



Networking Services Developer's Reference Library
微软网络编程与开发丛书

Microsoft® Press

微软独家授权 揭示微软网络技术内幕
深入网络编程与开发
IT 企业和网络编程人员必备

The essential reference set for developing with
Microsoft® Windows® networking technologies

David Iseminger
Series Editor
www.iseminger.com



[美] Microsoft 公司 著

**Windows® Sockets
and QOS**

Winsock 和 QOS

(影印版)

北京大学出版社
<http://cbs.pku.edu.cn>

BASED ON
msdn™ library

微软网络编程与开发影印丛书

Windows Sockets and QOS
Winsock 和 QOS

Microsoft 公司 著

江苏工业学院图书馆
藏书章

北 京 大 学 出 版 社

内 容 简 介

本书是 NETWORKING SERVICES DEVELOPER'S REFERENCE LIBRARY(微软网络编程与开发影印从书)中的一本。Windows Sockets 2 (Winsock)和 Quality of Service(QOS)是 Windows 2000 所支持的两个联网标准,而本书讲述的就是关于这二者的重要标题性信息。Winsock 提供了访问多个传输协议的简单方法。使程序员能够创建高级 Web 以及和网络相关的应用程序,实现与协议元关的数据传送。而业界广泛采用的 QOS 使开发人员能够创建或改进任务必需的应用程序(无论网络状况怎样,这些应用程序在运行时总感觉状况良好)。

Copyright (2000) by Microsoft Corporation

Original English language edition Copyright © 2000 (year of first publication by author)

By Microsoft Corporation (author)

All rights published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A.

北京市版权局著作权合同登记号:图字 01-2000-0287

图书在版编目(CIP)数据

Winsock 和 QOS.第1卷:英文/美国微软公司(Microsoft)著.一影印本.一北京:北京大学出版社,2000.4

ISBN 7-301-02040-6

I. W... II. 美... III. 操作系统, Windos 2000-程序设计-英文 IV. TP316.7

中国版本图书馆 CIP 数据核字(2000)第 07607 号

书 名: Winsock 和 QOS

责任著作者: Microsoft 公司 著

标准书号: ISBN 7-301-02040-6/TP · 0169

出 版 者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

网 址: <http://cbs.pku.edu.cn>

电 话: 出版部 62752015 发行部 62765126 62754140 编辑室 62765127

电 子 信 箱: wdzh@mail.263.net.cn

印 刷 者: 中国科学院印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 57 印张 1359 千字

2000 年 4 月第 1 版 2000 年 4 月第 1 次印刷

定 价: 148.00 元

前 言

在我们的生活和工作越来越离不开网络的今天，网络的速度、性能以及如何充分利用现有网络的容量都是至关重要的问题。Windows 2000 的几乎每一个新特性和新组件都必须直接或间接地使用网络。为了帮助用户最大限度地利用 Windows 2000 的网络特性，我们组织影印了这套 5 卷本的丛书，《微软网络编程与开发影印丛书》。本丛书可为编程人员提供完整的、考虑周全的参考资料。本丛书将最广泛使用的、最重要的和最及时的 Windows 网络技术信息搜集整理成册，便于开发人员开发更好的、与网络紧密集成的、在 Windows 平台上运行的应用程序。

本丛书的每一卷都侧重于 Windows 网络技术的不同领域，并结合 Windows 程序员的实际需要，精心编排了完整的索引，便于阅读和查找信息。本丛书的精心设计还有助于程序员方便地找到 Microsoft 的其他参考资料，包括印刷资料和电子版资料。每一本书还包括一个对整套丛书的概述，一个关于编程要素的附录，以及关于所引用的 Microsoft 技术的索引。但本丛书的内容远不止这些。本丛书是关于 Windows 网络特性、协议和服务的最佳资源。

