



Windows Sockets

An Open Interface for Network Programming under Microsoft Windows

Version 1.1
20 January 1993

Note

This HTML version of the Windows Socket 1.1 Specification is provided by Mark Towfiq. It may be freely redistributed, either as provided or in modified form. Winsock providers may integrate it into their product documentation without incurring any obligation.

Acknowledgements

I am indebted and thankful to:

- [Joel Golberger](#) of InfoMagic, Inc. for the work he did on the [Windows Help version of the Windows Sockets Specification](#). I used [his .RTF file](#) as the source file for this document.
- [Chris Hector](#) of Cray Research Inc. for his package [rtftohtml](#), which was used to generate this document.

Mark Towfiq (towfiq@East.Sun.Com)

Table of Contents

- [Authors and Copyright](#)
- [Acknowledgments](#)
- [Introduction](#)
 - [What is Windows Sockets](#)
 - [Berkeley Sockets](#)
 - [Microsoft Windows and Windows-specific extensions](#)
 - [The Status of this Specification](#)
 - [Revision History](#)
 - [Windows Sockets Version 1.0](#)
 - [Windows Sockets Version 1.1](#)
- [Programming with Sockets](#)
 - [Windows Sockets Stack Installation Checking](#)
 - [Sockets](#)
 - [Basic concepts](#)
 - [Client-server model](#)

- Out-of-band data
 - Broadcasting
- Byte Ordering
- Socket Options
- Database Files
- Deviation from Berkeley Sockets
 - socket data type and error values
 - select() and FD *
 - Error codes - errno, h_errno & WSAGetLastError()
 - Pointers
 - Renamed functions
 - Blocking routines & EINPROGRESS
 - Maximum number of sockets supported
 - Include files
 - Return values on API failure
- Raw Sockets
- Windows Sockets in Multithreaded Versions of Windows
- Socket Library Overview
 - Socket Functions
 - Blocking/Non blocking & Data Volatility
 - Database Functions
 - Microsoft Windows-specific Extension Functions
 - Asynchronous select() Mechanism
 - Asynchronous Support Routines
 - Hooking Blocking Methods
 - Error Handling
 - Accessing a Windows Sockets DLL from an Intermediate DLL
 - Internal use of Messages by Windows Sockets Implementations
 - Private API Interfaces
- Socket Library Reference
 - Socket Routines
 - accept()
 - bind()
 - closesocket()
 - connect()
 - getpeername()
 - getsockname()
 - getsockopt()
 - htonl()
 - htons()
 - inet_addr()
 - inet_ntoa()
 - ioctlsocket()
 - listen()
 - ntohl()
 - ntohs()
 - recv()
 - recvfrom()
 - select()
 - send()
 - sendto()
 - setsockopt()

- [shutdown\(\)](#)
- [socket\(\)](#)
- [Database Routines](#)
 - [gethostbyaddr\(\)](#)
 - [gethostbyname\(\)](#)
 - [gethostname\(\)](#)
 - [getprotobyname\(\)](#)
 - [getprotobynumber\(\)](#)
 - [getservbyname\(\)](#)
 - [GetServByPort\(\)](#)
- [Microsoft Windows-specific Extensions](#)
 - [WSAAsyncGetHostByAddr\(\)](#)
 - [WSAAsyncGetHostByName\(\)](#)
 - [WSAAsyncGetProtoByName\(\)](#)
 - [WSAAsyncGetProtoByNumber\(\)](#)
 - [WSAAsyncGetServByName\(\)](#)
 - [WSAAsyncGetServByPort\(\)](#)
 - [WSAAsyncSelect\(\)](#)
 - [WSACancelAsyncRequest\(\)](#)
 - [WSACancelBlockingCall\(\)](#)
 - [WSACleanup\(\)](#)
 - [WSAGetLastError\(\)](#)
 - [WSAIsBlocking\(\)](#)
 - [WSASetBlockingHook\(\)](#)
 - [WSASetLastError\(\)](#)
 - [WSAStartup\(\)](#)
 - [WSAUnhookBlockingHook\(\)](#)
- [Appendices](#)
 - [Error Codes and Header Files](#)
 - [Error Codes](#)
 - [Header Files](#)
 - [Berkeley Header Files](#)
 - [Windows Sockets Header File - winsock.h](#)
 - [Notes for Windows Sockets Suppliers](#)
 - [Introduction](#)
 - [Windows Sockets Components](#)
 - [Development Components](#)
 - [Run Time Components](#)
 - [Multithreadedness and blocking routines.](#)
 - [Database Files](#)
 - [FD_ISSET](#)
 - [Error Codes](#)
 - [DLL Ordinal Numbers](#)
 - [Validation Suite](#)
 - [For Further Reference](#)
 - [Background Information](#)
 - [Origins of Windows Sockets](#)
 - [Legal Status of Windows Sockets](#)
 - [The Story Behind the Windows Sockets Icon](#)

Authors and Copyright

Martin Hall

Mark Towfiq

Geoff Arnold

David Treadwell

Henry Sanders

**Copyright (c) 1992 by Martin Hall, Mark Towfiq
Geoff Arnold, David Treadwell and Henry Sanders**

All rights reserved.

