MAY 1996

NOVELL® RESEARCH

NetWare Client 32 for DOS/Windows: Overview of Architecture and Features

EDWARD A. LIEBING
Research Engineer
Systems Research Department

NetWare Client 32 for DOS/Windows represents the next step in the evolution of the client software. It combines the best elements of Novell's previous client architectures (the NETX Shell and the NetWare DOS Requester) and adds some significant enhancements. This AppNote provides an overview of the NetWare Client 32 architecture and describes the various components included in the software. It also summarizes the key features of the NetWare Client 32 for DOS/Windows.

ACKNOWLEDGEMENTS
Thanks to Ian Stiles and Brad Young of Novell for their help with this AppNote.

TRADEMARKS
NetWare and Novell are registered trademarks and NetWare Client 32, NetWare Directory Services, NDS, NetWare DOS Requester, NetWare Loadable Module, NLM, Virtual Loadable Module, and VLM are trademarks of Novell, Inc in the United States and other countries.

Macintosh is a registered trademark of Apple Computer, Inc. IBM and OS/2 a re registered trademarks of International Business Machines Corporation. Microsoft, MS-DOS, and Windows are registered trademarks of Microsoft Corporation. All other product names mentioned are trademarks of their respective companies or distributors.

DISCLAIMER
Novell, Inc. makes no representations or warranties with respect to the contents or use of these Application Notes (AppNotes) or of any of the third-party products discussed in the AppNotes. Novell reserves the right to revise these AppNotes and to make changes in their content at any time, without obligation to notify any person or entity of such revisions or changes. These AppNotes do not constitute an endorsement of the third-party product or products that were tested. Configuration(s) tested or described may or may not be the only available solution. Any test is not a determination of product quality or correctness, nor does it ensure compliance with any federal, state, or local requirements. Novell does not warranty products except as stated in applicable Novell product warranties or license agreements.

Novell, Inc.
122 East 1700 South
Provo, Utah 84606 USA

Introduction

NetWare Client 32 for DOS/Windows represents the next step in the evolution of the client software. It combines the best elements of Novell's previous client architectures (the NETX Shell and the NetWare DOS Requester, known by many as VLMs) and adds some significant enhancements. NetWare Client 32 for DOS/Windows offers 16-bit desktop users full 32-bit access to NetWare services, including Novell Directory Services (NDS). Built on the same code- base as the recently released NetWare Client 32 for Windows 95, the NetWare Client 32 for DOS/Windows provides increased speed through 32-bit access. Users also benefit from Client 32 features such as full automatic reconnection to all network files and resources, simultaneous access to multiple Directory trees, and improved local file caching routines.

The NetWare Client 32 for DOS/Windows is designed for workstations running MS-DOS 5.0 or above, Windows 3.1 or above, and Windows for Workgroups 3.11 or above. The NetWare Client 32 for DOS/Windows supports connections to NetWare 2.2, 3.1x, and 4.x servers. At this time, Client 32 does not support peer-to- peer connections via Personal NetWare.

This AppNote provides an overview of the NetWare Client 32 architecture and describes the various components included in the software. It also summarizes the key features of the NetWare Client 32 for DOS/Windows.

NetWare Client 32 Architecture Overview

Novell programmers designed the underlying architecture of the 32-bit clients to be dynamic, modular, and portable. Their main goal was to create a consistent, cross-platform architecture that could be integrated as tightly as possible to the native DOS, Windows 3.1x, and Windows 95 environments. Even though each desktop operating environment is unique, Novell programmers hoped to supply similar features for 32-bit access, as well as client utilities that have a consistent look-and-feel across platforms.

The NetWare Client 32 for DOS/Windows is designed to take full advantage of the 32-bit capabilities of Intel's 386, 486, and Pentium microprocessors. Unlike the 286 microprocessor, which addresses memory in 16-bit chunks, these newer microprocessors address memory in 32-bit chunks and can therefore run 32-bit programs. Because 32-bit processors can move data in bigger chunks, a program that is optimized for 32-bit processors can be smaller and execute much faster than the same program written to run on 16-bit computers.