本丛书的主编 David Iseminger 是 Microsoft 的独立顾问，作为网络和路由器性能分析师、通信专家和性能工具开发者，自 Windows NT 3.5 以来，他就一直从事 Windows NT 和 Windows 2000 的研究与开发工作。最近，他作为程序员，正在与开发人员文档组 (Developer Documentation Group) 一道，创建和维护 MSDN 建立的和新兴的网络技术，包括服务质量 (Quality of Service)。作为计算机图书作家，David 的著作颇丰，包括他最新创作、Microsoft 出版的《Active Directory Services for Microsoft Windows 2000 Technical Reference》。David 毕业于华盛顿大学，现居住在华盛顿州的 Puget Sound 城。欲详细了解 David 的工作以及本丛书的其他作品，请访问站点：www.iseminger.com。

本丛书由以下 5 本书组成：

《Windows Sockets and QOS》(《Winsock 和 QOS》)

《Network Protocols and Interface》(《网络协议和接口》)

《RPC and Windows Networking》(《RPC 和 Windows 网络》)

《Remote Access Services》(《远程访问服务》)

《Routing》(《路由》)

Contents

Acknowledgements	iii
-------------------------------	------------

Part 1

Chapter 1: Getting Around in the Networking Services Library	1
How the Networking Services Library Is Structured	2
How the Networking Services Library Is Designed	3
Chapter 2: What's In This Volume?	5
Winsock	6
Quality of Service	7
Chapter 3: Using Microsoft Reference Resources	9
The Microsoft Developer Network	10
Comparing MSDN with MSDN Online	11
MSDN Subscriptions	13
MSDN Library Subscription	13
MSDN Professional Subscription	14
MSDN Universal Subscription	14
Purchasing an MSDN Subscription	14
Using MSDN	15
Navigating MSDN	16
Quick Tips	18
Using MSDN Online	20
Navigating MSDN Online	22
MSDN Online Features	23
MSDN Online Registered Users	29
The Windows Programming Reference Series	30
Chapter 4: Finding the Developer Resources You Need	31
Developer Support	31
Online Resources	33
Internet Standards	34
Learning Products	35
Conferences	37
Other Resources	37

Chapter 5: Writing Great IrDA Applications (with Winsock)	39
What Is an Ad-Hoc Networking-Enabled Application?	39
What Is IrDA?	40
What Is IrDA-C (Previously Known as IrBus)?	40
What Is Unique about IrDA?	41
IrDA Core Protocols and Services	41
Serial IrDA (SIR) Physical Layer (115 Kb/s)	41
Fast IrDA (FIR) Physical Layer (4 Mb/s)	41
IrLAP Data Link Layer	42
IrLMP and TinyTP	42
IrCOMM	43
IrCOMM Modes	44
No IrCOMM Virtual Serial Ports on Windows 2000	45
Windows 2000 Support for IrCOMM Through Winsock	45
IrDA and the Windows Sockets API	46
Talking to Non-Windows Devices	46
Application Addressing	46
Data Transfer and Connection Close	47
IrDA and Winsock Reference	49
WSAStartup	49
af_irda.h	49
socket	49
SOCKADDR_IRDA Structure	50
bind	50
listen	51
accept	51
send and recv	52
closesocket	52
getsockopt(, IRLMP_ENUMDEVICES,,) and connect()	52
IAS	54
IrCOMM Client	57
Windows 2000 IrDA Architecture	60
IrDA Hardware Drivers	60
Windows 2000 Multiple-Adapter Support	61

