

<<Visual C#网络编程>>

图书基本信息

书名：<<Visual C#网络编程>>

13位ISBN编号：9787121146466

10位ISBN编号：7121146460

出版时间：2011-10

出版时间：电子工业出版社

作者：郑阿奇 编

页数：416

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<Visual C#网络编程>>

### 内容概要

本书以visual studio 2008为平台，介绍visual c#网络编程。

全书共12章，分三个部分。

第一部分（第1~2章）为“c#语言网络编程基础”，主要介绍开发环境、网络编程常识和c#与网络编程有关的特性。

第二部分（第3~6章）为“网络传输编程”，也就是人们通常所说的socket编程，包括tcp、udp等基于网络传输层接口的编程。

由于各种网络应用的协议都是建立在传输层之上的，因此这部分内容是网络编程的基础。

第三部分（第7~12章）为“internet应用编程”，是本书的主体部分，介绍各种典型的网络应用的编程开发方法。

本书可作为计算机及相关专业本科、高职高专网络编程的教材或参考书。

同时，为广大电脑爱好者、网络爱好者、编程爱好者、软件发烧友及电脑网络diy玩家成为网络高手打好基础。

## 书籍目录

## 第一部分 c#语言网络编程基础

## 第1章 visualstudio2008开发环境和网络编程常识

- 1.1 visualstudio2008(c#)开发环境
- 1.2 网络编程常识
  - 1.2.1 什么是网络编程
  - 1.2.2 网络程序工作机制
  - 1.2.3 网络应用编程界面

## 第2章 c#网络编程常用特性

- 2.1 c#委托机制
  - 2.1.1 初识委托
  - 2.1.2 为什么要使用委托
  - 2.1.3 委托的意义
- 2.2 c#多线程
  - 2.2.1 线程概述
  - 2.2.2 线程的创建
  - 2.2.3 线程支持的实时gui
  - 2.2.4 线程支持的并发任务
- 2.3 c#方法回调
  - 2.3.1 回调实现的一般过程
  - 2.3.2 方法回调与委托、线程的综合应用

## 第二部分 网络传输编程

## 第3章 tcp编程

- 3.1 tcp简介
  - 3.1.1 tcp的工作过程
  - 3.1.2 tcp的主要特点
  - 3.1.3 套接字的tcp通信流程
- 3.2 同步与异步
- 3.3 c#中的tcp编程类
  - 3.3.1 tcpListener类
  - 3.3.2 tcpClient类
- 3.4 tcp基础编程的一般步骤
  - 3.4.1 编写服务器端程序的一般步骤
  - 3.4.2 编写客户端程序的一般步骤
- 3.5 同步tcp编程
  - 3.5.1 界面设计
  - 3.5.2 客户端程序
  - 3.5.3 服务器端程序
  - 3.5.4 同步tcp的性质
- 3.6 异步tcp编程
  - 3.6.1 异步程序设计
  - 3.6.2 界面设计
  - 3.6.3 客户端程序
  - 3.6.4 服务器端程序
  - 3.6.5 异步tcp的性质
- 3.7 两种tcp程序的区别

## &lt;&lt;Visual C#网络编程&gt;&gt;

- 3.7.1 同步和异步程序的对比演示
- 3.7.2 同步与异步机制的优劣与适用场合

## 第4章 udp编程

- 4.1 udp简介
  - 4.1.1 udp原理
  - 4.1.2 udp与tcp的区别
  - 4.1.3 udp的优势
- 4.2 c#对udp编程的支持
  - 4.2.1 udpclient类
  - 4.2.2 发送数据
  - 4.2.3 接收数据
- 4.3 udp进程通信
  - 4.3.1 udp进程通信演示程序
  - 4.3.2 udp的重要特性
- 4.4 udp广播和组播
  - 4.4.1 广播和组播的基本概念
  - 4.4.2 加入和退出组播组
  - 4.4.3 udp广(组)播演示程序
  - 4.4.4 udp广播和组播的实验演示

## 第5章 p2p编程

- 5.1 p2p基础知识
  - 5.1.1 p2p的发明
  - 5.1.2 p2p架构概述
  - 5.1.3 p2p基本原理
- 5.2 net对p2p编程的支持
  - 5.2.1 对等名称解析协议 ( pnrp )
  - 5.2.2 peertopeer命名空间
- 5.3 p2p资源发现原理与编程
  - 5.3.1 典型的p2p资源发现过程
  - 5.3.2 p2p编程环境配置
  - 5.3.3 资源发现程序示例
  - 5.3.4 资源发现演示
- 5.4 p2p在现实互联网中的应用方式
  - 5.4.1 迅雷下载
  - 5.4.2 qq查找联系人

