

<<Windows Sockets网络编程>>

图书基本信息

书名：<<Windows Sockets网络编程>>

13位ISBN编号：9787111392743

10位ISBN编号：7111392744

出版时间：2012-8

出版时间：机械工业出版社

作者：Bob Quinn,Dave Shute

译者：徐磊,腾婧,张莹,何慧

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>

## <<Windows Sockets网络编程>>

### 内容概要

《华章专业开发者丛书：Windows Sockets网络编程》是Windows Sockets网络编程领域公认的经典著作，由Windows Sockets 2.0规范解释小组负责人亲自执笔，权威性毋庸置疑。

它结合大量示例，对Windows Sockets规范进行了深刻地解读，系统讲解了Windows Sockets网络编程及其相关的概念、原理、主要命令、操作模式，以及开发技巧和可能的陷阱，从程序员的角度给出了大量的建议和最佳实践，是学习Windows Sockets网络编程不可多得的参考书。

全书分为三部分：第一部分（第1～6章），提供了翔实的背景知识和框架方面的概念，借助于此框架，读者可理解WinSock的具体细节，包括Windows Sockets概述、OSI网络参考模型、TCP/IP协议簇中的协议和可用的服务、WinSock网络应用程序的框架及其工作机制、WinSock的三种操作模式、socket通信机制等；第二部分（第7～12章），以FTP客户端实例为基础介绍了函数实例库，还介绍了客户端程序、服务器程序和DLL中间构件及它们的相应函数，并涵盖socket命令和选项及移植BSD Sockets相关事项等；第三部分（第13～17章），介绍了应用程序调试技术和工具，针对应用编程中的陷阱的建议和措施，WinSock API的多种操作系统平台，WinSock规范的可选功能和WinSock规范2.0中的所有新功能。

## <<Windows Sockets网络编程>>

### 作者简介

作者：（美国）奎因（Bob Quinn）（美国）舒特（Dave Shute）译者：徐磊 腾婧 张莹 何慧 奎因（Bob Quinn），WinSock 2.0规范解释小组的负责人，Windows Sockets技术的开拓者之一，曾为WinSock规范1.1版本做出重要贡献，从事TCP/IP网络开发多年，经验颇丰。

舒特（Dave Shute），Windows Sockets技术的先驱者和布道者，WinSock问世之初便与Bob一起服务于著名的FTP软件公司，对WinSock技术和TCP/IP等网络技术都有非常透彻的研究，实践经验极为丰富。