Part 2

Chapter 6: Winsock 2 API Overview	63
Welcome to Windows Sockets 2	63
Using the Windows Sockets 2 API Document	63
Overview of Windows Sockets 2	63
Windows Sockets 2 Features	64
Conventions for New Functions	65
Microsoft Extensions and the Windows Sockets 2 API	65
Socket Handles for Windows Sockets 2	65
New Concepts, Additions, and Changes for Windows Sockets 2	66
Windows Sockets 2 Architecture	66
Simultaneous Access to Multiple Transport Protocols	66
Backward Compatibility for Windows Sockets 1.1 Applications	67
Making Transport Protocols Available to Windows Sockets	69
Layered Protocols and Protocol Chains	69
Using Multiple Protocols	70
Multiple Provider Restrictions on Select	71
Function Extension Mechanism	72
Debug and Trace Facilities	72
Name Resolution	73
Overlapped I/O and Event Objects	73
Event Objects	74
Receiving Completion Indications	75
Asynchronous Notification Using Event Objects	76
Flow Specification Quality of Service	77
QOS Templates	77
Default Values	77
Socket Groups	77
Shared Sockets	77
Enhanced Functionality During Connection Setup and Teardown	78
Extended Byte-Order Conversion Routines	79
Support for Scatter/Gather I/O in the API	79
Protocol-Independent Multicast and Multipoint	79
Summary of New Socket Options	80
Summary of New Socket ioctl Opcodes	81
Summary of New Functions	82
Windows Sockets Programming Considerations	85
Deviation from Berkeley Sockets	85

Socket Data Type	85
Select and FD_*	86
Error Codes—errno, h_errno and WSAGetLastError	86
Pointers	87
Renamed Functions	87
Maximum Number of Sockets Supported	87
Include Files	88
Return Values on Function Failure	88
Service Provided Raw Sockets	88
Byte Ordering	89
Windows Sockets Compatibility Issues	89
Default State for a Socket's Overlapped Attribute	90
Windows Sockets 1.1 Blocking Routines and EINPROGRESS	90
Graceful Shutdown, Linger Options, and Socket Closure	92
Protocol-Independent Out-of-Band Data	93
Summary of Windows Sockets 2 Functions	96
Socket Functions	96
Microsoft Windows-Specific Extension Functions	97
Registration and Name Resolution	99
Protocol-Independent Name Resolution	100
Name Resolution Model	100
Summary of Name Resolution Functions	103
Name Resolution Data Structures	105
Compatible Name Resolution for TCP/IP in the Windows Sockets 1.1 API	108
Basic Approach for GetXbyY in the API	109
getprotobyname and getprotobyname Functions in the API	109
getservbyname and getservbyport Functions in the API	109
gethostbyname Function in the API	110
gethostbyaddr Function in the API	110
gethostname Function in the API	111
Multipoint and Multicast Semantics	111
Multipoint Taxonomy	111
Windows Sockets 2 Interface Elements for Multipoint and Multicast	112
Attributes in WSAPROTOCOL_INFO Structure	113
Flag Bits for WSASocket	113
SIO_MULTIPPOINT_LOOPBACK Command Code for WSAIoctl	114
SIO_MULTICAST_SCOPE Command Code for WSAIoctl	114
Semantics for Joining Multipoint Leaves	114
Using WSAJoinLeaf	115

Semantic Differences Between Multipoint Sockets and Regular Sockets	116
How Existing Multipoint Protocols Support These Extensions	117
IP Multicast	117
ATM Point to Multipoint	118
Additional Windows Socket Information	119
Windows Sockets 2 API Header File—Winsock2.h	119
Socket Options Specific to Microsoft Service Providers	119
Socket Option for Windows NT 4.0 Only	119
Socket Option for Windows NT 4.0 and Windows 95	120
Additional Documentation	121
Chapter 7: Error Codes in the Winsock API	123
Error Codes	123
Chapter 8: Winsock 2 Functions	133
Windows Sockets 2 Functions	133
Chapter 9: Winsock 2 Structures and Enumerations	377
Windows Sockets Structures in the API	377
Windows Sockets Enumeration in the API	413
Chapter 10: Winsock 2 SPI Overview	415
Welcome to Windows Sockets 2 SPI	415
Using the SPI Document	415
Overview of the Windows Sockets 2 SPI	415
Windows Sockets 2 SPI Features	416
Microsoft Extensions and the Windows Sockets 2 SPI	417
Socket Handles for the Windows Sockets 2 SPI	417
Windows Sockets 2 Architectural Overview	418
Windows Sockets 2 as a WOSA Component	418
Windows Sockets 2 DLLs	419
Function Interface Model	419
Naming Conventions	420
Windows Sockets 2 Service Providers	420
Transport Service Providers	420
Namespace Service Providers	422
Windows Sockets 2 Identifiers	424
Data Transport Providers	424
Transport Division of Responsibilities Between DLL and Service Providers	424
Transport Mapping Between API and SPI Functions	426

