



chapter

# 1

## **About the NetWare DOS Requester**

### Overview

This chapter introduces and briefly describes the NetWare DOS Requester.

The following topics are covered in this chapter.

Topic	Page
The New Workstation Software Components	1
Features of the NetWare DOS Requester	2
About the NetWare DOS Requester	3

### The New Workstation Software Components

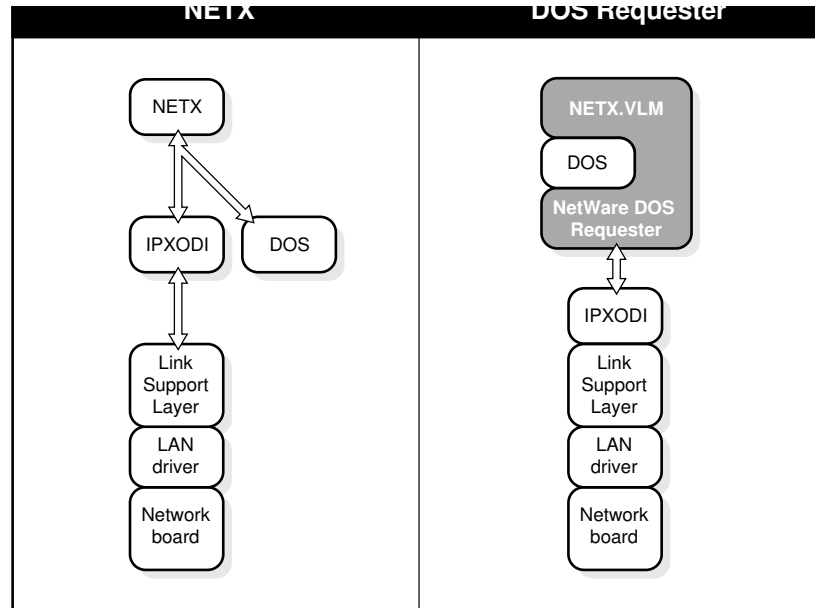
The key components of the NetWare DOS and Windows environment are four terminate-and stay-resident (TSR) programs:

- ◆ LSL™ (Link Support Layer™)
- ◆ A LAN driver (example: NE2000™)
- ◆ IPXODI (Internetwork Packet Exchange™ Open Data Link Interface)
- ◆ NetWare DOS Requester (VLM.EXE and associated.VLM files)

The NetWare DOS Requester replaces NETX. Almost all activity at the workstation now involves these four components.

Figure 1-1 shows the difference between the NETX and the NetWare DOS Requester architectures.

Figure 1-1  
Comparison between  
NETX and NetWare  
DOS Requester



## Features of the NetWare DOS Requester

The NetWare DOS Requester provides several improvements over NETX. It does the following:

- ◆ Provides a modular architecture that has advantages for current and future applications.
- ◆ Takes advantage of memory-swapping technology and DOS redirection capability.
- ◆ Includes Packet Burst protocol and Large Internet Packet (LIP).
- ◆ Supports the installed base of NetWare users by providing backward compatibility with NETX.
- ◆ Supports NetWare Directory Services™ in NetWare v4.0.

## About the NetWare DOS Requester

The NetWare DOS Requester consists of a number of files that provide NetWare support for a DOS workstation. Unlike NETX or other previous versions of the shell, the NetWare DOS Requester consists of a number of Virtual Loadable Modules™ (VLMs).

See “Technical Information” on page 73 for more information about the NetWare DOS Requester and its associated VLMs.

### NETX

Previously, NETX, the NetWare workstation shell for DOS, intercepted requests for DOS services. At the workstation, NETX hooked interrupts in front of DOS and provided network services for files and printing.

### NetWare DOS Requester

The NetWare DOS Requester takes advantage of DOS redirection to provide file and print services. It also uses shell technology to add features to DOS and provide compatibility with NETX.

The VLM manager (VLM.EXE) is responsible for loading the required modules and disbursing requests for individual VLMS. Thus, instead of loading NETX, EMSNETX, XMSNETX, or BNETX, the user loads only VLM.EXE.

### Memory Considerations

The NetWare DOS Requester works with extended memory, expanded memory, and conventional memory. When loading in extended memory, the Requester loads in LIM XMS v2.0 memory, not in the High Memory Area (HMA), meaning that extended memory can coexist with DOS loaded high.

The VLM.EXE automatically selects the best possible memory use: extended memory first; expanded memory second; and conventional memory only in the absence of enhanced memory options.

The user can override the VLM.EXE defaults with command line options. See “VLM Manager” on page 79 for VLM command line options.

## Backward Compatibility with NETX

NETX.VLM is provided for backward compatibility with NETX and older shell versions. NETX and other versions of the shell shouldn't be loaded with the NetWare DOS Requester.

The NetWare DOS Requester supports DOS v3.1 and above.



*chapter* **2** ***Configuring Your Workstation***

## Overview

The NET.CFG file is a configuration file that you use to specify settings for your workstation other than the defaults.

This chapter explains only how to create a NET.CFG file. See “NET.CFG File Parameters” on page 91 for a complete description of NET.CFG options.

The following topics are covered in this chapter.

Topic	Page
When to Use a NET.CFG File	6
Creating a NET.CFG File	6
Entering Options into the NET.CFG File	7
Sample NET.CFG File	8

## When to Use a NET.CFG File

Use entries in the NET.CFG file to change the workstation's network environment or configuration. For example, you might want to change the configuration in these cases:

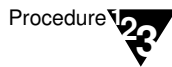
- ◆ You changed the default hardware settings on the network board.
- ◆ You are using multiple protocols.
- ◆ You are using Novell's LAN Workplace.



Older versions of the DOS workstation software used the SHELL.CFG file, which contained network configuration information specific to the shell and not to protocols or other layers. You can specify SHELL.CFG options in the NET.CFG file.

You may need to see the documentation for your protocol for additional configuration information.

## Creating a NET.CFG File



1. Use a DOS text editor to type section headings and options in an existing NET.CFG file or a NET.CFG file that you create to set up your workstation configuration.
2. After a NET.CFG file is created, copy the NET.CFG file to the workstation diskette or directory.

If all workstations use the same NET.CFG file, you can save time by copying the NET.CFG file onto a master workstation diskette.

If each workstation requires a unique NET.CFG file, you must copy a unique NET.CFG file to each workstation diskette or directory.

## Entering Options into the NET.CFG File

Use the following conventions to create or modify a NET.CFG file:

- ◆ Type one heading or option per line.

Headings and options are not case-sensitive. Blank lines are ignored, but they can be helpful in separating the section headings or options to make the NET.CFG file easier to read.

- ◆ Enter section headings at the left margin of the file with no spaces before or after them.

Each NET.CFG section heading may have several options.

- ◆ Enter options, one per line, below the section heading to which they apply, and indent each one.

Use the Space bar or <Tab> key to indent options. Options *must* be indented at least one space.

- ◆ Place a hard return at the end of every line in the file, *including the last line*.

If you don't put a return at the end of the last line, the line is ignored.

- ◆ Precede comment lines with a semicolon.

## Sample NET.CFG File

Following is a sample NET.CFG file that

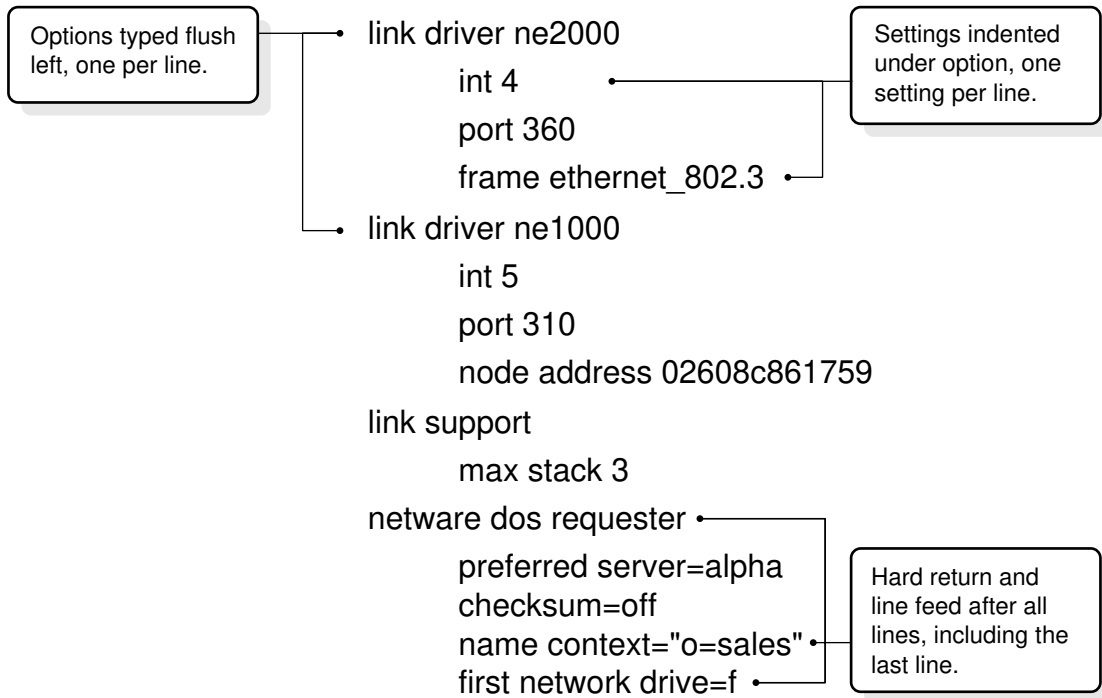
- ◆ Changes the IRQ on the link driver to 4;
- ◆ Changes the port to 340;
- ◆ Sets up F: as the first network drive when you log in to the network.

```
LINK DRIVER NE2000  
  ; Change the interrupt (IRQ) to 4  
INT #1 4  
  ; Change the port to 340 (hex)  
PORT #1 340  
  
NETWARE DOS REQUESTER  
  ; Set up F: as the first drive on network  
First Network Drive = F
```

For an illustration of the NET.CFG file format, see Figure 1-1 on page 9.



Figure 0-1  
Format of NET.CFG  
file







*chapter* **3** ***Improving Speed and Security on Your Workstation***

## Overview

This chapter explains how you can improve workstation speed by using the Packet Burst protocol and Large Internet Packets (LIP). It also explains how you can protect information on your DOS and Windows workstations. The following topics are covered in this chapter.

Topic	Page
Using Packet Burst to Increase Speed	12
Requirement for Packet Burst	12
How Packet Burst Works	12
Disabling Packet Burst	12
Using Large Internet Packets (LIP) to Increase Speed	13
How LIP Works	13
Disabling LIP	13
Using NCP Packet Signature to Improve Security	13
How NCP Packet Signature Works	14
When to Use NCP Packet Signature	14
NCP Packet Signature Options	15
Installing NCP Packet Signature	18
Troubleshooting NCP Packet Signature	19
Using Other Workstation Security Guidelines	20

## Using Packet Burst to Increase Speed

The Packet Burst protocol is designed to transmit multipacket messages efficiently over the internetwork. Packet Burst is enabled automatically in the NetWare DOS Requester.

### Requirement for Packet Burst

The Packet Burst protocol code requires about 6 K of memory. However, as a default, the NetWare DOS Requester uses ODI for Packet Burst and doesn't require additional workstation memory.

### How Packet Burst Works

At connection time, maximum burst sizes are negotiated with each server. Since Packet Burst is established with each connection, it's possible to "burst" with one server but not with another.

Once you establish a Packet Burst connection between a workstation and a NetWare server, the workstation automatically uses the Packet Burst service whenever an application requests to write more than one physical packet of data.

### Disabling Packet Burst

Packet Burst is enabled by default in the NetWare DOS Requester.

To disable Packet Burst at the workstation, add this line to the workstation's NET.CFG file under the "NetWare DOS Requester" section heading:

```
PB BUFFERS = 0
```

## Using Large Internet Packets (LIP) to Increase Speed

Large Internet Packet (LIP) functionality allows the packet size to be increased from the default of 576 bytes.

Previously, the size of packets that cross bridges or routers on NetWare networks was limited to 576 total bytes. Some network architectures (Ethernet and token ring) allow larger packets to be sent over the network.

By allowing the NetWare packet size to be increased, LIP enhances the throughput over bridges and routers if the routers aren't limited to the smaller packet size.

### How LIP Works

In previous versions of NetWare, the workstation initiated a negotiation with the NetWare server to determine an acceptable packet size.

If the NetWare server detected a router between it and the workstation, the server returned a maximum packet size of 576 bytes to the workstation.

In the current version of NetWare, the workstation still initiates packet size negotiation. However, the NetWare server doesn't return a packet size of 576 bytes when a router is detected.

Instead, the workstation negotiates with the NetWare server to agree on a packet size.

### Disabling LIP

LIP is enabled by default in the NetWare DOS Requester.

To disable LIP functionality in the workstation, enter the following line in the NET.CFG file under the "NetWare DOS Requester" section heading:

```
LARGE INTERNET PACKETS = OFF
```

## Using NCP Packet Signature to Improve Security

NCP packet signature is an enhanced security feature that protects servers and workstations using the NetWare Core Protocol™ by preventing packet forgery.

NCP packet signature is optional because the packet signature process consumes CPU resources and slows performance, both for the workstation and the NetWare server.

Without the NCP packet signature installed, a knowledgeable network workstation can pose as a more privileged workstation to send a forged NCP request to a NetWare server. By forging the proper NCP request packet, an intruder can gain rights to access all network resources.

## How NCP Packet Signature Works

NCP packet signature prevents forgery by requiring the server and the workstation to “sign” each NCP packet. The packet signature changes with every packet.

NCP packets with incorrect signatures are discarded without breaking the workstation’s connection with the server. However, an alert message about the source of the invalid packet is sent to the error log, the affected workstation, and the NetWare server console.

If NCP packet signature is installed on the server and all its workstations, it is virtually impossible to forge a valid NCP packet.

## When to Use NCP Packet Signature

NCP packet signature is not required for every installation. Some network supervisors may choose not to use NCP packet signature because they can tolerate certain security risks.

## Tolerable Security Risks

The following situations are examples of network situations that may not need NCP packet signature:

- ◆ Only executable programs reside on the server.
- ◆ All workstation users on the network are known and trusted by the supervisor.
- ◆ Data on the NetWare server is not sensitive; loss or corruption of this data will not affect operations.

## Serious Security Risks

NCP packet signature is recommended for security risks such as these:

- ◆ Unauthorized users on a workstation on the network.
- ◆ Easy physical access to the network cabling system.
- ◆ An unattended, publicly accessible workstation.

## NCP Packet Signature Options

Several signature options are available, ranging from never signing NCP packets to always signing NCP packets. NetWare servers and network workstations both have four signature levels.

The default NCP packet signature level is 1 for both workstations and NetWare servers. In general, this setting provides the most flexibility while still offering protection from forged packets.

The signature options for servers and workstations combine to determine the level of NCP packet signature on the network.



Some combinations of server and workstation packet signature levels may slow performance. However, low CPU-demand systems may not show any performance degradation.

Network supervisors can choose the packet signature level that meets both their performance needs and their security requirements.

## Workstation Packet Signature Options

Workstation signature levels are assigned by a new NET.CFG parameter:

**SIGNATURE LEVEL = *number***

Replace *number* with 0, 1, 2, or 3. The default is 1.

Number	Explanation
0	Workstation doesn't sign packets.
1	Workstation signs packets <i>only</i> if the server requests it (server option is 2 or higher).
2	Workstation signs packets if the server is capable of signing (server option is 1 or higher).
3	Workstation signs packets and requires the server to sign packets (or logging in will fail).

## Effective Packet Signature Level

The packet signature levels for the server and the workstation interact to create the “effective” packet signature. Some combinations of server and workstation levels don't allow logging in.

Table 1-1, “Effective packet signature of server and workstation,” on page 17 shows the interactive relationship between the server packet signature levels and the workstation signature levels.



Table 0-1  
Effective packet signature of server and workstation

IF	Server = 0	Server = 1	Server = 2	Server = 3
Workstation = 0	No packet signature	No packet signature	No packet signature	<i>No logging in</i>
Workstation = 1	No packet signature	No packet signature	Packet signature	Packet signature
Workstation = 2	No packet signature	Packet signature	Packet signature	Packet signature
Workstation = 3	<i>No logging in</i>	Packet signature	Packet signature	Packet signature

## Examples of Packet Signature Levels

This section includes some examples of when you would use different signature levels.

### All Information on the Server Is Sensitive

If an intruder gains access to *any* information on the NetWare server, it could damage the company.

The network supervisor sets the server to level 3 and all workstations to level 3 for maximum protection.

### Sensitive and Nonsensitive Information Reside on the Same Server

The NetWare server has a directory for executable programs and a separate directory for corporate finances (such as accounts receivable).

The network supervisor sets the server to level 2, and the workstations that need access to accounts receivable to level 3. All other workstations remain at level 1.

### Users Often Change Locations and Workstations

The network supervisor is uncertain which employees will be using which workstations, and the NetWare server contains some sensitive data.

The network supervisor sets the server to level 3. Workstations remain at level 1.

#### Workstation Is Publicly Accessible

An unattended workstation is set up for public access to non-sensitive information, but another server on the network contains sensitive information.

The network supervisor sets the sensitive server to level 3 and the unattended workstation to level 0.

## Installing NCP Packet Signature

To install NCP packet signature on a DOS or Windows workstation, add the following parameter, under the “NetWare DOS Requester” heading, to the NET.CFG file of each workstation:

**SIGNATURE LEVEL** = *number*

Replace *number* with 0, 1, 2, or 3. The default is 1.

See Table 1-1, “Effective packet signature of server and workstation,” on page 17 for a chart of the levels.

## Troubleshooting NCP Packet Signature

This section describes some solutions to problems that may be associated with using NCP packet signature.

### Workstations Are Not Signing Packets

SECURITY.VLM isn't loaded. SECURITY.VLM loads by default when the workstation signature level is set to 1.

### Workstations Cannot Log In

SECURITY.VLM isn't loaded. SECURITY.VLM loads by default when the workstation signature level is set to 1.

Make sure the packet signature levels on the server and the workstation are correct. See "Effective Packet Signature Level" on page 16.

The following situations do not allow logging in:

- ◆ Server packet signature = 3, workstation signature = 0
- ◆ Server packet signature = 0, workstation signature = 3
- ◆ The LOGIN utility is an older version that doesn't support packet signature
- ◆ The Requester or shell is an older version that doesn't support packet signature.

### "Error Receiving from the NetWork" Appears

The workstation is using a version of LOGIN.EXE file that doesn't include NCP packet signature. Make sure the new LOGIN.EXE and other new utilities are installed on all servers on the network.

### Third-Party NLMs Do Not Work

If the SET parameter "Allow Change to Workstation Rights" is set to OFF, some third-party NLMs may not function. Set this parameter to ON.

## Unsecure Workstations Log In to a Secure Server

The workstations are using an old LOGIN.EXE file that does not include NCP packet signature. Make sure the new LOGIN.EXE and other new utilities are installed on all servers on the network.

Use the “Preferred Server” option in the NET.CFG file for all workstations that have access to secure servers (level 3).

## Using Other Workstation Security Guidelines

In addition to installing NCP packet signature, network supervisors can use other NetWare security features and protective measures to keep their workstations secure.

We suggest the following security guidelines for workstations:

- ◆ Use only the most current versions of system software, workstation software, and patches.
- ◆ Check for viruses regularly.
- ◆ Use the SECURITY utility to detect vulnerable access points to the server.
- ◆ Enable intruder detection and lockout.
- ◆ Advise users to log out when their workstations are unattended.
- ◆ Secure unattended workstations.
- ◆ Require passwords of at least five characters on all accounts.
- ◆ Force password changes at least every three months.
- ◆ Require unique passwords.
- ◆ Limit the number of grace logins.
- ◆ Limit concurrent connections.
- ◆ Enforce LOGIN time restrictions and station restrictions.



chapter

# 4

## ***Running NetWare with Other Network Operating Systems***

### Overview

ODINSUP allows you to connect to disparate networks from your workstation and use them as if they were one network.

The following topics are covered in this chapter.

Topic	Page
How ODINSUP Works	24
Installing ODINSUP on DOS	24

### ODINSUP: Connectivity for Disparate Networks

ODINSUP (Open Data-link Interface/Network Driver Interface Specification Support) is an interface that allows coexistence of the two main network driver interfaces: the Network Driver Interface Specification (NDIS) and the Open Data-link Interface (ODI) Specification.

For example, after you load ODINSUP on your workstation, you can log in to a network running 3+Share® or LAN Manager™, and also log in to a NetWare network. You can then copy files and run applications as if you were on one network.

When ODINSUP is loaded, you can use a wider variety of programs without compatibility problems. You also don't have to reconfigure or reboot your workstation to switch from one type of network to another.

## Prerequisites

To complete ODINSUP installation, you must have access to the NDIS protocols and documentation. You should also have a working knowledge of the NDIS protocol.

## How ODINSUP Works

ODINSUP functions as a default protocol stack by accepting requests from the ODI Link Support Layer (LSL) that are not specifically marked for another registered protocol (such as IPX™ or TCP/IP).

When ODINSUP receives requests, it passes them to the NDIS protocol stack. With ODINSUP, the NDIS Protocol Manager can communicate with a network interface board without being aware of the details of the transmission on that board (such as frame type).

Instead, the details are handled at the ODI driver level and then transmissions are passed to the Link Support Layer, which in turn passes them to the correct protocol stack or to ODINSUP. ODINSUP translates the request for the Protocol Manager.

## Installing ODINSUP on DOS

Following are instructions for installing ODINSUP on a workstation for DOS and Windows. You must modify the CONFIG.SYS, AUTOEXEC.BAT, NET.CFG, and PROTOCOL.INI files.