If you are familiar with the NetWare 3 and 4 server architecture, you will recognize many of the same architectural elements implemented in Client 32 for DOS/Windows. These elements include protected-mode execution, a flat memory model, advanced memory management, and even client NetWare Loadable Modules (NLMs) and 32-bit Open Data-link Interface (ODI) LAN drivers.

Novell's previous clients)the NetWare Shell (NETX) and the NetWare DOS Requester (VLMs))operate in 16-bit real mode, which requires conventional memory. In a typical DOS configuration, all DOS-based software must run in a fixed amount of conventional memory (640KB) or in the Upper Memory Blocks (UMBs) between 640 and 1024KM (1MB). As a result, network connectivity software has to contend with other DOS-based applications and utilities for available conventional/UMB memory. In multiple-protocol configurations, network connectivity can take up quite a bit of this memory, leaving precious little for applications.

NetWare Client 32 provides a 32-bit protected-mode environment for DOS/Windows clients. When Client 32 loads, it switches the CPU to protected mode. In this mode, all of the computer's extended memory (beyond 1MB) is addressable as one continuous range of addresses. This flat memory model makes memory allocation and management more efficient and flexible.

Client 32 works with an extended memory manager (HIMEM.SYS or its equivalent) to make extended memory available for loading the NetWare Client 32 modules. The resulting memory environment can accommodate client-based NLMs and LAN drivers, enabling them to be loaded and unloaded in extended memory as needed, just like NLMs and LAN drivers on NetWare servers. Since the majority of the Client 32 modules are loaded in extended memory, Client 32 requires a mere 4KB of conventional or UMB memory.

The Client 32 architecture is comprised of three main components that make up the 32-bit client environment:

- The NetWare I/O Subsystem (NIOS.EXE)
- Communications protocols and LAN drivers (LSLC32.NLM, CMSM.NLM, CNE2000.LAN, IPX.NLM, TCPIP.NLM, SNMP.NLM, and so on)
- The NetWare Client 32 Requester (CLIENT32.NLM)

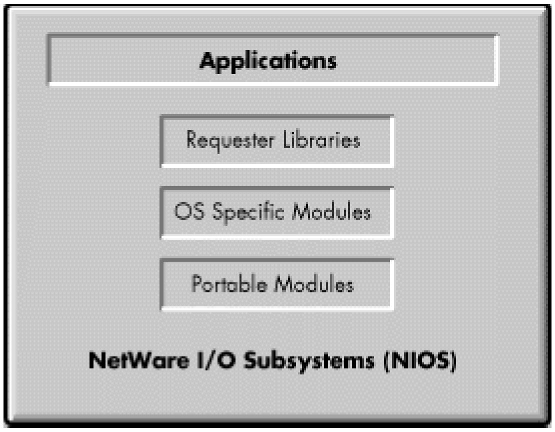
In addition, NetWare Client 32 for DOS/Windows comes with a number of client utilities, including the GUI Login utility (LOGINW31.EXE), the NetWare User Tools utility (NWUSER.EXE and NETWARE.DRV), and the NetWare Application Manager utility (NAL.EXE). You can also obtain a copy of the Client 32 Configuration Utility (NWSETUP.EXE), a shareware utility for making NET.CFG settings in Windows.

The following sections of this AppNote describe the Client 32 architectural components listed above. The client utilities are discussed in other AppNotes in this issue.

The NetWare I/O Subsystem (NIOS)

The NetWare I/O Subsystem (NIOS.EXE) is the main component around which all the other Client 32 components revolve. In fact, Client 32 is often referred to as the "NIOS" client. NIOS serves as the interface layer between the client operating system and the 32-bit client services provided by NetWare (see Figure 1).

Figure 1: *Architecture of the NetWare I/O Subsystem (NIOS).*



NIOS is the first Client 32 component that is loaded. It is essentially a DOS-extender program that creates a protected-mode, flat memory environment for loading the other Client 32 modules and drivers. You can think of NIOS as the client equivalent of NetWare's SERVER.EXE program, which runs first on a NetWare server to set up the protected-mode environment for the NetWare operating system. Whereas SERVER.EXE is a module launcher and memory manager rolled into one, NIOS is a module launcher that communicates with an extended memory manager that is already loaded on the workstation.