Function Extension Mechanism in the SPI	427
Transport Configuration and Installation	428
Name Resolution Providers	429
Name Resolution Model for the SPI	429
Name Resolution Division of Responsibilities Between DLL and Service Providers	432
Name Resolution Mapping Between API and SPI Functions	433
Name Resolution Configuration and Installation	433
Windows Sockets 2 Transport Provider Requirements	434
Service Provider Activation	434
Initialization	434
Cleanup	436
Error Reporting and Parameter Validation	436
Byte Ordering Assumptions	436
Socket Creation and Descriptor Management	437
Descriptor Allocation	437
Socket Attribute Flags and Modes	438
Closing Sockets	438
Blocking Operations	438
Pseudo vs. True Blocking	439
Blocking Hook	439
Canceling Blocking Operations	440
Event Objects in the Windows Sockets 2 SPI	440
Creating Event Objects	440
Using Event Objects	441
Destroying Event Objects	441
Notification of Network Events	441
Selects	442
Windows Messages	442
Event Object Signaling	442
Socket Groups in the Windows Sockets 2 SPI	442
Socket Group Operations	442
Required Socket Grouping Behavior	442
Recommended Socket Grouping Behavior	442
Quality of Service in the Windows Sockets 2 SPI	443
Socket Connections on Connection-Oriented Protocols	443
Binding to a Local Address	443
Protocol Basics: Listen, Connect, Accept	443
Determining Local and Remote Names	444

Enhanced Functionality at Connect Time	444
Connection Shutdown	445
Socket Connections on Connectionless Protocols.....	448
Connecting to a Default Peer	448
Reconnecting and Disconnecting.....	448
Using Sendto While Connected	448
Socket I/O	448
Blocking Input/Output.....	449
Nonblocking Input/Output.....	449
Overlapped Input/Output.....	449
Support for Scatter/Gather Input/Output in the SPI	453
Out-of-Band Data in the SPI	453
Shared Sockets in the SPI	455
Multiple Handles to a Single Socket.....	456
Reference Counting	456
Precedence Guidelines	457
Protocol-Independent Multicast and Multipoint in the SPI	457
Multipoint Taxonomy and Glossary.....	458
Multipoint Attributes in the WSAPROTOCOL_INFOW Structure.....	459
Multipoint Socket Attributes.....	459
SIO_MULTIPOINT_LOOPBACK ioctl.....	460
SIO_MULTICAST_SCOPE ioctl	460
SPI Semantics for Joining Multipoint Leaves	460
Using WSPJoinLeaf	461
Semantic Differences Between Multipoint Sockets and Regular Sockets in the SPI	462
Socket Options and IOCTLs.....	463
Summary of Socket ioctl Opcodes.....	465
Summary of SPI Functions.....	466
Generic Data Transport Functions	466
Upcalls Exposed by Windows Sockets 2 DLL	468
Installation and Configuration Functions	471
Name Resolution Service Provider Requirements.....	471
Summary of Namespace Provider Functions.....	471
Namespace Provider Configuration and Installation.....	472
Namespace Provider Initialization and Cleanup	472
Service Installation in the Windows Sockets 2 SPI.....	472
Service Query.....	473
Helper Functions in the SPI	473