## 书籍目录

译者序 序 前言 第1章 Windows Sockets概述 1.1 什么是Windows Sockets 1.2 Windows Sockets的发展历史 1.3 Windows Sockets的优势 1.3.1 Windows Sockets是一个开放的标准 1.3.2 Windows Sockets提供源代码可移植性 1.3.3 Windows Sockets支持动态链接 1.3.4 Windows Sockets的优点 1.4 Windows Sockets的前景 1.5 结论 第2章 Windows Sockets的概念 2.1 OSI网络模型 2.2 WinSock网络模型 2.2.1 信息与数据 2.2.2 应用协议 2.3 WinSock中的OSI层次 2.3.1 应用层 2.3.2 表示层 2.3.3 会话层 2.3.4 传输层 2.3.5 网络层 2.3.6 数据链路层 2.3.7 物理层 2.4 模块化的层次框 2.5 服务和协议 2.6 协议和API 第3章 TCP/IP协议服务 3.1 什么是TCP / IP 3.2 TCP/IP的发展历史 3.3 传输服务 3.3.1 无连接的服务：UDP 3.3.2 面向连接的服务：TCP 3.3.3 传输协议的选择：UDP与TCP的对比 3.4 网络服务 3.4.1 IP服务 3.4.2 ICMP服务 3.5 支持协议和服务 3.5.1 域名服务 3.5.2 地址解析协议 3.5.3 其他支持协议 3.6 TCP/IP的发展前景 第4章 网络应用程序工作机制 4.1 客户端—服务器模型 4.2 网络程序概览 4.3 socket的打开 4.4 socket的命名 4.4.1 sockaddr结构 4.4.2 sockaddr\_in结构 4.4.3 端口号 4.4.4 本地IP地址 4.4.5 什么是socket名称 4.4.6 客户端socket名称是可选的 4.5 与另一个socket建立关联 4.5.1 服务器如何准备建立关联 4.5.2 客户端如何发起一个关联 4.5.3 服务器如何完成一个关联 4.6 socket之间的发送与接收 4.6.1 在“已连接的”socket上发送数据 4.6.2 在“无连接的”socket上发送数据 4.6.3 接收数据 4.6.4 socket解复用器中的关联 4.7 socket的关闭 4.7.1 closesocket ( ) 4.7.2 shutdown ( ) 4.8 客户端和服务端概览 第5章 操作模式 5.1 什么是操作模式 5.1.1 不挂机，等待：阻塞 5.1.2 挂机后再拨：非阻塞 5.1.3 请求对方回拨：异步 5.2 阻塞模式 5.2.1 阻塞socket 5.2.2 阻塞函数 5.2.3 伪阻塞的问题 5.2.4 阻塞钩子函数 5.2.5 阻塞情境 5.2.6 撤销阻塞操作 5.2.7 阻塞操作中的超时 5.2.8 无最少接收限制值 5.2.9 代码示例 5.3 非阻塞模式 5.3.1 怎样使socket成为非阻塞的 5.3.2 成功与失败不是绝对的 5.3.3 探询而非阻塞 5.3.4 显式地避让 5.3.5 代码示例 5.4 异步模式 5.4.1 认识异步函数 5.4.2 撤销异步操作 5.4.3 代码示例 5.4.4 AU\_Time应用 5.5 性能比较 5.6 平台比较 5.7 真正的阻塞 5.7.1 线程化的应用程序 5.7.2 其他适用性说明 第6章 socket状态 6.1 什么是socket状态 6.1.1 数据报socket的状态 6.1.2 流socket的状态 6.2 socket状态的检测方法 6.2.1 根据函数调用的成功或失败检测 6.2.2 同步检测 6.2.3 异步检测 6.3 WSAAsyncSelect ( ) 6.3.1 WSAAsyncSelect ( ) 的工作原理 6.3.2 什么是WSAAsyncSelect ( ) 事件 6.3.3 避免机会丢失的窗口 6.3.4 什么是重使能函数 6.3.5 什么是通知消息 6.3.6 为函数失败做好准备 6.3.7 在响应时设置循环 6.3.8 撤销异步通知 6.3.9 事件描述 6.3.10 AS\_Echo应用程序 6.4 select ( ) 6.4.1 怎样使用select ( ) 6.4.2 select ( ) 的宏 6.4.3 select ( ) 的输入参数 6.4.4 不要把select ( ) 作为定时器 6.4.5 注意0超时 6.4.6 代码示例 6.5 查看数据 6.5.1 ioctlsocket ( ) FIONREAD命令 6.5.2 带有MSG\_PEEK标志的recv ( ) 6.6 带外数据 6.6.1 TCP的局限 6.6.2 ioctlsocket ( ) 函数和SIOCATMARK命令 6.6.3 代码示例 6.7 监听状态 第7章 应用程序和函数实例库 7.1 FTP客户端实例 7.1.1 文件传输协议概述 7.1.2 FTP有限状态机 7.1.3 AC\_FTP应用程序 7.1.4 操作中的AC\_FTP 7.2 函数实例库 7.2.1 CenterWnd ( ) 7.2.2 CloseConn ( ) 7.2.3 CreateLclFile ( ) 7.2.4 Dlg\_About ( ) 7.2.5 GetAddr ( ) 7.2.6 GetBuf ( ) 7.2.7 GetHostID ( ) 7.2.8 GetLclDir ( ) 7.2.9 GetPort ( ) 7.2.10 WSAErrStr ( ) 7.2.11 WSAperror ( ) 7.3 WinSockx.h 7.3.1 Globalsec 7.3.2 WSA\_Xtra.h 第8章 主机名称与主机地址 8.1 主机名称与主机地址解析 8.1.1 hostent结构 8.1.2 主机名称解析 8.1.3 地址解析 8.2 主机表、域名系统和网络信息服务 8.2.1 主机表 8.2.2 域名系统 8.2.3 网络信息服务 8.3 本地主机信息 8.4 网络地址与格式化 8.4.1 in\_addr结构 8.4.2 inet\_addr ( ) 8.4.3 inet\_ntoa ( ) 8.5 协议族与地址族 第9章 socket信息与控制 9.1 socket控制 9.1.1 FIONBIO 9.1.2 FIONREAD 9.1.3 SIOCATMARK 9.1.4 其他控制命令 9.2 socket选项 9.2.1 getsockopt ( ) 与setsockopt ( ) 9.2.2 SO\_ACCEPTCONN 9.2.3 SO\_BROADCAST 9.2.4 SO\_DEBUG 9.2.5 SO\_DONTLINGER 9.2.6 SO\_DONTROUTE 9.2.7 SO\_ERROR 9.2.8 SO\_KEEPALIVE 9.2.9 SO\_LINGER 9.2.10 SO\_OOINLINE 9.2.11 SO\_RCVBUF与SO\_SNDBUF 9.2.12 SO\_REUSEADDR 9.2.13 SO\_TYPE 9.2.14 TCP\_NODELAY 9.2.15 不支持的选项 9.2.16 未来会使用的选项 9.3 阻塞钩子 9.3.1 WSASetBlockingHook ( ) 9.3.2 WSAUnhookBlockingHook ( ) 9.3.3 阻塞钩子函数 9.3.4 阻塞状态 9.4 socket名称 9.4.1 getsockname ( ) 9.4.2 getpeername ( ) 第10章 支持例程 10.1 启动与清除 10.1.1 WSAStartup ( ) 10.1.2 WSACleanup ( ) 10.2 字节顺序 10.3 服务名称与端口号 10.3.1 servent结构 10.3.2 服务解析 10.3.3 端口解析 10.3.4 服务数据库 10.4 协议名称与协议号 10.4.1 protoent结构 10.4.2 协议名称解析 10.4.3 协议号解析 10.4.4 协议数据库