### Requirements

- ◆ The ODINSUP.COM file from the *WSDOS\_1* diskette must be in the same directory as the NET.CFG file.
- ◆ The LSL.COM file must be v1.10 or above.

To check the version number, move to the directory containing the file and type

**LSL?**

LSL.COM v1.10 and v1.21 require that ODINSUP.COM be loaded from the same directory as the LSL.COM file.

- ◆ The ODI LAN drivers must be dated later than May 21, 1991.

To check the date of the drivers, type

**DIR** *drivername*

- ◆ If you are running a DOS shell with a DOS redirector or the NetWare DOS Requester, you can't run Windows in Standard or Enhanced mode.

After it is installed, ODINSUP.COM uses about 4900 bytes of DOS memory. Each additional network board that ODINSUP binds to increases memory usage by about 2800 bytes.

The installed LSL.COM and the ODI LAN driver usually take about the same amount of memory as the corresponding NDIS PROTMAN and NDIS MAC drivers. Using ODINUP.COM instead of an NDIS MAC driver means you need an additional 4900 bytes of memory, depending on the LAN driver used.

Note



ODINSUP.COM cannot be unloaded because NDIS v1.0 doesn't provide a dynamic interface for its protocols and MACs. Because it must be loaded after the LSL.COM and the ODI LAN drivers, these modules cannot be unloaded, either.

If you want to be able to unload the ODI protocol modules, load them after the NETBIND.EXE and the NDIS protocols.

## Connecting to a 3+Share Server

The instructions for installing ODINSUP on a DOS workstation that you want to connect to a 3+Share server are similar to those for installing ODINSUP on a workstation. The differences are documented in the steps that follow.

## Changing the CONFIG.SYS File

Edit the CONFIG.SYS file to

- ◆ Set the LASTDRIVE variable to indicate which drive letters can be used by redirectors and which drive letters can be used by NetWare.
- ◆ Load the NDIS Protocol Manager.

Procedure



1. Set the LASTDRIVE variable by including a line similar to the following to the CONFIG.SYS file:

**LASTDRIVE** = *drive of your selection*

NetWare uses drive letters after the LASTDRIVE variable, while redirectors use drive letters that are prior or equal to the LASTDRIVE variable.

For example, in a 3+Share network, you move to the NetWare network by typing the next drive letter after the letter you specified in LASTDRIVE. If you specified the LASTDRIVE as H:, you would type I: to go to the NetWare network.

To move from the NetWare system to the 3+Share system, you would type a drive letter prior to the one you specified in LASTDRIVE.

2. Load the NDIS Protocol Manager by adding a line similar to the following to the CONFIG.SYS file:

**DEVICE** = *path* **PROTMAN.DOS** /**I:***path*

where *I:* tells the CONFIG.SYS file to look for the PROTOCOL.INI file in the following pathname, and

where *path* refers to the directory location of the PROTOCOL.INI file.

Note



If you want to connect to a 3+Share server that uses the XNX protocol, you must also include the following line in the CONFIG.SYS file:

**DEVICE** = *path* **XNSTP.DOS**



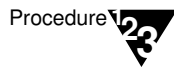
3. Remove all references to the NDIS LAN drivers from the CONFIG.SYS file.

For example, if your CONFIG.SYS file has a “DEVICE=*path* ELNKII.DOS” statement to load an ELNKII.DOS NDIS driver, you would remove it from the file.

4. Save your changes and exit the CONFIG.SYS file.

## Changing the AUTOEXEC.BAT File

Edit the AUTOEXEC.BAT file to load the Link Support Layer (LSL), the ODI LAN driver, the ODINSUP protocol, the protocol stacks, and the NetWare DOS Requester.



1. Put a “Change Directory” (CD) statement in the AUTOEXEC.BAT file to the directory where the ODINSUP.COM and NET.CFG files are located.

See the example in “Step 2.” The ODINSUP protocol cannot execute unless it can access the NET.CFG file.

2. Add the following lines to load the ODI components.

You must load LSL and the ODI LAN driver before you load the ODINSUP protocol.

For a DOS workstation, use the following:

```
CD path  
LSL ODI LAN driver (such as 3C503)  
ODINSUP  
NETBIND  
NETBIOS  
IPXODI  
VLM
```

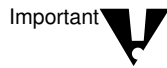
Replace *path* with the directory where the ODINSUP.COM and NET.CFG files are found.

For a DOS workstation to connect to a 3+Share server, use the following:

```
CD path
LSL ODI LAN driver (such as 3C503.COM)
ODINSUP
NETBIND
XNSTP
NETBIOS
MINSES12
MSREDIR
SETNAME $$3COM$$
3CLOGIN username
IPXODI A
VLM
next available drive
```

Replace *path* with the directory where the ODINSUP.COM and NET.CFG files are found, and replace *next available drive* with the next drive available after the LASTDRIVE setting.

For example, if LASTDRIVE=H:, the next available drive is I:. After the NetWare DOS Requester is loaded, log in to the NetWare server by accessing the drive indicated by LASTDRIVE + 1.



Use XNSTP if you are using XNS. Use IPXODI to load IPX only, not SPX.

3. Save your changes and exit the AUTOEXEC.BAT file.

## Changing the NET.CFG File

Edit the NET.CFG file to

- ◆ Enable the required Ethernet and token-ring frame types. (ODINSUP supports only ODI LAN drivers that are compatible with Ethernet and token ring.)
- ◆ Bind the ODINSUP protocol to a particular ODI driver.



1. Enable the frame types that ODINSUP requires.

If you are using token-ring network boards, type the following lines in the NET.CFG, substituting the name of your ODI LAN driver for *drivername*:

```
LINK DRIVER drivername  
FRAME TOKEN-RING  
FRAME TOKEN-RING_SNAP
```

(Indent the lines beginning with *frame* and type them in the order shown.)

If you are using Ethernet network boards, type the following lines in the NET.CFG, substituting the name of your ODI LAN driver for *drivername*:

```
LINK DRIVER drivername  
FRAME ETHERNET_802.3  
FRAME ETHERNET_II  
FRAME ETHERNET_802.2  
FARME ETHERNET_SNAP
```

2. Bind ODINSUP to one or more ODI drivers.

Do this by using the “Protocol” option with the name of the ODI driver that is used with your network board.

For example, if you have one NE2000 network board in your workstation, you would type

```
PROTOCOL ODINSUP  
BIND NE2000
```

If you don't bind ODINSUP to an ODI driver, ODINSUP searches for any Ethernet or token-ring drivers that are loaded. ODINSUP binds to and uses the first driver of these types that it locates.

When ODINSUP binds to a driver, the network board for that driver is the board used for transmissions to and from the network.

If you have two or more network interface boards of the same type in your workstation, ODINSUP binds only to the driver for the first board unless you specify differently.

You can specify exactly which driver to bind to, or you can bind ODINSUP to multiple drivers by typing an “instance” number with the “Protocol” option.

For example, if you have two NE2000 network interface boards in your workstation, you could bind ODINSUP to each board's driver by typing a 2 for the second board:

```
PROTOCOL ODINSUP  
  BIND NE2000  
  BIND NE2000 2
```

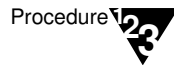
ODINSUP can bind to a maximum of four ODI drivers.

3. Save your changes and exit the NET.CFG file.

## Changing the PROTOCOL.INI File

Edit the PROTOCOL.INI file to

- ◆ Bind the NDIS protocol stack to the ODI drivers.
- ◆ Remove NDIS-related information.



1. Bind each ODI driver you are using to the NDIS protocol, using the “BINDINGS = *drivername*” parameter.

For example, to bind an NE2000 to the XNS NDIS protocol, you would modify the existing statement to read as follows:

```
[XNS]  
  
.  
.  
BINDINGS = NE2000
```

To bind an NDIS protocol to more than one ODI driver, type both driver names on the same line, separated by a comma.

For example, to bind the Etherand protocol to both an NE2000 driver and an NE1000™ driver, you would type the following:

```
[ETHERAND]
.
.
BINDINGS = NE2000, NE1000
```

Following is a sample PROTOCOL.INI file showing binding instructions for a DOS workstation connecting to a 3+Share server using the XNS protocol:

```
[PROTOCOL_MANAGER]
DRIVERNAME = PROTMAN$

[XNS]
DRIVERNAME = XNSTP$
BINDINGS = NE2000
```



In the PROTOCOL.INI file, driver names can't start with a number. Put an X in front of 3Com drivers and other drivers that start with a number.

To bind an NDIS driver to an instance of an ODI driver other than the first instance, type the instance number at the end of the driver name (for example, NE20004). Don't put a space between the driver name and the instance number.

If you don't know the name of the ODI driver you are using, reboot your machine and read the startup messages carefully. The driver name you should use is displayed.

2. (Optional) Remove all NDIS driver-specific information from the PROTOCOL.INI file.

ODINSUP doesn't require this information.

3. Save your changes and exit the PROTOCOL.INI file.
4. Reboot your machine for changes to CONFIG.SYS, NET.CFG, and PROTOCOL.INI to take effect.





chapter

# 5

## ***Setting Up Additional Protocol Support***

### Overview

This chapter explains how to set up the workstation to use the Named Pipes protocol.

The following topics are covered in this chapter.

Topic	Page
How to Use the Named Pipes Extender for DOS	33
Loading the Extender into Memory	34
Unloading the Extender from Memory	34
Loading the Extender Automatically	35
Configuring the Named Pipes Extender	35

### How to Use the Named Pipes Extender for DOS

The Named Pipes extender for DOS is a terminate-and-stay-resident (TSR) program that extends the capability of DOS to include use of remote Named Pipes. You must have an OS/2® Named Pipes server to use this extender.

This section explains how to install and configure the extender.

## Prerequisite

The workstation on which you install the extender must be running DOS v3.3 or above.

## Loading the Extender into Memory

Your OS/2 workstation may require up to a minute after you run the extender for DOS before it is ready to run a Named Pipes application.

To load the extender, complete these steps:

Procedure



1. Insert the **WSDOS\_1** diskette into drive A:.
2. Change to drive A:.
3. Type the following:

```
DOSNP <Enter>
```

The extender is now loaded in memory.

4. Remove the diskette.
5. If you want a report of the status of the Named Pipes extender, type the following at the DOS command line:

```
DOSNP /I <Enter>
```

## Unloading the Extender from Memory

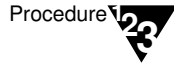
To unload the extender, type the following at the DOS command line:

```
DOSNP /U <Enter>
```



## Loading the Extender Automatically

To have the extender loaded into memory each time you boot your workstation, follow these steps:



1. Insert the **WSDOS\_1** diskette into drive A:.
2. Copy the following file to the boot drive (hard drive or floppy diskette):

**Copy A:\DOSNP.EXE**

3. Add a line to your AUTOEXEC.BAT file that executes the copy of DOSNP.EXE that you put on the boot drive.

This line should be placed *after* the line that loads IPX.COM and *before* the line that loads the NetWare DOS Requester (usually called VLM.EXE).

4. Remove the diskette.

Each time you boot the workstation, the extender is loaded into memory.

## Configuring the Named Pipes Extender

You can use the NET.CFG file to specify various options of the Named Pipes extender.

The following characteristics of the extender can be specified in NET.CFG:

- ◆ Maximum machine names
- ◆ Maximum open Named Pipes
- ◆ Maximum communication buffers

These parameters are explained in the following sections.

## Maximum Machine Names

### Description

Controls the number of Named Pipe servers with which the extender can communicate.

### Syntax

**NP MAX MACHINE NAMES = *number***

Replace *number* with the number of Named Pipe servers cached on the DOS machine.

Range: 4 to 50

Default: 10

### Example

The following line in the NET.CFG file changes the number of Named Pipes servers with which the workstation can communicate from 10 to 14:

**NP MAX MACHINE NAMES = 14**

## Maximum Open Named Pipes

### Description

Specifies the maximum number of Named Pipes the workstation can have open simultaneously.

### Syntax

**NP MAX OPEN NAMED PIPES = *number***

Replace *number* with the maximum number of Named Pipes that can be open simultaneously.

Range: 4 to 128

Default: 4

### Example

The following line in the NET.CFG file changes the maximum number of pipes that the workstation can have open simultaneously from the default to 6:

```
NP MAX OPEN NAMED PIPES = 6
```

## Maximum Communication Buffers

### Description

Specifies the number of communication buffers that the extender can use to transmit data to and receive data from the Named Pipes server.

### Syntax

```
NP MAX COMM BUFFERS = number
```

Replace *number* with the maximum number of communication buffers the extender can use to communicate with the Named Pipes server.

Range: 4 to 40

Default: 6

### Example

The following line in the NET.CFG file changes the number of communication buffers that the extender can use to communicate with the Named Pipes server from 6 to 10:

```
NP MAX COMM BUFFERS = 10
```





chapter

# 6

## **Backing Up the DOS Workstation**

### Overview

This chapter describes how to back up the information on your workstation's local disk drives to a NetWare server. The information provided is for use at the server console and workstation. See *Server Backup* for information and procedures.

The following topics are covered in this chapter.

Topic	Page
TSA_SMS Features	39
Supported Storage Devices and Drivers	40
Backing Up Your Workstation	40
Command Line Options	42

### TSA\_SMS Features

The TSA\_SMS has the following features:

- ◆ Several levels of security
- ◆ Workstation TSR that runs in the background, allowing the user to use other applications
- ◆ Workstation TSR that occupies less than 7 K if the system defaults are used
- ◆ Support for MS DOS® (v3.3 and above) and DR DOS® (v5.0 and above)

- ◆ An optional switch that increases the number of read buffers (this increases TSR size) to improve backup performance.
- ◆ An optional switch that allows users to specify stack size to reduce the overall TSR size.

## Supported Storage Devices and Drivers

NetWare currently supports 1/4", 4mm, and 8mm tape technologies, and the list of supported devices continues to grow.

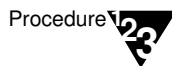
There are several ways to get information about which devices are supported:

- ◆ You can call your local NetWare reseller or consultant, your local Novell office, or 1-800-NETWARE (in the United States) for the list of currently supported devices.
- ◆ You can download a list of currently supported devices on NetWare<sup>SM</sup>. See your authorized dealer for more information.

## Backing Up Your Workstation

To back up your workstation, you must load the following files in the order listed:

- ◆ Driver for controller and storage device
- ◆ SBACKUP and TSA files



### 1. Load the device drivers.

Refer to the manufacturer's instructions.

If the SCSI controller and its device driver are ASPI compatible, you can use TAPEDAI.DSK to run a tape device connected to the controller. For more information, see "Supported Storage Devices and Drivers" on page 40.

To load TAPEDAI.DSK, at the server console prompt, type

**LOAD TAPEDAI**

2. Load TSA\_DOS at the host server. Type

**LOAD TSA\_DOS**

If the workstation user specified the "Password" option while loading the files, the person performing the backup must know the workstation password to back up any data that resides on that workstation.

3. Load TSA\_SMS, from the **WSDOS\_1** diskette, at the workstation that you want to back up. Type

**TSA\_SMS [options]**

See "TSA\_SMS command line options" Table 1-2, on page 42 for a description of options you can use.

4. Load SBACKUP at the NetWare host server. Type

**SBACKUP [options]**

See "SBACKUP command line options" Table 1-1, on page 42 for options you can use.

## Command Line Options

SBACKUP and TSA\_SMS have options that you can use at the command line.

### SBACKUP Options

Table 1-1 shows the range of values for SBACKUP options and their default values.

Table 0-1  
SBACKUP command line options

Option	Range
Buffer Size	16, 32, 64, 128, or 256 KB (Default: 64 KB)
Number of Buffers	2 to 10 (Default: 4)

When using SBACKUP command line options, the syntax is

**LOAD SBACKUP SIZE=XXX BUFFER=X <Enter>**

### TSA\_SMS Options

Table 1-2 lists and explains the TSA\_SMS options.

Table 0-2  
TSA\_SMS command line options

Option	Syntax	Explanation
Help	/H	Displays all options.
Buffers	/B= <i>n</i>	TSA 1 K buffers ( <i>n</i> = 2 through 30). The default is 1. Increasing the number increases throughput speed, but requires more RAM on the workstation.



Table 0-2 *continued*  
TSA\_SMS command line options

Option	Syntax	Explanation
Drive	<i>/D=x</i>	Indicates the DOS drives where data resides that you want to back up, or to which you want to restore data. Replace <i>x</i> with the drive letter. Don't type a colon after the drive letter. Example: <i>/D = CE</i> where "C" stands for the C: drive and "E" stands for the E: drive.
Name	<i>/N=workstation name</i>	Sets the workstation's unique name. The limit is 10 characters. You need to specify this option only after NetWare Directory Services for NetWare v 4.0 networks has been initialized (or reinitialized) on a server.
Password	<i>/P=password</i>	Sets a password for the workstation. If you use this option, anyone doing a backup must know the password. The "Trust" option can be used instead.
Remove	<i>/R=server name, workstation name</i>	Removes the workstation address from NetWare Directory Services.
Server	<i>/SE=server name</i>	Specifies the name of the server you want this workstation to connect to for backup and restore services.
Stack	<i>/ST=n</i>	Specifies the stack size represented as a decimal (512 through 4096 bytes). The default is 2048 bytes. Do not change this number unless RAM is extremely limited or you receive "Stack Overflow" messages.

Table 0-2 *continued*

TSA\_SMS command line options

Option	Syntax	Explanation
Trust	/T	If you use this option, anyone doing a backup must have SUPERVISOR-equivalent rights on the server where TSA_DOS resides. This option can be used instead of the "Password" option.
Unload	/U	Unloads TSA_SMS from the workstation's memory.

Example

Following is an example of a command that you could execute at a DOS workstation:

**TSA\_SMS /SE=SALES /T /D=C /B=30 /N=LISA**

This command indicates the following:

- ◆ TSA\_SMS is the .COM filename.
- ◆ SALES is the server name.
- ◆ The "Trust" option is set.
- ◆ The drive is set to drive C:.
- ◆ There are 30 TSA buffers (1 KB each).
- ◆ LISA is the workstation name.



chapter **7** **Using Your DOS Workstation on a  
Token-Ring Network**

## Overview

This section explains the use of the IBM Token-Ring Source Routing Driver with DOS ODI.

The following topics are covered in this chapter.

Topic	Page
How to Use the Token-Ring Source Routing Driver	45
Installing the Source Routing Driver with DOS ODI	47
Setting Alternate Parameters for ROUTE.COM	47
Sample ROUTE.COM	51

## How to Use the Token-Ring Source Routing Driver

The IBM Token-Ring Source Routing Driver enables communication across IBM Token-Ring network bridges. Any type of DOS ODI protocol stack can use this source routing feature to communicate across IBM Token-Ring network bridges.

All nodes that need to communicate across a source route bridge must be running the IBM Token-Ring Source Routing Driver.

Figure 1-1 on page 46 shows an example network configuration using IBM source routing bridges.

Figure 0-1  
Network configuration  
using IBM source  
routing bridges

LAN Drivers	Options												
	DMA	INT	MEM	PORT	NODE ADDRESS	SLOT	FRAME	PROTOCOL	SAPS	LINK STATIONS	ALTERNATE	MAX FRAME SIZE	DOUBLE BUFFER OFF
EXOS		I	I	I	■	M	■	■					
LANSUP							■	■	■	■	■	■	
NE2					■	■	■	■					
NE2_32					■	■	■	■					
NE1000		■		■	■		■	■					
NE2000		■		■	■		■	■					
NE2100	■	■		■	■		■	■					
NE3200						E	■	■					E
NTR2000					I		I	I			I	I	
PCN2L					■	M	■	■			■		
TOKEN					■	M	■	■			■	■	
TRXNET		I	I	I		M		■					

E = EISA, extended industry architecture  
M = MCA, microchannel architecture  
I = ISA, industry standard architecture

## Installing the Source Routing Driver with DOS ODI

To install source routing on workstations, complete the following steps.

Procedure



1. Copy the ROUTE.COM file from the **WSDOS\_1** diskette to the boot disk.

The command to load ROUTE.COM should be added after LANSUP.COM or TOKEN.COM, but before the protocol stack you are using (for example, IPXODI.COM).

Note



If you are using ROUTE.COM with remote boot, you must load ROUTE.COM before the LAN driver in the AUTOEXEC.BAT. See "LAN Driver" on page A-83 for more information.

2. Set the parameters for ROUTE.COM.

The default parameters for ROUTE.COM can be used with most network communications.

For additional ROUTE.COM parameters, see "Setting Alternate Parameters" below for ROUTE.COM. Add the parameters at the command line.

Note



The ROUTE.COM module can be removed from memory by specifying the u command line switch, as in the following example:

```
ROUTE /U
```

## Setting Alternate Parameters for ROUTE.COM

The following parameters can be entered when ROUTE.COM is first loaded:

```
BOARD = number
CLEAR
DEF
GBR
MBR
NODES = number
REMOVE = number
```

The parameters can also be changed by loading ROUTE.COM a second time. For online help, type

```
ROUTE ? <Enter>
```

Most of the parameters have default values that should work with simple configurations for IBM bridges. If you have installed parallel IBM bridges, you can change some of the parameters to reduce traffic on some of the paths. Each parameter is described below.

For an example of a ROUTE.COM file, see "Sample ROUTE.COM" on page 51.

**BOARD = *number***

Specifies a board number for the network board.

If this parameter is not specified, the default of 1 is used.

If you have loaded more than one LAN driver in the workstation, the board number is determined by the order in which the LAN drivers are loaded (the first board is 1). See your batch file for the order you specified.

