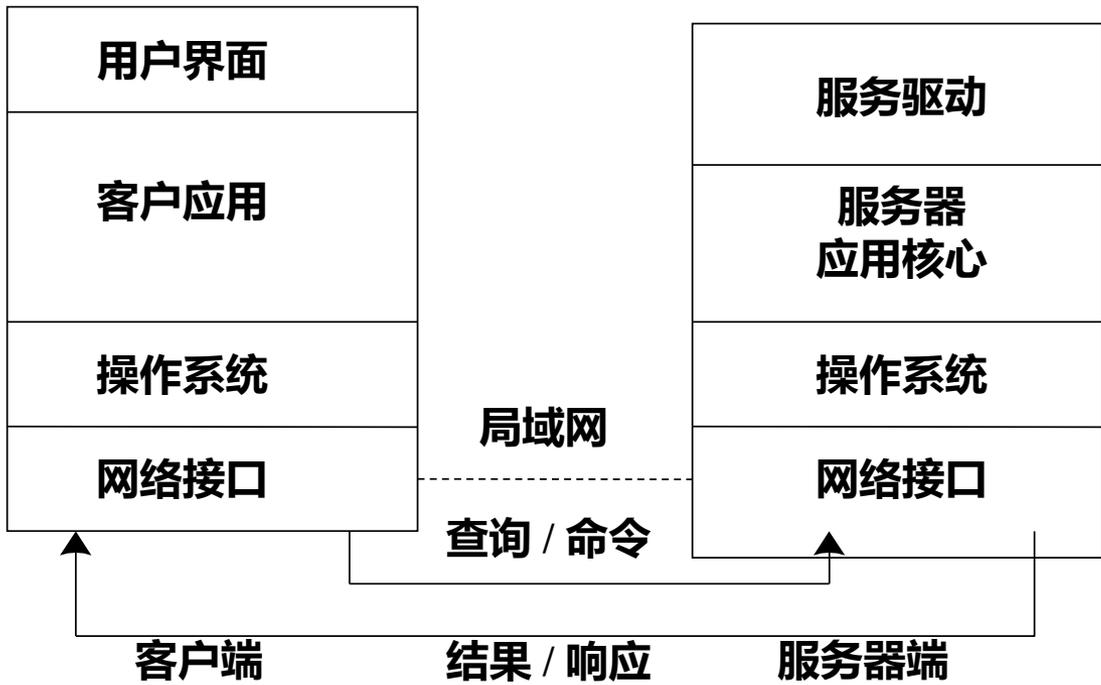
传统的网络基本服务基本上都是基于客户机 - 服务器模式

如: Telnet, WWW, E-Mail, FTP等。





7.1.2 网络应用模式





7.1.2 网络应用模式

C/S 应用模式

- 应用层的许多协议都是基于客户服务器 (C/S) 方式。
- 客户端 (Client) 和服务端 (Server) 都是指通信中所涉及的两个应用进程。
- 所描述的是进程之间服务和被服务的关系。客户端是服务请求方，服务端是服务提供方。

Socket (套接字) 系统调用

进程 A
(client)



IP addr + port



connect ()



send ()



进程 B
(server)



IP addr + port



bind (IPaddr, port)

listen ()

accept ()

recv ()

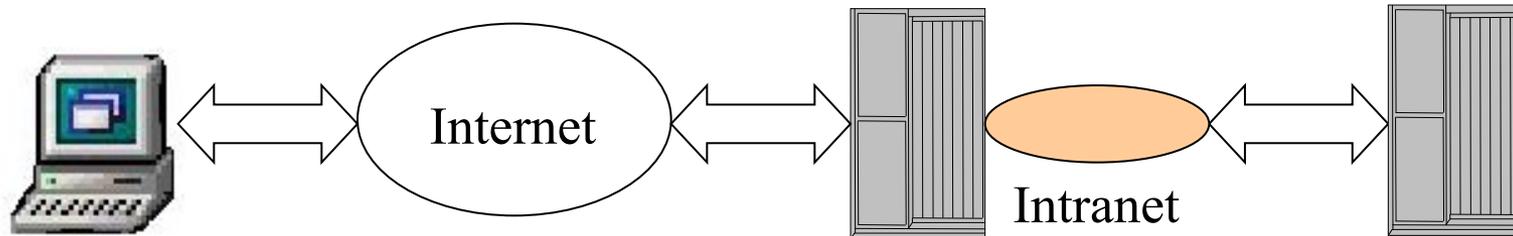


7.1.2 网络应用模式

4. 基于Web的客户机/服务器应用模式

Java虚拟客户机

网页 (Homepage)



Web Browser

HTTP

Web Server

DB Server

- MS-IE
- Netscape Navigator
- Firefox
- Mosaic
- Gopher

- **PHP**
- **ASP**
- **HTML**
- **XML**

- **BBS**
- **FTP**
- **DNS**
- **E-mail**



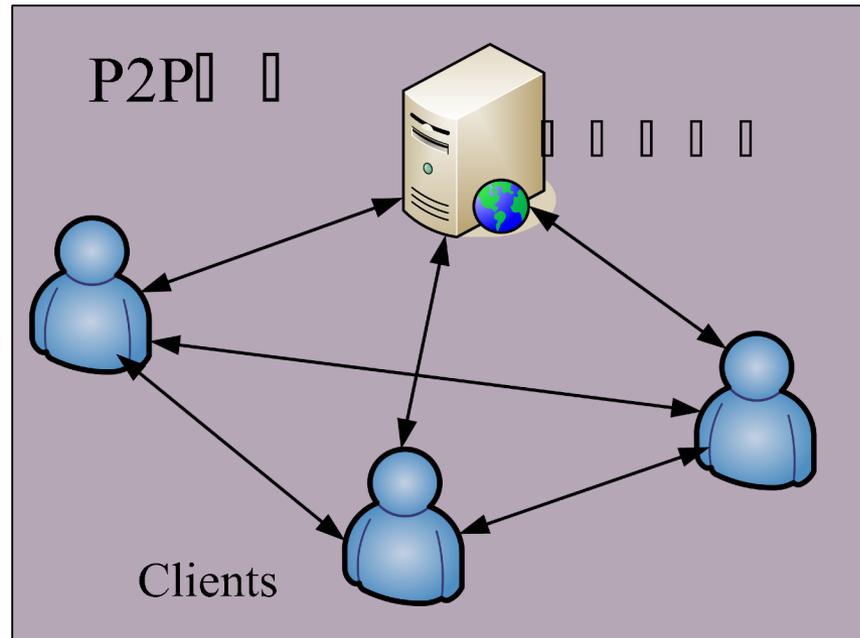
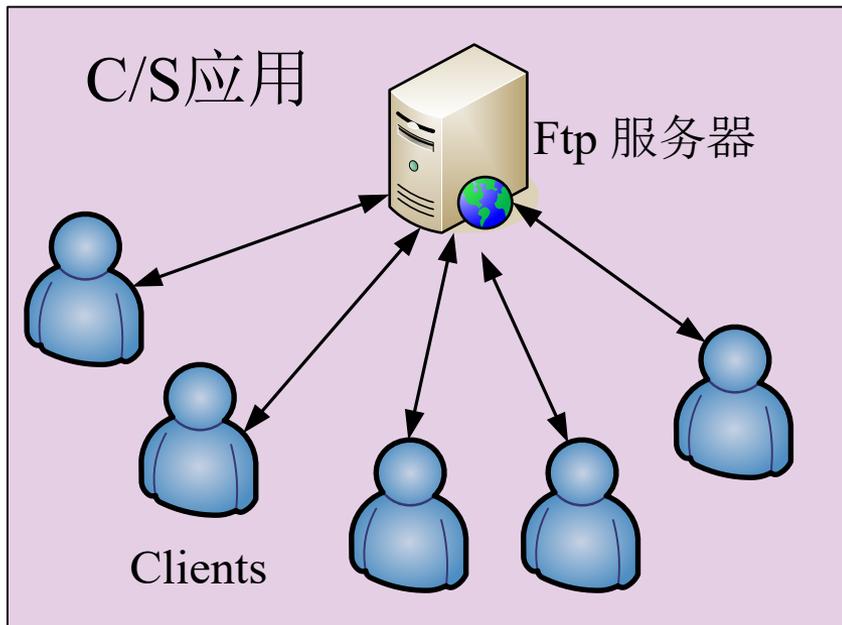
7.1.2 网络应用模式

5. 对等网络应用模式 (Peer-to-Peer, P2P)

- 在底层物理网络拓扑的基础上，在应用层构建P2P覆盖网络的虚拟拓扑结构。
- 在对等网络应用模式中，端系统主机既充当客户机，又充当服务器。
- P2P应用：文件共享，音/视频流分发
- P2P系统，如：BitTorrent，Skype，迅雷等。

7.1.2 网络应用模式

P2P与C/S系统结构的比较



内容提要

Transition Page



01 应用层协议与网络应用模式

02 域名系统 (DNS)

03 远程登录 (Telnet)

04 文件传输协议 (FTP)

05 引导程序协议与
动态主机配置协议

06 电子邮件系统和SMTP

07 万维网与HTTP



问题的提出

7.2.1 域名系统概念

- 一般而言，人们可以比较轻松的记住7位数字
- 然而，IP地址一般为12位数字
- 我们如何能够更轻松的记住IP地址呢？



解决方案

7.2.1 域名系统概念

- 使用英文名字来代表IP地址
 - E.g., www.njupt.edu.cn
- 那么我们就需要一个“字典”来匹配IP地址与英文名字
- 这个过程称之为地址解析



“字典”类型

7.2.1 域名系统概念

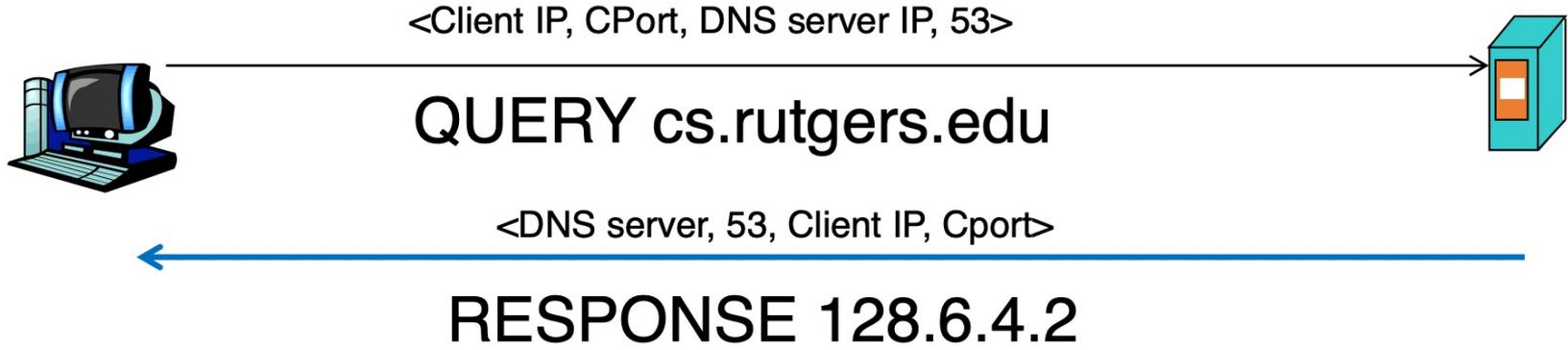
- 简单设计：
 - 中心式“字典”
 - 询问每一个人（泛洪法）
 - 然后告诉每一个人
- 可扩展的分布式设计：
 - 分层的命名空间（e.g., Domain Name System (DNS)）
 - 平坦的命名空间（e.g., 分布式哈希表）



简单设计

7.2.1 域名系统概念

| DOMAIN NAME | IP ADDRESS |
|-------------------|----------------|
| spotify.com | 98.138.253.109 |
| cs.rutgers.edu | 128.6.4.2 |
| www.google.com | 74.125.225.243 |
| www.princeton.edu | 128.112.132.86 |



核心思想： 部放一台服务器来负责查表



简单设计

7.2.1 域名系统概念

思考：

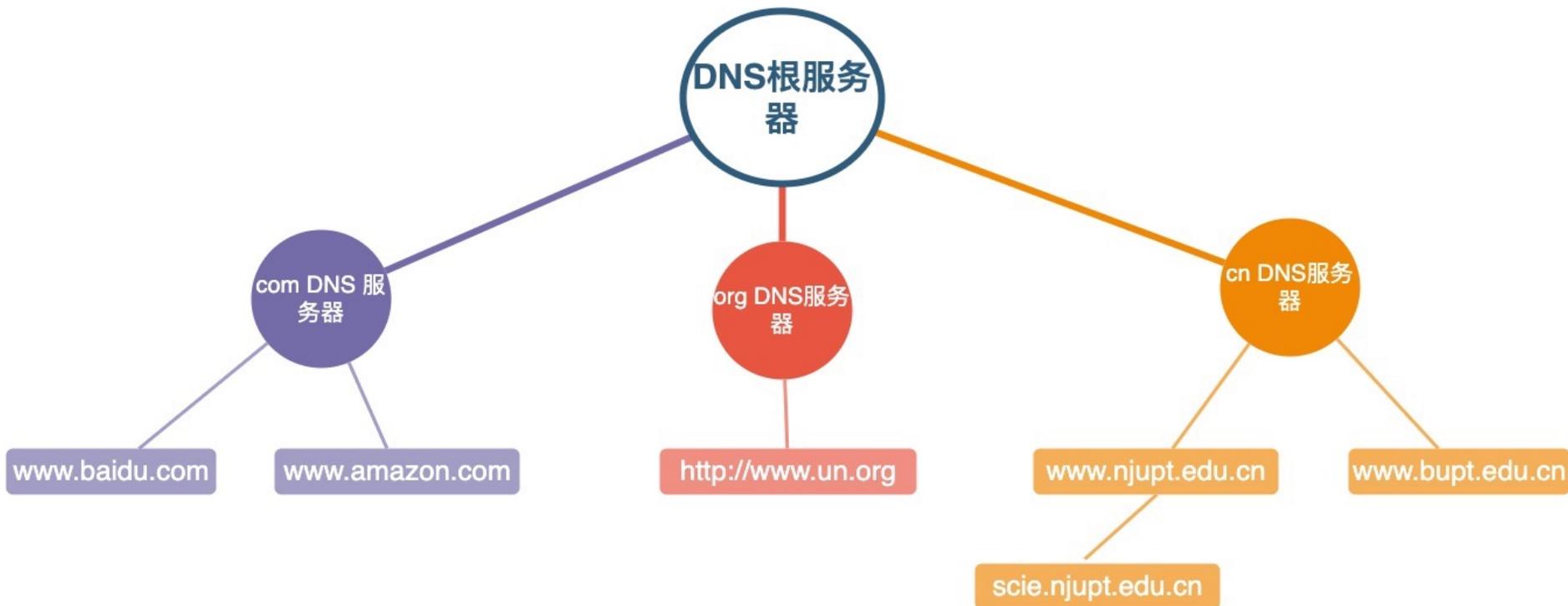
这样的设计可扩展吗？

- 每台主机都需要在表中
- 性能：这台服务器可以为数十亿人服务吗？
- 如果服务器或者数据库崩了怎么办？
- 如何保障这台服务器的安全？



分布式设计

7.2.1 域名系统概念





7.2.2 因特网的域名结构

- 许多因特网采用了**层次树状结构**的命名方法。
- 任何一个连接在因特网上的主机或路由器，都有一个**惟一**的层次结构的**名字**，即**域名**。
- 域名的结构由若干个分量组成，各分量之间用**点**隔开：