After you load NIOS.EXE, you will not be able to unload it unless you reboot the workstation. However, once NIOS is loaded, it makes the NetWare client environment flexible and dynamic. You can load and unload the other Client 32 modules and drivers at any time. Through the NWSETUP.EXE utility, you can dynamically allocate and deallocate memory for many of the NET.CFG settings that you make.

Note: For a discussion of changing NET.CFG parameters, including the NET.CFG parameters that affect the NIOS module, see "Setting NET.CFG Parameters Through the NWSETUP.EXE Utility" in this issue.

By using DOS-specific application program interfaces (APIs), NIOS creates an extended-memory environment that allows 32-bit NLMs and *.LAN drivers that are running in protected mode to execute real-mode operations when necessary. In addition, NIOS manages real-mode to protected-mode context switches and provides hooks to real-mode interrupts. NIOS also provides a Windows transition architecture that allows environment-dependent modules to correctly handle transitions from DOS to Windows and from Windows to DOS.

In the NetWare Client 32 for DOS/Windows, NIOS is the only module that loads as an executable (EXE) file. All of the other modules are implemented as client NLMs and are loaded with the LOAD command from the STARTNET.BAT file.

How NIOS Allocates Memory

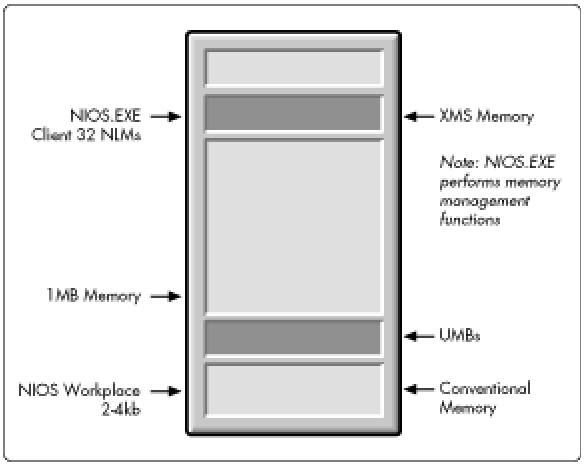
At load time, NIOS communicates with HIMEM.SYS (or equivalent XMS extended memory manager) to find out how much extended memory is available. Upon initialization, NIOS takes about 25 percent of the available extended memory for its own file caching routines and other internal tasks. (You can configure caching and other NIOS features in the NET.CFG file. See the NetNote entitled "Workstation File Caching with NetWare Client 32" in this issue.)

Working from the flat memory model, NIOS communicates with the extended memory manager to allocate memory for each of the remaining Client 32 modules as they load. The bulk (99.35 percent) of the NetWare Client 32 software loads in extended memory. Client 32 uses only about 4KB of memory for a "shim" in either UMBs or conventional memory. If NIOS can find 4KB available in UMBs (which is usually the case), the shim loads there. If NIOS cannot find 4KB of UMBs available, it will place the shim in conventional memory.

As mentioned previously, NIOS manages memory allocation for the communications protocols, LAN drivers, and other client-based NLMs that are loaded after it. These modules can use another 800KB to 1.3MB of extended memory, depending on the number of NLMs and types of LAN drivers you have loaded and how many NET.CFG parameters are set differently from their defaults. NIOS allocates this memory starting from the top of available extended memory and working downward (see Figure 2).

Having only a 4KB real-mode footprint for the network connectivity software frees up conventional memory for DOS-based applications. This is especially beneficial for workstations that load multiple frame types and protocols.

Figure 2: *Client 32 for DOS/Windows memory usage.*



As an example of Client 32's memory usage, we set up a multimedia workstation with a sound card and a CD-ROM drive. (We also loaded a 4KB SNIPPER utility to make the screen shots.) This machine initially ran the NetWare DOS Requester to connect to the network. After a strenuous memory configuration exercise, we were able to squeeze out 580KB of conventional memory, with 40KB of UMBs available (see Figure 3).