Load ROUTE.COM for each logical board number (frame type) enabled by the Token-Ring LAN driver.

**CLEAR**

Clears the Source Routing table and forces a dynamic rebuilding of the table by sending a default frame to each node in the network.

Use this parameter when an IBM bridge on the network has gone down and an alternate route is available.

## DEF

Prevents frames that have unknown (default) destination addresses from being sent across Single Route IBM bridges.

- ◆ If this parameter is specified, these frames are forwarded as All Routes Broadcast frames.
- ◆ If this parameter is not specified, all frames that have addresses not in the workstation's Source Routing table are forwarded as Single Route Broadcast frames.
- ◆ If ROUTE.COM has already been loaded with the DEF parameter, reloading ROUTE.COM with this parameter broadcasts all subsequent Single Route Broadcast frames as All Routes Broadcast frames.

## GBR

Specifies that all General Broadcast frames are to be sent as All Routes Broadcast frames.

- ◆ If this parameter is not specified when ROUTE.COM is loaded, all General Broadcast frames are broadcast as Single Route Broadcast frames.
- ◆ If ROUTE.COM has already been loaded with the GBR parameter, reloading ROUTE.COM with this parameter broadcasts all subsequent General Broadcast frames as All Routes Broadcast frames.

## MBR

Specifies that all Multicast Broadcast frames are to be sent as All Routes Broadcast frames.

- ◆ If the parameter is not specified when ROUTE.COM is loaded, all Multicast Broadcast frames are broadcast as Single Route Broadcast frames.
- ◆ If ROUTE.COM has already been loaded with the MBR parameter, reloading ROUTE.COM with this parameter broadcasts all subsequent Multicast Broadcast frames as All Routes Broadcast frames.

NODES = *number*

Specifies the number of table entries in the Source Routing table.

- ◆ You must enter this parameter the first time ROUTE.COM is loaded.
- ◆ The parameter can't be changed by reloading ROUTE.COM.
- ◆ Replace *number* with a value from 8 to 255. The default is 16.

REMOVE = *number*

Removes a node address from the workstation's Source Routing table.

- ◆ Use the parameter when a bridge has gone down.
- ◆ Removing the node from the Source Routing table forces the workstation to determine an alternate path.
- ◆ Replace *number* with a 12-digit (6-byte) hexadecimal number.

If fewer than 9 digits are entered, ROUTE.COM prefixes the address with 4000h. For example, REMOVE=2 is interpreted as REMOVE=400000000002.



## Sample ROUTE.COM

The following command shows one possible ROUTE.COM setting:

```
ROUTE BOARD=3, DEF, GBR, MBR
```

If you set these parameters, ROUTE.COM would do the following:

- ◆ Send packets through logical board 3
- ◆ Send all frames that are not addressed in the workstation's source routing table as All Routes Broadcast
- ◆ Send General Broadcast frames as All Routes Broadcast
- ◆ Send Multicast Broadcast frames as All Routes Broadcast

When the NODES parameter is not set, ROUTE.COM assumes the default of 16.

The CLEAR and REMOVE parameters are not needed for the initial load of ROUTE.COM. However, if a bridge goes down, you can reload ROUTE.COM with these parameters to reconfigure the Source Routing table.

Note



For more information on using source routing, see the *IBM Token-Ring Network Architecture Reference* manual.





chapter

# 8

## ***Running Multiple Applications in a Task-Switching Environment***

### Overview

NetWare's task-switching files provide the data buffers needed to support IPX and SPX requests made from applications running in a DOS session.

Note



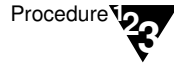
If the only IPX/SPX™ application you are using is the NetWare DOS Requester, you don't need to load the task-switching files.

The following topics are covered in this chapter.

Topic	Page
Task Switching in DOS	54
Task Switching in Windows	54
Using Batch Files with TBMI2	56
Using Command Line Parameters with TBMI2 or TASKID	57
Troubleshooting TBMI2	58

## Task Switching in DOS

To use NetWare's task-switching files in DOS, do the following:



1. Copy TBMI2.COM from the **WSWIN\_1** diskette to any directory.  
You must be able to run TBMI2 from this directory later.
2. From the directory where TBMI2 is located, type  
  
**TBMI2** <Enter>
3. Start MS-DOS 5.0 DOSSHELL or DR DOS 6.0 TASKMAX normally.

## Task Switching in Windows

NetWare's task-switching files help non-Windows IPX and SPX programs work in a multitasking environment.

You *must* use the task-switching files if

- ◆ You will be switching between DOS sessions, *and*
- ◆ Your application bypasses the NetWare shell (NETX) and accesses IPX or SPX directly, *and*
- ◆ You are running in standard or real mode.

If you are running Windows 3.0, see "Task Switching with Windows 3.0" on page 55. If you are running Windows 3.1, see "Task Switching with Windows 3.1" on page 56.

*Do not* use the task-switching files if

- ◆ You will not be switching between DOS sessions, *or*
- ◆ You are running in enhanced mode, *or*
- ◆ Your application goes through the NetWare shell (NETX) to access IPX or SPX.

If your application requires task-switching files and you don't use them, the session fails and your workstation may hang.



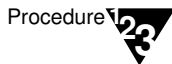
If you aren't sure your application needs task-switching support, go ahead and run the task-switching files; they use only a small amount of memory.

After running the application for a period of time, enter the command line parameter /D and look at the number in the "Far Call Usage" field. If this number is zero, your application has not used the task-switching files; you can run it without them.

## Task Switching with Windows 3.0

The task-switching files for Windows v3.0 include TBMI2.COM and TASKID.COM.

Load TBMI2 at the command line *before* you begin Windows; load TASKID in Windows after opening a DOS prompt.



1. Expand and copy TBMI2.COM and TASKID.COM from the **WSWIN\_1** diskette to any directory.

To expand a file, type

```
UNPACK SOURCE: filename DESTINATION: filename
```

For example, to expand TBMI2.COM, type

```
UNPACK A:TBMI2.CO_ C:TBMI2.COM
```

2. From the directory where TBMI2 is located, type

```
TBMI2 <Enter>
```

3. Start Windows.
4. Start a DOS session.

5. At the new DOS prompt, load TASKID. Type

```
TASKID <Enter>
```

For each DOS prompt you open, repeat Step 5 before running an application from that prompt.

6. When you close a DOS session with the EXIT command, unload TASKID by typing "TASKID /U <Enter>."

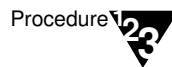


If you don't unload TASKID before you close the session, your computer may hang. You do not need to unload TBMI2 after you exit Windows unless you want to free memory.

## Task Switching with Windows 3.1

The task-switching file for Windows 3.1 is TBMI2.COM.

Load TBMI2 at the command line *before* you begin Windows.



1. Expand and copy TBMI2.COM from the **WSWIN\_1** diskette to any directory.

To expand a file, type

```
UNPACK SOURCE: filename DESTINATION: filename
```

For example, to expand TBMI2.COM, type

```
UNPACK A:TBMI2.CO_ C:TBMI2.COM
```

2. From the directory where TBMI2 is located, type

```
TBMI2 <Enter>
```

3. Start Windows.

## Using Batch Files with TBMI2

You can include TBMI2 in a batch file to ensure that it is started before entering a task-switching environment and that it is unloaded afterwards.

For example, the batch file could include the following:

```
TBMI2  
DOSSHELL (or TASKMAX or WIN)  
TBMI2 /U
```

## Using Command Line Parameters with TBMI2 or TASKID

You can use any of the following command line parameters with TBMI2 or TASKID.

Table 0-1  
TBMI2 or TASKID command line options

Parameter	Syntax	Explanation
Help	<i>/?</i> or <i>/H</i>	Displays help or usage information.
Configuration File	<i>/Cfilename</i>	Specifies an alternate configuration file TBMI2 should use.  The filename is NET.CFG by default. Do not put a space between <i>/C</i> and the filename. For example, you would type "TBMI2/ <i>CTBMI2.CFG</i> " at the DOS prompt.  For information about configuration file parameters, see "Configuring Your Workstation" on page 5.
Diagnostic Information	<i>/D</i>	Displays diagnostic information.
Version Information	<i>/I</i>	Displays version information.
Unload	<i>/U</i>	Unloads TBMI2 after you exit Windows.

## Troubleshooting TBMI2

If you have problems while using TBMI2, you may need to change configuration parameters in the NET.CFG file. For an explanation of the NET.CFG file, see “Configuring Your Workstation” on page 5.

View configuration parameters by typing “TBMI2 /D” at the command line.

Check the values associated with

- ◆ “Max Buffers Used,” which tells you how many buffers are used.
- ◆ “Configured Data ECBs,” which tells you how many buffers are available.

If the number of buffers used approaches or equals the number of buffers available, increase the number of buffers available using the “ECB Count” and “Data ECB Count” parameters in the configuration file.

- ◆ Also check the “Unavail Buffer Count.”

If it is more than zero, increase the number of buffers available using the “ECB Count” and “Data ECB Count” parameters in the NET.CFG file.



The COMCHECK and RCONSOLE utilities use too many buffers to be used with TBMI2. To use either of these utilities under Windows, don't load TBMI2 and don't switch out of the DOS session running the utility before exiting the utility.





chapter

# 9

## ***Booting DOS Workstations from the Network***

### Overview

This chapter explains how to set up the network so that workstations (mostly diskless workstations) can boot from the Remote Boot disk image files stored on the NetWare server's hard disk.

The following topics are covered in this chapter.

Topic	Page
About RPL.NLM	60
Features of RPL.NLM	60
Running RPL with Enhanced Remote Boot PROMS	61
Running RPL with Older Remote Boot ROMS	70
Running RPLFIX.COM for Older ROMs with DOS Versions Above 5.x	70
How to Use RPLFIX	71
Troubleshooting RPL with Older ROMS	71

## About RPL.NLM

RPL.NLM is a NetWare Loadable Module™ that acts as a protocol stack and responds to the IBM architected Remote Program Load (RPL) frames as defined in the *IBM Remote Program Load User's Guide*.

RPL.NLM is used in networks that have diskless workstations installed with the RPL BIOS module. Currently, RPL is supported on the following network adapters:

- ◆ IBM Ethernet adapters (MCA and Model 25SX adapters)
- ◆ IBM PC Network adapters
- ◆ IBM Token-Ring Network adapters
- ◆ Novell adapters and the OEM adapters with the Enhanced Remote Boot PROM installed

Specifically, RPL.NLM responds to the frames as shown in Table 1-1.

Table 0-1  
RPL frame response

RPL.FRAME	RPL.NLM Response
FIND	FOUND
SEND.FILE.REQUEST	FILE.DATA.RESPONSE

## Features of RPL.NLM

RPL.NLM comes with five bootstrap programs:

- ◆ ETHER.RPL for networks using the IBM Ethernet adapter.
- ◆ F1ETH.RPL for IBM adapters.
- ◆ PCN2L.RPL for networks using the IBM PC Network adapter.
- ◆ TOKEN.RPL for networks using the IBM Token-Ring adapter.
- ◆ RBOOT.RPL for Novell adapters and OEM adapters with the Enhanced Remote Boot PROM.

Together, these files offer the following features and fixes:

- ◆ The ability to use BOOTCONF.SYS wildcard characters (\* and ?) in specifying node addresses. Also, more than one disk image file name is allowed per node address. BOOTCONF.SYS is parsed by RPL.NLM at the NetWare server to minimize the amount of network traffic.

See “RPL Parameters and Extensions” on page 141 for a complete description of how to use these features.

- ◆ The ability to boot across a source routing bridge.

## Running RPL with Enhanced Remote Boot PROMS

### What You Need

To install the disk image files on the NetWare server, you need the following:

- ◆ A worksheet to record settings for node addresses and network addresses
- ◆ A workstation with a floppy or hard disk drive
- ◆ The boot diskettes for each remote boot workstation

## 1. Install RPL.NLM on the NetWare Server

Use the SYS:\SYSTEM directory. Type

**LOAD RPL**

at the server console.

## 2. Ensure Bootstrap Program Files Are in the SYS:\LOGIN Directory

The following bootstrap programs come with RPL.NLM and should be installed in the SYS:\LOGIN directory of the NetWare server:

Bootstrap program	Adapter type
ETHER.RPL	IBM MCA Ethernet
F1ETH.RPL	IBM Model 25SX Ethernet
PCN2L.RPL	IBM PC Network
TOKEN.RPL	IBM Token-Ring Network
RBOOT.RPL	Adapters using Novell Boot ROM Kit

## 3. Bind RPL.NLM to the Network Board

Since RPL.NLM is a protocol stack, it must be bound to all network boards that have RPL clients attached to them.

To bind the board, use this syntax:

**BIND RPL TO BOARD**  
[ACK], [FRAME=*ff*], [GNS], [NODEFAULT],  
[PROTECT], [PS=*server*], [TRO], [WAIT  
TIME=*sss*]

where BOARD is the name of any NetWare LAN driver that is configured for IEEE 802.2 frame type.

See "RPL.NLM BIND Time Parameters" on page 141 for a complete description of the parameters you can use.

## 4. Create a Single or Multiple Remote Boot Disk Image File(s)

If you want to create a single Remote Boot disk image file for a single workstation, see “Creating a Single Remote Boot Disk Image File” on page 63.

If you need to create multiple Remote Boot disk image files for workstations using different network boards or operating environments, see “Creating Multiple Remote Boot Disk Image Files” on page 65.

### Creating a Single Remote Boot Disk Image File

Procedure



1. Boot a workstation from floppy or hard disk, and log in as SUPERVISOR.
2. Insert the boot diskette for the Remote Boot workstation into drive A:.
3. Map drive F: to SYS:SYSTEM. Type  
**MAP F:=SYS:SYSTEM** <Enter>
4. Map drive G: to SYS:LOGIN. Type  
**MAP G:=SYS:LOGIN** <Enter>
5. Change to the SYS:LOGIN directory. Type  
**G:** <Enter>
6. Run DOSGEN. Type  
**F :DOSGEN** <Enter>

Your screen will look similar to Figure 1-1.

Figure 0-1  
Single disk image file  
screen

```
Floppy Type f9 = Quad Density, 15 Sectors per track
Total Floppy Space 2400 Sectors
Setting Up System Block
Setting Up FAT Tables
Setting Up Directory Structures
Traversing Directory Structures
Processing IO SYS
Processing MSDOS COM
Processing COMMAND COM
Processing IPXODI COM
Processing LSL COM
Processing NE1000 COM
Processing ULM
Processing NET CFG
Processing AUTOEXEC BAT
Transferring Data to NET$DOS.SYS
```

DOSGEN creates a disk image file called NET\$DOS.SYS (a copy of the files on the boot diskette) in the SYS:LOGIN directory.

7. Copy the AUTOEXEC.BAT file from the boot diskette into the SYS:LOGIN directory and into the default directory specified in the user's login script.

You may get a "Batch file missing" error when you log in if the AUTOEXEC.BAT file is not copied to *both* the SYS:LOGIN directory and the default directory.

8. Flag the NET\$DOS.SYS file in SYS:LOGIN as Shareable. Type

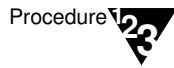
```
FLAG NET$DOS.SYS S <Enter>
```

Flagging the NET\$DOS.SYS file Shareable insures that the file is not locked by another workstation when booting.

## Creating Multiple Remote Boot Disk Image Files

You must create different boot image files on the NetWare server for each Remote Boot workstation that has different boot file requirements.

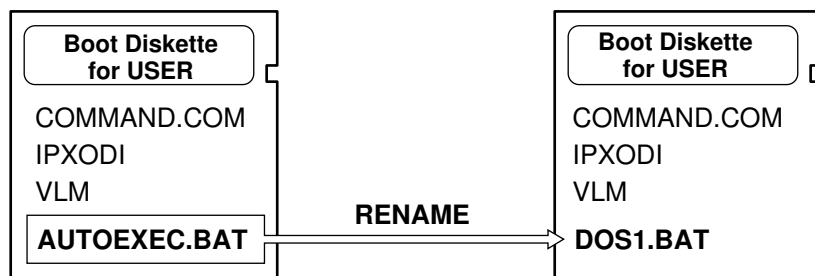
For every boot image file, rename NET\$DOS.SYS file to a unique name.



1. Boot a workstation from a floppy or hard disk, and log in as SUPERVISOR.
2. Rename the AUTOEXEC.BAT file on each boot diskette to a unique name with a .BAT extension. See Figure 1-2.

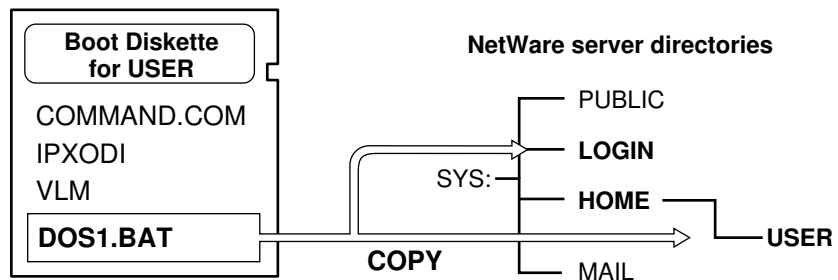
List this information, with the network addresses and node addresses, on your worksheet. You need this information to create the BOOTCONF.SYS file.

Figure 0-2  
Rename the  
AUTOEXEC.BAT file



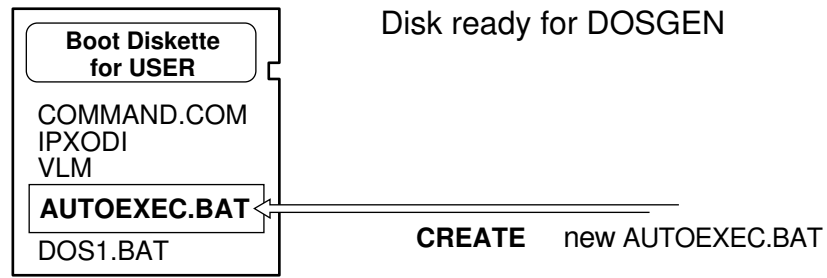
3. Copy the renamed .BAT file (DOS1.BAT, in this example) from the boot diskette into the SYS:LOGIN directory. See Figure 1-3.

Figure 0-3  
Copy the renamed  
.BAT file



4. Copy the renamed .BAT file (DOS1.BAT) into the default directory specified in the login script.
5. Create a new AUTOEXEC.BAT file on each boot diskette to execute the renamed batch file. See Figure 1-4.

Figure 0-4  
Create a new  
AUTOEXEC.BAT file



6. Map drives to the directories that DOSGEN writes to.
  - 6a. Map drive F: to SYS:SYSTEM:
 

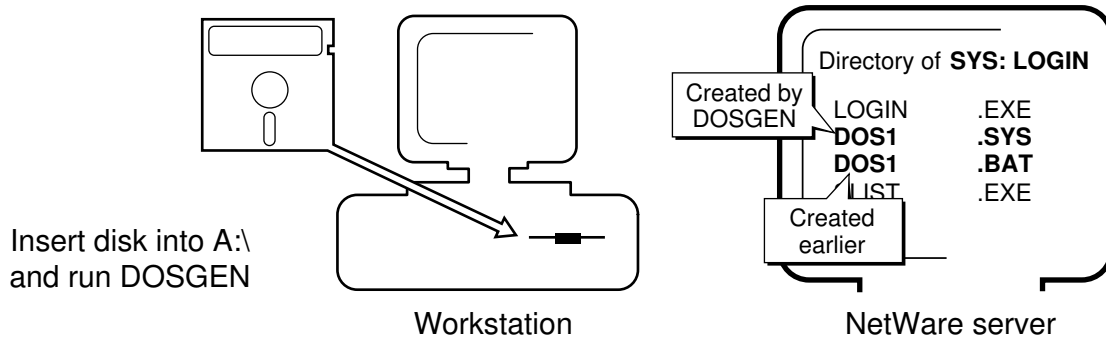
```
MAP F: =SYS:SYSTEM <Enter>
```
  - 6b. Map drive G: to SYS:LOGIN:
 

```
MAP G: =SYS:LOGIN <Enter>
```
  - 6c. Change to the SYS:LOGIN directory:
 

```
G: <Enter>
```
7. Use DOSGEN from the *WSDOS\_1* diskette to create a uniquely named .SYS file in the SYS:LOGIN directory for each boot diskette. See Figure 1-5.



Figure 0-5  
Run DOSGEN to  
create a .SYS file



7a. From drive G: (and with the boot diskette in drive A:), type a command similar to the following:

```
F:DOSGEN A: DOS1.SYS <Enter>
```

DOSGEN creates the new disk image file in the SYS:LOGIN directory. Your screen will look similar to Figure 1-6.

Figure 0-6  
Multiple disk image  
file screen

```
Floppy Type f9 = Quad Density, 15 Sectors per track
Total Floppy Space 2400 Sectors
Setting up System Block
Setting Up FAT Tables
Setting Up Directory Structures
Traversing Directory Structures
Processing IO SYS
Processing MSDOS COM
Processing COMMAND COM
Processing IPXODI COM
Processing LSL COM
Processing NE1000 COM
Processing VLM
Processing NET CFG
Processing AUTOEXEC BAT
Transferring Data to DOS1.SYS
```

7b. Record the network number and node address of the workstation that will use the disk image file you just created.

You'll use this information when you create the BOOTCONF.SYS file.