This document may be freely redistributed in any form, electronic or otherwise, provided that it is distributed in its entirety and that the copyright and this notice are included. Comments or questions may be submitted via electronic mail to WinSock@MailBag.Intel.Com. Requests to be added to the Windows Sockets mailing list should be addressed to MajorDomo@MailBag.Intel.Com. This specification, archives of the mailing list, and other information on Windows Sockets are available via anonymous FTP from the host [SunSite.UNC.Edu, directory /pub/micro/pc-stuff/ms-windows/winsock](ftp://SunSite.UNC.Edu/directory/pub/micro/pc-stuff/ms-windows/winsock). Questions about products conforming to this specification should be addressed to the vendors of the products.

Acknowledgments

The authors would like to thank their companies for allowing them the time and resources to make this specification possible: JSB Corporation, Microdyne Corporation, FTP Software, Sun Microsystems, and Microsoft Corporation.

Special thanks should also be extended to the other efforts contributing to the success of Windows Sockets. The original draft was heavily influenced by existing specifications offered and detailed by JSB Corporation and NetManage, Inc. The "version 1.0 debate" hosted by Microsoft in Seattle allowed many of the members of the working group to hash out final details for 1.0 vis-a-vis.

Sun Microsystems was kind enough to allow first time implementors to "plug and play" beta software during the first Windows Sock-A-Thon of Windows Sockets applications and implementations at Interop Fall '92. Microsoft has shared WSAT (the Windows Sockets API Tester) with other Windows Sockets implementors as a standard Windows Sockets test suite to aid in testing their implementations. Finally, Sun Microsystems and FTP Software plan to host the Windows Sock-A-Thon II in Boston February '93.

Without the contributions of the individuals and corporations involved in the working group, Windows Sockets would never have been as thoroughly reviewed and completed as quickly. In just one year, several competitors in the networking business developed a useful specification with something to show for it! Many thanks to all which participated, either in person or on e-mail to the Windows Sockets effort. The authors would like to thank everyone who participated in any way, and apologize in advance for anyone we have omitted.

List of contributors:

Martin Hall	(Chairman)	JSB Corporation	martinh@jsbus.com
<u>Mark Towfig</u>	(Coordinator)	Microdyne Corporation	towfig@microdyne.com
Geoff Arnold	(Editor 1.0)	Sun Microsystems	geoff@east.sun.com
David Treadwell	(Editor 1.1)	Microsoft Corporation	davidtr@microsoft.com
Henry Sanders		Microsoft Corporation	henrysa@microsoft.com
J. Allard		Microsoft Corporation	jallard@microsoft.com
Chris Arap-Bologna		Distinct	chris@distinct.com
Larry Backman		FTP Software	backman@ftp.com
Alistair Banks		Microsoft Corporation	alistair@microsoft.com
Rob Barrow		JSB Corporation	robb@jsb.co.uk
Carl Beame		Beame & Whiteside	beame@mcmaster.ca
Dave Beaver		Microsoft Corporation	dbeaver@microsoft.com
Amatzia BenArtzi		NetManage, Inc.	amatzia@netmanage.com
Mark Beyer		Ungermann-Bass	mbeyer@ub.com
Nelson Bolyard		Silicon Graphics, Inc.	nelson@sgi.com
Pat Bonner		Hewlett-Packard	p_bonner@cnd.hp.com
Derek Brown		FTP Software	db@wco.ftp.com
Malcolm Butler		ICL	mcab@oasis.icl.co.uk
Mike Calbaum		Frontier Technologies	mike@frontiertech.com
Isaac Chan		Microsoft Corporation	isaacc@microsoft.com
Khoji Darbani		Informix	khoji@informix.com
Nestor Fesas		Hughes LAN Systems	nestor@hls.com
Karanja Gakio		FTP Software	karanja@ftp.com
Vikas Garg		Distinct	vikas@distinct.com
Gary Gere		Gupta	ggere@gupta.com
Jim Gilroy		Microsoft Corporation	jamesg@microsoft.com
Bill Hayes		Hewlett-Packard	billh@hpchdpc.cnd.hp.com
Paul Hill		MIT	pbh@athena.mit.edu
Tmima Koren		NetManage, Inc.	tmima@netmanage.com
Hoek Law		Citicorp	law@dcc.tti.com
Graeme Le Roux		Moresdawn P/L	-
Kevin Lewis		Novell	kevinl@novell.com
Roger Lin		3Com	roger_lin@3mail.3com.com
Terry Lister		Hewlett-Packard	tel@cnd.hp.com
Jeng Long Jiang		Wollongong	long@twg.com
Lee Murach		Network Research	lee@nrc.com
Pete Ostenson		Microsoft Corporation	peteo@microsoft.com
David Pool		Spry, Inc.	dave@spry.com
Bob Quinn		FTP Software	rcq@ftp.com
Glenn Reitsma		Hughes LAN Systems	glennr@hls.com
Brad Rice		Age	rice@age.com
Allen Rochkind		3Com	-
Jonathan Rosen		IBM	jrosen@vnet.ibm.com
Steve Stokes		Novell	stoke@novell.com
Joseph Tsai		3Com	joe_tsai@3mail.3com.com
James Van Bokkelen		FTP Software	jbvb@ftp.com
Miles Wu		Wollongong	wu@twg.com
Boris Yanovsky		NetManage, Inc.	boris@netmanage.com

Introduction

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.