Name Resolution Data Structures in the SPI.....	474
Compatible Name Resolution for TCP/IP in the Windows Sockets 1.1 SPI.....	477
Basic Approach for getXbyY in the SPI	478
getprotobyname and getprotobyname Functions in the SPI	478
getservbyname and getservbyport Functions in the SPI	478
gethostbyname Function in the SPI.....	479
gethostbyaddr Function in the SPI.....	479
gethostname Function in the SPI.....	480
Sample Code for a Service Provider	480
Additional Windows Sockets 2 SPI Concerns	495
Service Provider Ordering.....	495
Windows Sockets SPI Header File - Ws2spi.h	496
Chapter 11: Winsock 2 SPI Reference	497
Winsock 2 SPI Reference	497
Chapter 12: Winsock 2 Protocol-Specific Annex	657
Using the Annex.....	657
Overview of Windows Sockets 2.....	657
Microsoft Extensions and Windows Sockets 2	658
Socket Handles for Windows Sockets 2	658
TCP/IP	658
TCP/IP Introduction.....	658
TCP/IP Overview	659
TCP/IP Data Structures	659
TCP/IP Controls	660
UNIX ioctl's	660
TCP/IP Socket Options	660
TCP/IP Function Details.....	663
Multicast.....	663
TCP/IP Raw Sockets	663
IPv6 Support	664
Text Representation of IPv6 Addresses	665
TCP/IP Header File.....	666
IPX/SPX	666
IPX/SPX Introduction	666
IPX/SPX Overview	666
AF_IPX Address Family.....	667
IPX Family of Protocol Identifiers.....	667
Broadcast to Local Network.....	668

All Routes Broadcast.....	668
Directed Broadcast.....	668
About Media Packet Size	669
How Packet Size Affects Protocols	669
IPX/SPX Data Structures	670
IPX/SPX Controls	674
NSPROTO_IPX Socket Options	675
DECnet	676
DECnet Overview	676
DNPROTO_NSP Protocol Family	677
AF_DECnet Address Families	677
SOCK_SEQPACKET Socket Type	678
DECnet Data Structures	678
Manifest Constants (Winsock2.h)	678
Manifest Constants (Ws2dnet.h)	678
Data Structures (Ws2dnet.h)	678
DECnet Function Details	680
Connections Using Accept/WSAAccept/WSPAccept	680
Structure Information for Bind/WSPBind	682
Connections Using Connect/WSAConnect/WSPConnect	682
Addressing with GetPeerName/WSPGetPeerName	684
Receiving Local Name with getsockname/WSPGetSockName	684
Using Getsockopt/WSPGetSockOpt	685
Using Socket/WSASocket/WSPSocket	686
DECnet Out-of-Band Data	686
DECnet-Specific Extended Functions Identifiers	686
dnet_addr	687
dnet_eof	687
dnet_getacc	688
dnet_getalias	688
dnet_htoa	688
dnet_ntoa	689
getnodeadd	689
getnodebyaddr	689
getnodebyname	690
getnodename	690
DECnet Header File	691
Open Systems Interconnection (OSI)	691
OSI Introduction	691

International Organization for Standardization (IOS).....	691
OSI Expedited Data	692
ISO Qualified Data	692
ISO Reset	692
OSI Quality of Service.....	692
Option Profiles	692
Address Format	693
OSI Data Structures	693
OSI Controls.....	693
ioctl.....	694
Socket Options.....	694
OSI Function Specifics.....	695
Quality of Service	695
OSI Header File	695
ATM-Specific Extensions	695
ATM Introduction.....	695
ATM Overview.....	696
ATM Data Structures	696
Using the ATM_ADDRESS Structure	698
ATM_BLLI Structure and Associated Manifest Constants.....	699
ATM_BHLI Structure and Associated Manifest Constants	701
ATM Controls	701
ATM Function Specifics	702
ATM-Specific Quality of Service Extension	702
AAL Parameters.....	703
ATM Traffic Descriptor	704
Broadband Bearer Capability.....	705
Broadband High Layer Information.....	706
Broadband Lower Layer Information	706
Called Party Number.....	707
Called Party Subaddress	707
Calling Party Number	707
Calling Party Subaddress	708
Quality of Service Parameter	708
Transit Network Selection	708
Cause	709
ATM Header File	711