... .三级域名.二级域名.顶级域名

- 各分量分别代表不同级别的域名。



顶级域名

7.2.2 因特网的域名结构

- (1) 国家顶级域名 nTLD: 如: .cn 表示中国, .us 表示美国, .uk 表示英国, 等等。
- (2) 国际顶级域名 iTLD: 采用.int。国际性的组织可在 .int 下注册。
- (3) 通用顶级域名 gTLD: 最早的顶级域名是:
 - .com** 表示公司企业
 - .net** 表示网络服务机构
 - .org** 表示非赢利性组织
 - .edu** 表示教育机构 (美国专用)
 - .gov** 表示政府部门 (美国专用)
 - .mil** 表示军事部门 (美国专用)



新通用顶级域名

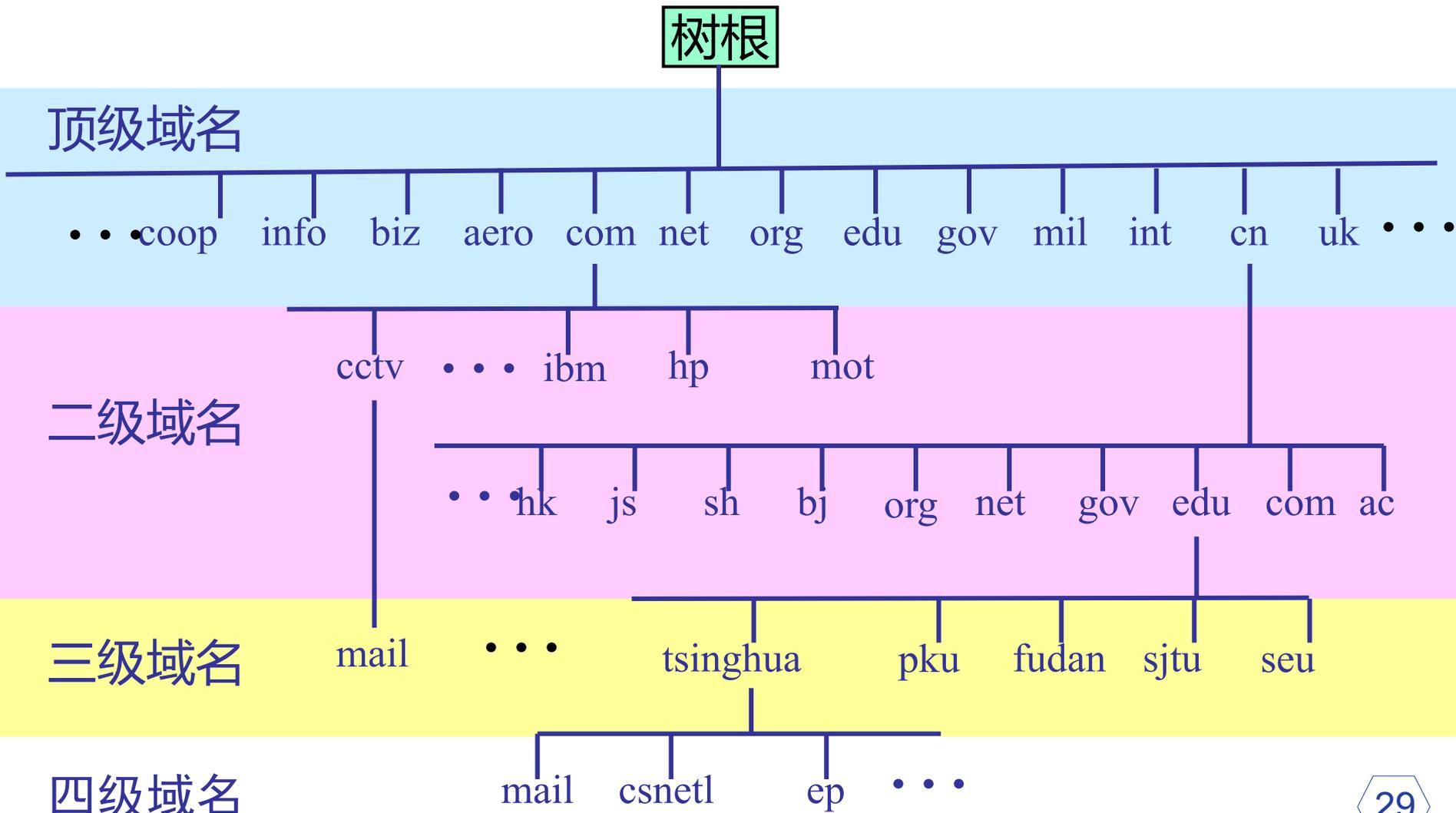
7.2.2 因特网的域名结构

- .aero 用于航空运输企业
- .biz 用于公司和企业
- .coop 用于合作团体
- .info 适用于各种情况
- .museum 用于博物馆
- .name 用于个人
- .pro 用于会计、律师和医师等自由职业者



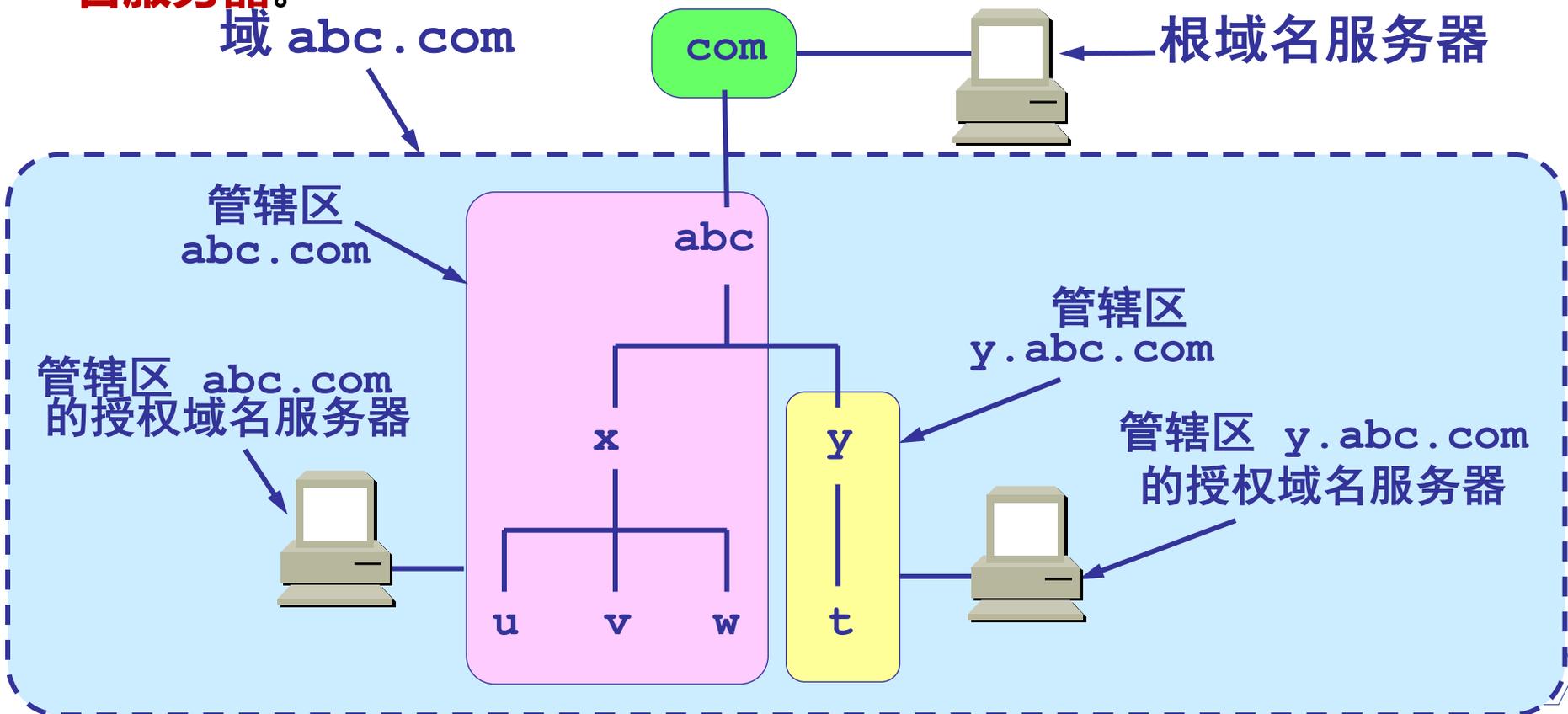
因特网的名字空间

7.2.2 因特网的域名结构



7.2.3 域名解析服务

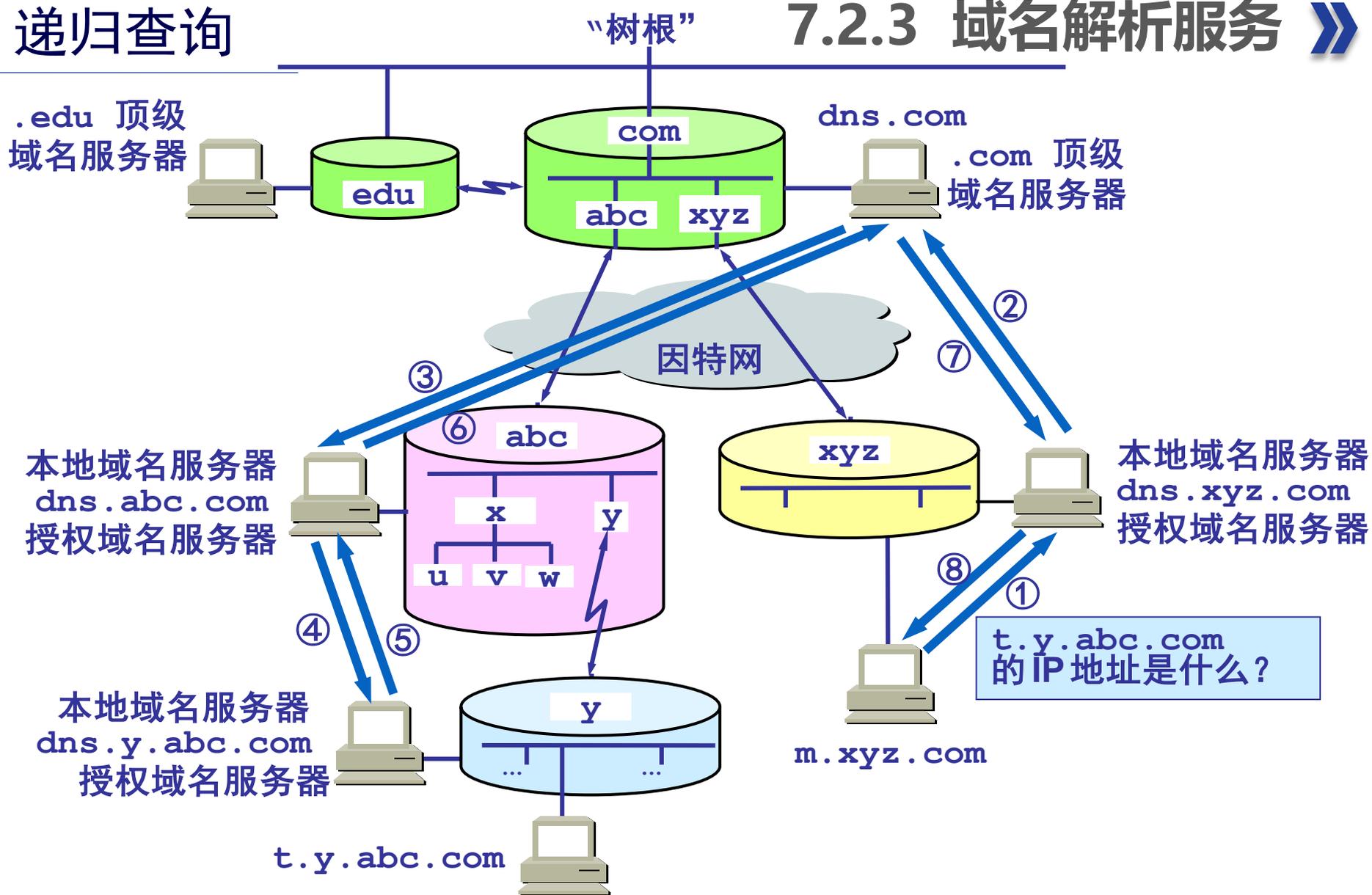
- 因特网允许各个单位根据具体情况将本单位的域名划分为若干个域名服务器**管辖区**(zone), 并在各管辖区中设置相应的**授权域名服务器**。





递归查询

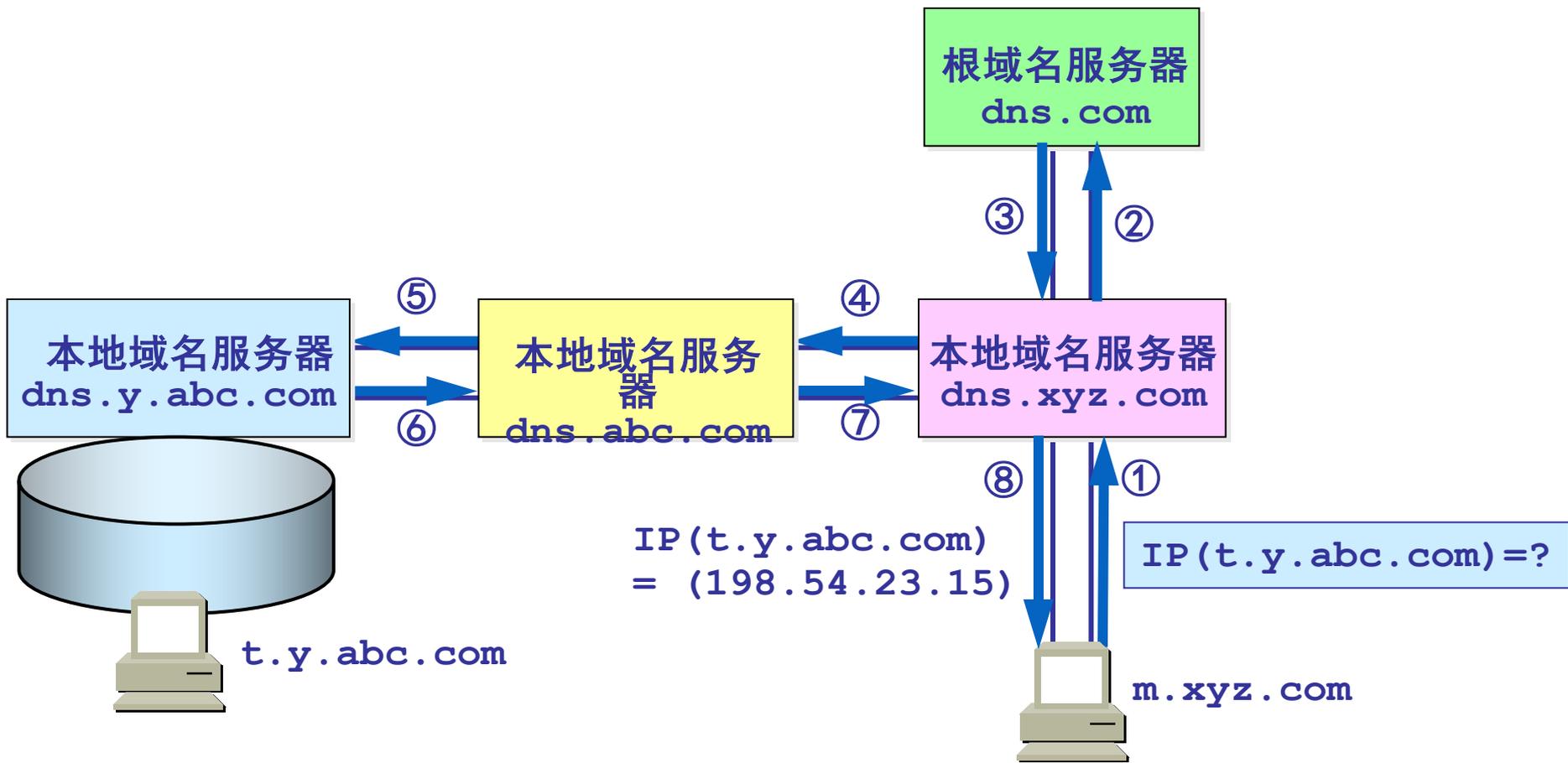
7.2.3 域名解析服务





递归与迭代相结合的查询

7.2.3 域名解析服务



内容提要

Transition Page



01 应用层协议与网络应用模式

02 域名系统 (**DNS**)