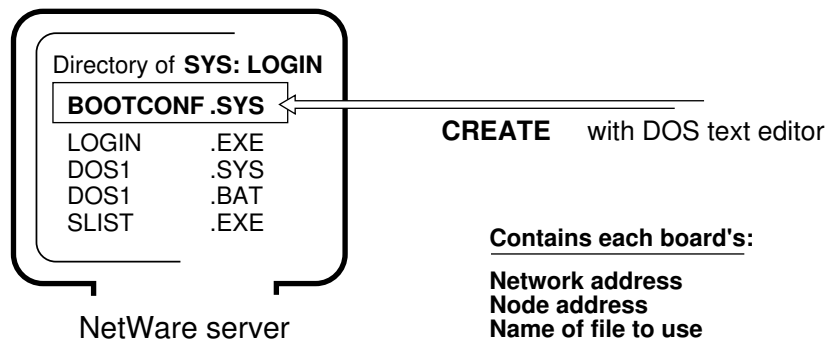
When you have finished running DOSGEN for two boot diskettes, your list will look similar to the following:

```
DOS1.SYS: Network#=DOC20 Node=5a003b77
DOS2.SYS: Network#=DOC20 Node=1b0276a3
```

- 7c. Repeat Steps 7a and 7b for each boot diskette.
8. Create a BOOTCONF.SYS file in the SYS:LOGIN directory. See Figure 1-7.

The BOOTCONF.SYS file tells the workstation which boot image file to use.

Figure 0-7  
Create a  
BOOTCONF.SYS file



When you create multiple Remote Boot disk image files, you also need a BOOTCONF.SYS file in the SYS:LOGIN directory that lists

- ◆ All custom Remote Boot disk image files. (This doesn't include the default NET\$DOS.SYS file.)
- ◆ The network address and node address of each workstation that uses the customized boot image files.

Add the new entries to the existing file with your DOS text editor.

- 8a. Move to the SYS:LOGIN directory.
- 8b. Use a DOS text editor to create the BOOTCONF.SYS file in the SYS:LOGIN directory.

Include a line for each Remote Boot image file you created, using an entry format containing the following information:

- ◆ 0x (the number zero plus x)
- ◆ The network address
- ◆ A comma (,)
- ◆ The node or station address
- ◆ An equal sign (=)
- ◆ The boot disk image filename

An example for two boot diskettes follows:

```
0xDOC20,5a003b77=DOS1.SYS [RPL.NLM BIND  
Override Parameters for this node]  
0xDOC20,1b0276a3=DOS2.SYS [RPL.NLM BIND  
Override Parameters for this node]
```

See “RPL.NLM BIND Override Parameters in the BOOTCONF.SYS File” on page 144 for a complete description of the parameters you can use in the example above.

See “BOOTCONF.SYS Extensions” on page 146 for the following information:

- ◆ Using wildcard characters in the BOOTCONF.SYS file
- ◆ Specifying more than one disk image file per node address
- ◆ Allowing multiple lines per node address

9. Flag the .SYS files in SYS:LOGIN as Shareable.

Type commands similar to the following:

```
FLAG *.SYS S <Enter>
```

```
FLAG *.BAT S <Enter>
```

## Running RPL with Older Remote Boot ROMS

If you use the older Remote Boot ROMS, modify the steps described for running RPL on Enhanced Remote Boot ROMS described in the sections that follow.

- ◆ For single remote boot image files, see “Creating a Single Remote Boot Disk Image File” on page 63 for instructions.
- ◆ For multiple remote boot image files, see “Creating Multiple Remote Boot Disk Image Files” on page 65 for instructions.

Important



You can't use RPL.NLM BING override parameters in the BOOTCONF.SYS and BOOTCONF.SYS for older remote Boot PROMS.

## Running RPLFIX.COM for Older ROMs with DOS Versions Above 5.x

RPLFIX allows workstations to remote program load (RPL) properly with DOS 5.x and above. Run it after you create the boot image file. RPLFIX modifies the boot image file, so you need to run it only once.

The remote workstation may hang during the reset process if you are resetting using DOS 5.x or above and the boot image file hasn't been modified by RPLFIX.

## How to Use RPLFIX

RPLFIX is found on the *WSDOS\_1* diskette. After you locate RPLFIX.COM, map a drive to the LOGIN subdirectory that contains the boot image file (usually named NET\$DOS.SYS). If you renamed your boot image file, you must use the new filename with RPLFIX:

```
RPLFIX [d:] boot image file <Enter>
```

Replace [d:] with the drive letter where the image file is located. Replace *boot image file* with the name of the file created with the DOSGEN utility.

If the drive you mapped to SYS:LOGIN was drive F:, you would enter the following:

```
RPLFIX F:NET$DOS.SYS <Enter>
```

## Troubleshooting RPL with Older ROMS

- ◆ If you get the error message “Error opening boot disk image file,” you are probably attaching to another file server that does not contain the Remote Boot disk image file.

Either log in to the other possible default file servers as SUPERVISOR and run DOSGEN on each, or copy the .SYS and .BAT files from the default file server to the other file servers on the network.

- ◆ If you get the error “Batch file missing,” make sure the AUTOEXEC.BAT file is in

SYS:LOGIN	For every file server you could possibly attach to;
-----------	---

Default (or first mapped drive)	For every file server you normally log in to.
---------------------------------	---

- ◆ If one user can log in but other users are unsuccessful when trying to log in at the same time, make sure the .SYS files are flagged Shareable and that users are assigned the Modify right to SYS:SYSTEM.
- ◆ If you are using a Remote Reset PROM on a Token-Ring network board and you can't boot a workstation, ensure that you have loaded the Remote

Program Load loadable module (LOAD TOKENRPL.NLM) at the NetWare server console before booting the workstation.

- ◆ Use TRACK ON at the NetWare server console and watch for “Get Nearest Server Requests” from the workstation to determine if the boot PROM on the workstation is sending packets.
- ◆ Load MONITOR at the NetWare server console and watch to see if the workstation opens the BOOTCONF.SYS file, the NET\$DOS.SYS file, or other boot disk image files.
- ◆ If a workstation using the boot PROM doesn’t boot, and you have another workstation with a diskette drive configured the same as the first workstation (has the same type of network board using the same configuration options), see if the second workstation will boot with the boot diskette you used with DOSGEN.

Booting with the boot diskette on the second workstation should be the same as booting from the NetWare server on the first workstation.



*appendix*

# **A**

## **Technical Information**

### Overview

This appendix describes the architecture of the NetWare DOS Requester and its Virtual Loadable Modules (VLMs). Also included is a discussion on how networking tasks work with Windows.

The following topics are covered in this appendix.

Topic	Page
Virtual Loadable Modules (VLMs)	74
NetWare DOS Requester	76
NETX.VLM	81
IPXODI	82
LSL	83
LAN Driver	83
TBMI2 (Task-Switched Buffer Manager)	84
Windows Technical Overview	85
Accessing the Network from a Windows Application	86
Running DOS Applications in Real or Standard Mode	87
Running DOS Applications in Enhanced Mode	88
Receiving Broadcast Messages	89
Printing to Network Queues	90

## Virtual Loadable Modules (VLMs)

The NetWare DOS Requester is composed of several individual functional modules called Virtual Loadable Modules (VLMs).

A VLM is a modular executable program with a set of logically grouped features or APIs. For example, transport related functions (send packets, receive packets, etc.) fit logically into a VLM.

The two types of VLMs are these:

- ◆ Child VLMs
- ◆ Multiplexors

### Child VLMs

Child VLMs handle a particular implementation of a logical grouping of functionality. For example, each of the NetWare server types has its own child VLM:

- ◆ BIND.VLM for NetWare v2.x and v3.x bindery servers
- ◆ NDS.VLM for NetWare v4.0 NetWare Directory Services servers
- ◆ PNW.VLM for Personal NetWare (NetWare desktop) servers

Various implementations of transport protocols also have their individual child VLMs. For example, IPXNCP.VLM handles IPX services. Other transport protocols may be defined in the future.

### Multiplexors

A multiplexor is a special VLM that routes calls to the correct child VLM. Requester multiplexors can be considered parent VLMs, ensuring that requests to their child VLMs reach the appropriate VLM.



For example, NWP.VLM is the multiplexor that handles calls to and from the child VLMs, BIND.VLM, NDS.VLM, and PNW.VLM. Similarly, TRAN.VLM is the multiplexor that coordinates communication at the transport layer when its child VLMs (IPXNCP.VLM and other transport protocols) are concurrently loaded.

Table A-1 describes the individual components of the DOS workstation environment.

Table A-1  
NetWare DOS Requester modules

Module	Description
VLM.EXE and associated VLMs	Provides network services for file and print redirection, as well as their own APIs for connection maintenance and other NetWare-specific support.
IPXODI	Delivers requests and replies between a workstation and the network.  Also handles packet sequencing and acknowledgment for the workstation-server connection (for example, SPX).  Takes workstation requests that the NetWare DOS Requester has determined are for the network, "packages" them with transmission information (such as their destination), and hands them to the LSL.
LSL	Puts the packaged requests from IPXODI into the proper format for transmission on the particular physical network that the workstation is running on.  Also takes replies for the workstation from the network (via the LAN driver), removes the network-specific information it has added, and passes the reply to IPX.
LAN driver	Takes requests from the LSL and sends them to the network.  Also receives replies from the network for the workstation and passes them to the LSL.

Table A-1 *continued*  
NetWare DOS Requester modules

Module	Description
TBM12	Handles direct calls to IPX from switched peer-to-peer applications.  Also takes replies from IPX and gives them to the correct switched application.
NETX.VLM	Provides backwards compatibility with the older NETX shell.

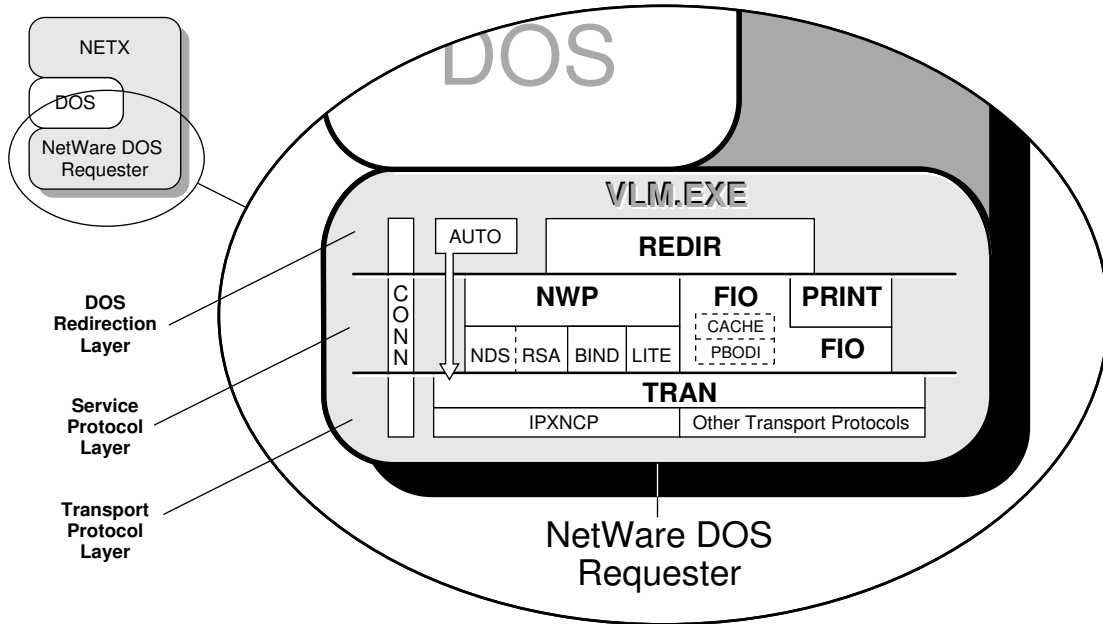
## NetWare DOS Requester

The NetWare DOS Requester provides various services grouped into three layers:

- ◆ DOS Redirection Layer
- ◆ Service Protocol Layer
- ◆ Transport Protocol Layer

Figure A-1 on page 77 shows how these layers and modules fit together. Individual pieces are discussed in further detail in the sections that follow.

Figure A-1  
 NetWare DOS Requester layers and modules



## DOS Redirection Layer

This layer provides DOS file services through the DOS redirector; its VLM is REDIR.VLM. REDIR performs all the DOS-specific callouts.

Previously, NETX provided its own file services; the NetWare DOS Requester uses the DOS redirector for most file services.

## Service Protocol Layer

This layer consists of the following parallel services.

### NetWare Services Provided by NWP.VLM

NWP, the NetWare protocol multiplexor, handles each particular network implementation through these child VLMs:

- ◆ NDS, NetWare Directory Services (NetWare v4.0)
- ◆ BIND, bindery (NetWare v2.x and v3.x)
- ◆ PNW, NetWare desktop (Personal NetWare)

The NWP services also include establishing and destroying connections, logins and logouts, and broadcast messages.

### Security Services Provided by RSA.VLM

RSA (Rivest, Shamir, and Adleman, developers of this particular public key encryption system) provides system-level background authentication for the workstation.

NetWare uses this encryption system to strengthen user authentication and access control functions.

### File Services Provided by FIO.VLM

FIO, the file input/output module, is a separate-but-parallel piece that implements a basic file-transfer protocol. This includes cached or noncached read/write, burst mode-based read/write, and large internet packet read/write.

The Packet Burst protocol in FIO.VLM is based on ODI and is referred to as PBODI. This functionality allows users to choose performance over memory use and vice versa.

## Print Services Provided by PRINT.VLM

The PRINT module provides printing services using the FIO module for its file writes. Print redirection is handled in the following ways, depending on entries in the NET.CFG file:

- ◆ Noncached
- ◆ Cached
- ◆ Redirected via Packet Burst protocol to the server

Printer redirection using FIO is faster.

## Transport Protocol Layer

The transport layer is responsible for maintaining connections, providing packet transmissions between connections, and performing other transport-specific functions.

The IPXNCP module uses IPX as a transport mechanism.

The AUTO.VLM module reconnects a workstation to a server and rebuilds the workstation's environment—excluding file-specific items—to its pre-connection-loss state.

## VLM Manager

The VLM Manager (VLM.EXE) controls communication and memory issues between the individual VLMs, multiplexors, and child VLMs.

The VLM Manager decides whether a given VLM uses expanded memory, extended memory, conventional memory, or any other memory type supported, without affecting the individual VLMs.

The user can override the VLM.EXE defaults with command line options or entries in the NET.CFG file:

**VLM** [*option*]

Table A-2  
VLM options

Option	Syntax	Description
Help	/?	Displays help screen
Unload	/U	Unloads VLM.EXE from memory
Configuration File	/C	Indicates alternate filename for configuration information.  <i>/C = [path]filename</i>
Conventional Memory	/MC	Uses conventional memory when loading VLM.EXE
Extended Memory	/MX	Uses extended memory when loading VLM.EXE
Expanded Memory	/ME	Uses expanded memory when loading VLM.EXE
Diagnostics	/D	Displays VLM.EXE file diagnostics

## Connection Manager

The Connection Table Manager (CONN.VLM) spans all the layers of the NetWare DOS Requester architecture.

CONN allows a workstation running the NetWare DOS Requester to establish a configurable number of connections with multiple NetWare servers, by using a NET.CFG parameter. See "CONNECTIONS" on page 119 for a description this parameter.

## NETX.VLM

The NETX.VLM provides backwards compatibility with NETX and other older versions of the shell.

## NET.CFG File

NET.CFG is a specialized text file that you create with any ASCII text editor and include on a workstation boot diskette with any other necessary boot files. The NET.CFG file replaces the SHELL.CFG file.

Like the DOS CONFIG.SYS file, the NET.CFG file contains configuration values of the LAN drivers and NetWare DOS Requester that are read and interpreted when your workstation starts up. These values adjust the operating parameters of the NetWare DOS Requester, IPX, and other workstation software.

You may want to change the values of certain NetWare DOS Requester parameters to modify the workstation's reaction in certain routines and processes. For a comprehensive list of NET.CFG options see "NET.CFG File Parameters" on page 91.

Applications such as database, multitasking, or NETBIOS (involved in peer-to-peer communications or distributed processing) may require parameter values different from the default values to function properly on the network.

To find out which parameters you might need to modify, consult the setup reference for each application used on your network. You might also find that printing, file retrieval, and other network problems can be solved by adjusting workstation parameters.

To create a NET.CFG file and modify the various parameters, see "Configuring Your Workstation" on page 5.

## IPXODI

Although the NetWare DOS Requester intercepts and prepares requests for network transmission, the actual delivery is made by the IPXODI.COM program (Inter-Packet Exchange Open Data-Link Interface).

IPXODI attaches a header to each data packet. The header specifies necessary information for targeted network delivery, announcing where the packet came from, where it's going, and what happens after delivery.

The targeting ability of IPXODI is important, but by itself does not guarantee successful delivery of a data transmission.

IPXODI transmits data packets as datagrams (self-contained packages that move independently from source to destination), and thus can deliver the packets only on a best-effort basis. Delivery is guaranteed only when using the SPX protocol.

## SPX

SPX (Sequenced Packet Exchange) is a protocol within IPXODI. SPX that is derived from Novell's IPX using the Xerox® Sequenced Packet Protocol. It enhances the IPX protocol by supervising data sent out across the network. SPX verifies and acknowledges successful packet delivery to any network destination by requesting a verification that the data was received.

Within this verification must be a value that matches the value calculated from the data before transmission. So SPX ensures not only that the data packet arrived, but that it arrived intact. It can verify one transmission, or each transmission in a series.

SPX can track data transmissions consisting of a series of separate packets. If an acknowledgment request brings no response within a specified time, SPX retransmits it. After a reasonable number of retransmissions fail to return a positive acknowledgment, SPX assumes the connection has failed and warns the operator of the failure.



## LSL

The LSL (Link Support Layer) is an implementation of the Open Data-Link Interface specification. The LSL serves as an intermediary between the LAN drivers and communication protocols like IPXODI, AFP, or TCP/IP.

The LSL allows one network board to service several communications protocol stacks. The LSL also allows several network boards to service the same protocol stack.

## LAN Driver

Every transmission prepared by the NetWare DOS Requester must pass through the network board connecting the workstation to the network. The LAN driver makes the connection between the physical network board and the logical routines and programs that use it.

A LAN driver is linked both to the specific network board and to the IPX program. Through the LAN driver, IPX can transmit requests regardless of the type of interface board or network protocol.

In effect, different drivers allow the same operating system to use many different kinds of boards. This gives an installer flexibility to choose the most effective network topology.

## TBMI2 (Task-Switched Buffer Manager)

The Task-Switched Buffer Manager for IPX/SPX (TBMI2) allows IPX and SPX programs to work in a multitasking environment (such as Microsoft Windows 3.1, DR DOS 6.0, or MS-DOS 5.0).

The multitasking environment in real and standard modes allows application task switching (swapping). Each application runs in a separate DOS session (DOS Prompt) in available memory.

Global memory, available to all, contains drivers and TSRs such as COMMAND.COM and, if you are running NetWare, IPXODI.COM and the NetWare DOS Requester. Local memory, available to only the current DOS session or prompt, contains the application and application data.

The multitasking environment switches from one DOS application to another by moving the contents of the current DOS session from conventional memory to disk, and then loading the contents of the new DOS session into conventional memory.

Only the local memory is switched; the global memory with its drivers and TSRs stays intact and is used with the new session. This means that separate local memory segments exist, one for each DOS session, while only one global memory segment exists.

You don't need to use TBMI2 if

- ◆ The application uses the NetWare DOS Requester to access IPX or SPX;
- ◆ You will not switch between sessions.

You must use TBMI2 if

- ◆ You will switch between sessions; and
- ◆ The application bypasses the NetWare DOS Requester and accesses IPX or SPX directly.

If your application requires TBMI2 and you don't use it, the session will fail when the DOS session is swapped out or task-switched.



Note

If you aren't sure your application needs TBMI2, run TBMI2; it uses only a small amount of memory. After running the application for a period of time, enter the command TBMI2 /D, and look at the number in the "Far Call Usage" field. If this number is zero, then your application is not using TBMI2; you can run your application without it.

You must update to IPXODI before you can use TBMI2.

## Windows Technical Overview

This section contains conceptual information about the NetWare Workstation for Windows components. The following topics are included.

Topic	Page
Accessing the Network from a Windows Application	86
Running DOS Applications in Real or Standard Mode	87
Running DOS Applications in Enhanced Mode	88
Receiving Broadcast Messages	89
Printing to Network Queues	90

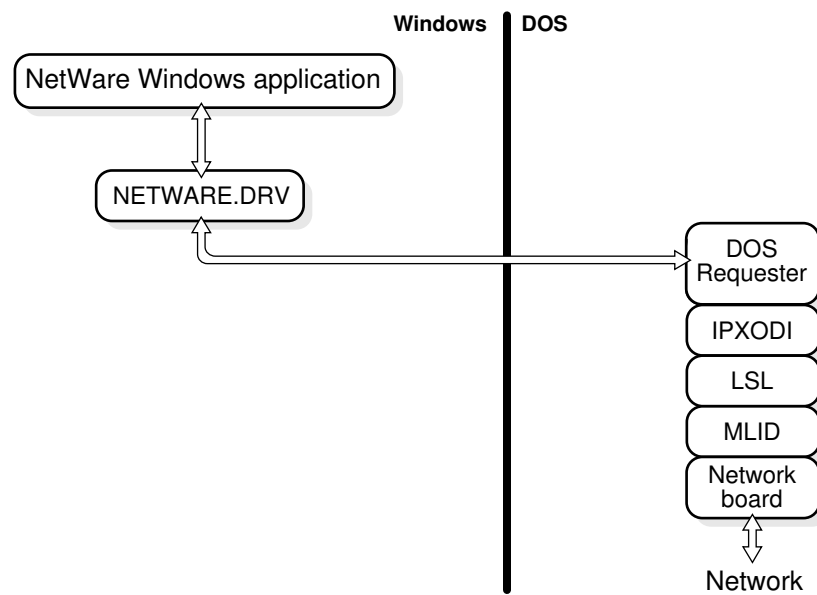
## Accessing the Network from a Windows Application

Figure A-2 illustrates which NetWare components are used to access the network from Windows.