Figure 3: *Memory consumption before upgrading to NetWare Client 32 for DOS/Windows.*

```
Modules using memory below 1MB:

Name           Total       =   Conventional  +    Upper Memory

MSDOS       19,341 (19K)      19,341 (19K)          0  (0K)
HIMEM        1,168  (1K)       1,168  (1K)          0  (0K)
EMM386       4,144  (4K)       4,144  (4K)          0  (0K)
COMMAND      3,312  (3K)       3,312  (3K)          0  (0K)
DOSEDIT      2,176  (2K)       2,176  (2K)          0  (0K)
NE2000       6,240  (6K)       6,240  (6K)          0  (0K)
VLM         71,984 (70K)      71,984 (70K)     51,136 (50K)
SNIPPER      3,776  (4K)       3,776  (4K)          0  (0K)
```

```
SETVER          608  (1K)           0  (0K)          608  (1K)
ANSI          4,208  (4K)           0  (0K)        4,208  (4K)
MDSCD_FD     16,400 (16K)           0  (0K)       16,400 (16K)
MOUSE        17,088 (17K)           0  (0K)       17,088 (17K)
MSCDEX       40,368 (39K)           0  (0K)       40,368 (39K)
SMARTDRV     30,368 (30K)           0  (0K)       30,368 (30K)
LSL           5,360  (5K)           0  (0K)        5,360  (5K)
IPXODI       12,992 (13K)           0  (0K)       12,992 (13K)
Free        635,488(621K)     594,064(580K)       41,424 (40K)
```

This same machine running Client 32 has 610KB of conventional memory and 105KB of UMBs available. This leaves much more room for DOS applications that are running ın native DOS or in a DOS box in Windows.

Figure 4: Since the bulk of Client 32 loads in extended memory, more conventional memory is freed up for DOS-based applications.

```
Modules using memory below 1MB:

Name           Total       =   Conventional  +    Upper Memory

MSDOS        19,341 (19K)      19,341 (19K)            0  (0K)
HIMEM         1,168  (1K)       1,168  (1K)            0  (0K)
EMM386        4,144  (4K)       4,144  (4K)            0  (0K)
COMMAND       3,312  (3K)       3,312  (3K)            0  (0K)
DOSEDIT       2,192  (2K)       2,192  (2K)            0  (0K)
Client32      3,360  (3K)           0  (0K)        3,360  (3K)
SETVER          608  (1K)           0  (0K)          608  (1K)
ANSI          4,208  (4K)           0  (0K)        4,208  (4K)
MDSCD_FD     16,400 (16K)           0  (0K)       16,400 (16K)
MOUSE        17,088 (17K)           0  (0K)       17,088 (17K)
MSCDEX       40,368 (39K)           0  (0K)       40,368 (39K)
SMARTDRV     30,368 (30K)           0  (0K)       30,368 (30K)
Free        732,448(715K)     624,896(610K)      107,552(105K)
```

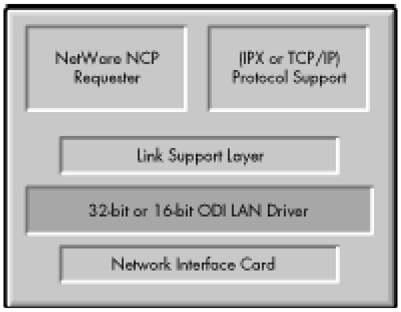
The tradeoff, of course, is that less extended memory is left when you run the NetWare Client 32 for DOS/Windows. Our multimedia workstation came with 20MB of memory, of which almost 6MB was allocated to SmartDrive and Client 32, leaving 14MB for Windows programs. On a workstation with less memory installed, the amount of extended memory taken by Client32 can become an issue. However, Novell chose this architecture because you can add more extended memory to a machine (by adding RAM), but you cannot increase the 640KB conventional memory limitation for DOS.

Note: Enabling Client 32's local file cache and other features also affects memory usage. For more information, see the NetNote entitled "Workstation File Caching with NetWare Client 32" in this issue.

Communications Protocols and LAN Drivers

NetWare Client 32 uses the ODI (Open Data-link Interface) architecture to communicate to the network (see Figure 5). The NetWare Client 32's Link Support Layer, IPX, and TCP/IP components are similar to their NetWare DOS Requester counterparts. However, the NetWare Client 32 components have been rewritten as client NLMs rather than as DOS executables. This way, the NetWare Client 32 components can take advantage of the flat memory model that NIOS provides.