## 第6章 传输应用编程：即时通信聊天工具

- 6.1 即时通信系统
  - 6.1.1 生活中的即时通信
  - 6.1.2 系统架构的方式
- 6.2 传输层的技术
  - 6.2.1 同步与异步tcp
  - 6.2.2 匿名udp
  - 6.2.3 实名udp
  - 6.2.4 各种传输技术的适用场合
- 6.3 原理、流程与协议设计
  - 6.3.1 会话流程设计
  - 6.3.2 用户管理

## &lt;&lt;Visual C#网络编程&gt;&gt;

## 6.3.3 协议设计

## 6.4 系统实现

## 6.4.1 界面设计

## 6.4.2 服务线程

## 6.4.3 登录/注销

## 6.4.4 即时聊天

## 6.4.5 试运行

## 6.5 源代码清单

## 6.6 设计思想与创新

## 第三部分 internet应用编程

## 第7章 web浏览器

## 7.1 web 浏览器概述

## 7.1.1 互联网的入口

## 7.1.2 历史与主流浏览器

## 7.1.3 浏览器的结构和功能

## 7.2.net 对浏览器开发的支持

## 7.2.1 webbrowser控件

## 7.2.2 net请求 / 响应模型及其类库

## 7.3 web浏览器开发实战

## 7.3.1 搜集素材和设计图形界面

## 7.3.2 基本功能实现

## 7.3.3 高级功能编程

## 7.3.4 附加功能完善

## 7.3.5 完整源代码

## 7.3.6 程序试运行

## 7.4 web浏览器增强功能展示

## 7.4.1 百度搜索

## 7.4.2 保存web页

## 7.4.3 网页源代码显示和保存

## 7.4.4 网页源html的两种编码方式

## 第8章 ftp文件上传下载

## 8.1 ftp概述

## 8.1.1 文件传输的基本概念

## 8.1.2 ftp服务的工作过程

## 8.1.3 匿名ftp服务

## 8.2 ftp客户端开发相关类

## 8.2.1 ftpwebrequest类

## 8.2.2 ftpwebresponse类

## 8.2.3 networkcredential类

## 8.3 ftp文件上传下载器的制作

## 8.3.1 ftp工具常用功能及工作流程

## 8.3.2 ftp客户端界面设计

## 8.3.3 主程序代码框架

## 8.3.4 各功能模块代码

## 8.3.5 程序试运行

## 8.4 ftp上传下载软件测试

## 第9章 电子邮件应用

## &lt;&lt;Visual C#网络编程&gt;&gt;

- 9.1 邮件应用概述
  - 9.1.1 电子邮件基本原理
  - 9.1.2 电子邮件相关协议
  - 9.1.3 传统邮件系统
  - 9.1.4 基于web的邮件系统
  - 9.1.5 邮件服务商与邮箱产品
  - 9.1.6 申请免费邮箱
- 9.2 邮件客户端的功能与界面
  - 9.2.1 典型的收发信过程
  - 9.2.2 邮件系统的常用功能
  - 9.2.3 邮件客户端界面设计
- 9.3 邮件发送功能编程
  - 9.3.1 smtp协议
  - 9.3.2 邮件发送相关类
  - 9.3.3 邮件发送程序设计
  - 9.3.4 发送功能演示
- 9.4 邮件接收功能编程
  - 9.4.1 pop3协议
  - 9.4.2 登录中的pop3会话
  - 9.4.3 jmail组件
  - 9.4.4 邮件接收功能的实现
- 9.5 e-mail收发器的完整源代码
  - 9.5.1 完整源代码
  - 9.5.2 “关于”对话框
- 第10章 http编程：web服务器
  - 10.1 http原理
    - 10.1.1 万维网的工作过程
    - 10.1.2 超文本传输协议http
  - 10.2 web服务器的开发
    - 10.2.1 界面设计
    - 10.2.2 实现流程
    - 10.2.3 基于http协议的服务器编程
    - 10.2.4 web服务器完整源代码
  - 10.3 自制浏览器访问web服务器的实验
    - 10.3.1 实验准备
    - 10.3.2 访问web服务器虚拟目录下的网页和图片
- 第11章 ftp编程：ftp服务器
  - 11.1 ftp协议原理
    - 11.1.1 文件传输的主要功能
    - 11.1.2 文件传输模式
    - 11.1.3 ftp规范
  - 11.2 ftp服务器的实现
    - 11.2.1 界面设计
    - 11.2.2 ftp编程
    - 11.2.3 ftp服务器完整源代码
  - 11.3 完整的ftp软件套件
    - 11.3.1 ftp上传下载器的改造