For example, when you use the “Drives” tools to connect a network drive, your workstation uses the components shown below.

The NetWare Windows driver, NETWARE.DRV, translates between Windows and DOS.

Figure A-2  
Accessing the  
network from a  
Windows application



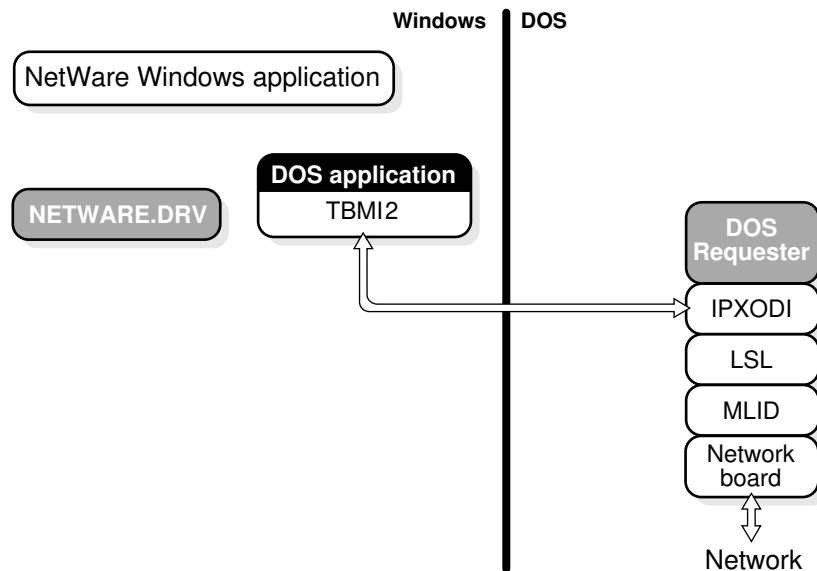
## Running DOS Applications in Real or Standard Mode

Figure A-3 illustrates which NetWare components are used to run DOS applications in real or standard mode.

For example, if you are running Windows on a 286 machine and you choose the DOS prompt icon to go to a DOS window and check your email, your workstation uses the components shown below.

NetWare's task-switching file, TBMI2, synchronizes network calls and responses for DOS sessions.

Figure A-3  
Running a DOS  
application in real or  
standard mode



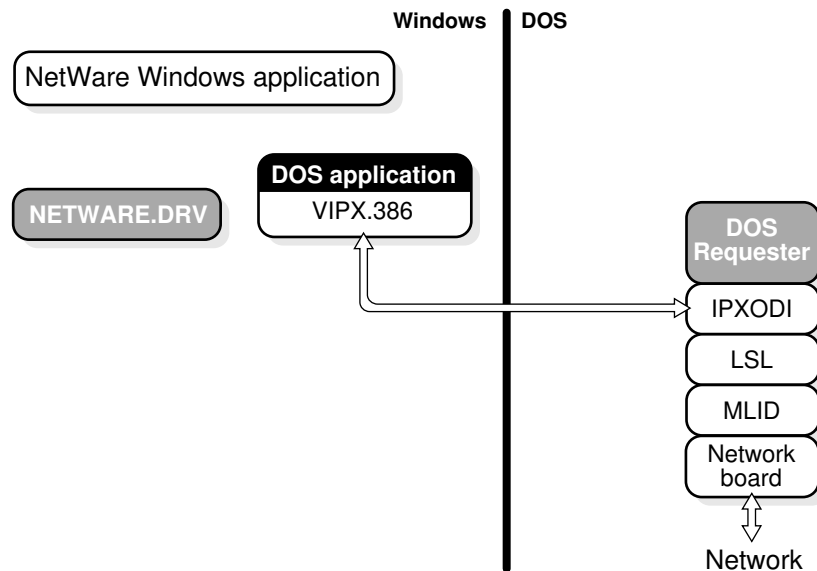
## Running DOS Applications in Enhanced Mode

Figure A-4 illustrates which NetWare components are used to run DOS applications in enhanced mode.

For example, if you are running Windows in enhanced mode and you choose the DOS prompt icon to go to a DOS window and check your email, your workstation uses the components shown below.

NetWare's task-switching file for enhanced mode, VIPX.386, synchronizes network calls and responses for DOS sessions.

Figure A-4  
Running a DOS  
application in  
enhanced mode



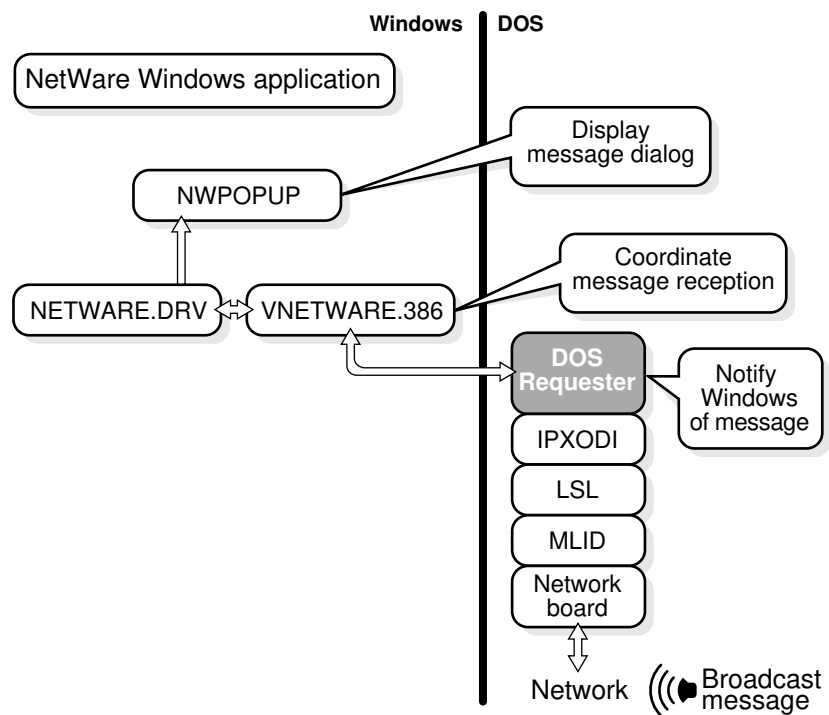
## Receiving Broadcast Messages

Figure A-5 illustrates which NetWare components are used when your workstation receives a broadcast message.

For example, if the secretary sends a message informing you that paychecks have arrived, your workstation uses the components shown below.

VNETWARE.386 works with NETWARE.DRV and NWPOPUP.EXE to coordinate and display message dialogs.

Figure A-5  
Receiving a  
broadcast message

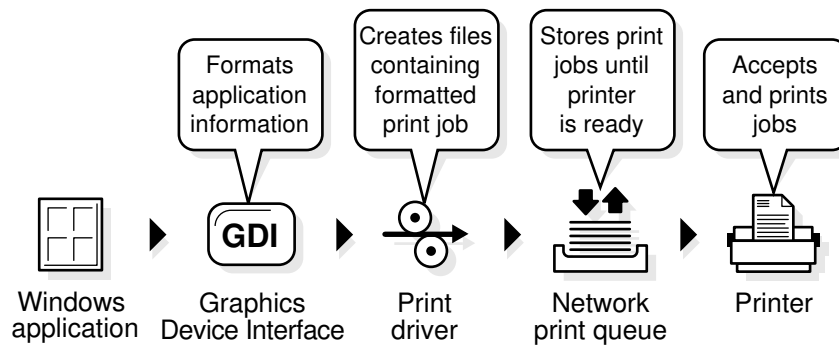


## Printing to Network Queues

Figure A-6 illustrates Windows and NetWare components that are used when you print from a Windows application.

For example, if you print a text file from the Windows Notebook application, your workstation uses the components shown below.

Figure A-6  
Printing path







*appendix*

# **A** **NET.CFG File Parameters**



## Overview

The NET.CFG file is a configuration file that you use to specify settings for your workstation other than the defaults. See “Configuring Your Workstation” on page 5 for instructions on creating the file.

This chapter contains only descriptions of the parameters you can use in the NET.CFG file. Figure A-4 on page 137 lists all the parameters, with defaults.

The following types of parameters are explained in this appendix.

Topic	Page
Link Driver Parameters	92
Link Support Parameters	103
NetBIOS Parameters	105
Protocol IPXODI Parameters	111
NetWare DOS Requester Parameters	116
TBMI2 Parameters	132
TBMI2 Command Line Parameters	134
Command Line Parameters	134

## Link Driver Parameters

Use the “Link Driver” section to name your driver and specify various hardware and software settings for this driver. The “Link Driver” section has the following format:

```
LINK DRIVER drivename
  ALTERNATE
  DMA [#1| #2] channel number
  FRAME frame_type
  INT [#1| #2] interrupt_request_number
  LINK STATIONS number
  MAX FRAME SIZE number
  MEM [#1| #2] hex_starting_address_ [hex_length]
  NODE ADDRESS hex_address
  PORT [#1| #2] hex_starting_address_ [hex_number_of_ports]
  PROTOCOL name hex_protocol_ID_frametype
  SAPS number
  SLOT number
```

To use hardware options, see page 94.

To use software options, see page 99.



If you are using the driver's defaults, you don't need to enter parameters for that driver.

In Figure A-1 on page 93, square bullets indicate options available to both ISA (industry standard architecture) and MCA (microchannel architecture).

Options marked with an I are available to ISA only; options marked with an M are available to MCA only; options marked with an E are available to EISA only.

Figure A-1  
NET.CFG options  
available for LAN  
drivers

Options LAN Drivers	DMA	INT	MEM	PORT	NODE ADDRESS	SLOT	FRAME	PROTOCOL	SAPS	LINK STATIONS	ALTERNATE	MAX FRAME SIZE	DOUBLE BUFFER OFF
EXOS		I	I	I	■	M	■	■					
LANSUP							■	■	■	■	■	■	
NE2					■	■	■	■					
NE2_32					■	■	■	■					
NE1000		■		■	■		■	■					
NE2000		■		■	■		■	■					
NE2100	■	■		■	■		■	■					
NE3200						E	■	■					E
NTR2000					I		I	I			I	I	
PCN2L					■	M	■	■			■		
TOKEN					■	M	■	■			■	■	
TRXNET		I	I	I		M		■					

E = EISA, extended industry architecture  
M = MCA, microchannel architecture  
I = ISA, industry standard architecture



Note If you are using a LAN driver not listed above, refer to its documentation for parameter applicability.

To use the “Link Driver” section, replace *drivername* with the name of the driver you are using; it is typically the LAN driver’s filename.

Replace *drivername* with one of the following driver names:

- ◆ EXOS® (for Novell EXOS205 and EXOS215)
- ◆ LANSUP™ (for the IBM LAN Support program only)
- ◆ NE2™ (for Novell Ethernet NE/2)

- ◆ NE2\_32™ (for Novell Ethernet NE/2-32)
- ◆ NE1000™ (for Novell Ethernet NE1000)
- ◆ NE2000™ (for Novell Ethernet NE2000)
- ◆ NE2100™ (for Novell Ethernet NE2100)
- ◆ NE3200™ (for Novell Ethernet 3200)
- ◆ NTR2000™ (for Novell NTR2000 Token-Ring)
- ◆ PCN2L™ (for IBM PC Network II and II/A)
- ◆ TOKEN™ (for IBM Token-Ring)
- ◆ TRXNET™ (for Novell RX-Net and Turbo RX-Net)

After specifying a *drivername*, place settings for both hardware commands and software commands under the “Link Driver” heading.

You can specify options for each network board you are using, but you must have a separate “Link Driver” *drivername* heading for each network board.

## Hardware Options for the “Link Driver” Section

The hardware options for the “Link Driver” section are as follows:

```
LINK DRIVER drivername
  DMA [#1| #2] channel number
  INT [#1| #2] interrupt_request_number
  MEM [#1| #2] hex_starting_address_ [hex_length]
  NODE ADDRESS hex_address
  PORT [#1| #2] hex_starting_address_ [hex_number_of_ports]
  SLOT number
```

These options are described in the sections that follow. For the DMA, INT, MEM, and PORT options, you don't need to include the characters "#1" if you are specifying a value for only one network board.

For example, if you are using the first configurable DMA channel on a network board as DMA channel 3, you can place the following lines in your NET.CFG file:

```
LINK DRIVER drivername  
DMA 3
```

or

```
LINK DRIVER drivername  
DMA #1 3
```

If, however, you are using more than one DMA, INT, MEM, or PORT value, you must include the characters "#2" for the second value. This abbreviation convention is used for all NET.CFG options that use the [#1| #2] notation.

The following sections contain an explanation of each option's function and some possible reasons for changing the setting. In the explanations, these conventions are used:

[ ]	An optional element
<i>number</i>	A decimal number
<i>hex</i>	A hexadecimal number

#### DMA [#1 | #2] *channel\_number*

Specifies the hardware setting of the network board used in the workstation. It allows DMA channels to be configured.

Enter the channel number to be used. The default is the first configurable channel (#1).

You can also specify two “DMA” values. For example, if the first configurable DMA channel on your board uses DMA channel 3 and the second configurable channel uses channel 4, place the following lines in your NET.CFG file:

```
LINK DRIVER driver name  
DMA #1 3  
DMA #2 4
```

**INT** [#1 | #2] *interrupt\_request\_number*

Specifies which interrupt (IRQ) the network board uses. Use the interrupt request number set on the board.

For example, to specify interrupt 5 on an NE2100 board, place the following lines in your NET.CFG file:

```
LINK DRIVER NE2100  
INT #1 5
```

To specify more than one *interrupt\_request\_number*, place the following lines in your NET.CFG file:

```
LINK DRIVER driver name  
INT #1 5  
INT #2 3
```

The second INT channel also requires you to use the pound sign (#).

**MEM** [#1 | #2] *hex\_starting\_address* [*hex\_length*]

Specifies a memory range to be used by the network board.

Use the hexadecimal physical (absolute) address of the memory used by the board. This *hex\_starting\_address* must match the starting address configured on the board.

Enter the *hex\_length* in hexadecimal paragraphs (a paragraph is 16 bytes) of the memory address range used by the board.

For example, if you address a board from D0000 to D4000 (bytes), the starting address is D0000 and the range is 400 hexadecimal paragraphs. In this case, for an EXOS board (for example, EXOS205), you would place the following lines in your NET.CFG file:

```
LINK DRIVER EXOS  
MEM D0000 400
```

Usually, the length ("*hex\_length*") is not needed.

#### NODE ADDRESS *hex\_address*

Overrides any hard-coded node address in the network board, if the hardware allows it. Use the *hex\_address* parameter to specify a hexadecimal address number.

Note



Changing the node address on a board can create conflicts with other network boards. Use the hard-coded node address whenever possible.

The example below shows how to change the node address on a network board:

```
LINK DRIVER driver name  
NODE ADDRESS 12D43
```

#### PORT [#1 | #2] *hex\_starting\_address* [*hex\_number\_of\_ports*]

Specifies the starting port (*hex\_starting\_address*) and number of ports in the range (*hex\_number\_of\_ports*). All values must be written in hexadecimal notation.

For example, suppose you want to specify the starting port and the number of ports in the first range on your board. The starting I/O port is 300; 16 ports are in the first range. You would place the following lines in your NET.CFG file:

```
LINK DRIVER driver name  
PORT #1 300 16
```

The number of ports is optional. The following example specifies two ranges with 32 ports in each range:

```
LINK DRIVER driver name  
PORT #1 300 32  
PORT #2 340 32
```

#### SLOT *number*

In slot-based machines, the driver usually locates the board by scanning through the slots in order from lowest to highest. This option directs the driver to locate the board in the specified slot, instead of scanning for it.

Use the number of the slot into which you inserted the board. The slot number is usually found on the back of the computer.

For example, if you were using two NE/2 boards in the same workstation and you inserted one board into slot 1 and the other into slot 2, you would place the following lines in your NET.CFG file:

```
LINK DRIVER NE2  
SLOT 1
```

```
LINK DRIVER NE2  
SLOT 2
```

Then you would place the following lines in your batch file:

```
LSL  
NE2  
NE2
```

The “Slot” option directs the first NE2 driver to use an NE2 board in slot 1 and directs the second NE2 driver to use the NE2 board in slot 2.



## Software Options for the “Link Driver” Section

The NET.CFG software options for the “Link Driver” section are as follows:

LINK DRIVER *drivername*  
ALTERNATE  
FRAME *frame\_type*  
LINK STATIONS *number*  
MAX FRAME SIZE *number*  
PROTOCOL *name hex\_protocol\_ID frame\_type*  
SAPS *number*

### ALTERNATE

Specifies an alternate board. Normally, the LANSUP, IBM Token-Ring and NTR2000, and PCN2L drivers use the primary board.

Specify ALTERNATE under the appropriate driver section heading, as shown below:

```
LINK DRIVER LANSUP  
ALTERNATE
```

### FRAME *frame\_type*

Enables the frame types used by the network board. Use it with boards that support multiple frame types or when you want to overwrite the default frame type.

For example, to enable the Ethernet II frame type on an NE1000 board, you would place the following lines in the NET.CFG file:

```
LINK DRIVER NE1000  
FRAME ETHERNET_II
```

Ethernet LAN drivers now default to the Ethernet\_802.2 frame type; Token-Ring LAN drivers default to the Token-Ring frame type.

Important



Previous versions of NetWare defaulted to the Ethernet\_802.3 frame type.

Figure A-2 lists the supported frame types for each LAN driver supplied by Novell.

Figure A-2  
Supported frame  
types for LAN drivers

LAN Drivers	Frame Types								
	ETHERNET_802.3	ETHERNET_802.2	ETHERNET_II	ETHERNET_SNAP	TOKEN-RING	TOKEN-RING_SNF	IBM_PCN2_802.2	IBM_PCN2_SNAP	NOVELL_RX-NET
EXOS	■	■	■	■					
LANSUP					■	■			
NE2	■	■	■	■					
NE2-32	■	■	■	■					
NE1000	■	■	■	■					
NE2000	■	■	■	■					
NE2100	■	■	■	■					
NE3200	■	■	■	■					
NTR2000					■	■			
PCN2L							■	■	
TOKEN					■	■			
TRXNET									■

LINK STATIONS *number*

Specifies the number of link stations needed for the LANSUP driver. Set “Link Stations” to allow for all applications using the IBM LAN Support Program. The maximum value depends on the type of board used.

Default: 1



Note LINK STATIONS is ignored if another application already opened the adapter before the LANSUP.COM driver was loaded.

MAX FRAME SIZE *number*

Sets the maximum number of bytes that can be put on the network by the LAN driver.

The NTR2000.COM driver’s default size is 4216 bytes. But if your board has 8 KB of shared RAM available, the default size is 2168 bytes.

The LANSUP.COM driver's default is 1144 bytes. But if you're using the IBM LAN Support Program with an Ethernet device driver, the maximum size is 1496 bytes.

For NTR2000.COM and LANSUP.COM, the value for *number* must be a multiple of 8. It must include the number of bytes for the data packet (usually 1, 2, 4, or 8 KB), for adapter overhead (6 bytes), and for the largest possible header (currently, 40 bytes LAN header + 74 bytes protocol header = 114 bytes).

For example, to use 2 KB data packets, you would calculate the value for *number* as

$$2048 + 6 + 35 + 5 + 74 = 2168$$

If the total isn't a multiple of 8, then the NTR2000.COM driver rounds it up to the next multiple of 8.

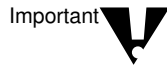
The NET.CFG file for a NTR2000 link driver with a maximum size of 2168 would have the following lines:

```
LINK DRIVER NTR2000  
MAX FRAME SIZE 2168
```

Note



When the driver loads, the sign-on message doesn't reflect the 6 bytes of overhead. Therefore, the "Max Frame" for the above example would be displayed as 2162.



Important

If you are running Windows in enhanced mode, the VIPX.386 device driver requires the maximum frame size to be less than 8000 bytes. If the frame size exceeds 8000 bytes, network communication will fail.

If you are running TBMI2.COM or TASKID, you can't use MAX FRAME SIZE.

If the line speed is 16 Mbps, the value for *number* must be between 632 and 17,960. If the line speed is 4 Mbps, the value must be between 632 and 4464.

PROTOCOL *name hex\_protocol\_ID frame\_type*

Allows existing LAN drivers to handle new network protocols.

Replace *name* with the name of the new protocol.

Replace *hex\_protocol\_ID* with the assigned hexadecimal protocol ID that the protocol is assigned.

Replace *frame\_type* with the frame type that the new protocol ID applies to.

For example, to use a new protocol XYZ with an NE2-32 network board, you would place the following lines in NET.CFG file:

```
LINK DRIVER NE2_32  
FRAME ETHERNET_SNAP  
PROTOCOL XYZ 904A ETHERNET_SNAP
```

SAPS *number*

Specifies the *number* of Service Access Points (SAPS) needed for the LANSUP driver. Set this option to allow for all applications using the IBM LAN Support Program. The maximum value depends on the type of board used.

Default: 1



Note

TSAPS is ignored if another application already opened the board before the LANSUP.COM driver was loaded.

## Link Support Parameters

Use the “Link Support” section to configure receive buffers, the size of the memory pool buffers, and the number of boards and stacks.

The “Link Support” section has the following format:

```
LINK SUPPORT
  BUFFERS number [size]
  MAX BOARDS number
  MAX STACKS number
  MEMPOOL number
```

You can also specify command line parameters for this section. See Figure A-3 “Command line parameters” on page 135 for a these parameters.