03 远程登录 (**Telnet**)

04 文件传输协议 (**FTP**)

05 引导程序协议与

动态主机配置协议 (**DHCP**)

06 电子邮件系统和**SMTP**

07 万维网与**HTTP**



7.3 远程登录

Telnet是一种**应用层**协议，使用于**互联网**及**局域网**中，使用**虚拟终端**的形式，提供双向、以文字字符串为主的**命令行接口**交互功能。属于**TCP/IP**协议族的其中之一，是**互联网**远程登录服务的标准协议和主要方式，常用于**服务器**的远程控制，可供用户在本地主机执行远程主机上的工作。

Telnet在1969年开发出来，在RFC 15定义，RFC 854定义了扩展。**互联网工程任务组** (IETF)，在**STD 8**中，将其加以标准化，是最早形成的互联网协议之一。

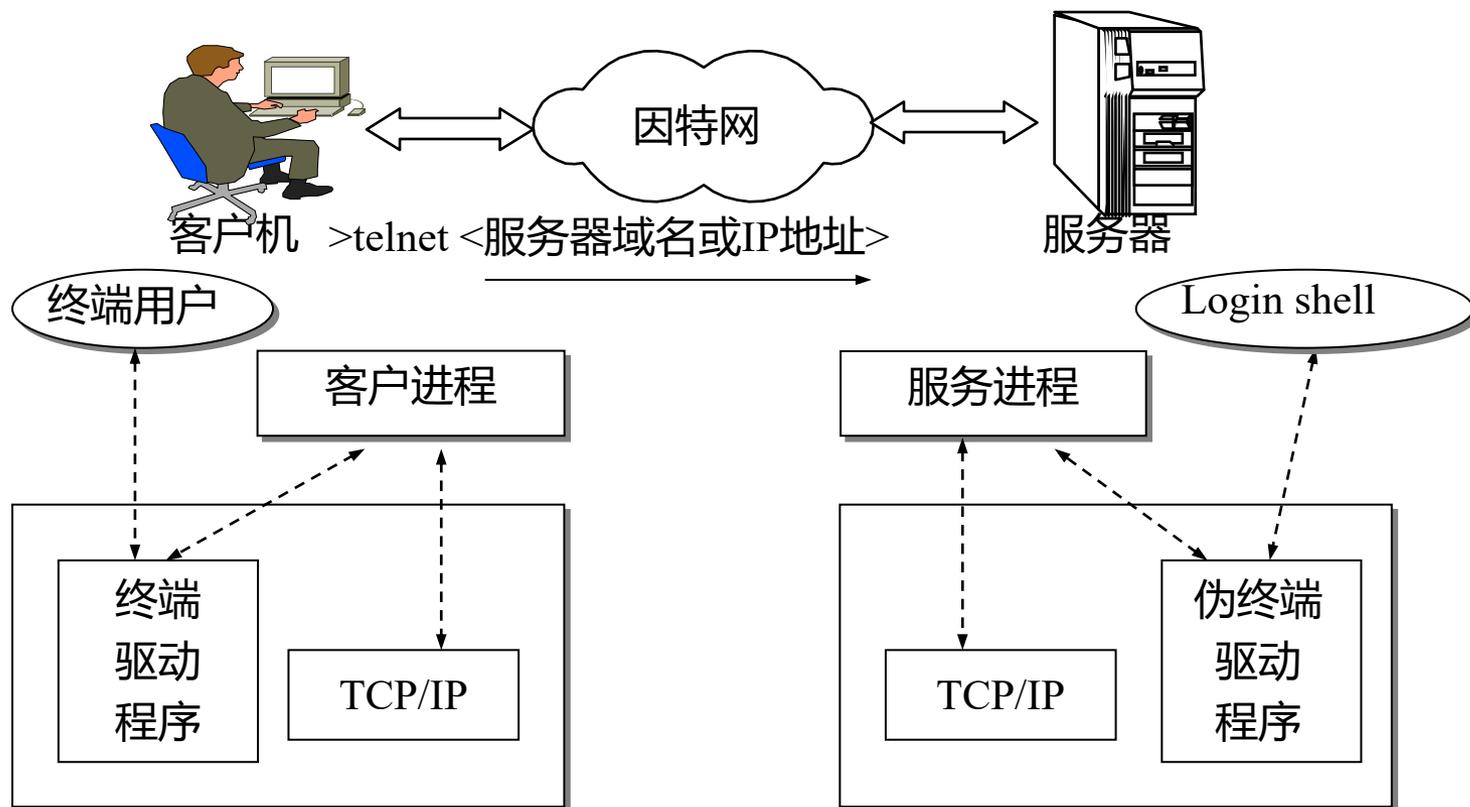


7.3 远程登录

- Telnet采用客户/服务器模式。
- 用户首先在电脑执行Telnet程序，连线至目的地**服务器**，然后输入账号和密码以验证身份。用户可以在本地主机输入命令，然后让已连接的远程主机执行，就像直接在对方的控制台上输入一样。
- 传统Telnet会话所传输的资料并未**加密**，账号和密码等敏感资料容易会被窃听，因此很多服务器都会屏蔽Telnet服务，改用更安全的**SSH**。

Telnet协议工作流程

7.3 远程登录



- 图中客户进程通过面向连接的TCP服务发到远程服务器，并显示从TCP连接上收到的数据。



网络虚拟终端

7.3 远程登录

- 服务器的操作系统内核中的伪终端驱动程序提供一个网络虚拟终端 (NVT, Network Virtual Terminal), 供操作系统和服务进程在NVT上建立注册, 以及与用户进行交互操作。
- 服务器上的应用程序可以不必考虑实际终端的类型。



NVT

7.3 远程登录

- NVT的格式定义：所有的通信使用8bit的字节。
- 在传送时，NVT采用7位的ASCII码传数据，高位置1时作控制命令。
- NVT只使用ASCII码的几个控制字符，而所有可打印的95个字母、数字和标点符号，NVT的定义与ASCII码一致的。

内容提要

Transition Page



01 应用层协议与网络应用模式

02 域名系统 (**DNS**)

03 远程登录 (**Telnet**)

04 文件传输协议 (**FTP**)

05 引导程序协议与
动态主机配置协议 (**DHCP**)

06 电子邮件系统和**SMTP**

07 万维网与**HTTP**



交互式访问

7.4 文件传输协议 (FTP)

- **文件传送协议 FTP**(File Transfer Protocol) 是 Internet 上使用得最为广泛的文件传送协议。
- 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。
- FTP屏蔽了各个计算机系统的差异，适合于在异构网络中计算机之间传送文件。



设计难点

7.4.1 文件传输协议

- 网络环境中的一项基本应用就是将文件从一台计算机中复制到另一台可能相距很远的计算机中。
- 初看起来，在两个主机之间传送文件是很简单的事情。
- 其实这往往非常困难。原因是众多的计算机厂商研制出的文件系统多达数百种，且差别很大。



网络环境下复制文件的复杂性

7.4.1 文件传输协议

- (1) 计算机**存储数据的格式**不同。
- (2) 文件的**目录结构和文件命名**的规定不同。
- (3) 对于相同的文件存取功能，操作系统使用的**命令**不同。
- (4) **访问控制方法**不同。



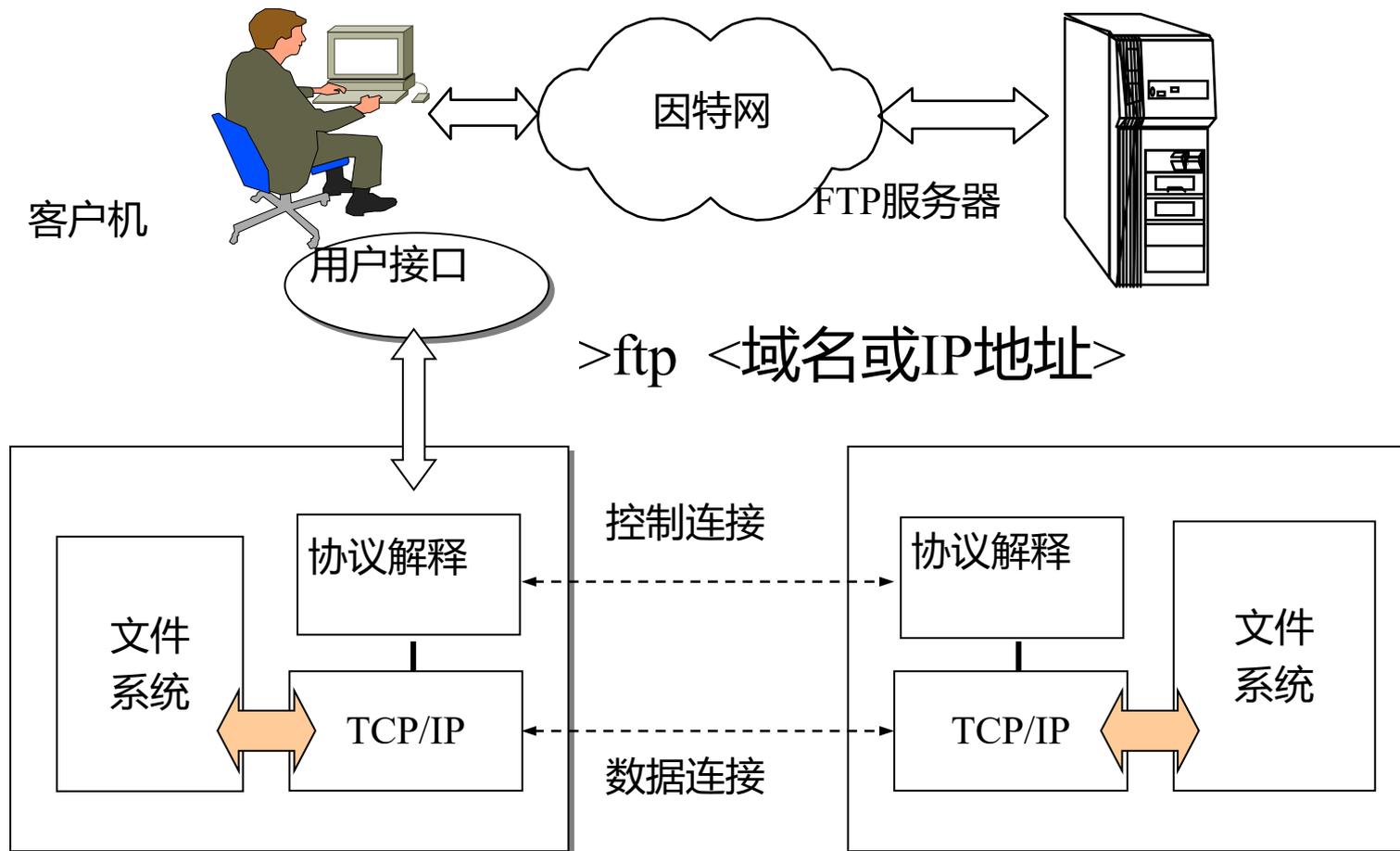
FTP特点

7.4.1 文件传输协议

- 文件传送协议 FTP 只提供文件传送的一些基本的服务，它使用 TCP 可靠的运输服务。
- FTP 的主要功能是减少或消除在不同操作系统下处理文件的不兼容性。
- FTP 使用**客户服务器方式**。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**，负责接受新的请求；另外有若干个**从属进程**，负责处理单个请求。

FTP功能模块与连接

7.4.1 文件传输协议





主进程的工作步骤

7.4.1 文件传输协议

- 打开熟知端口（端口号为21），使客户进程能够与服务器建立连接。
- 等待客户进程发出连接请求。
- 启动从属进程来处理客户进程发来的请求。从属进程对客户进程的请求处理完毕后即终止，但从属进程在运行期间根据需要还可能创建其他一些子进程。
- 回到等待状态，继续接受其他客户进程发来的请求。主进程与从属进程的处理是并发地进行。



两个连接

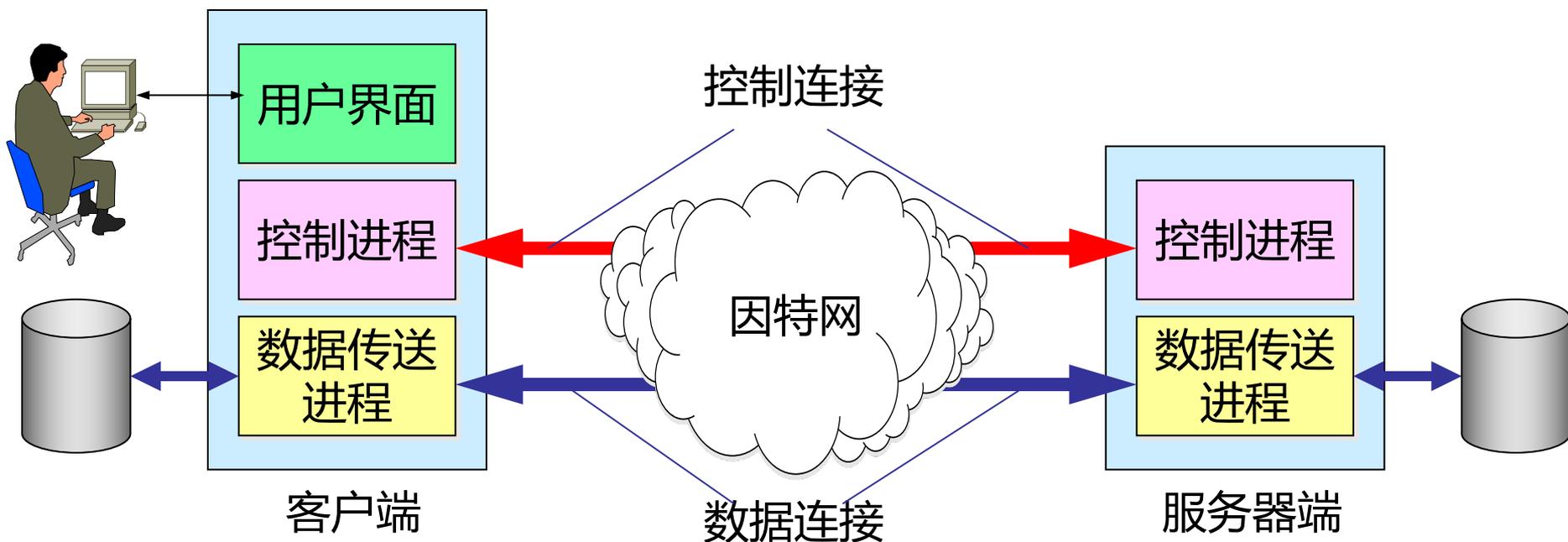
7.4.1 文件传输协议

- **“控制连接”** 在整个会话期间一直保持打开，FTP 客户发出的传送请求通过控制连接发送给服务器端的控制进程，但控制连接不用来传送文件。
- 实际用于传输文件的是 **“数据连接”**。服务器端的控制进程在接收到 FTP 客户发送来的文件传输请求后创建 **“数据传送进程”** 和 **“数据连接”**，用来连接客户端和服务器的数据传送进程。
- 数据传送进程实际完成文件的传送，在传送完毕后关闭 **“数据传送连接”**，并结束运行。



FTP 使用的两个 TCP 连接

7.4.1 文件传输协议 >>





两个不同的端口号

7.4.1 文件传输协议

- 当客户进程向服务器进程发出建立连接请求时，要寻找连接服务器进程的**熟知端口(21)**，同时还要告诉服务器进程自己的另一个端口号码，用于建立数据传送连接。
- 接着，服务器进程用自己传送数据的熟知端口(20)与客户进程所提供的端口号码建立**数据传送连接**。
- 由于FTP使用了两个不同的端口号，所以数据连接与控制连接不会发生混乱。