Other Windows Sockets 2 Considerations	711
Secure Sockets Layer (SSL)	711
RSVP	711
Chapter 13: QOS Overview	713
QOS Documentation Structure	713
Determining Which Discussion Is for You	714
Additional Information on QOS	715
About Quality of Service	715
Introduction to QOS	715
Quality of Service Defined	716
Windows 2000 Quality of Service Defined	716
What QOS Solves	716
How Windows 2000 QOS Works	718
Windows 98 QOS Notes	718
QOS Header Files	720
QOS Components	720
Application-Driven QOS Components	721
Network-Driven QOS Components	724
Policy-Driven QOS Components	726
RSVP and QOS	729
Chapter 14: QOS Programming	731
Basic QOS Operations	731
QOS-Enabling Your Application	731
Opening a QOS-Enabled Socket	732
Invoking the RSVP SP	732
Providing the RSVP SP with QOS-specific Parameters	733
Receiving QOS-Enabled Data	734
Sending QOS-Enabled Data	735
Closing the QOS Connection	736
QOS Templates	736
Enumerating Available QOS Templates	737
Applying a QOS Template	737
Installing a QOS Template	738
Removing a QOS Template	738
Built-in QOS Templates	738
RSVP SP Error Codes	739
Error Codes	739
Error Values	740

Service Types	749
Primary Service Types.....	750
BEST EFFORT	750
CONTROLLED LOAD.....	750
GUARANTEED	750
QUALITATIVE.....	751
Secondary Service Types	751
SERVICETYPE_NOTRAFFIC	751
SERVICETYPE_GENERAL_INFORMATION	752
SERVICETYPE_NOCHANGE	752
SERVICE_NO_TRAFFIC_CONTROL.....	752
SERVICE_NO_QOS_SIGNALING	753
Using Service Types	753
Directional Implications of Service Types	753
Examples of Setting the Service Type.....	754
Using the ProviderSpecific Buffer	755
Structure of the ProviderSpecific Buffer.....	755
Use of the ProviderSpecific Buffer as a Receiver.....	755
Use of the ProviderSpecific Buffer as a Sender	755
Understanding Traffic Control	756
How the RSVP SP Invokes TC	756
Using SIO_CHK_QOS	757
Disabling Traffic Control.....	758
QOS Events	758
Listening for FD_QOS Events	759
Using WSAEventSelect or WSAAsyncSelect.....	759
Using Overlapped WSALocatl(SIO_GET_QOS)	759
QOS Event Codes	760
RSVP SP and RSVP	761
Basic RSVP Operations.....	761
Invoking RSVP.....	761
Using the RSVP_RESERVE_INFO Object.....	762
Confirming RSVP Reservations.....	762
Disabling RSVP Signaling	763
RSVP Reservation Styles	763
Base RSVP Reservation Styles.....	763
Default RSVP Filter Style Settings.....	764
Overriding Default RSVP Filter Style Settings	765
Mapping RSVP SP Parameters to RSVP	766

RSVP PATH and RESV Messages.....	767
Tspec, FlowSpec, and Adspec.....	769
Mapping QOS Call Sequences to RSVP	771
Sending Applications.....	771
Receiving Applications	775
Receiver Reservation Semantics	778
Using WSAConnect to Join Unicast RSVP Sessions	778
Using WSJoinLeaf to Join Multicast RSVP Sessions.....	780
Use of Sendto and WSASendTo by Multicast Senders	781
Using WSALocatl(SIO_SET_QOS) During RSVP Sessions	781
Chapter 15: QOS API Reference.....	783
QOS Functions	783
QOS Structures	791
QOS Objects	798
Chapter 16: Traffic Control API Reference	807
Traffic Control Functions	807
Entry Points Exposed by Clients of the Traffic Control Interface	829
Traffic Control Structures	834
Traffic Control Objects.....	855
Chapter 17: Local Policy Module API Reference	859
LPM Functions.....	859
LPM Structures.....	875

Part 3

Index: Networking Services Programming Elements – Alphabetical Listing.....	879
--	------------