Figure 5: Client 32 is based on the ODI architecture for the NetWare operating system.


NetWare NCP Requester
(IPX or TCP/IP) Protocol Support
Link Support Layer
32-bit or 16-bit ODI LAN Driver
Network Interface Card

LSLC32.NLM is the Client 32 version of the Link Support Layer (LSL). It is a ported version of the LSL for the NetWare OS LAN architecture that has been rewritten in ANSI C code. LSLC32.NLM is not platform specific, so it can be used on other platforms supported by NIOS.

CMSM.NLM is a C-based version of the Media Support Module in the ODI architecture. Client 32 also uses Topology Support Modules, or TSMs, which are components of the NetWare OS LAN driver architecture. TSMs provide an intermediate layer between the actual LAN driver and the LSL. TSMs are topology-dependent)if you have an Ethernet topology, you load the ETHERTSM.NLM; if you have Token-Ring, you load the TOKENTSM.NLM file; if you have an FDDI topology, you load the FDDI.NLM file. (There are no 32-bit support modules available for ARCnet.)

The TSMs are customized to each individual desktop operating system. Instead of using the traditional LSL buffer pool approach that the NetWare OS TSMs use, the Client 32 TSMs use NIOS's Receive Look-Ahead method. Receive Look-Ahead allows protocols to preview packet header information, and then allocate buffers directly to incoming packets.

LAN Drivers for Client 32

The NetWare Client 32 software supports 32-bit LAN drivers as well as the older 16-bit ODI LAN drivers. You can thus choose from three basic sets of available LAN drivers:

- 32-bit ODI LAN drivers included with NetWare Client 32
- Many of the 32-bit ODI Server LAN drivers included with NetWare 4.1 and above
- 16-bit ODI LAN drivers from previous NetWare client software

Note: For best results, you should use 32-bit LAN drivers whenever possible. See the AppNote entitled "Installing NetWare Client 32 for DOS/Windows" in this issue for more information.

32-bit ODI LAN Drivers Included with Client 32. The 32-bit ODI LAN drivers provided with the NetWare

Client 32 for DOS/Windows software have been rewritten in C especially for Client 32. They are based on version 1.10 of Novell's 32-bit ODI driver specification. (Future versions of these drivers will contain additional functionality to provide the most benefit from using the NetWare Client 32 for DOS/Windows, especially if you use 32-bit network interface boards.)

32-bit Server LAN Drivers Included with NetWare 4.1. You can also use many of the 32-bit binary *.LAN drivers included with the NetWare 4.1 operating system. (NetWare 3.x server drivers are not compatible with NIOS.) This is possible because the new client architecture uses NIOS, a derivative of SERVER.EXE. These LAN drivers are based on version 3.20 of Novell's assembly- based driver specification.

Not all of the assembly-based server *.LAN drivers written to the v3.20 driver specification will work with Client 32, but many of them will. If you cannot find a 32-bit LAN driver for your network interface board, try using the assembly-based server .LAN drivers that come with NetWare 4.1 and above.

Note: The binary *.LAN files that are being developed for the Green River release of NetWare 4 are written to version 3.30 of Novell's assembly driver specification, and are designed to work with Client 32 as 32-bit LAN drivers.

16-bit ODI LAN Drivers. If you do not have server .LAN drivers for your network interface board, or if the server drivers you have do not work with Client 32, you can continue to use the 16-bit ODI LAN drivers that work with the NetWare DOS Requester. These 16-bit LAN drivers require a special shim to communicate on the network via Client 32. A different set of support drivers will be loaded when you bring up Client 32 on the workstation. (The differences between loading 16-bit and 32-bit LAN drivers is explained in the AppNote entitled "Installing NetWare Client 32 for DOS/Windows" in this issue.)