使用两个不同的端口号形成两个独立连接

7.4.1 文件传输协议

- 使用两个独立的连接的主要特点
 - 使协议更加简单和更容易实现。
 - 在传输文件时还可以利用控制连接（例如，客户发送请求终止传输）。
- TCP依靠两个端点标识一个连接，因此主进程创建从属进程时，可以不使用临时端口。即服务器可以使用20、21端口建立多个控制连接和数据连接。



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] Request login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 login ok, access restrictions apply.
[08] ftp> get rfc1261.txt nicinfo
[09] 250 get rfc1261.txt successful.
[10] ftp>
```

[01] 用户要用 FTP 和远地主机(网络信息中心 NIC 上的主机)建立连接。

```
local: nicinfo remote: rfc1261.txt
4488 bytes received in 15 seconds (0.3 Kbytes/s).
[14] ftp> quit
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 User not login ok, access restrictions apply.
[08] ftp> cd /
[09] 250 CWD successful.
[10] ftp> get rfc1261.txt nicinfo
[11] (128.36.12.27,1401) (4318 bytes).
[12] 226 ASCII Transfer complete.
    local: nicinfo remote: rfc1261.txt
    4488 bytes received in 15 seconds (0.3 Kbytes/s).
[13] ftp> quit
[14] 221 Goodbye.
```

[02] 本地 FTP 发出的连接成功信息。



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 31 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 200 Guest login ok, access restrictions apply.
[08] ftp> get nicinfo.txt
[09] 250 local: nicinfo remote: rfc1261.txt
[10] ftp> get nicinfo.txt
[11] 200 local: nicinfo remote: rfc1261.txt
[12] 4488 bytes received in 15 seconds (0.3 Kbytes/s).
[13] 226 ASCII Transfer complete.
[14] ftp> quit
[15] 221 Goodbye.
```

[03] 从远地服务器返回的信息，220 表示“服务就绪”。



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] Guest login ok, access restrictions apply.
[08] ftp> get rfc
[09] 250 Command successful.
[10] ftp> get 1261.txt nicinfo
[11] 200 Local file transferred successfully.
[12] 150 ASCII connection for rfc1261.txt
```

[04] 本地 FTP 提示用户键入名字。用户键入的名字表示“匿名”。用户只需键入 anonymous 即可。

```
[14] ftp> quit
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] > cd rfc
[09] CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 command successful.
[12] 150 opening connection for rfc1261.txt
    (128...01) (4318 bytes).
```

[05] 数字 331 表示“用户名正确”，需要口令。

```
[14] ftp> quit
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] ftp> cd rfc
[09] 200 CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 GET command successful.
[12] 150 data connection for rfc1261.txt
      (128... 401) (4318 bytes).
[13] 226 ASCII transfer complete.
```

[06] 本地 FTP 提示用户键入口令。用户这时可键入 guest 作为匿名的口令，也可以键入自己的电子邮件地址，即耶鲁大学数学系名为 xyz 的主机上的 abc。



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] ftp> cd rfc
[09] 250 CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] PORT command successful.
[12] 150 opening BINARY data connection for rfc1261.txt
    (192.168.27,1401) (4318 bytes).
[13] 226 Transfer complete.
    local: rfc1261.txt
    remote: rfc1261.txt
    (4318 bytes/s).
```

[07] 数字 230 表示用户已经注册完毕。



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] ftp> cd rfc
[09] 250 CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 PORT command successful.
[12] 150 opening data connection for rfc1261.txt
      (128.1.1.1401) (4318 bytes).
[13] 226 A file transfer complete.
```

“ftp>”是 FTP 的提示信息。用户键入的是将目录改变为包含 RFC 文件的目录。

```
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] ftp> cd rfc
[09] 250 CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 PORT command successful.
[12] ASCII data connection for rfc1261.txt
[13] 6.12.27,1401) (4318 bytes).
[14] 226 Transfer complete.
[15] local file is fo remote: rfc1261.txt
[16] 448 bytes received in 15 seconds (0.3 Kbytes/s).
```

[09] 字符 CWD 是 FTP 的标准命令，
代表 Change Working Directory。



FTP

[10] 用户要求将名为 rfc1261.txt 的文件复制到本地主机上，并改名为 nicinfo。

```
[01]
[02] connecte...mil
[03] 220 nic... (Sunos 4.1) ready.
[04] Name:
[05] 331 ... login ok, send ident as password.
[06] Pa... abc@xyz.math.yale.edu
[07] 2... dest login ok, access restrictions apply.
[08] ftp> cd rfc
[09] 250 CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 PORT command successful.
[12] 150 ASCII data connection for rfc1261.txt
    (128.36.12.27,1401) (4318 bytes).
[13] 226 ASCII Transfer complete.
    local: nicinfo remote: rfc1261.txt
    4488 bytes received in 15 seconds (0.3 Kbytes/s).
[14] ftp> quit
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

[11] 字符 PORT 是 FTP 的标准命令，表示要
建立数据连接。200 表示“命令正确”。

```
[01]
[02]
[03]
[04] name. an
[05] 331 Guest, send ident as password.
[06] Password: xyz.math.yale.edu
[07] 230 Login ok, access restrictions apply.
[08] ftp> cd rfc
[09] CWD command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 PORT command successful.
[12] 150 ASCII data connection for rfc1261.txt
    (128.36.12.27,1401) (4318 bytes).
[13] 226 ASCII Transfer complete.
    local: nicinfo remote: rfc1261.txt
    4488 bytes received in 15 seconds (0.3 Kbytes/s).
[14] ftp> quit
[15] 221 Goodbye.
```

FTP 的屏幕信息举例

7.4.1 文件传输协议

[12] 数字 150 表示“文件状态正确，即将建立数据连接”。

```
[01]
[02]
[03]
[04] name.
[05] 331 Go ahead and send ident as password.
[06] Password: xyz@xyz.math.yale.edu
[07] 230 Login ok, access restrictions apply.
[08] ftp> rfc
[09] 200 PORT command successful.
[10] ftp> get rfc1261.txt nicinfo
[11] 200 PORT command successful.
[12] 150 ASCII data connection for rfc1261.txt
(128.36.12.27,1401) (4318 bytes).
[13] 226 ASCII Transfer complete.
    local: nicinfo remote: rfc1261.txt
    4488 bytes received in 15 seconds (0.3 Kbytes/s).
[14] ftp> quit
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] [13] 数字 226 是“释放数据连接”。  
[02] 现在一个新的本地文件已产生。  
[03]  
[04] name.  
[05] 331 Get password ok, send ident as password.  
[06] Password: xyz@xyz.math.yale.edu  
[07] 230 Login ok, access restrictions apply.  
[08] ftp> rfc  
[09] 200 PORT command successful.  
[10] ftp> get rfc1261.txt nicinfo  
[11] 200 PORT command successful.  
[12] 150 ASCII data connection for rfc1261.txt  
    (128.36.12.27,1401) (4318 bytes).  
[13] 226 ASCII Transfer complete.  
    local: nicinfo remote: rfc1261.txt  
    4488 bytes received in 15 seconds (0.3 Kbytes/s).  
[14] ftp> quit  
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] ftp>
[09] 25 [14] 用户键入退出命令。
[10] ftp>
[11] 200 PORT command successful.
[12] 150 Accepted data connection for rfc1261.txt
    (128.112.27,1401) (4318 bytes).
[13] 227 ASCII Transfer complete.
    local: nicinfo remote: rfc1261.txt
    4488 bytes received in 15 seconds (0.3 Kbytes/s).
[14] ftp> quit
[15] 221 Goodbye.
```



FTP 的屏幕信息举例

7.4.1 文件传输协议

```
[01] ftp nic.ddn.mil
[02] connected to nic.ddn.mil
[03] 220 nic FTP server (Sunos 4.1) ready.
[04] Name: anonymous
[05] 331 Guest login ok, send ident as password.
[06] Password: abc@xyz.math.yale.edu
[07] 230 Guest login ok, access restrictions apply.
[08] ft
[09] 25
[10] ft
[11] 200 PORT command successful.
[12] 150 ASCII data connection for rfc1261.txt
    (128.2.27,1401) (4318 bytes).
[13] 226 ASCII Transfer complete.
    local: nicinfo remote: rfc1261.txt
    488 bytes received in 15 seconds (0.3 Kbytes/s).
[14] ftp> quit
[15] 221 Goodbye.
```

[15] 表明 FTP 工作结束。



例题

7.4.1 文件传输协议

- 下面关于文件传送协议FTP的叙述，错误的是（ ）。
- A、FTP使用客户服务器方式
- B、一个FTP服务器进程可同时为多个客户进程提供服务
- C、FTP的客户和服务器之间要建立两个连接，即控制连接和数据连接
- D、FTP基于UDP协议



例题

7.4.1 文件传输协议

- FTP客户和服务端之间传递FTP命令时，使用的连接是（ ）
- A.建立在TCP之上的控制连接
- B.建立在TCP之上的数据连接
- C.建立在UDP之上的控制连接
- D.建立在UDP之上的数据连接

内容提要

Transition Page



01 应用层协议与网络应用模式

02 域名系统 (**DNS**)

03 远程登录 (**Telnet**)

04 文件传输协议 (**FTP**)

05 引导程序协议与
动态主机配置协议 (**DHCP**)

06 电子邮件系统和**SMTP**

07 万维网与**HTTP**



7.5 引导程序协议与动态主机配置协议 >>

在系统初始化（引导程序中）需要解决的问题

- 主机如何获取其IP地址？
- 主机如何知道其DNS服务器？
- 主机如何知道其子网掩码？
- 主机如何获悉如何连到其他网络？



7.5 引导程序协议与动态主机配置协议

主机如何获得IP地址？

- 方法一：人工配置
 - e.g., Linux: /etc/network/interfaces
 - Bootp协议
- 方法二：系统动态获取
 - Dynamic Host Configuration Protocol (DHCP)



7.5 引导程序协议与动态主机配置协议

7.5.1 引导程序协议编 (BOOTP)

BOOTP使用UDP为无盘工作站提供自动获取配置信息服务。

BOOTP使用C/S服务模式。为了获取配置信息，协议软件广播一个BOOTP请求报文，使用全1广播地址作为目的地址，而全0作为源地址。



7.5.1 引导程序协议编 (BOOTP)

- 收到请求报文的BOOTP服务器查找该计算机的各项配置信息 (如IP地址, 子网掩码, 默认路由器的IP地址, 域名服务器的IP地址) 后, 将其放入一个BOOTP响应报文, 可以采用广播方式回送给提出请求的计算机,
- 或使用收到广播帧上的硬件地址 (网卡地址) 进行单播。



7.5.1 引导程序协议编 (BOOTP)

- BOOTP是一个**静态配置协议**。
- 当BOOTP服务器收到某主机的请求时，就在其数据库中查找该主机已确定的地址绑定信息。
- 一旦当主机移动到其他网络时，则BOOTP不能提供服务，除非管理员人工添加或修改数据库信息。



7.5.2 动态主机配置协议

- 动态主机配置协议 (DHCP, Dynamic Host Configuration Protocol) 是与BOOTP兼容的协议
- 即插即用连网 (Plug-and-Play networking)
- DHCP允许一台计算机加入新网可自动获取IP地址, 不用人工参与。



DHCP使用C/S服务模式。 7.5.2 动态主机配置协议

- 当某主机需要IP地址，启动时向DHCP服务器发送广播报文（目的IP地址为全1，源IP地址置全0），命名为**广播发现报文**（DHCPDISCOVER），主机成为DHCP客户。
- 在本地网络的所有主机均能收到该广播发现报文，惟有DHCP服务器对此报文予以响应。
- DHCP服务器先在其数据库中查找该计算机配置信息，若找到，则采用**提供报文**（DHCPOFFER）将其回送到主机；若找不到，则从服务器的IP地址池中任选一个IP地址分配给主机。

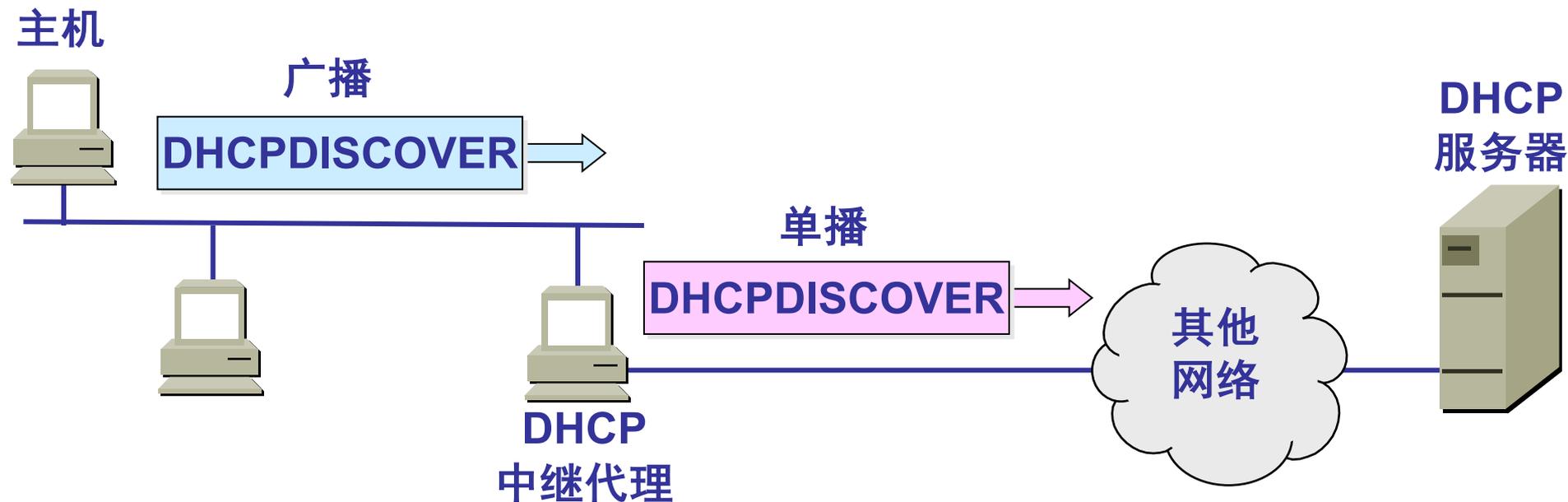


- 并不是每个网络上都有 DHCP 服务器，这样会使 DHCP 服务器的数量太多。现在是每一个网络至少有一个 DHCP **中继代理**，它配置了 DHCP 服务器的 IP 地址信息。
- 当 DHCP 中继代理收到主机发送的发现报文后，就以单播方式向 DHCP 服务器转发此报文，并等待其回答。收到 DHCP 服务器回答的提供报文后，DHCP 中继代理再将此提供报文发回给主机。



DHCP 中继代理 以单播方式转发发现报文

7.5.2 动态主机配置协议



注意：DHCP 报文只是 UDP 用户数据报中的数据。

内容提要

Transition Page



01 应用层协议与网络应用模式

02 域名系统 (**DNS**)

03 远程登录 (**Telnet**)

04 文件传输协议 (**FTP**)

05 引导程序协议与
动态主机配置协议 (**DHCP**)