11.3.2 自制ftp客户端与服务器的对接实验

第12章 c#.net与互联网应用的未来

12.1 互联网的3个发展阶段

12.2 网络应用与c#.net的进化

12.3 互联网应用的未来

12.3.1 常用网络应用

12.3.2 中国网民的群体划分

12.3.3 互联网应用的未来

## 章节摘录

版权页：插图：看了上述的这个即时通信聊天软件的实现过程，也许细心的读者已经发现，它在客户端、服务器之间通信的不同阶段，使用了各不相同的连接方式和传输层技术：新用户发起登录时使用匿名UDP；传输用户列表使用同步TCP；聊天内容消息通过匿名UDP直接发往实名UDP套接口（P2P方式）；在线用户变动的消息由服务器匿名UDP广播转发……这一系列发送和接收的传输方式，其实都是经过了笔者事先精心设计的。

下面简要地阐述这样设计的理由。

1.匿名UDP与安全性.在匿名UDP方式中，消息发送方与接收方不建立正式的直接连接，这样有利于系统安全。

新用户登录并不是直接向服务器的某个监听端口发起连接，而只是发出一个通告消息，服务器允许后才在应答中将自己的监听端口告知用户，这样做避免了监听端口在网络上的暴露，能够有效地防止DDOS攻击的发生（在这种攻击中，黑客会控制大批“肉鸡”恶意向服务器端口发出连接请求，最终导致服务器资源耗尽而死机）。

2.同步TCP与可靠性在客户端通过验证后（服务器返回“Accept”时），才获得向服务器发起TCP请求的权限，服务器通过TCP连接向客户端传输当前在线用户的列表信息，确保了所传输用户列表数据的准确无误，保证了新加入用户从一开始就与服务器保持同步，提高了可靠性。

不仅如此，假若该系统还要扩充其他功能，在需要传输大数据量并且对数据可靠性要求较高的时候，也同样要由这种同步TCP承担工作。

3.实名UDP与P2P对等识别在本例中，为每一个加入系统的用户终端都启动一个实名UDP线程，这种套接字与特定的用户端点（地址）绑定，使得消息发送方能够很容易地定位到每个用户端，从而实现全系统的P2P通信。

用户之间的聊天内容信息无须经由服务器集中转发，既减轻了中央服务器的负担，同时也有效地保护了用户间的谈话隐私。

虽然经UDP收发的消息数据报可能延迟或遗失，但这对于实时性要求很高的聊天类应用来说并不是问题，事实上，大家生活中使用的QQ就是采用这种基于UDP的消息机制。

4.广播通知由匿名UDP转发的高效性在本例中，服务器虽然要担任全系统的组织者和协调者的角色，但它却并不与每个客户端时刻保持TCP连接，整个过程中，在服务器上只维持两个常驻线程（一个实名UDP接收线程和一个TCP监听线程）。

当在线用户发生变动（登录或注销）时，服务器临时通过创建匿名的UDP进行广播通知。

而只有在客户端需要获取最新的用户列表时，才与服务器建立TCP连接。

即使是这个连接也是临时的，在传完用户表后就立即与客户端断开了。



## <<Visual C#网络编程>>

### 编辑推荐

《Visual C#网络编程》：循序渐进、深入浅出地介绍不同层次网络编程。  
精心设计实例，通过观察运行结果，深刻体验和领悟网络编程的原理和工作机制。  
通过创新开发即时通信软件，综合运用各种传输层编程技术；并与当下流行的QQ产品作比较。  
在《Visual C#网络编程》的实例程序之间进行互操作，如客户端和服务器对接、程序之间整合集成成为套件，使书中的实例不再是孤立实体。  
通过所讲实例延伸，介绍网络应用的流行趋势，提出很多新观点。

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>