As of this writing, the list of 16- and 32-bit LAN drivers that are included with Client 32 is as follows (this list is subject to change):

16-bit LAN Driver	Board(s)
NE2.COM	Novell/Eagle NE/2
NE2_32.COM	Novell/Eagle NE/2-32
NE1000.COM	Novell/Eagle NE1000
NE1500T.COM	Novell Ethernet NE1500T
NE2000.COM	Novell/Eagle/Microdyne NE2000, National Semiconductor NE2000 InfoMover
NE2100.COM	Novell Ethernet NE2100
NE3200.COM	Novell/Eagle/Microdyne NE3200, Intel EtherExpress32
PCIMLID.COM	Accton EN1203 PCI Ethernet, Accton EN1207(-TX) PCI Fast Ethernet
MPX2PNP.COM	Accton EN166X MPX-Series Ethernet
PCMMLID.COM	Accton EN2218 Ethernet+Modem/Fax Adapter
E22ODI.COM	Cabletron Systems Ethernet E22
F70ODI.COM	Cabletron Systems FDDI F70
DE22X.COM	D-Link DE-220 Family
DLKPCI.COM	D-Link PCI Ethernet
HPFEODIM.COM	Hewlett-Packard 10/100VG Bus Master LAN Adapter
E100BODI.COM	Intel EtherExpress PRO/100 LAN Adapter
NTR2000.COM	Novell/Microdyne NTR2000 Token-Ring Adapter, IBM Token-Ring Network
LANSUP.COM	ODI Module for the IBM LAN Support Program

MADGEODI.COM	Madge Smart 16/4 Ringnode Adapter
OCE2XODI.COM	Olicom OC-2231 Ethernet PCMCIA Card
OCTOK16.COM	Olicom Token-Ring PCI 16/4 Adapter
ILANPCI.COM	Racal InterLan PCI Ethernet MLID
NI6510.COM	Racal InterLan XLerator/EB/NI6510 Ethernet MLID
SMC8000.COM	SMC Ethernet LAN Adapter
SMC8100.COM	SMC TokenCard Elite Family Adapter
SMC9232.COM	SMC Ether100 Ethernet and Fast Ethernet EISA Adapter
SMCPWR.COM	SMC 8432/8434/9332 PCI Adapter
TCTOKSH.COM	Thomas-Conrad TC4035/TC4045/TC4046 Adapters
TC4048SH.COM	Thomas-Conrad TC4048 Adapter
TCVG45.COM	Thomas-Conrad VG45 ISA Adapter
TCVG47.COM	Thomas-Conrad VG47 EISA Adapter
XPSODI.COM	Xircom Performance Series
TRXNET.COM	Novell RX-Net/RX-Net II, RX-Net/2

32-bit LAN Driver	**Board(s)**
CNE2.LAN	Novell/Eagle NE/2
CNE2_32.LAN	Novell/Eagle NE/2-32
NE1000.LAN	Novell/Eagle NE1000
CNE1500T.LAN	Novell Ethernet NE1500T
CNE2000.LAN	Novell/Eagle/Microdyne NE2000, National Semiconductor NE2000 InfoMover, IBM Ethernet Credit Card Adapter II
CNE2100.LAN	Novell Ethernet NE2100
CNE3200.LAN	Novell/Eagle/Microdyne NE3200, Intel EtherExpress32
CNTR2000.LAN	Novell/Microdyne NTR2000 Token-Ring Adapter
PCI40.LAN	Accton EN1203, EN1207 PCI Series Ethernet (960111)
ACC41.LAN	Accton EN166X PnP Adapter
NE100SP.LAN	Microdyne NE10/100SP Ethernet Adapter
OCTOK162.LAN	Olicom 16 bit Token-Ring Adapters (ISA adapter in EISA bus) Olicom 16 bit Token-Ring and Token-Ring PCI Adapters

The 32-bit C-based versions of the LAN drivers usually have a filename beginning with "C")for example, CNE2000.LAN is the C-based version of the NE2000.LAN server driver. Since these are NLM files, they are loaded using the `LOAD` command. Configuration settings such as frame type, interrupt, I/O port, and retries are specified in the `LOAD` command.

Protocols

NetWare Client 32 for DOS/Windows supports both IPX/SPX and TCP/IP protocols equally. The IPX, TCP/IP, and NetWare/IP protocol stacks have been ported to NLM format (IPX.NLM, TCPIP.NLM and NWIP.NLM) for use with Client 32. Some optional modules, such as SNMP.NLM, are implemented as NLMs as well. Others, such as NetBIOS, 3270 emulation, and Btrieve support, remain in their current terminate-and-stay-resident (TSR) implementation for the initial release of Client 32.