06 电子邮件系统和**SMTP**

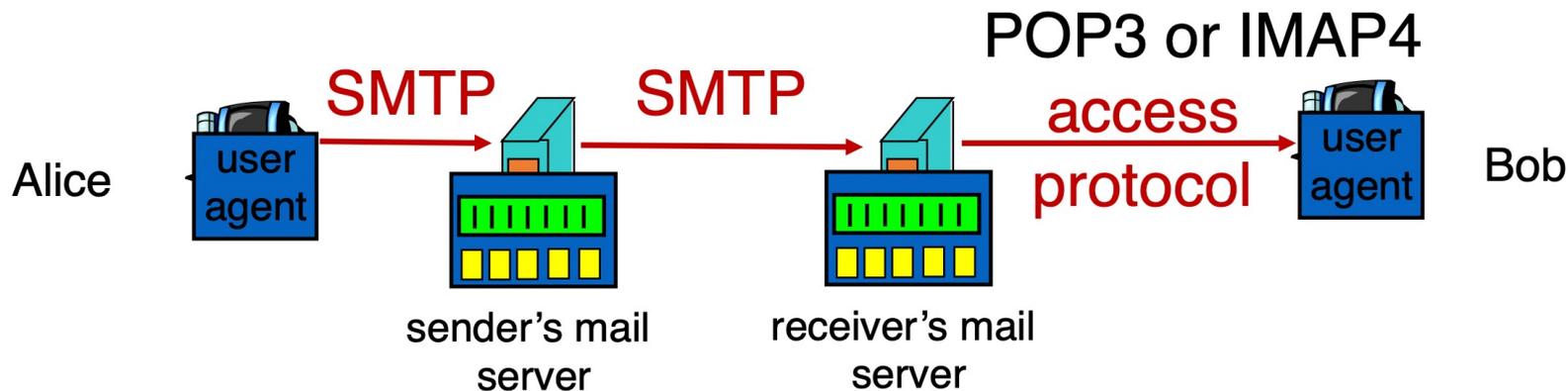
07 万维网与**HTTP**



7.6 电子邮件系统与SMTP

- 电子邮件(e-mail)是因特网上使用得最多的和最受用户欢迎的一种应用。
- 电子邮件把邮件发送到 ISP 的邮件服务器，并放在其中的收信人邮箱中，收信人可随时上网到 ISP 的邮件服务器进行读取。
- 电子邮件不仅使用方便，而且还具有**传递迅速和费用低廉**的优点。
- 现在电子邮件不仅可传送文字信息，而且还可附上声音和图像。

7.6 电子邮件系统与SMTP



- SMTP (simple mail transfer protocol): 发送邮件至邮件服务器上。工作方式为推送(focused on **push**)。
- 邮件接收协议(Mail access protocol) : 从邮件服务器中拉取信息(focused on **pull**)
 - 邮局协议POP: Post office protocol [RFC 1939]
 - IMAP: Internet Mail Access Protocol [RFC 1730]
 - HTTP: Gmail, outlook, etc.



7.6.1 电子邮件系统的组成

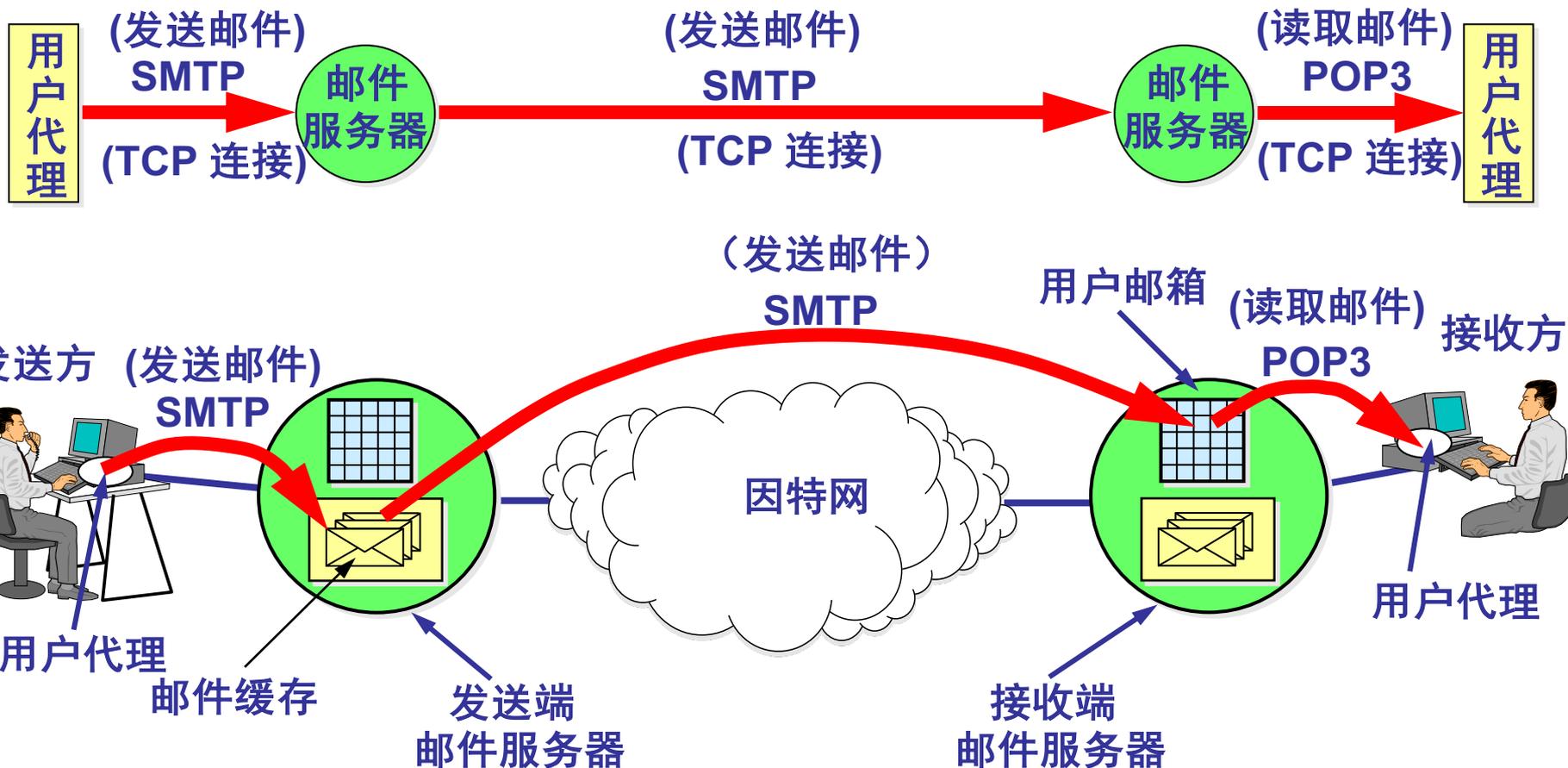
- **用户代理 UA** 就是用户与电子邮件系统的接口。
- 用户代理的功能是：撰写、显示和处理。
- **邮件服务器**的功能是发送和接收邮件，同时还要向发信人报告邮件传送的情况（已交付、被拒绝、丢失等）。
- 邮件服务器按照客户服务器方式工作。邮件服务器需要使用两个不同的协议。
 - **SMTP 协议用于发送邮件。**
 - **Mail access protocol (POP or IMAP)用于接收邮件。**

7.6.1 电子邮件系统的组成

- 一个邮件服务器既可以作为客户，也可以作为服务器。
- 例如，当邮件服务器 A 向另一个邮件服务器 B 发送邮件时，邮件服务器 A 就作为 SMTP 客户，而 B 是 SMTP 服务器。
- 当邮件服务器 A 从另一个邮件服务器 B 接收邮件时，邮件服务器 A 就作为 SMTP 服务器，而 B 是 SMTP 客户。

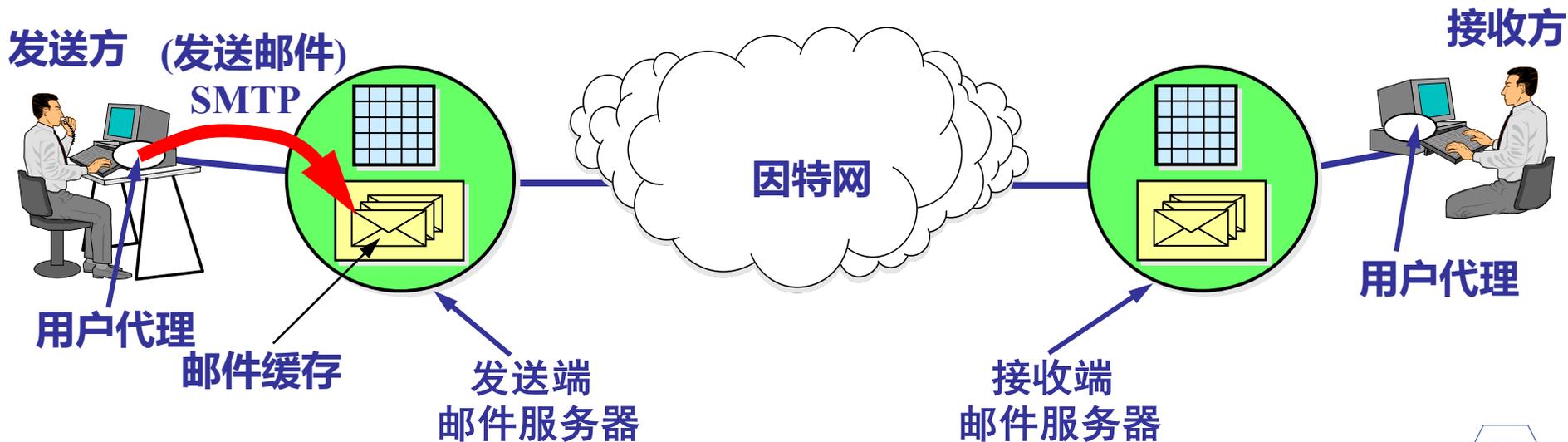
电子邮件的最主要的组成构件

7.6.1 电子邮件系统的组成



电子邮件的发送和接收过程 7.6.1 电子邮件系统的组成

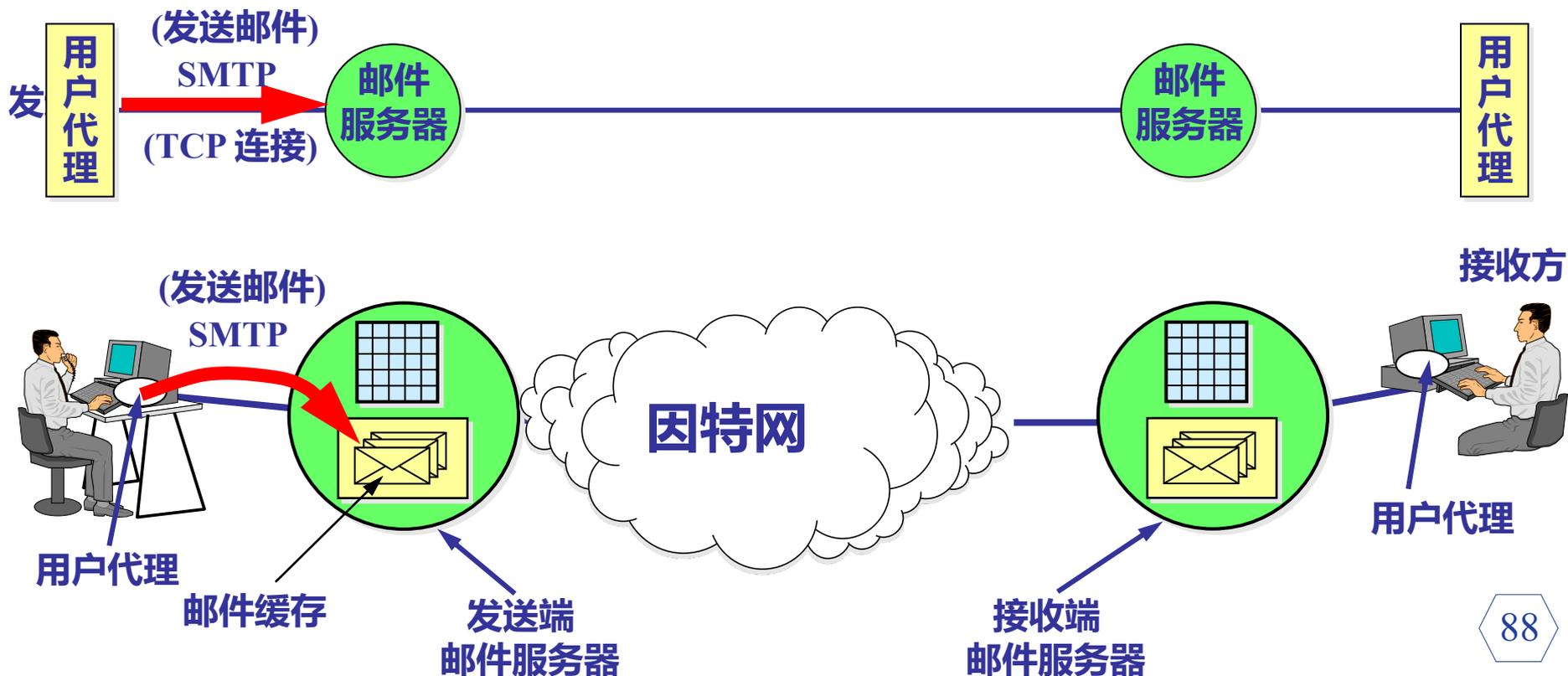
(1) 发信人调用用户代理来编辑要发送的邮件。用户代理用 SMTP 把邮件传送给发送端邮件服务器。