10.5 报错 10.5.1 WSAGetLastError ( ) 10.5.2 WSASetLastError ( ) 10.5.3 错误文本显示 第11章 WinSock上的DLL 11.1 创建一个新的API 11.1.1 仿真的API 11.1.2 应用协议API 11.1.3 封装的WinSock API 11.2 DLL的问题 11.2.1 重入 11.2.2 任务管理 11.2.3 与WinSock DLL的链接 11.3 DLL实例 11.3.1 WSASimpl应用程序 11.3.2 WSASimpl DLL 第12章 移植BSD Sockets 12.1 Windows Sockets与Berkeley Sockets的差异 12.1.1 socket不是文件句柄 12.1.2 signal ( ) 不可用 12.1.3 域名不可用 12.1.4 地址信息和操作不可用 12.1.5 syslog ( ) 不可用 12.1.6 变量errno不可用 12.1.7 perror ( ) 和strerror ( ) 不可用 12.1.8 数据库文件操作不可用 12.1.9 bcmp ( )、bcopy ( ) 和bzero ( ) 不可用 12.1.10 本地IPC不可用 12.1.11 OOB API不同 12.1.12 setsockopt ( ) SO\_DEBUG是可选的 12.2 16位Windows的局限性 12.2.1 16位Windows不清理现场 12.2.2 16位Windows是非抢占式的 12.2.3 16位Windows文件系统不安全 12.2.4 16位Windows使用分段寻址 12.2.5 fork ( ) 不可用 12.3 移植Berkeley Sockets的其他事项 12.3.1 包含文件 12.3.2 socket域 12.3.3 socket类型 12.3.4 socket句柄 12.3.5 错误值 12.4 函数列表 ..... 第13章 调试 第14章 该做和不该做的 第15章 系统平台 第16章 可选特性 第17章 WinSock 2 附录A TCP/IP协议首部 附录B 快速参考 附录C 错误信息参考 附录D 用户必备

## 章节摘录

版权页：插图：显然，2.0版本在1.1版本的基础上更进了一大步，通过提高其可用性和扩展其范围进一步巩固了规范的现有地位。

WinSock 2增加的众多新功能使得API更加灵活和强大，当然，也使其复杂化了。

幸运的是，它对于1.1版本的核心API的支持保持不变。

在本书中，我们重点关注WinSock 1.1版本的API，在第17章中将详细讲述WinSock 2中的一些有用的增补。

其中，有些新的特性在WinSock 1.1的某些实现版本中是以选项的形式出现的。

基于本书所提供的对WinSock 1.1的概念和方法的透彻讲解，你将能够更好地掌握和利用WinSock 2实现中的扩展新功能。

1.5 结论 Windows Sockets 是网络应用开发舞台上不可或缺的角色，它并不完美，在不同的实现版本中还存在一些问题，但是，在应用开发中避免这些差异带来的问题是可能的。

关键是采用防御性程序设计技术，规避规范中的模糊地带。

在本书中，我们将向读者展示，为了创建能在任何WinSock实现上成功运行的Windows Sockets应用，什么可做，什么不可做，分析在Windows环境（16位和（或）32位）下开发一个Win-Sock应用的需求。

Windows Sockets最初的承诺是：提供创建通用应用的能力。

这一目标已经接近实现，在进一步扩展其功能以实现新的承诺的同时，Windows Sockets将保留现有的API。

WinSock API承诺将满足新的网络媒体和其他协议的需求，更不必说那些迄今为止尚未出现的一些全新网络应用。

计算机正朝着移动的方向发展，WinSock也如是。

由于带宽在增加，网络通道也越来越宽阔，WinSock将使带宽的利用更加有效。

Internet迫切地需要能提供导航、查找、易用的搜索功能的应用软件，而WinSock正是帮助这类应用普及和提高的完美平台。

我们目前所触及的还只是网络多媒体分发的一些浅层应用。

伴随着更多的发现，我们会看到WinSock大有用武之地。

WinSock是一个让技术贴近用户的工具，它就像是引导者和推动者，已经为这个任重道远的使命做好了准备。

### 编辑推荐

《Windows Sockets网络编程》是Windows Sockets网络编程领域公认的经典著作，由Windows Sockets 2.0规范解释小组负责人亲自执笔，权威性毋庸置疑。

它结合大量示例，对Windows Sockets规范进行了深刻地解读，系统讲解了Windows Sockets网络编程及其相关的概念、原理、主要命令、操作模式，以及开发技巧和可能的陷阱，从程序员的角度给出了大量的建议和最佳实践，是学习Windows Sockets网络编程不可多得的参考书。

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>