### **BUFFERS** *number* [*size*]

Configures the number and size of receive buffers that the Link Support Layer (LSL) will maintain.

The *number* of communication buffers must be large enough to hold all media headers and the maximum data size.

Default: 0

Refer to the documentation for the third-party protocol stacks for possible settings.

Buffer *size* is optional. The minimum size is 618 bytes. The total buffer space must fit into approximately 59 KB (*number* times *size*).

Default: 1130

### MAX BOARDS *number*

Configures the maximum number of logical boards the LSL.COM can handle.

Each LAN driver logical board uses one board resource. For example, if you had the NE1000.COM configured to load all possible Ethernet frame types (ETHERNET\_II, ETHERNET\_802.3, ETHERNET\_802.2, ETHERNET\_SNAP), four board resources would be used. Therefore, to load all four frame types, "Max Boards" must be set to a value of 4 or higher.

Default: 4

Range: 1 to 16

This is an example of the NET.CFG option for this setting:

```
LINK SUPPORT  
MAX BOARDS 1
```

### MAX STACKS *number*

Configures the maximum number of logical protocol stack IDs the LSL.COM can handle.

Each protocol stack uses one or more stack ID resources.

If a protocol stack fails to load because of an out-of-resource condition, increase the value for this option.

The amount of resident memory the LSL.COM consumes increases with larger "Max Stacks" values and decreases with smaller values. Thus, you can conserve some memory by reducing the value to the actual number of stack IDs used by a particular system.

Default: 4

Range: 1 to 16

This is an example of the NET.CFG option for this setting:

```
LINK SUPPORT  
MAX STACKS 3
```

MEMPOOL *number[k]*

Some protocols use the “Mempool” option to configure the size of the memory pool buffers that the LSL will maintain.

Thus, the *k* notation means multiply by 1024.



Note

The IPXODI protocol stack does not use the “Mempool” buffers.

Refer to the documentation for other protocol stacks for recommended settings.

## NetBIOS Parameters

Use the “NetBIOS” section to configure changes in the NetBIOS.

You can also specify command line parameters for NetBIOS. See Figure A-3 “Command line parameters” on page 135

### NETBIOS ABORT TIMEOUT

Adjusts the amount of time, in ticks, that NetBIOS waits without receiving any response from the other side of a session before it terminates the session.

Increase this value (along with “NetBIOS Retry Count” and “NetBIOS Retry Delay”) if there are NetBIOS nodes across asynchronous lines or large internetworks.

The timeout number is in ticks (18.21 ticks per second on IBM PCs and compatibles).

Default: 540 ticks (about 30 seconds)

## NETBIOS BROADCAST COUNT

When multiplied by the “NetBIOS Broadcast Delay” number, determines the total time needed to broadcast a name resolution packet across the network.

This number reflects the size of your network or internetwork.

Increase this value if you

- ◆ Have many LAN segments on the network with nodes that need NetBIOS support;
- ◆ Cannot attach to a gateway.

This value is automatically reduced if “NetBIOS Internet” = OFF.

Default: When “NetBIOS Internet” = ON: 4

Default: When “NetBIOS Internet” = OFF: 2

Range: 2 to 65,535

## NETBIOS BROADCAST DELAY

When multiplied by “NetBIOS Broadcast Count,” determines the total time (in ticks) needed to broadcast a name resolution packet across the network.

This value reflects the traffic of the network. The default is sufficient in most cases.

Increase this value if the packet loss rate is high or if the traffic is high; reduce “NetBIOS Broadcast Count” by a similar amount to maintain the same name resolution timeout value.

This value is automatically reduced if “NetBIOS Internet” = OFF.

Default: When “NetBIOS Internet” = ON: 36

Default: When “NetBIOS InterNET” = OFF: 18

Range: 18 to 65,535

## NETBIOS COMMANDS

Sets the number of NetBIOS command available.



The number of commands needed may vary between applications. In most cases, the default setting of 12 is sufficient. However, if a NetBIOS command error 22 occurs, increase the parameter.

Most applications ignore this error and retry if a NetBIOS command is unavailable. In this case, increasing the number of commands improves application performance.

For specifics, contact the application's software developers to find out how many commands are required.

Default: 12 commands

Range: 4 to 250

## NETBIOS INTERNET

- ◆ If you are running NetBIOS applications on a single network with a dedicated NetWare server, this parameter speeds up the delivery of name resolution and datagram packets when its value is set to OFF.
- ◆ If you are running on more than one network or LAN segment and will be communicating through bridges, or if you are running a nondedicated NetWare server, the value must remain at the default ON.

Default: ON

## NETBIOS LISTEN TIMEOUT

Adjusts the time that NetBIOS waits (when no packets are received from the other side of a session) before it requests a keep-alive packet from the other side to assure the session is still valid.

If NetBIOS has not heard from the other side within this time period, it requests that the other side respond immediately.

NetBIOS continues to send these requests to the other side at an interval specified by “NetBIOS Verify Timeout” until the timeout specified by “NetBIOS Abort Timeout” has expired.

The timeout number is in ticks.

Default: 108 ticks (about 6 seconds)

Range: 1 to 65, 535

## NETBIOS RECEIVE BUFFERS

Configures the number of IPX receive buffers that NetBIOS uses.

Increase this parameter when has:

- ◆ A workstation-server oriented
- ◆ A many-to-one relationship
- ◆ An incoming burst traffic situations, such as a 3270 gateway.

Default: 6 buffers

Range: 4 to 20 buffers

## NETBIOS RETRY COUNT

Determines the number of times NetBIOS resends a packet to establish a session with a remote partner. Affects NetBIOS session commands only.

Adjust the parameter with “NetBIOS Retry Delay” to vary the timeout on establishing NetBIOS sessions. This number reflects the size of the network or internetwork.

Increase this value if you

- ◆ Have many LAN segments on the network with nodes that need NetBIOS support;
- ◆ Can't attach to a gateway.

This value is automatically reduced if “NetBIOS Internet” = OFF.

Default: When “NetBIOS Internet” = ON: 20

Default: When “NetBIOS Internet” = OFF: 10

Range: 4 to 20

## NETBIOS RETRY DELAY

Configures the delay (in ticks) between packets that NetBIOS sends during an attempt to establish a session. Affects NetBIOS session commands only.

NetBIOS sends the request (and waits to send the next request) the number of times indicated by “NetBIOS Retry Count.” This value reflects the size of the network or internetwork you are working on.

Increase this value if you

- ◆ Have many LAN segments on the network with nodes that need NetBIOS support;
- ◆ Can’t attach to a gateway.

Default: 10 ticks (about .5 second)

Range: 10 to 65, 535 ticks

## NETBIOS SEND BUFFERS

Configures the number of IPX send buffers that NetBIOS uses.

Increase this parameter when there exist in the software has

- ◆ A workstation-server oriented
- ◆ A many-to-one relationship
- ◆ An incoming burst traffic situations such as a 3270 gateway.

Default: 6 buffers

Range: 4 to 250 buffers

## NETBIOS SESSION

Configures the maximum number of virtual circuits that NetBIOS can support at the same time.

**Default:** 32 sessions

**Range:** 4 to 250 sessions

## NETBIOS VERIFY TIMEOUT

Adjusts the frequency at which NetBIOS sends a keep-alive packet to the other side of a session to preserve the session.

If no packets are being exchanged on the NetBIOS session by the software that established the session, NetBIOS sends packets at regular intervals to make sure that the session is still valid.

The timeout number is in ticks (18.21 ticks per second on IBM PCs and compatibles).

**Default:** 54 ticks (about 3 seconds)

**Range:** 4 to 65,535 ticks

## NPATCH

Patches any location in the NETBIOS.EXE data segment with any value.

**Default:** None

## Protocol IPXODI Parameters

Use the “Protocol IPXODI” section to change the IPXODI protocol.

You can also specify command line parameters for Protocol IPXODI. See Figure A-3 “Command line parameters” on page 135.

**BIND** *board\_name*

Usually a protocol binds to the first network board it finds. “Bind” forces the protocol to bind to the boards you specify.

Board numbers are displayed when you load the LAN drivers.

Replace *board\_name* with the name of the board you want the protocol to bind to. For example, if you wanted protocol IPXODI to bind to your NE2000 board, the NET.CFG would look like this:

```
PROTOCOL IPXODI  
  BIND NE2000
```

If you were to bind protocol IPXODI to the second and third logical boards, the NET.CFG would look like this:

```
PROTOCOL IPXODI  
  BIND #2, #3
```



Some protocols do not support binding to multiple boards. Refer to the protocol documentation for binding information.

**CONFIG OPTION**

Overrides the configuration options chosen during the WSGEN program. It does not permanently change the IPX file, just the configurations within the workstation RAM.

Use this parameter to change the network board’s configuration options temporarily without running WSGEN or DCONFIG (for example, when you find a possible conflict and want to see if a new configuration option will solve the problem).

If the configuration option must be changed permanently, reconfigure the IPX file with the WSGEN or DCONFIG program.

Default: option set during WSGEN or DCONFIG

#### INT64

Allows applications to use interrupt 64h to access IPX services.

IPX now uses interrupt 64h to maintain compatibility with earlier versions of NetWare.

If an application's documentation requests interrupt 64h or if you have an application that works on earlier versions of NetWare but hangs on NetWare v3.1, set this parameter to OFF.

Default: ON

#### INT7A

Allows applications to use interrupt 7Ah to access IPX services.

IPX now uses interrupt 7Ah to maintain compatibility with NetWare v2.0a.

If an application's documentation requests interrupt 7Ah or if an application works on earlier versions of NetWare but hangs on NetWare v3.1, set this parameter to OFF.

Default: ON

#### IPATCH

Allows any address in the IPXODI.COM file to be patched with any specified byte offset value.

Default: None

## IPX PACKET SIZE LIMIT

Reduces the maximum packet size set by each LAN driver.

Even though a LAN driver could send 16 KB packets on the wire, the wasted memory for most operations may be unacceptable.

The optimum packet size for Token-Ring drivers is 4160 bytes. For Ethernet, the optimum is 1500 bytes. Reduce the maximum packet size if you receive out-of-memory errors at the workstation.

“IPX Packet Size Limit” is a new feature, and not all drivers support it.

**Default:** The lesser of either 4160 or the size specified by the LAN driver

**Range:** 576 to 6500 (bytes)

## IPX RETRY COUNT

Sets the number of times the workstation resends a packet. On networks that lose many packets, this retry count may need to be increased.

IPX does not actually resend a packet. It uses this count to recommend the number of retries to the DOS shell and SPX.

Increasing this number causes a longer delay for some network functions, such as establishing a NetBIOS session or registering a NetBIOS name.

**Default:** 20 retries

## IPX SOCKETS

Specifies the maximum number of sockets that IPX can have open at the workstation.

An IPX-specific program, such as LANSchool™ from LAN Systems Inc.®, may require more than the default number of sockets.

**Default:** 20 sockets

## MINIMUM SPX RETRIES

Determines how many unacknowledged transmit requests are allowed before assuming the connection has gone bad.

SPX applications have two methods to specify a transmit retry count to SPX: first, the application can specify a retry value at the time of connection; second, if the application doesn't specify a retry value, SPX uses the configured "IPX Retry Count" value (defaults to 20).

This option is especially useful if an application that uses SPX is abnormally losing its connections. This may be due to a low value specified in the default retry count in the application or the "IPX Retry Count."

This parameter is supported by IPXODI.COM v2.00 and above.

Range: 0 to 255

### SPX ABORT TIMEOUT

Adjusts the amount of time that SPX waits without receiving any response from the other side of the connection before it terminates the session.

The timeout number is in ticks (18.21 ticks per second on IBM PCs and compatibles).

Default: 540 ticks (about 30 seconds)

### SPX CONNECTIONS

Specifies the maximum number of SPX connections a workstation can use at one time.

Default: 15 connections

### SPX LISTEN TIMEOUT

Adjusts the time that SPX waits without receiving a packet from the other side of the connection before it requests the other side to send a packet to assure the connection is still valid.

If SPX has not heard from the other side of the connection within this time, it sends packets to the other side asking for verification that the connection still exists.

The timeout number is in ticks (18.21 ticks per second on IBM PCs and compatibles).



Default: 108 ticks (about 6 seconds)

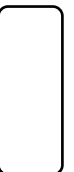
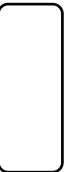
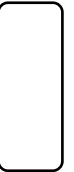
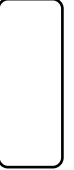
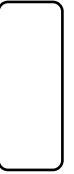
### SPX VERIFY TIMEOUT

Adjusts the frequency at which SPX sends a packet to the other side of a connection to indicate that it is still alive.

If no packets are being exchanged on the SPX connection by the software that established the session, SPX sends packets at regular intervals to make sure that the connection is still working.

The timeout number is in ticks (18.21 ticks per second on IBM PCs and compatibles).

Default: 54 ticks (about 3 seconds)



# NetWare DOS Requester Parameters

Use the “NetWare DOS Requester” section to change the NetWare DOS Requester and its VLMs. The following sections describes each parameter, its defaults, and the VLM file that uses it.

## AUTO LARGE TABLE

Allocates a small table of 34 bytes per connection for bindery reconnects. When the setting is enabled, AUTO.VLM allocates a large connection table of 178 bytes per connection for bindery reconnects.

If a username and password is longer than 16 characters, set this parameter to ON.

Default: OFF

Modules: AUTO.VLM and BIND.VLM



For this parameter to work, also set BIND RECONNECT = ON.

## AVERAGE NAME LENGTH

Allows the NetWare DOS Requester to set aside space for a table of NetWare server names based on the “Average Name Length” and the value for “Connections.” For shorter NetWare server names, you can save some memory by setting the average length to a lower number.

Default: 48 characters

Range: 2 to 48 characters

Module: CONN.VLM

## AUTO RECONNECT

When this option is set to ON, AUTO.VLM reconnects a workstation to a NetWare server and rebuilds the workstation’s environment (excluding file-specific items) prior to connection loss.

When this option is set to OFF, and a connection is lost, reconnecting to a NetWare server becomes the user’s responsibility.

Default: ON

Modules: AUTO.VLM, NDS.VLM, PNW.VLM



Note

You must include the parameter VLM = AUTO.VLM to enable VLM.EXE to load AUTO.VLM.

## AUTO RETRY

Number of seconds AUTO.VLM waits before attempting a retry after receiving a network critical error.

Default: 0

Range: 0/3640

Modules: AUTO.VLM, NDS.VLM, BIND.VLM, PNW.VLM



Note

When this parameter is 0, AUTO.VLM makes no retry attempts.

## BIND RECONNECT

Automatically rebuilds bindery connections and restores drive mappings and printer connections.

Default: OFF

Modules: AUTO.VLM and BIND.VLM



Note

For this parameter to work, also set AUTO RECONNECT = ON.

## CACHE BUFFERS

Sets how many cache buffers the NetWare DOS Requester uses for local caching of nonshared, nontransaction-tracked files.

Each buffer allocated allows the NetWare DOS Requester to cache one file.

You can increase the number of cache buffers to speed up the process of sequential reads/writes. However, increasing this parameter also increases memory use.

Default: 5 cache blocks

Range: 0 to 64 cache blocks

Module: FIO.VLM

## CACHE BUFFER SIZE

Sets the buffer size for the cache buffers that the FIO module uses.

Increasing this parameter allows you to cache larger amounts of data, thereby increasing performance and also memory use.

When specifying this parameter, don't exceed the maximum packet size of the network board.

Default: 512 bytes

Range: 64 to 4096 bytes

Module: FIO.VLM

## CACHE WRITES

Sets the cache write requests to ON or OFF.

Setting this parameter to OFF increases data integrity but decreases performance. Leaving this parameter set to ON can cause data loss if the NetWare server runs out of disk space between write requests.

Default: ON

Module: FIO.VLM

## CHECKSUM

Provides a higher level of data integrity by validating NCP packets.

Setting this parameter to 2 or 3 increases data integrity but decreases performance. The checksum values are these:

- 0 = disabled
- 1 = enabled but not preferred
- 2 = enabled and preferred
- 3 = required

Default: 1

Module: IPXNCP.VLM and NWP.VLM

Note



Ethernet frame type 802.3 doesn't support checksums. The Ethernet drivers must be configured to use a frame type other than 802.3 to use checksums.

## CONNECTIONS

Sets the maximum number of connections the NetWare DOS Requester supports.

The NetWare DOS Requester supports up to 50 connections in its connection table. A value larger than needed uses memory unnecessarily.

Default: 8 connections

Range: 2 to 50 connections

Module: CONN.VLM and FIO.VLM

Note



Values other than 8 may decrease NETX compatibility.

## DOS NAME

Sets the name of the operating system used in the shell.

The %OS variable in the login script uses this variable when mapping a search drive to the network DOS directory. This parameter uses a maximum of five characters.

Default: MSDOS

Module: NETX.VLM and GENERAL.VLM



The NetWare DOS Requester automatically recognizes DRDOS and sets this option. However, setting this option overrides the auto-detect feature.

## FIRST NETWORK DRIVE

Sets the first network drive to the letter of choice when the NetWare DOS Requester makes a connection to the NetWare server.

This parameter accepts only the drive letter and not the colon.

Default: First available drive

Range: A to Z

Module: GENERAL.VLM

## HANDLE NET ERRORS

Determines the default method for handling network errors.

A network error is generated when the workstation doesn't receive a response from the NetWare server. The two values are as follows:

ON = interrupt 24 handles network errors

OFF = return NET\_RECV\_ERROR (example: 8805h)

Default: ON

Module: IPXNCP.VLM



Some applications may not work properly if this parameter is set to OFF.

## LARGE INTERNET PACKETS

In the past, NetWare communicated across routers and bridges with a 576-byte maximum packet size. However, Ethernet and Token- Ring are capable of using larger packets for communication.

When this parameter is set to ON, the maximum packet size negotiated between the NetWare server and the workstation is used, even across routers and bridges.

Default: ON

Module: IPXNCP.VLM



Note Some routers and bridges have been hardcoded to use 576-byte packets. In this case, the NetWare DOS Requester can use only 576-byte packets, regardless of this parameter.

## LOAD CONN TABLE LOW

When using the initial release of NetWare v4.0 utilities, you must set the “Load Conn Table Low” parameter to ON. This loads the connection table low which increases conventional memory requirements required by the initial release of NetWare v4.0 utilities.

If you are not using an initial release of the NetWare v4.0 utilities, the default setting (OFF) will give you better memory performance. The default setting loads the connection table in an UMB, if available.

Default: OFF

Modules: CONN.VLM

## LOAD LOW CONN

By default, the connection manager, CONN.VLM, is loaded in conventional memory. If this parameter is set to OFF, CONN.VLM loads in upper memory, saving memory but sacrificing performance.

Default: ON

Module: CONN.VLM

## LOAD LOW IPXNCP

By default, the transport protocol implementation for IPX, IPXNCP.VLM, is loaded in conventional memory. If this parameter is set to OFF, IPXNCP.VLM loads in upper memory, saving memory but sacrificing performance.

Default: ON

Module: IPXNCP.VLM

## LOCAL PRINTERS

Overrides the number of local printers on the workstation, which is normally determined by the BIOS. The BIOS defaults to a local printer for each parallel port.

By setting the number of local printers to 0, you can prevent a workstation from hanging if you press <Shift> and <Print Screen> while the workstation isn't capturing and doesn't have a local printer.

Default: 3

Range: 0 to 9

Module: PRINT.VLM

## LONG MACHINE TYPE

Tells the NetWare DOS Requester what type of machine is being used each time the %MACHINE variable is accessed. Use this parameter to set the machine's search path to the correct version of DOS.

Default: IBM\_PC

Maximum: 6 letters

Module: NETX.VLM and GENERAL.VLM



## MAX TASKS

Configures the maximum number of tasks that can be active at one time.

Certain multitasking applications, such as Windows and DESQview™, allow several programs to run at one time. If you have problems running a new program, increase the value of this parameter.

Default: 31 tasks

Range: 20 to 128 tasks

Module: CONN.VLM

## MESSAGE LEVEL

Sets how you want to display load time messages. Each message level implies the previous level's message (example: 1 implies 0).

The values are as follows:

0 = Always display copyright message and critical errors

1 = Display warning messages

2 = Display program load information for VLMs

3 = Display configuration information

4 = Display diagnostic information

Default: 1

Module: NWP.VLM

## MESSAGE TIMEOUT

Defines timeout in ticks before broadcast messages are cleared from the screen without user intervention.

Default: 0 (wait for user)

Range: 0 to 10 000 (approximately six hours)

Module: NWP.VLM

## NAME CONTEXT = "*name context*"

Allows you to set your current position in the Directory tree structure. This parameter applies only to workstations connecting to a NetWare v4.0 network.

The default is the root, which may cause confusion if duplicate usernames exist. The quotation marks are required in this parameter.

Module: NDS.VLM

## NETWORK PRINTERS

Sets the number of LPT ports the NetWare DOS Requester can capture. This parameter allows you to capture and redirect LPT1 through LPT9.

Increasing the number increases memory use. Setting this parameter to 0 specifies that PRINT.VLM does not load.

Default: 3

Range: 0 to 9

Module: PRINT.VLM

## PB BUFFERS

Controls the use of Packet Burst protocol for file input/output.

Packet Burst is automatically enabled in the NetWare DOS Requester. For more information on Packet Burst, see "Using Packet Burst to Increase Speed" on page 12.

The values are as follows:

0 = off

non-zero = on

Setting this option to 0 decreases memory use and, in some cases, decreases performance.

Default: 3

Range: 0 to 10

Module: FIO.VLM

#### PREFERRED SERVER = *servername*

Sets the NetWare server you attach to first and helps guarantee your connection to the network. If the server specified has a connection available, the NetWare DOS Requester attaches to that server.