电子邮件的发送和接收过程

7.6.1 电子邮件系统的组成

(2) 发送端邮件服务器将邮件放入邮件缓存队列中，等待发送。

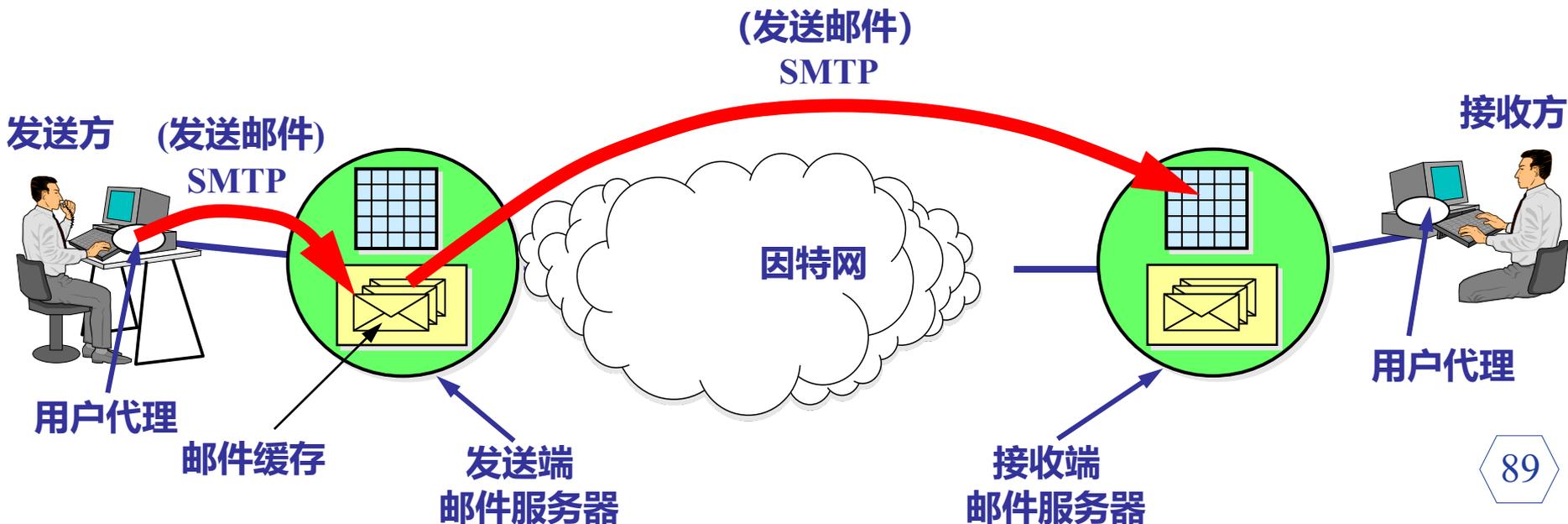




电子邮件的发送和接收过程

7.6.1 电子邮件系统的组成

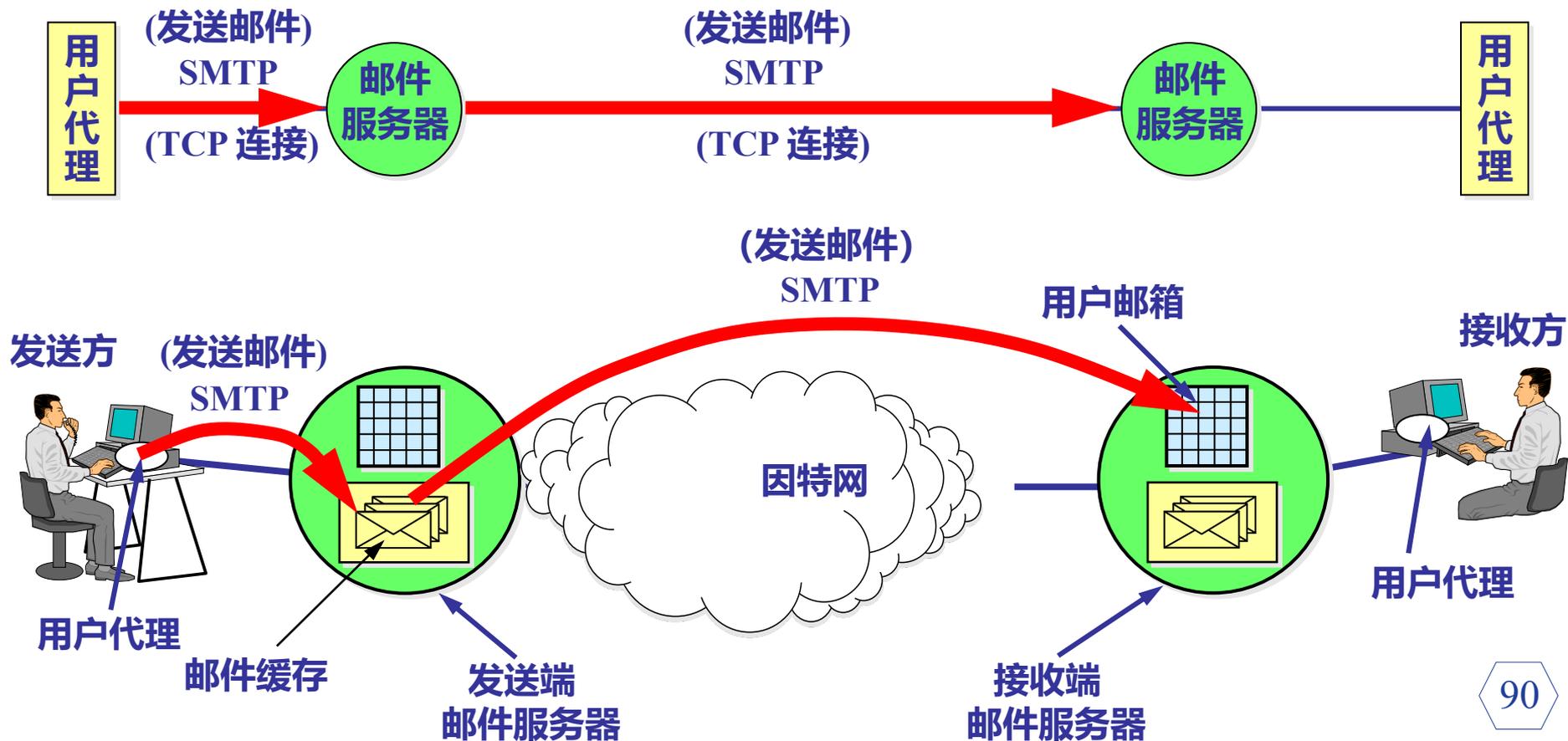
(3) 运行在发送端邮件服务器的 SMTP 客户进程，发现在邮件缓存中有待发送的邮件，就向运行在接收端邮件服务器的 SMTP 服务器进程发起 TCP 连接的建立。



电子邮件的发送和接收过程

7.6.1 电子邮件系统的组成

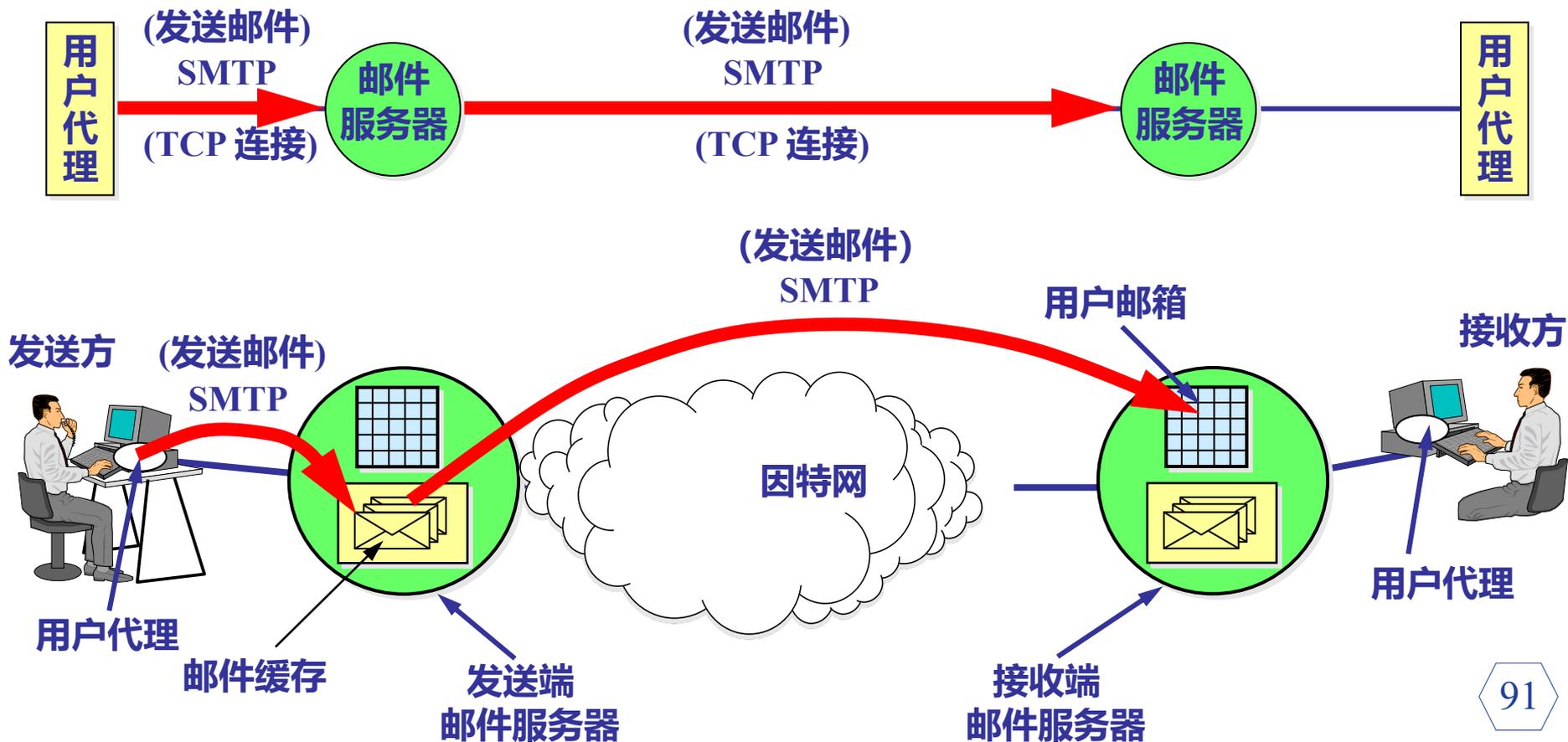
(4) TCP 连接建立后, SMTP 客户进程开始向远程的 SMTP 服务器进程发送邮件。当所有的待发送邮件发完了, SMTP 就关闭所建立的 TCP 连接。



电子邮件的发送和接收过程

7.6.1 电子邮件系统的组成

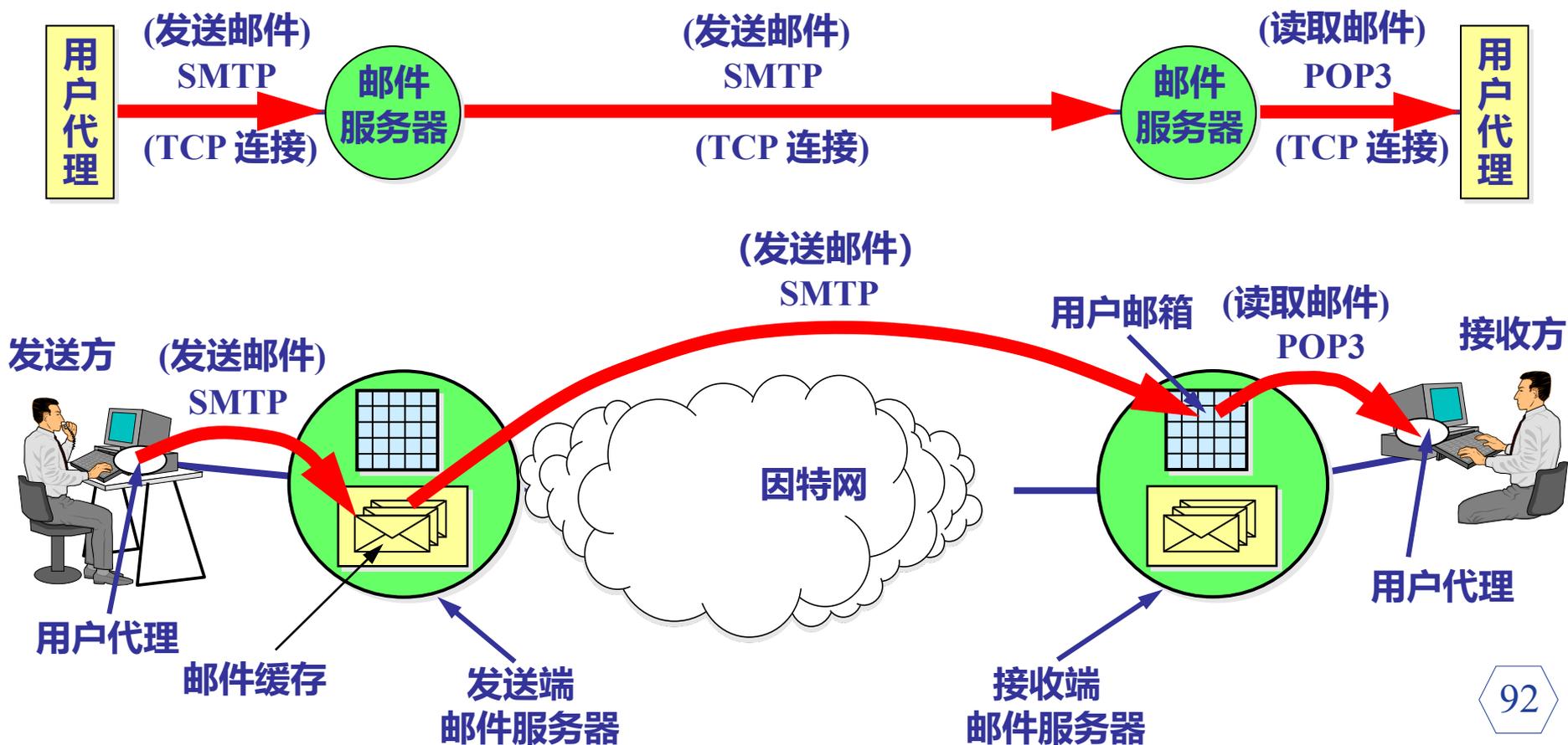
(5) 运行在接收端邮件服务器中的 SMTP 服务器进程收到邮件后，将邮件放入收信人的用户邮箱中，等待收信人在方便时进行读取。



电子邮件的发送和接收过程

7.6.1 电子邮件系统的组成

(6) 收信人在打算收信时，调用用户代理，使用 POP3 (或 IMAP) 协议将自己的邮件从接收端邮件服务器的用户邮箱中的取回 (如果邮箱中有来信的话)。





电子邮件的组成

7.6.1 电子邮件系统的组成

- 电子邮件由**信封(envelope)**和**内容(content)**两部分组成。
- 电子邮件的传输程序根据邮件信封上的信息来传送邮件。用户在从自己的邮箱中读取邮件时才能见到邮件的内容。
- 在邮件的信封上，最重要的就是收信人的地址。
- 电子邮件信封中的相关信息，可以自动从内容中获得。

电子邮件地址的格式

7.6.1 电子邮件系统的组成

- TCP/IP 体系的电子邮件系统规定电子邮件地址的格式如下：

收信人邮箱名@邮箱所在主机的域名

- 符号 “@” 读作 “at”，表示 “在” 的意思。
- 例如，电子邮件地址 zhezhang@njupt.edu.cn

这个用户名在该域名的范围内是唯一的。

邮箱所在的主机的域名在全世界必须是唯一的



7.6.2 简单邮件传送协议SMTP

- SMTP 所规定的就是在两个相互通信的 SMTP 进程之间应如何交换信息。
- 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 规定了 14 条命令和 21 种应答信息。每条命令用 4 个字母组成，而每一种应答信息一般只有一行信息，由一个 3 位数字的代码开始，后面附上（也可不附上）很简单的文字说明。



SMTP 通信的三个阶段

7.6.2 简单邮件传送协议SMTP

- 1. 连接建立：连接是在发送主机的 SMTP 客户和接收主机的 SMTP 服务器之间建立的。SMTP 不使用中间的邮件服务器。
- 2. 邮件传送
- 3. 连接释放：邮件发送完毕后，SMTP 应释放 TCP 连接。



电子邮件的信息格式

7.6.2 简单邮件传送协议SMTP



- 一个电子邮件分为**信封**和**内容**两大部分。
- [RFC 822]只规定了邮件内容中的**首部**(header)格式，而对邮件的**主体**(body)部分则让用户自由撰写。
- 用户写好首部后，邮件系统将自动提取信封所需的信息，并写在信封上。所以用户不需要填写电子邮件信封上的信息。
- 邮件内容首部包括一些关键字，后面加上冒号。**最重要的关键字**是：
To 和 Subject。



邮件内容的首部

7.6.2 简单邮件传送协议SMTP

- **“To:”** 后面填入一个或多个收信人的电子邮件地址。用户只需打开地址簿，点击收信人名字，收信人的电子邮件地址就会自动地填入到合适的位置上。
- **“Subject:”** 是邮件的主题。它反映了邮件的主要内容，便于用户查找邮件。
- 抄送 **“Cc:”** 表示应给某某人发送一个邮件副本。
- **“From”** 和 **“Date”** 表示发信人的电子邮件地址和发信日期。
“Reply-To” 是对方回信所用的地址。



7.6.4 通用因特网邮件扩充MIME

- **SMTP 有以下缺点：**
- SMTP 不能传送可执行文件或其他的二进制对象。
- SMTP 限于传送 7 位的 ASCII 码。许多其他非英语国家的文字（如中文、俄文，甚至带重音符号的法文或德文）就无法传送。
- SMTP 服务器会拒绝超过一定长度的邮件。
- 某些 SMTP 的实现并没有完全按照[RFC 821]的 SMTP 标准。



电子邮件的一些标准

7.6 电子邮件系统与SMTP

- 在 1982 年制定出**简单邮件传送协议 SMTP** (Simple Mail Transfer Protocol) 和因特网文本报文格式, 它们都已成为因特网的正式标准。
- 1993 年提出了**通用因特网邮件扩展 MIME** (Multipurpose Internet Mail Extensions)。
- MIME 在其邮件首部中说明了邮件的数据类型(如文本、声音、图像、视像等)。**在 MIME 邮件中可同时传送多种类型的数据。**