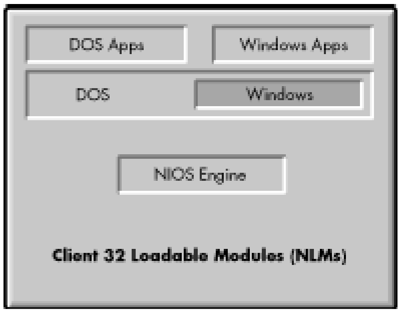
The Client 32 protocols contain built-in support for Packet Burst and Large Internet Packets (LIP) for optimized performance on both local and wide area networks. They also support binding to multiple network adapters and auto-binding to dynamically loaded LAN drivers. The TCP/IP stack includes support for DHCP (Dynamic Host Configuration Protocol), while NetWare/IP 2.2 includes support for BootP+ (Novell's implementation of DHCP).

The Client 32 Requester

In Novell's new client architecture, the NetWare Client 32 Requester (CLIENT32.NLM) takes the place of the NETX shell or the NetWare DOS Requester (VLMs). The Client 32 Requester provides the internal tables and services necessary to track network resources, file caching, and automatic reconnection levels.

Figure 6 shows how the core NLMs included in the Client 32 Requester work with NIOS.

Figure 6: NIOS and core NLMs.



As with the DOS Requester, the NetWare Client 32 Requester is a set of subcomponents, or modules. However, the modules are all contained within the CLIENT32.NLM file instead of as separate files. Following is a listing of the modules contained within CLIENT32.NLM:

Module	Name	Description
CONNMAN	Connection Manager	Used by Client 32 to keep track of the connection tables, and to store other connection information.
TASKMAN	Task Manager	Keeps track of each task in the system, whether these are WINDOS, WIN16, or WIN32 applications or activities.
FILEDIR	File and Directory	Handles all file system functions.
PRINT	Printing	Set up network printing capabilities for Client 32.
SESSMUX	Session Multiplexor	Currently supports only NCP(NetWare Core Protocol) sessions attaching and accessing NetWare servers. However, future releases of Client 32 may include support for Microsoft's SMB (Server Message Block) sessions.
NCP	NetWare Core Protocol	A child module that ties to the Session Multiplexor.
NSMUX	Name Services Multiplexor	Coordinates activities between the child modules, or

		processes, that you load.
NDS	Novell Directory Services	A child module to the Name Service Multiplexor module. Provides the name resolution and authentication processes for Directory Services connectivity.
BINDERY	Bindery Services	Also a child module to the Name Service Multiplexor. It provides the name resolution and authentication processes for Bindery connectivity.
MOCKNW	Mock NetWare	Catches raw NCPs sent from applications and redirects them to Client 32 functions in order to take advantage of Client 32's caching and automatic reconnection settings.
POLYPROC	Poly-procedure	Contains a set of generic miscellaneous routines that didn't specifically fit in any of the other CLIENT32.NLM modules.
NETX	DOS Interrupt 21h	Used to redirect all Interrupt 21h (NETX) function calls to redirector Client 32 functions.
VLMMAP	VLM Mapper	Used to redirect all DOS Requester (VLM) function calls to Client 32 functions. The VLMMAP module is used to support applications and utilities that are written to NETX and DOS Requester APIs.

The NetWare Client 32 Requester is fully backward-compatible with the NETX shell and the NetWare DOS Requester, so NetWare-aware applications taking advantage of the earlier client APIs should run with no modification under the NetWare Client 32. This compatibility is accomplished through the NETX and VLMMAP modules that are a part of CLIENT32.NLM.

Features of NetWare Client 32 for DOS/Windows

NetWare Client 32 for DOS/Windows includes all the features users are familiar with in the NetWare clients they are using today. Client 32 builds on these features to deliver a number of advanced capabilities. This section provides a summary of the key features included in the first release of NetWare Client 32 for DOS/Windows.

Full NetWare Connectivity
With Client 32, users can access NetWare 2.x, 3.x and 4.x resources throughout the network, with full connectivity and backwards compatibility with all services and information. The Client 32 Requester supports simultaneous multiple frame types, protocols, and name services (NDS and Bindery).