Default: no preferred server

Module: BIND.VLM



Note

If both preferred tree (for NetWare Directory Services) and preferred server (for bindery services) are specified, then the first protocol to successfully build an attachment is used.

#### PREFERRED TREE = *tree name*

Sets the tree you first want to connect to in a NetWare v4.0 network if you have multiple trees. If the tree specified has a server with a free connection, the NetWare DOS Requester attaches to that tree.

Default: No preferred tree

Module: NDS.VLM



Note

If both preferred tree (for NetWare v4.0) and preferred server (for NetWare v2.x and v3.x) are specified, then the first protocol to successfully build an attachment is used.

#### PRINT BUFFER SIZE

Determines the size, in bytes, for the print buffer. The print buffer acts as a cache for 1-byte print requests, which increases the size of some jobs.

Increasing this option increases memory use, while increasing some printing output.

Default: 64 bytes

Range: 0 to 256 bytes

Module: PRINT.VLM

## PRINT HEADER

Sets the size of the buffer that holds the information used to initialize a printer for each print job.

If you send print jobs with many instructions in the header (such as initializing a printer for an emulated mode or changing defaults, font selections, page length, or orientation) and the printer is not delivering all the requested attributes, increase the size of the print header.

Default: 64 bytes

Range: 0 to 1024 bytes

Module: PRINT.VLM

## PRINT TAIL

Sets the size of the buffer that holds the information used to reset the printer after a print job.

If your printer is not clearing out the buffer completely or resetting after each print job, increase the "Print Tail" size.

Default: 16 bytes

Range: 0 to 1024 bytes

Module: PRINT.VLM

## READ ONLY COMPATIBILITY

Determines whether a file marked Read Only can be opened with a read/write access call. Certain applications require this parameter to be set to ON.

Prior to NetWare v2.1, a program could open a Read Only file with write access without getting an error, though any attempt to write to the file produced an error.

To be compatible with DOS, NetWare v2.1 and above does not allow a Read Only file to be opened for write access. Setting the “Read Only Compatibility” to ON causes the shell to revert to the old mode and allow the open request to succeed.

Default: OFF  
Module: REDIR.VLM

## SEARCH MODE

Alters the NetWare DOS Requester’s method for finding a file if it is not in the current directory.

Most .EXE and .COM files use a specific search mode to locate files in a directory structure. The search mode set in the NET.CFG file is global and affects all .EXE and .COM files.

When using “Search Mode,” select the search mode that works correctly with most of your .EXE and .COM files.

If you want to set a search mode for one particular .EXE or .COM, use the “Search Mode” option in FLAG (see “FLAG” in *Utilities Reference*).

Default: 1  
Module: GENERAL.VLM

Valid search modes are explained below.

- 0 No search instructions. Default setting for executable files.
- 1 If a directory path is specified in the executable file, the executable file searches only that path. If a path is not specified, the executable file searches the default directory and network search drives.
- 2 The executable file searches only the default directory or the path specified.
- 3 If a directory path is specified in the executable file, the executable file searches only that path. If a path is not specified and the executable file opens data files flagged Read Only, the executable file searches the default directory and search drives.
- 4 Reserved.
- 5 The executable file searches the default directory and NetWare search drives whether or not the path is specified in the executable file. If a search mode is set, the shell allows searches for any .xxx extension; otherwise DOS searches only for .EXE, .COM, and .BAT.
- 6 Reserved.
- 7 If the executable file opens data files flagged Read Only, the executable file searches the default directory and search drives whether or not the path is specified in the executable file.

In previous NetWare workstation software versions, the default drive had to be a network drive for “Search Mode” to function. The NetWare DOS Requester, as mentioned, is global and affects all .EXE and .COM files, regardless of the current drive.

## SET STATION TIME

Synchronizes the workstation date and time with that of the NetWare server to which the workstation initially attaches.

Setting this option to OFF disables the synchronization feature.

Default: ON  
Module: VLM.EXE

## SHOW DOTS

The NetWare server doesn't have directory entries for . and .. as DOS does. To see . and .. in directory listings, use SHOW DOTS = ON.

Default: OFF (for Windows 3.x, set SHOW DOTS = ON)  
Module: REDIR.VLM

Note



SHOW DOTS is supported only by NetWare versions 2.11 and above.

## SHORT MACHINE TYPE

This option is similar to the "Long Machine Type," except that it is used specifically with overlay files.

Use it when the %SMACHINE variable is accessed. Examples of files using this parameter include the IBM\$RUN.OVL file for the windowing utilities and the CMPQ\$RUN.OVL file that uses a default black-and-white color palette for NetWare menus.

Default: IBM  
Maximum: 4 letters  
Module: NETX.VLM and GENERAL.VLM

## SIGNATURE LEVEL

Designates the level of enhanced security support.

Enhanced security includes the use of a message digest algorithm and a per connection/per request session state. The values are as follows:

- 0 = disabled
- 1 = enabled but not preferred
- 2 = preferred
- 3 = required

Setting this option to 2 or 3 increases security but decreases performance.

Default: 1

Module: NWP.VLM

## TRUE COMMIT

Selects whether the commit NCP is sent on DOS commit requests.

OFF = opts for performance over integrity

ON = opts for integrity over performance

Set this option to ON when processing critical data to guarantee data integrity (for example, database applications).

Default: OFF

Module: FIO.VLM



## USE DEFAULTS

Overrides the default VLMs that VLM.EXE loads. Without it, the VLM will attempt to load these VLMs:

CONN  
IPXNCP  
TRAN  
SECURITY  
NDS  
BIND  
NWP  
FIO  
GENERAL  
REDIR  
PRINT  
NETX

If you specify VLMs that are normally loaded by default in the NET.CFG file and don't set "Use Defaults" to OFF, any default VLMs you specify will attempt to load twice, producing an error during load.

Default: ON  
Module: VLM.EXE

## VLM = *path VLM*

Specifies a .VLM file that VLM.EXE should load. This option allows VLMs not listed in the default for VLM.EXE to be added. The following VLMs are not listed in the default load table:

RSA  
AUTO  
NMR

You must specify the complete file name, including the .VLM extension.

Maximum: 50 VLMs  
Module: VLM.EXE

## TBMI2 Parameters

Use the TBMI2 parameters to change the task-switching environment.

You can also specify command line parameters for TBMI2. See Table B-1, “TBMI2 Command Line Parameters” on page 134.

### DATA ECB COUNT

Specifies how many data ECBs are allocated for use by DOS programs needing virtualization.

These ECBs apply to most IPX and SPX send-and-receive packets. If a nondata ECB request is made when none are available, a data ECB is used.

Each allocated data ECB requires 628 bytes of memory; the 60 ECB default requires 37,680 bytes.

The maximum allocation also depends upon available memory; the total size of all ECBs must be less than 64 KB, which normally limits the data ECB count to less than 255.

Default: 60

Maximum: 89

Minimum: 10

### ECB COUNT

Specifies how many nondata event control blocks (ECBs) are allocated for use by DOS programs needing virtualization.

These ECBs apply to most AES (asynchronous events services) events. If TBMI2 runs out of nondata ECBs, data ECBs can be allocated for use.

Each allocated ECB requires 52 bytes of memory; the 20 ECB default requires 1,040 bytes.

The maximum allocation also depends upon available memory; the total size of all ECBs must be less than 64 KB, which normally limits the ECB count to less than 255.

Default: 20

INT64

Maximum: 255  
Minimum: 10

Allows applications to use interrupt 64h to access IPX and SPX services.

IPX and SPX now use interrupt 64h to maintain compatibility with earlier versions of NetWare. If an application's documentation requests interrupt 64h, set this parameter to OFF.

Default: ON

INT7A

Allows applications to use interrupt 7Ah to access IPX and SPX services.

IPX and SPX now use interrupt 7Ah to maintain compatibility with NetWare v2.0. If an application's documentation requests interrupt 7Ah, set this parameter to OFF.

Default: ON

USE MAX PACKETS

Tells TBMI2 to use the maximum packet size as specified by the IPX "Max Packet" size.

USING WINDOWS 3.0

Tells TBMI2 to use TASKID. Setting this parameter tells NetWare to identify tasks in each DOS BOX as separate tasks.

## TBMI2 Command Line Parameters

Table A-1 shows the TBMI2 parameters that you can run at the command line.

Table A-1  
TBMI2 command line parameters

Parameter	Syntax	Description
Help	<i>/?</i> or <i>/H</i>	Displays help or usage information.
Configuration Files	<i>/C = filename</i>	Specifies configuration files. Example: Type <i>TBMI2/C = TBMI2.CFG</i> at the DOS prompt.
Diagnostic	<i>/D</i>	Displays diagnostic information.
Version	<i>/I</i>	Displays version information.
Unload	<i>/U</i>	Unloads TBMI2.

## Command Line Parameters

These parameters can be run at the command line, and can be used with NETX, IPXODI, LSL, NetBIOS and TBMI2 unless otherwise specified. None is case-sensitive.

Figure A-3 on page 135 shows the various parameters that you can use.

Figure A-3  
Command line  
parameters

Parameter	Description	Works with
<b>/C=</b>	Allows users to specify a configuration file for the DOS Requester to use as it loads. Otherwise, NET.CFG is used for configuration information. Type <b>vlm/c=[path]filename.ext &lt;Enter&gt;</b>  If no path is designated, the current directory is the default directory. If no path is specified, the configuration file must be in the current directory for the system to use that file.	All but NETBIOS
<b>/F</b>	Allows users to forcibly unload the shell even if other terminate-and-stay-resident programs (TSRs) were loaded after the shell. Type <b>netx/f &lt;Enter&gt;</b>  It doesn't allow the shell to be unloaded if any interrupts are hooked or if the workstation is running DESQview.	NETX only
<b>/PS=</b>	Allows user to choose a preferred server. Type <b>vlm=server name &lt;Enter&gt;</b>	VLM and NETX only
<b>/U</b>	Allows users to unload the DOS Requester if no other terminate-and-stay-resident programs (TSRs) are loaded above the DOS Requester. Type <b>vlm/u &lt;Enter&gt;</b>  It doesn't allow the DOS Requester to be unloaded if any interrupts are hooked or if the workstation is running DESQview.	All but IPX
<b>/?</b>	Displays usage information. Type <b>vlm/? &lt;Enter&gt;</b>	All but NETBIOS
<b>/H</b>	Displays usage information at the command line. Type <b>tbmi2/h &lt;Enter&gt;</b>	TBMI2 only
<b>/O#</b>	Load using hardware option #. Type <b>ipx/o# &lt;Enter&gt;</b>	IPX only

## NET.CFG Options

NetWare supports the following configuration options in your NET.CFG file

- ◆ Link Driver
- ◆ Link Support
- ◆ Protocol
- ◆ NetWare DOS Requester
- ◆ TBMI2

Figure A-4 on page 137 shows the various parameters and settings that you can use. Notice the following markings in figure text:

\* The default setting is the maximum value for NetWare v2.x and v3.x networks.

† This option is invalid for NetWare v2.x and v3.x networks.

Figure A-4  
NET.CFG options

NetWare Configuration Options	
Options and Settings	Defaults
<b>link driver</b> <i>drivename</i>	
<b>alternate</b>	
<b>dma</b> [#1] [#2] <i>channel_number</i>	
<b>frame</b> <i>frame_type</i>	
<b>int</b> [#1] [#2] <i>interrupt_request_number</i>	
<b>link stations</b> <i>number</i>	
<b>max frame size</b> <i>number</i>	
<b>mem</b> [#1] [#2] <i>hex_starting_address</i> [ <i>hex_length</i> ]	
<b>node address</b> <i>hex_address</i>	
<b>port</b> [#1] [#2] <i>hex_starting_address</i> [ <i>hex_number_of_ports</i> ]	
<b>protocol</b> <i>name</i> <i>hex_protocol_ID</i> <i>frame_type</i>	
<b>saps</b> <i>number</i>	
<b>slot</b> <i>number</i>	
<b>link support</b>	
<b>buffers</b> <i>communication_number</i> [ <i>size</i> ]	
<b>max boards</b> <i>number</i>	
<b>max stacks</b> <i>number</i>	
<b>mempool</b> <i>number</i> [ <i>k</i> ]	
<b>netbios abort timeout</b> <i>number</i> ..... 540 (30 seconds)	
<b>netbios broadcast count</b> <i>number</i> ..... 2	
<b>netbios broadcast delay</b> <i>number</i> ..... 18 (1 second)	
<b>netbios commands</b> <i>number</i> ..... 12	
<b>netbios internet</b> [on off] ..... on	
<b>netbios listen timeout</b> <i>number</i> ..... 108 (6 seconds)	
<b>netbios receive buffers</b> <i>number</i> ..... 6 (buffers)	
<b>netbios retry count</b> <i>number</i> ..... 10 (1 second)	
<b>netbios retry delay</b> <i>number</i> ..... 10 (1 second)	
<b>netbios send buffers</b> <i>number</i> ..... 6	
<b>netbios session</b> <i>number</i> ..... 32	
<b>netbios verify timeout</b> <i>number</i> ..... 54 (3 seconds)	
<b>npatch</b> <i>byte offset, value</i> .....	

Figure A-4 *continued*  
NET.CFG options

<b>NetWare Configuration Options (continued)</b>	
<b>Options and Settings</b>	<b>Defaults</b>
<b>protocol</b> <i>protocol_name</i>	
<b>bind</b> <i>board_number</i> .....	
<b>config option</b> <i>number</i> .....	
<b>int64</b> [on off] .....	<i>on</i>
<b>int7A</b> [on off] .....	<i>on</i>
<b>ipatch</b> <i>byte offset, value</i> .....	
<b>ipx packet size limit</b> <i>number</i> .....	<i>4160</i> (or set by each LAN driver)
<b>ipx retry count</b> <i>number</i> .....	<i>20</i> retries
<b>ipx sockets</b> <i>number</i> .....	<i>20</i> sockets
<b>minimum spx retries</b> <i>number</i> .....	
<b>spx abort timeout</b> <i>number</i> .....	<i>540</i> (≈30 seconds)
<b>spx connections</b> <i>number</i> .....	<i>15</i> (connections)
<b>spx listen timeout</b> <i>number</i> .....	<i>108</i> (≈6 seconds)
<b>spx verify timeout</b> <i>number</i> .....	<i>54</i> (≈3 seconds)
<b>netware dos requester</b>	
<b>auto large table</b> [on off] .....	<i>off</i>
<b>auto reconnect</b> [on off] .....	<i>on</i>
<b>auto retry</b> <i>number</i> .....	<i>0</i>
<b>average name length</b> <i>number</i> .....	<i>48</i>
<b>bind reconnect</b> [on off] .....	<i>off</i>
<b>cache buffers</b> <i>number</i> .....	<i>5</i> (cache blocks)
<b>cache buffers size</b> <i>number</i> .....	<i>512</i> (bytes)
<b>cache writes</b> [on off] .....	<i>on</i>
<b>checksum</b> <i>number</i> .....	<i>1</i>
* <b>connections</b> <i>number</i> .....	<i>8</i>
<b>dos name</b> <i>name</i> .....	<i>ms_dos</i>
<b>first network drive</b> <i>drive_letter</i> .....	(first available drive)
<b>handle net errors</b> [on off] .....	<i>on</i>
* <b>large internet packets</b> [on off] .....	<i>on</i>
<b>load low conn</b> [on off] .....	<i>on</i>
<b>load low ipxncp</b> [on off] .....	<i>on</i>
<b>local printers</b> <i>number</i> .....	<i>3</i>



Figure A-4 *continued*  
NET.CFG options

NetWare Configuration Options <i>(continued)</i>	
Options and Settings	Defaults
<b>long machine type</b> <i>name</i> .....	<i>ibm_pc</i>
<b>max tasks</b> <i>number</i> .....	<i>31</i>
<b>message level</b> <i>number</i> .....	<i>1</i>
<b>message timeout</b> <i>number</i> .....	<i>0</i>
† <b>name context</b> " <i>name</i> " .....	
* <b>network printers</b> <i>number</i> .....	<i>3</i>
<b>pb buffers</b> <i>number</i> .....	<i>3</i>
<b>pburst read window size</b> <i>number</i> .....	<i>16</i>
<b>pburst write window size</b> <i>number</i> .....	<i>10</i>
† <b>preferred server</b> <i>server_name</i> .....	(no preferred server)
* <b>preferred tree</b> <i>tree_name</i> .....	(no preferred tree)
* <b>print buffer size</b> <i>number</i> .....	<i>64</i> (bytes)
* <b>print header</b> <i>number</i> .....	<i>64</i> (bytes)
<b>print tail</b> <i>number</i> .....	<i>16</i> (bytes)
<b>read only compatibility</b> [on off] .....	<i>off</i>
<b>search mode</b> <i>number</i> .....	<i>1</i>
<b>set station time</b> [on off] .....	<i>on</i>
<b>show dots</b> [on off] .....	<i>off</i>
<b>short machine type</b> <i>name</i> .....	<i>ibm</i>
<b>signature level</b> <i>number</i> .....	<i>1</i>
<b>true commit</b> [on off] .....	<i>off</i>
<b>use defaults</b> [on off] .....	<i>on</i>
<b>vlm</b> <i>path vlm</i> .....	
<b>tbmi2</b>	
<b>data ecb count</b> <i>number</i> .....	
<b>ecb count</b> <i>number</i> .....	
<b>int64</b> [on off] .....	
<b>int7a</b> [on off] .....	
<b>use max packets</b> .....	
<b>using windows 3.0</b> .....	

\* The default setting is the maximum value for NetWare v2.x and 3.x networks





## appendix **C** *RPL Parameters and Extensions*

### Overview

This appendix contains the description of the RPL BIND parameters and the override parameters that you can specify in the BOOTCONF.SYS file.

The following topics are covered in this chapter.

Topic	Page
RPL.NLM BIND Time Parameters	141
RPL.NLM BIND Override Parameters in the BOOTCONF.SYS File	144
BOOTCONF.SYS Extensions	146

### RPL.NLM BIND Time Parameters

You use these parameters when you bind RPL.NLM to the network board when running RPL on enhanced Remote Boot PROMS. See “Bind RPL.NLM to the Network Board” on page 62 for further information.

These parameters are optional and are not case-sensitive. Enter the parameters in any order. Separate the parameters with blanks or commas.

## ACK

Configures the RPL BOOT ROM module to acknowledge FILE.DATA.RESPONSE frames sent by RPL.NLM. This parameter allows the adapter to keep pace with RPL.NLM.

RPL.NLM sends FILE.DATA.RESPONSE frames in Packet Burst mode.

## FRAME = *ff*

Configures the Bootstrap program to use a frame type other than the default to access the NetWare server. Valid selections are these:

- ◆ FRAME=802.2

This is the default. The “Frame” parameter is fully supported on BOOT ROMs that use the RBOOT.RPL Bootstrap program.

The “Frame” parameter is ignored on BOOT ROMs that use PCN2L.RPL and TOKEN.RPL.

- ◆ FRAME=EII

Configures the Bootstrap program for ETHERNET\_II. This option should be used only on ETHERNET networks.

- ◆ FRAME=SNAP

Configures the Bootstrap program for SNAP.

If you select FRAME=EII or FRAME=SNAP, RPL.NLM forces the Bootstrap program to issue a Get Nearest Server (GNS) request.

However, FRAME=EII or FRAME=SNAP configures ETHER.RPL to use ETHERNET\_II instead of 802.3. Specifying FRAME=EII on a Token-Ring BOOT ROM causes the parameter to be ignored.

## GNS

Specifies that you want the workstation to issue a Get Nearest Server request when the appropriate Bootstrap program gets downloaded.

Normally, RPL.NLM fills in the Bootstrap program with the NetWare server information, so that it doesn't need to issue a Get Nearest Server request.



GNS may cause the workstation to find a NetWare server other than the one where RPL.NLM is located.

## NODEFAULT

Tells RPL.NLM not to respond to a FIND frame unless the node address of the workstation is found in the BOOTCONF.SYS file.

This parameter is provided for security reasons.

The workstation won't boot until the network administrator inserts into the BOOTCONF.SYS file the node address and associated disk image filenames to use when booting the workstation.

## PROTECT

Tells RPL.NLM to configure the Bootstrap program so that it will protect itself in the workstation memory. It does this by adjusting the "Memory Size" variable in the BIOS data area (40:13) to reflect the amount of memory that it uses.



PROTECT reduces the amount of memory that the workstation has available for DOS by about 12 K. Don't use "Protect" unless necessary.

## PS = *server*

Tells the Bootstrap program to attach to a NetWare server other than the one where RPL.NLM is loaded.

Replace *server* with the name of a NetWare server that contains the disk image file for the workstation.

## TRO

Specifies that you want the Bootstrap program to do a This Ring Only count of three (3) on all Broadcast frames.

It is useful in a source routing environment where servers are available on the local ring.

## WAIT TIME = *ssss*

Specifies the number of seconds that you want the Bootstrap program to wait before automatically selecting the cursored disk image names specified in the BOOTCONF.SYS file.

Default: 0000, which means infinity

Range: 0000 to 65535.

## RPL.NLM BIND Override Parameters in the BOOTCONF.SYS File

When RPL.NLM parses BOOTCONF.SYS, it allows the user to override the BIND time parameters with parameters specific to a particular workstation that is being booted.

By default, the same parameters that you use with BIND apply to all workstations that are attached to the particular board specified in the BIND command.

See “RPL.NLM BIND Time Parameters” on page 141 for a description of these parameters.

In addition, the following commands are allowed on a per-node basis. They override the BIND time parameters.

## NOACK

Overrides the “ACK” parameter for BIND.

NOGNS

Overrides the “GNS” parameter for BIND.