- MIME 并没有改动 SMTP 或取代它，只是SMTP的一个扩展 (extension)。
- MIME 的意图是继续使用目前的[RFC 822]格式，但增加了邮件主体的结构，并定义了传送非 ASCII 码的编码规则。
- MIME实际上增加了SMTP的功能。



7.6.3 邮件读取协议 POP3 和 IMAP



- **邮局协议 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三个版本 **POP3**。
- POP 也使用客户服务器的工作方式。
- 在接收邮件的用户 PC 机中必须运行 POP 客户程序，而在用户所连接的 ISP 的邮件服务器中则运行 POP 服务器程序。



- IMAP 也是按客户服务器方式工作，现在较新的版本是 IMAP4。
- 用户在自己的 PC 机上就可以操纵 ISP 的邮件服务器的邮箱，就像在本地操纵一样。
- IMAP 是一个联机协议。当用户 PC 机上的 IMAP 客户程序打开 IMAP 服务器的邮箱时，用户就可看到邮件的首部。若用户需要打开某个邮件，则该邮件才传到用户的计算机上。



IMAP 的特点

7.6.3 邮件读取协议 POP3 和 IMAP

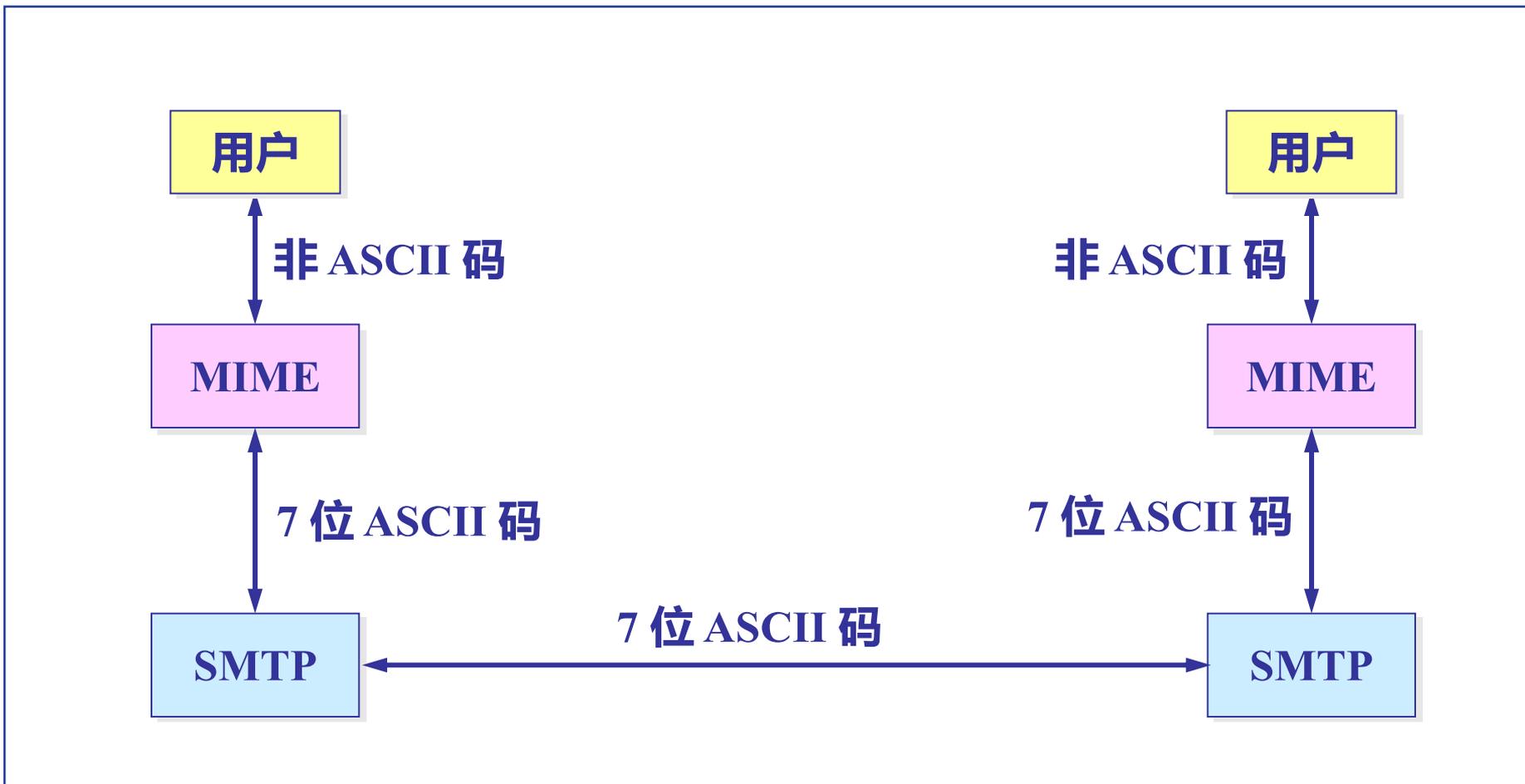


- IMAP最大的好处就是用户可以在不同的地方使用不同的计算机随时上网阅读和处理自己的邮件。
- IMAP 还允许收信人只读取邮件中的某一个部分。例如，收到了一个带有视像附件（此文件可能很大）的邮件。为了节省时间，可以先下载邮件的正文部分，待以后有时间再读取或下载这个很长的附件。
- IMAP 的缺点是如果用户没有将邮件复制到自己的 PC 机上，则邮件一直是存放在 IMAP 服务器上。因此用户需要经常与 IMAP 服务器建立连接。



MIME 和 SMTP 的关系

7.6.3 邮件读取协议 POP3 和 IMAP



内容提要

Transition Page



- 01 应用层协议与网络应用模式
- 02 域名系统 (DNS)
- 03 远程登录 (Telnet)
- 04 文件传输协议 (FTP)
- 05 引导程序协议与
动态主机配置协议 (DHCP)
- 06 电子邮件系统和SMTP
- 07 万维网与HTTP



7.7 域名解析服务

万维网 WWW (World Wide Web)并非某种特殊的计算机网络。

万维网是一个大规模的、联机式的信息储藏所。

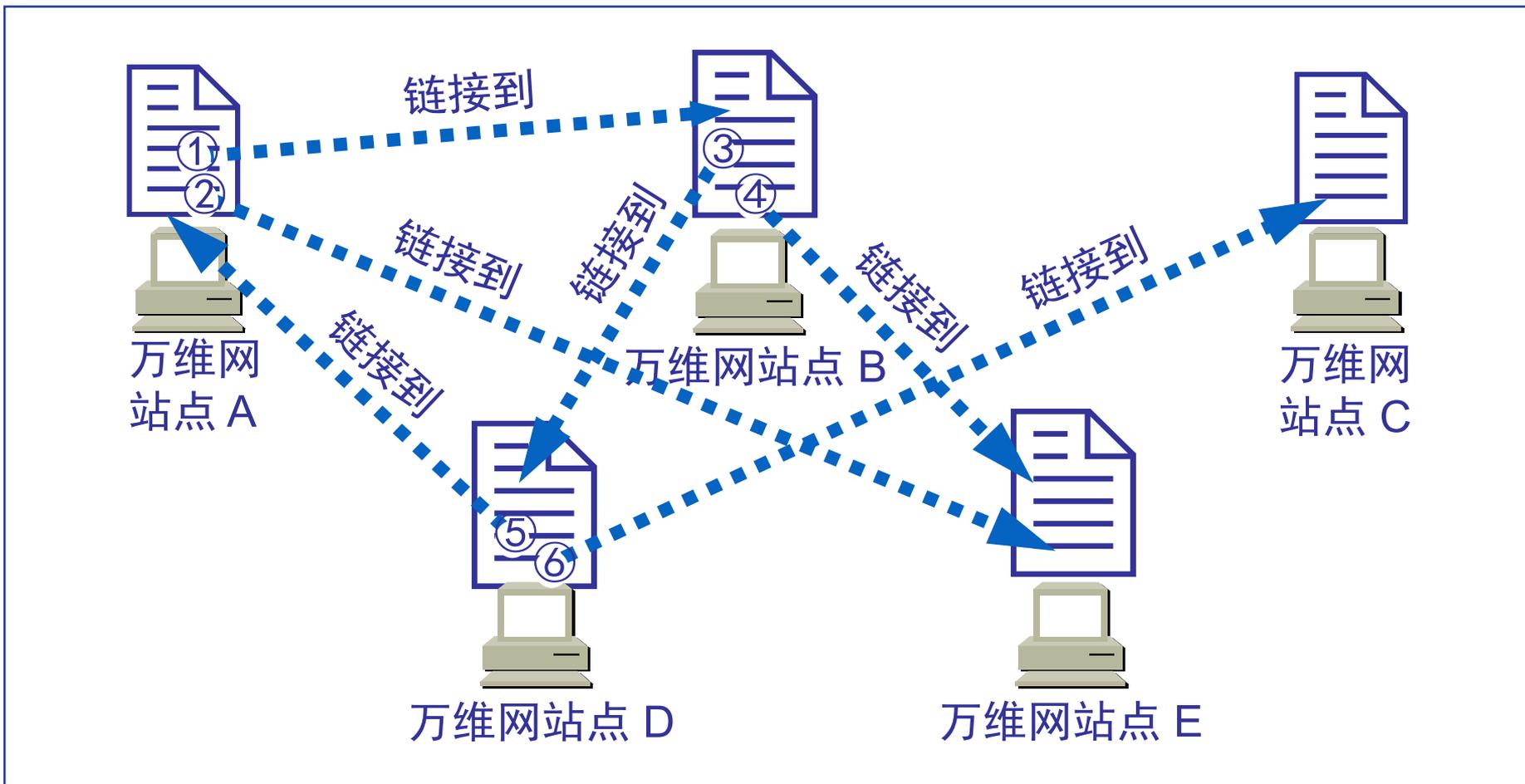
万维网用链接的方法能非常方便地从因特网上的一个站点访问另一个站点，从而主动地按需获取丰富的信息。

这种访问方式称为“**链接**”。



万维网提供分布式服务

7.7 域名解析服务





7.7.1 超文本传送协议HTTP

万维网是分布式**超媒体** (hypermedia) 系统，它是**超文本** (hypertext) 系统的扩充。

一个超文本由多个信息源**链接**成。利用一个链接可使用户找到另一个文档。这些文档可以位于世界上任何一个接在因特网上的超文本系统中。**超文本是万维网的基础。**

超媒体与超文本的区别是文档内容不同。超文本文档仅包含文本信息，而超媒体文档还包含其他表示方式的信息，如图形、图像、声音、动画，甚至活动视频图像。(web1.0->web2.0->web3.0)



万维网的工作方式

7.7.1 超文本传送协议HTTP

- 万维网以客户服务器方式工作。
- **浏览器**就是在用户计算机上的万维网**客户程序**。万维网文档所驻留的计算机则运行**服务器程序**，因此这个计算机也称为**万维网服务器**。
- 客户程序向**服务器程序**发出请求，服务器程序向客户程序送回客户所要的万维网文档。
- 在一个客户程序主窗口上显示出的万维网文档称为**页面**(page)。



万维网必须解决的问题 7.7.1 超文本传送协议HTTP

(1) 怎样标志分布在整个因特网上的万维网文档?

- 使用**统一资源定位符** URL (Uniform Resource Locator)来标志万维网上的各种文档。
- 使每一个文档在整个因特网的范围内具有唯一的标识符 URL。

(2) 用何协议实现万维网上各种超链的连接?

- 在万维网客户程序与万维网服务器程序之间进行交互所使用的协议，是**超文本传送协议** HTTP (HyperText Transfer Protocol)。
- HTTP 是一个应用层协议，它使用 TCP 连接进行可靠的传送。



7.7.1 超文本传送协议HTTP

(3) 怎样使各种万维网文档都能在因特网上的各种计算机上显示出来，同时使用户清楚地知道在什么地方存在着超链？

- **超文本标记语言** HTML (HyperText Markup Language)使得万维网页面的设计者可以很方便地用一个超链从本页面的某处链接到因特网上的任何一个万维网页面，并且能够在自己的计算机屏幕上将这些页面显示出来。

(4) 怎样使用户能够很方便地找到所需的信息？

- 为了在万维网上方便地查找信息，用户可使用各种的搜索工具（即搜索引擎）。



统一资源定位符 URL 7.7.1 超文本传送协议HTTP

- **统一资源定位符 URL** 是对可以从因特网上得到的资源的位置和访问方法的一种简洁的表示。
- URL 给资源的位置提供一种抽象的识别方法，并用这种方法给资源定位。
- 只要能够对资源定位，系统就可以对资源进行各种操作，如存取、更新、替换和查找其属性。
- URL 相当于一个文件名在网络范围的扩展。因此 URL 是与因特网相连的机器上的任何可访问对象的一个指针。



URL 的一般形式

7.7.1 超文本传送协议HTTP

- 由以冒号隔开的两大部分组成，并且在 URL 中的字符对大写或小写没有要求。
- URL 的一般形式是：

<URL的访问方式>://<主机>:<端口>/<路径>

ftp —— 文件传送协议 FTP

http —— 超文本传送协议 HTTP

News —— USENET 新闻



URL 的一般形式

7.7.1 超文本传送协议HTTP

- 由以冒号隔开的两大部分组成，并且在 URL 中的字符对大写或小写没有要求。
- URL 的一般形式是：

<URL的访问方式>://<主机>:<端口>/<路径> (8-2)