Improved Installation and Setup Procedures
Client 32 for DOS/Windows comes with both a DOS-based INSTALL.EXE and a Windows 3.x-based SETUP.EXE program. These installation programs allow easy upgrade of previous NetWare workstation clients, including NETX and VLMs. They will auto-detect previously loaded LAN drivers and configure the updated software to match the detected settings. They will also configure TCP/IP and NetWare/IP protocol information.

The client software is installable via floppy disk, local hard drive, network drive, or CD-ROM drive. The upgrade can be centrally administered via the ACU (Automated Client Update) facility.

Smaller Memory Footprint
Client 32 operates with a smaller conventional memory or Upper Memory Block (UMB) footprint (4KB in most environments using 32-bit LAN drivers). This frees up additional memory to run applications.

Support for UNC

Client 32 supports the use of UNC (Universal Naming Convention) for specifying volumes and directories on network servers. Directory Services names can be included in the UNC path; for example, note the NDS Volume object name in this UNC path:

```
\\NOVELL_INC\.NRD_SYS.NR.APPNOTES.Novell\Path
```

Expanded LAN and WAN Connectivity

Users can select from a large number of 32-bit certified server NLM drivers for connectivity between the workstation and the server. A shim is also provided to allow workstations to run a 16-bit ODI LAN driver, such as drivers for PCMCIA cards, in the event an equivalent NLM-style driver is not available.

WAN connectivity is provided via a 16-bit SLIP_PPP driver, shim and dialer. NetWare Client 32 for DOS/Windows does not include support for NDIS drivers.

Large Internet Packet (LIP) and Packet Burst Support

Support for LIP and Packet Burst comes enabled in Client 32 to provide improved LAN and WAN performance.

Multiple NDS Tree Support

Users can authenticate to and simultaneously browse resources on multiple NDS trees (graphically within Windows). (Multiple tree support *does not* include Linked Tree or Federated Tree support).

Improvements over 16-bit VLM Client

Client 32 extends the desktop with a full 32-bit NLM execution environment. It offers protected-mode operation via NIOS, so there is no swapping from extended to UMB or conventional memory, which results in greater performance. Client 32 uses the advanced NCPs found in NetWare 4 and features better error handing of critical errors (Int 24 conditions).

Dynamic Client-side Caching

Client 32 includes new NetWare caching algorithms for superior performance. Users can benefit from dynamic client-side (local) caching for faster performance of repetitive desktop functions.

Automatic Reconnection

Client 32 enables a complete restoration of the user environment after a network failure. The automatic reconnect capabilities include reestablishment of open files, mappings and file locks. This feature doesn't require a NetWare 4.x version of LOGIN.EXE.

Dynamic Reconfiguration of Client Settings

Client 32 accesses client settings via the NET.CFG configuration file for DOS and Windows 3.1x workstations. The client uses dynamic, self-configurable parameters, where possible, to minimize individual administration of workstation NET.CFG files.

GUI Login Utility

Client 32 comes with a GUI login utility that allows users, from within Windows, to access NetWare servers, update environment variables, and establish drive mappings.

Network Management Tools

The NetWare Client 32 for DOS/Windows provides tools to enable network administrators to dynamically setup and refresh client software across the network with Automatic Client Update capabilities, and to centrally administer applications accessed by users through NDS with NetWare Application Manager.

Client 32 also supports the NetWare Administrator utility, which provides full NDS administration, GroupWise and MHS service administration, control of all access rights, management of files, directories and volumes, and navigation of the network and searching of objects.

Client 32 features easier network node management through remote control and configuration management, DOS and Windows 3.x management agents, SNMP reporting via IPX or TCP/IP, and Plug-and-Play support.

Technical Support for NetWare Client 32

While Novell doesn't provide technical support for new Open Beta customers, during the Open Beta phase of development you can submit feedback to:

```
cl32dw@novell.com
```

Once the final version of NetWare Client 32 for DOS and Windows 3.1x is released, technical support will be provided by all the usual Novell channels.

At any time, you can submit enhancement requests for NetWare client software to:

```
enhclient@novell.com
```