有时可省略

<主机> 是存放资源的主机
在因特网中的域名



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 FTP 的 URL 举例

`ftp://rtfm.mit.edu/pub/abc.txt`

这表示使用 FTP 协议



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 FTP 的 URL 举例

`ftp://rtfm.mit.edu/pub/abc.txt`

冒号和两个斜线是规定的格式



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 FTP 的 URL 举例

ftp://rtfm.mit.edu/pub/abc.txt

这是网站的域名



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 FTP 的 URL 举例

ftp://rtfm.mit.edu/pub/abc.txt

这是路径和文件名



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP



- 使用 HTTP 的 URL 的一般形式

http://<主机>:<端口>/<路径>

↑
这表示使用 HTTP 协议



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 HTTP 的 URL 的一般形式

http://<主机>:<端口>/<路径>

冒号和两个斜线是规定的格式



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 HTTP 的 URL 的一般形式

http://<主机>:<端口>/<路径>

这里写主机的域名



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 HTTP 的 URL 的一般形式

http://<主机>:<端口>/<路径>

HTTP 的默认端口号是 80，通常可省略



使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 HTTP 的 URL 的一般形式

http://<主机>:<端口>/<路径>

若再省略文件的<路径>项, 则 URL 就指到因特网上的某个[主页](#)(home page)。

使用 FTP 的 URL

7.7.1 超文本传送协议HTTP

- 使用 HTTP 的 URL 的一般形式

`https://petrelli.github.io/zhe/download/`

若再省略文件的<路径>项, 则 URL 就指到因特网上的某个**主页**(home page)。

`https://petrelli.github.io/zhe/`



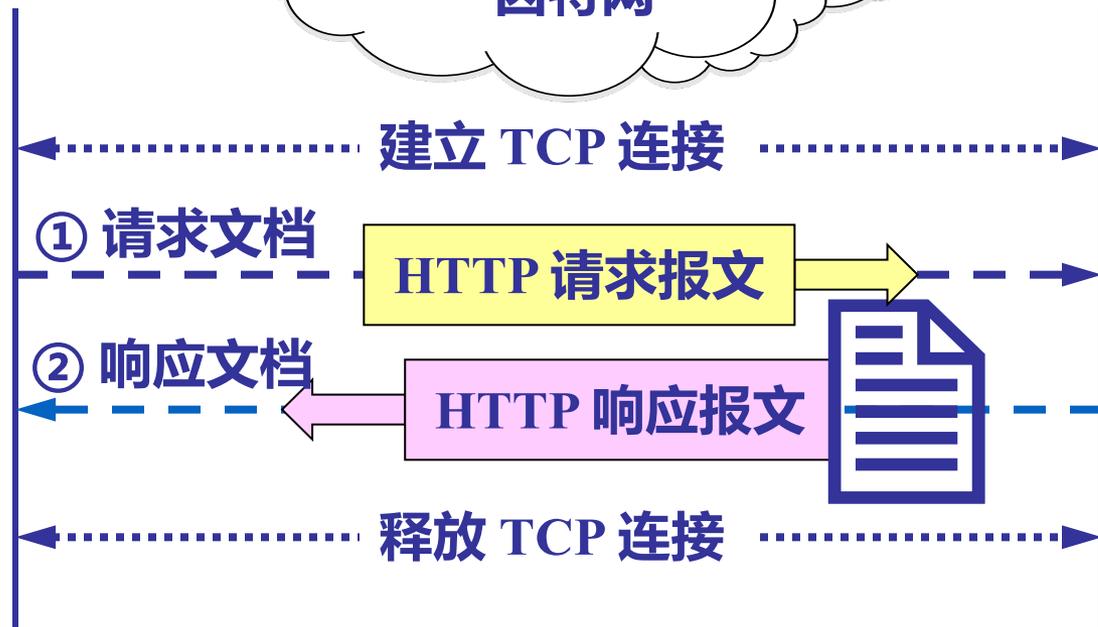
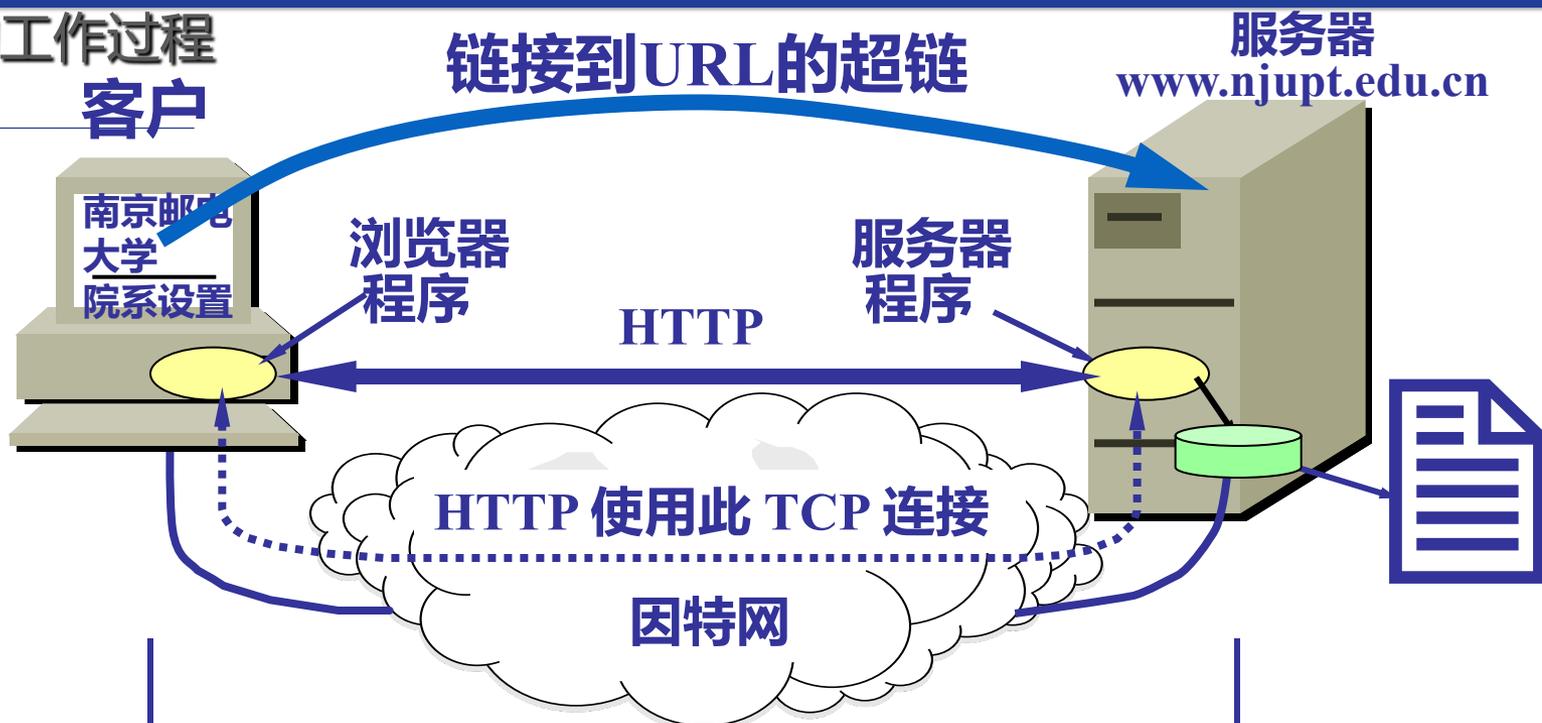
超文本传送协议 HTTP HTTP 的操作过程

7.7.1 超文本传送协议HTTP

- 为了使超文本的链接能够高效率地完成，需要用 HTTP 协议来传送一切必须的信息。
- 从层次的角度看，HTTP 是**面向事务的** (transaction-oriented)应用层协议，它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。



万维网的工作过程





- (1) 浏览器分析超链指向页面的 URL。
- (2) 浏览器向 DNS 请求解析 `www.njupt.edu.cn` 的 IP 地址。
- (3) 域名系统 DNS 解析出南京邮电大学服务器的 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：
`GET /chn/yxsx/index.htm`。
- (6) 服务器 给出响应，把文件 `index.htm` 发给浏览器。
- (7) TCP 连接释放。
- (8) 浏览器显示 “南京邮电大学院系设置” 文件 `index.htm` 中的所有文本。



浏览器中所看到的页面 7.7.1 超文本传送协议HTTP

南京邮电大学 Nanjing University of Posts and Telecommunications

科技改革30条 二十大专题网 理论学习平台 足迹 榜样 校庆专题网

旧版入口 | 外来访客 | 南邮校友 | 教职员工 | 在校学生

请输入关键字

首页 南邮概况 部门设置 学科建设 科学研究 招生就业 信息公开 人才培养 智慧校园 English

疫情防控 口罩先行

PLEASE WEAR A MASK

- Wear A Mask 佩戴口罩
- Wash Hands 勤快洗手
- Keep Distance 保持距离
- Keep Ventilation 保持通风

疫情防控千万条，戴好口罩第一条！

南邮要闻

更多 >>

- 【奋进新征程】校党委理论学习中心组专题学习《中国共产党章程...》 2022-11-02
- 我校召开江苏省高价值专利培育计划项目——“5G关键核心技术及应用...”项目启动会 2022-11-02
- 【奋进新征程】我校组织开展研究生“导学思政”下午茶系列活动 2022-10-29
- 通信与网络技术国家地方联合工程研究中心顺利通过验收 2022-10-28
- 射频集成与微组装技术国家地方联合工程实验室顺利通过验收 2022-10-28
- 我校杰出校友、国际电信联盟秘书长赵厚麟会见联合国机构负责人 2022-10-27

媒体印痕

更多 >>

- > 【新华日报】老有所学，课程更适老更多彩 2022-11-07
- > 【江苏教育报】敢为善为挑重担 务实落实扛重责 2022-11-04

通知公告

更多 >>

- 11-07 学校核酸采样安排（11月8日 周二）



HTTP 的主要特点

7.7.1 超文本传送协议HTTP

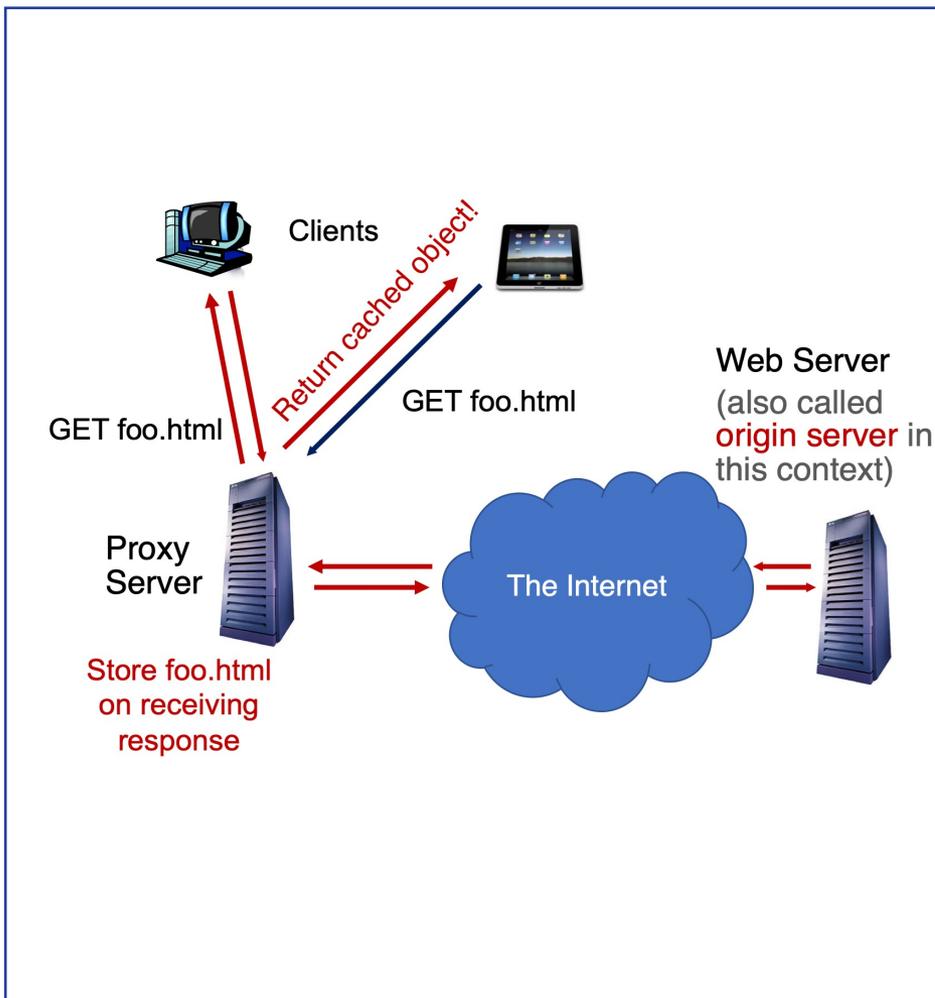
- HTTP 是面向事务的客户服务器协议。
- HTTP 1.0 协议是**无状态的**(stateless)。
- HTTP 协议本身也是无连接的，虽然它使用了面向连接的 TCP 向上提供的服务。
- 万维网浏览器就是一个 HTTP 客户，而在万维网服务器等待 HTTP 请求的进程常称为 HTTP daemon，有的文献将它缩写为 HTTPD。
- HTTP daemon 在收到 HTTP 客户的请求后，把所需的文件返回给 HTTP 客户。



7.7.1 超文本传送协议HTTP

- 万维网高速缓存代表浏览器发出 HTTP 请求，因此又称为 **代理服务器**(proxy server)。
- 万维网高速缓存将最近的一些请求和响应暂存在本地磁盘中。
- 当与暂时存放的请求相同的新请求到达时，万维网高速缓存就把暂存的响应发送出去，而不需要按 URL 的地址再去因特网访问该资源。

7.7.1 超文本传送协议HTTP



- 你可以在自己电脑中的网络设置中配置代理服务器。
- 如果配置了代理服务器，那么所有HTTP请求都会发给代理服务器。
- Hit: cache将所请求内容返回
- Miss: 从原服务器获取



7.7.1 超文本传送协议HTTP

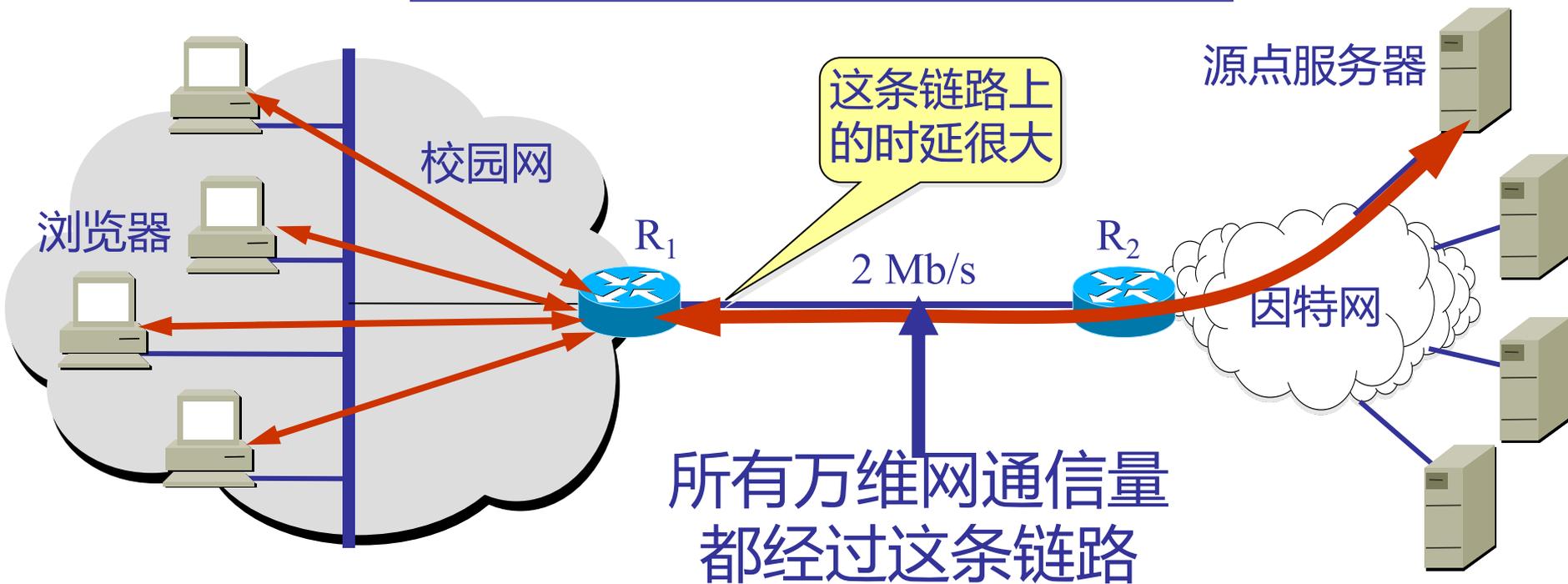
- Content distribution networks (CDN) 是一个web caches的全球网络。需注意，CDN需要DNS来重定向请求。
 - 运营商ISP
 - 或者内容提供商，如Netflix，Google。
- 可以显著降低网络流量
- 极大提升响应速度：因为CDN节点一般距离用户更近
- 减少带宽需求
- 降低维护服务器的成本。



使用高速缓存可减少
访问因特网服务器的时延

7.7.1 超文本传送协议HTTP

没有使用高速缓存的情况





HTTP的报文结构

7.7.1 超文本传送协议HTTP

HTTP 有两类报文：

- 请求报文——从客户向服务器发送请求报文。
- 响应报文——从服务器到客户的回答。



一些著名的搜索引擎

7.7.1 超文本传送协议HTTP

当前最有名的搜索引擎是：

Google (<https://www.google.com>)

Bing (<https://www.bing.com>)

Baidu (<https://www.baidu.com>)

Thank You

Have A Nice Day

南京邮电大学通信与信息工程学院

“**计算机通信与网络**” 国家精品课程组