NOPROTECT

Overrides the “Protect” parameter for BIND.

NOTRO

Overrides the “TRO” parameter for BIND.

REP*string1*|*string2*

Allows you to replace all occurrences of *string1* with *string2* in the disk image file. You need to use the | (ASCII 7Ch) delimiter to delimit the string values.

Use this parameter to reconfigure a disk image file during the RPL process. It is useful for tailoring files such as AUTOEXEC.BAT or CONFIG.SYS to a specific workstation.

These are the conventions for using REP:

- ◆ The search is case-sensitive.  
The Bootstrap program searches for *string1* exactly as it is entered in BOOTCONF.SYS file.
- ◆ All occurrences of *string1* are replaced with *string2* in the disk image file.
- ◆ *String2* must be equal to or shorter than *string1*.
- ◆ If *string2* is shorter than *string1*, the disk image file is padded with ASCII blanks when the substitution is made.
- ◆ *String2* must not contain embedded ASCII blanks.  
REP interprets the first blank as the end of the string.

### How Does REP Work?

RPL.NLM parses the node address line of BOOTCONF.SYS looking for these strings. If an entry is found, but doesn't match one of the strings, REP assumes it to be a disk image filename.

Therefore, you shouldn't have a disk image file named the same as any of these keywords.

An example of a BOOTCONF.SYS line using REP is as follows:

```
0x*1234 = NET$DOS.SY REP NODE ADDRESS ^^^^  
NODE ADDRESS 67890
```

The example shows that the string NODE ADDRESS ^^^^ will be replaced with NODE ADDRESS 67890 wherever it occurs in the disk image file.

## BOOTCONF.SYS Extensions

When RPL.NLM loads it searches the SYS:\LOGIN directory of the NetWare server for a BOOTCONF.SYS file. If it finds the file, it reads the file into a memory buffer so that it can parse the file when a FIND frame is received from a workstation.



Note

The parsing of BOOTCONF.SYS is done by RPL.NLM, and not the Bootstrap program, to minimize the amount of traffic on the network during the RPL process.

The extensions to BOOTCONF.SYS are given in this section.

### Using Wildcard Characters in BOOTCONF.SYS

Wildcard characters (\* and ?) are allowed in the line specifying the node address of the workstation. This allows the network administrator more flexibility in building the BOOTCONF.SYS file.



The rules for these wildcard characters are as follows:

- ◆ Use the asterisk (\*) character to specify a range of digits in the node address.

For example, if the node address of the workstation is 10005A123456, it may be specified as 0x\*123456 in BOOTCONF.SYS. In this example, RPL.NLM matches the node address with any node address that ends in 123456.

- ◆ Only one asterisk (\*) can appear in the node address.
- ◆ Use the question mark character to specify any single digit in the node address.

In the earlier example, the node address could be specified as 0x?????123456, which is equivalent to 0x\*123456.

You can use wildcard characters to specify a default disk image file for all workstations on the network that don't have a specific disk image file.

You do this by placing one of these lines as the last line in the BOOTCONF.SYS file:

**0x\* = DEFAULT.SYS**

**0x???????????? = DEFAULT.SYS**

Either one of these lines matches on all workstation node addresses. The DEFAULT.SYS (or any name you choose) disk image file is generated by DOSGEN, like any disk image file.

## Specifying More than One Disk Image File per Node Address

Each line in BOOTCONF.SYS that contains a node address can specify more than one disk image filename, separated by one or more blank characters.

The Bootstrap program presents the workstation user a prompt at boot up time to select the disk image file to boot.

For example, if a workstation's node address is 10005A123456, the line

**0x10005a123456 = ONE.SYS TWO.SYS THREE.DOS**

causes the Bootstrap program to display this on the screen:

```
ONE.SYS TWO.SYS THREE.DOS
```

The Bootstrap program then uses NCP calls to open the selected disk image file.

If the filenames don't exist, the screen displays the following:

```
Unable to OPEN Disk Image File
```

Then the Bootstrap program retries the operation.

You can enter up to 10 disk image filenames for each node address in BOOTCONF.SYS file.



The filenames must be separated by one or more blank characters, and they must all fit on one line.

## Allowing Multiple Lines per Node Address

The ASCII colon (:) can be used to allow for multiple lines when you specify a particular node address in BOOTCONF.SYS file. It is provided for convenience when specifying multiple parameters on the node address line.

To use this feature, place the ASCII colon (:) at the end of the line. Note that it must be preceded by at least one ASCII blank:

```
0x10005a460025 = NET$DOS.SYS DOS1.SYS :  
DOS2.SYS
```



## appendix **D** *System Messages*

### Understanding System Messages

System messages point to a software or hardware error that doesn't allow further processing. An explanation of the nature of the message and a recommended course of action follow each message listed below.

AUTO-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

**Explanation** The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

**Action** Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

AUTO-1.00-5: **<Module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

**Explanation** The **<module2>**.VLM requires that the **<module1>**.VLM be loaded first. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

AUTO-1.00-7: In order to load the **<module3>.VLM**, one of the following VLMs must be loaded: **<module1>.VLM**, **<module2>.VLM**.

**Explanation** The **<module3>.VLM** requires that **<module1>.VLM** or **<module2>.VLM** be loaded first. Either the current configuration has **<module3>.VLM** loading before **<module1>.VLM** or **<module2>.VLM**, or **<module1>.VLM** or **<module2>.VLM** did not load successfully.

**Action** Make sure that **<module1>.VLM** or **<module2>.VLM** loads successfully before **<module3>.VLM**.

AUTO-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

AUTO-1.00-43: There is a missing or invalid ON/OFF value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

AUTO-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

BIND-1.00-1: The message file <**name**> is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

BIND-1.00-5: <**Module1**>.VLM is not loaded. The <**module2**>.VLM file cannot be loaded before <**module1**>.VLM. Load the <**module1**>.VLM file first then try to load the <**module2**>.VLM file.

Explanation The <**module2**>.VLM requires that the <**module1**>.VLM be loaded first. Either the current configuration has <**module1**>.VLM and <**module2**>.VLM loading out of order, or <**module1**>.VLM did not load successfully.

Action Make sure that <**module1**>.VLM is configured to load before <**module2**>.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

BIND-1.00-7: In order to load the <**module3**>.VLM, one of the following VLMs must be loaded: <**module1**>.VLM, <**module2**>.VLM.

Explanation The <**module3**>.VLM requires that <**module1**>.VLM or <**module2**>.VLM be loaded first. Either the current configuration has <**module3**>.VLM loading before <**module1**>.VLM or <**module2**>.VLM, or <**module1**>.VLM or <**module2**>.VLM did not load successfully.

Action Make sure that <**module1**>.VLM or <**module2**>.VLM loads successfully before <**module3**>.VLM.

BIND-1.00-42: There is a missing or invalid value for '<**parameter**>' on line <**number**> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

BIND-1.00-43: There is a missing or invalid ON/OFF value for '<*parameter*>' on line <*number*> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

BIND-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

CONN-1.00-1: The message file <*name*> is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

CONN-1.00-5: <*Module1*>.VLM is not loaded. The <*module2*>.VLM file cannot be loaded before <*module1*>.VLM. Load the <*module1*>.VLM file first then try to load the <*module2*>.VLM file.

Explanation The <*module2*>.VLM requires that the <*module1*>.VLM be loaded first. Either the current configuration has <*module1*>.VLM and <*module2*>.VLM loading out of order, or <*module1*>.VLM did not load successfully.

Action Make sure that <*module1*>.VLM is configured to load before <*module2*>.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

CONN-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

**Explanation** The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded first. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

CONN-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

CONN-1.00-43: There is a missing or invalid ON/OFF value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

CONN-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

CONN-1.00-50: DOS version is not 3.1 or later. The NetWare Requester for DOS cannot be loaded. Reboot your computer with DOS v3.1 or later; then load the NetWare Requester for DOS files.

Explanation The DOS Requester requires DOS version v3.1 or above to operate. The current DOS version is not v3.1 or later so the DOS Requester cannot be loaded.

Action Upgrade the DOS version on your machine to v3.1 or above.

CONN-1.00-51: An older version of the shell is loaded. The NetWare Requester for DOS cannot be loaded. Unload the shell; then load the NetWare Requester for DOS files.

Explanation The DOS Requester cannot be loaded with the NetWare shell. The NetWare shell has been loaded in your machine. For NetWare shell compatibility, use NETX.VLM rather than loading the NetWare shell.

Action Type “NETX /U” to unload the NetWare shell. If the version of the NetWare shell in use does not support the “/U” parameter, reboot the machine without loading the NetWare shell. See “NetWare DOS Requester Parameters” on page 116 for more information.

CONN-1.00-52: The NetWare DOS Named Pipes Extender is currently loaded. The NetWare Requester for DOS cannot be loaded. Unload the NetWare DOS Named Pipes Extender; then load the NetWare Requester for DOS files.

Explanation The DOS Requester cannot be loaded after the NetWare DOS Named Pipes Extender. The NetWare DOS Pipes Extender has been loaded in this machine. To use both the DOS Requester and NetWare DOS Named Pipes Extender, load the DOS Requester first.

Action Type “DOSNP /U” to unload the DOS Named Pipes Extender; then load the DOS Requester before reloading the DOS Named Pipes Extender. See “NetWare DOS Requester Parameters” on page 116 for more information.

DOSCLINST-1.0-1: <**Filename**> could not be installed.

Explanation The indicated file could not be installed on the destination drive because the file was not found on the CD-ROM or master diskettes, or the INSTALL utility was unable to write.

Action Make sure that the file is on the master. Also make sure that the destination drive is not write-protected or has some other condition that would prevent the indicated file from being copied.



DOSCLINST-1.0-2: This program requires more disk space. Installation could not be completed.

**Explanation** Adequate disk space is not available on the destination drive. More space must be available for the installation to be completed.

**Action** Do one or more of the following:

- ◆ Delete unnecessary files from the hard disk.
- ◆ Change the SET server utility's "File Delete Wait Time" parameter so that files are purged immediately rather than being retained in a salvageable state on the volume. See information on SET parameters in "Managing Server Hard Disk Space" in *Supervising the Network*.
- ◆ Use FILER to purge deleted files if they cannot be purged automatically. (They are using up directory table space.)

- ◆ Increase the Hot Fix redirection area using INSTALL.NLM. The Hot Fix area typically contains 2% of the entire disk space. To change the Hot Fix on an existing drive, back up all the data on the partition, delete the volumes on the partition, and delete the partition; then recreate it. Assign the partition a different percentage to Hot Fix; then recreate the volumes and restore the data.
- ◆ Delete files or increase the percentage of disk space that can be used by a directory. If the disk or volume has space available, check the disk drives and channel to see if a failure has occurred.

See “Managing Server Hard Disk Space” in *Supervising the Network* for more information.

#### DOSCLINST-1.0-3: The INSTALL.OVL file could not be found.

- |             |   |
|-------------|---|
| Explanation | The INSTALL.OVL file could not be found. This file is an integral piece of the INSTALL utility; it must be in the same directory as the INSTALL.EXE file. |
| Action      | Make sure that INSTALL.OVL is in the same directory as the INSTALL.EXE file; then run INSTALL again.  |

#### DOSCLINST-1.0-4: A read error occurred while the program was reading INSTALL.OVL.

- |             |   |
|-------------|---|
| Explanation | The file INSTALL.OVL has been corrupted. The INSTALL utility requires INSTALL.OVL for successful installation.              |
| Action      | Copy INSTALL.OVL from the CD-ROM or master diskettes to the same directory as the INSTALL.EXE file; then run INSTALL again. |

#### FIO-1.00-1: The message file <name> is invalid. The program cannot be loaded.

- |             |   |
|-------------|---|
| Explanation | The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.                       |
| Action      | Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files. |

#### FIO-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

- |             |   |
|-------------|---|
| Explanation | The <module2>.VLM requires that the <module1>.VLM be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully. |
|-------------|---|

**Action** Make sure that *<module1>.VLM* is configured to load before *<module2>.VLM*. To do this, change the load order of the “VLM=” parameter in the NET.CFG file.

FIO-1.00-7: In order to load the *<module3>.VLM*, one of the following VLMs must be loaded: *<module1>.VLM*, *<module2>.VLM*.

**Explanation** The *<module3>.VLM* requires that *<module1>.VLM* or *<module2>.VLM* be loaded first. Either the current configuration has *<module3>.VLM* loading before *<module1>.VLM* or *<module2>.VLM*, or *<module1>.VLM* or *<module2>.VLM* did not load successfully.

**Action** Make sure that *<module1>.VLM* or *<module2>.VLM* loads successfully before *<module3>.VLM*.

FIO-1.00-42: There is a missing or invalid value for ‘*<parameter>*’ on line *<number>* of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

FIO-1.00-43: There is a missing or invalid ON/OFF value for ‘*<parameter>*’ on line *<number>* of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

FIO-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

FIO-1.00-60: Too much cache is configured. The FIO.VLM file will reduce the cache blocks by <**number**> blocks and load successfully. Check the CACHE BUFFERS and BUFFER SIZE parameters in the NET.CFG file; then load the FIO.VLM file.

**Explanation** The amount of cache available in DOS is limited by conventional memory. The parameters determining the amount of memory to be used are “Cache Buffers” and “Buffer Size.” These two parameters cannot be configured to use more than 64 KB of memory.

**Action** Reduce the “Cache Buffers” or “Buffer Size” parameter in the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

FIO-1.00-61: The IPX interface is not loaded. The FIO.VLM file will load successfully without packet burst support. Load the IPXODI v2.00 or later to use packet burst or add PB BUFFERS=0 to the NET.CFG file; then load FIO.VLM.

**Explanation** This message is only a warning. The DOS Requester will function properly without the use of Packet Burst. The Packet Burst protocol, which is a part of FIO.VLM, requires IPXODI version 2.00 or above to function properly. No IPX interface was loaded in this machine.

**Action** If you want Packet Burst, make sure that IPXODI.COM version 2.00 or above is loaded before the DOS Requester. If you do not want Packet Burst, add “PB BUFFERS=0” to the “NetWare DOS Requester” section of the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

FIO-1.00-62: The LSL interface is not loaded. The FIO.VLM file will load successfully without packet burst support. Load the LSL module or add PB BUFFERS=0 to the NET.CFG file; then load FIO.VLM.

**Explanation** This message is only a warning. The DOS Requester will function properly without the use of Packet Burst. The Packet Burst protocol, which is a part of FIO.VLM, requires the LSL.COM file to be loaded to function properly.

**Action** If you want Packet Burst, make sure that the LSL.COM file is loaded before the DOS Requester. If you do not want Packet Burst, add “PB BUFFERS=0” to the “NetWare DOS Requester” section of the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

FIO-1.00-63: The IPX socket for packet burst could not be opened. The FIO.VLM file will load successfully without packet burst support. configure the IPXODI.COM file for enough sockets in the NET.CFG file or add PB BUFFERS=0 to the NET.CFG file; then load the FIO.VLM file.

**Explanation** This message is only a warning. The DOS Requester will function properly without the use of Packet Burst. The Packet Burst protocol, which is a part of the FIO.VLM module, requires an IPX socket to function properly. The request to open a socket failed indicating that not enough IPX sockets are available.

**Action** If you want Packet Burst, make sure that the IPX interface is configured for enough sockets in the NET.CFG file before loading the DOS Requester. If you do not want Packet Burst, add "PB BUFFERS=0" to the "NetWare DOS Requester" section of the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

FIO-1.00-64: The IPX interface is not version 2.00 or later. The FIO.VLM file will load successfully without packet burst support. Load the IPXODI.COM file version 2.00 or later or add PB BUFFERS=0 to the NET.CFG file; then load the FIO.VLM file.

**Explanation** This message is only a warning. The DOS Requester will function properly without the use of Packet Burst. The Packet Burst protocol, which is a part of FIO.VLM, requires IPXODI version 2.00 or later to function properly. The version of IPXODI loaded was not version 2.00 or later.

**Action** If you want Packet Burst, make sure that the IPXODI.COM file version 2.00 or later is loaded before the DOS Requester. If you do not want Packet Burst, add "PB BUFFERS=0" to the "NetWare DOS Requester" section of the NET.CFG file.

FIO-1.00-65: The DOS Requester is being loaded in a Windows DOS box. The FIO.VLM file will load successfully without packet burst support.

**Explanation** This message is only a warning. The DOS Requester will function properly without the use of Packet Burst. The Packet Burst protocol, which is a part of FIO.VLM, cannot function when loaded in a Windows DOS box.

**Action** You can use Packet Burst if it is loaded before starting Windows. To do this, exit Windows and load the VLMs; then start Windows.

GENERAL-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

**Explanation** The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

**Action** Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

GENERAL-1.00-5: **<Module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

**Explanation** The **<module2>**.VLM requires that the **<module1>**.VLM be loaded first. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

GENERAL-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

**Explanation** The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded first. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

GENERAL-1.00-42: There is a missing or invalid value for '<*parameter*>' on line <*number*> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

GENERAL-1.00-43: There is a missing or invalid ON/OFF value for '<*parameter*>' on line <*number*> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time VLM.EXE is loaded, delete or correct the line specified by the error message.

GENERAL-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action Check and correct the parameter in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

IPXNCP-1.00-1: The message file <*name*> is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

IPXNCP-1.00-5: <**Module1**>.VLM is not loaded. The <**module2**>.VLM file cannot be loaded before <**module1**>.VLM. Load the <**module1**>.VLM file first then try to load the <**module2**>.VLM file.

- Explanation The <*module2*>.VLM requires that the <*module1*>.VLM be loaded first. Either the current configuration has <*module1*>.VLM and <*module2*>.VLM loading out of order, or <*module1*>.VLM did not load successfully.
- Action Make sure that <*module1*>.VLM is configured to load before <*module2*>.VLM. To do this, change the load order of the “VLM=” parameter in the NET.CFG file.

IPXNCP-1.00-7: In order to load the <**module3**>.VLM, one of the following VLMs must be loaded: <**module1**>.VLM, <**module2**>.VLM.

- Explanation The <*module3*>.VLM requires that <*module1*>.VLM or <*module2*>.VLM be loaded first. Either the current configuration has <*module3*>.VLM loading before <*module1*>.VLM or <*module2*>.VLM, or <*module1*>.VLM or <*module2*>.VLM did not load successfully.
- Action Make sure that <*module1*>.VLM or <*module2*>.VLM loads successfully before <*module3*>.VLM.

IPXNCP-1.00-42: There is a missing or invalid value for ‘<**parameter**>’ on line <**number**> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

- Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.
- Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

IPXNCP-1.00-43: There is a missing or invalid ON/OFF value for ‘<**parameter**>’ on line <**number**> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

- Explanation The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.



**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

**IPXNCP-1.00-45:** The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

**IPXNCP-1.00-53:** The IPX interface is not loaded. The IPXNCP.VLM file cannot be loaded. Load the IPXODI.COM file first and then try loading the IPXNCP.VLM file.

**Explanation** An attempt was made to load the IPXNCP.VLM file without having previously loaded the IPX interface.

**Action** Load IPXODI.COM before attempting to load IPXNCP.VLM.

**IPXNCP-1.00-54:** The IPX sockets could not be opened. The IPXNCP.VLM file cannot be loaded. Configure the IPXODI.COM file for enough sockets in the NET.CFG file and then try to load the IPXNCP.VLM file.

**Explanation** The IPXNCP.VLM file failed to open the IPX sockets needed in order to run. The IPXNCP.VLM module requires that four or more IPX sockets be available to run.

**Action** Increase the number of IPX sockets available by using the “IPX SOCKETS=” parameter in the NET.CFG file; then reload IPXODI.COM and IPXNCP.VLM.

**IPXNCP-1.00-55:** The IPX interface does not support checksums. The IPXNCP.VLM file will load successfully without using checksums. Make sure the installed IPXODI.COM is version 2.01 or later and that it is not bound to a board configured to use the ETHERNET\_802.3 frame format.

**Explanation** IPXNCP.VLM will load without supporting checksums. For checksum to be supported in IPXNCP.VLM, the loaded IPXODI.COM file must support checksums. Either the loaded version of IPXODI.COM does not have checksum support, or the protocol bound to IPX does not support checksums.

**Action** No action required at this point. However, to avoid receiving this message the next time the IPXNCP.VLM file is loaded, either load IPXODI.COM with checksum support enabled, or add "CHECKSUM=OFF" to the NET.CFG file.

IPXNCP-1.00-56: The IPX socket for large internet packets could not be opened. The IPXNCP.VLM file will load successfully without using large internet packets. Configure the IPXODI.COM file for enough sockets in the NET.CFG file or add LARGE INTERNET PACKETS=OFF to the NET.CFG file; then load the IPXNCP.VLM file.

**Explanation** This message is only a warning. The DOS Requester will function properly without the use of large internet packets. The large internet packet protocol, which is a part of the IPXNCP.VLM module, requires an IPX socket to function properly. The request to open a socket failed, indicating that not enough IPX sockets are available.

**Action** If you want large internet packets, make sure that the IPX interface is configured for enough sockets in the NET.CFG file before loading the DOS Requester. If you do not want large internet packets, add "LARGE INTERNET PACKETS=OFF" to the "NetWare DOS Requester" section of the NET.CFG file.

NDS-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

**Explanation** The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

**Action** Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

NDS-1.00-5: **<Module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

**Explanation** The **<module2>**.VLM requires that the **<module1>**.VLM be loaded first. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

NDS-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

**Explanation** The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded first. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

NDS-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NDS-1.00-43: There is a missing or invalid ON/OFF value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NDS-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

NETX-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

**Explanation** The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

**Action** Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

NETX-1.00-5: **<Module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

**Explanation** The **<module2>**.VLM requires that the **<module1>**.VLM be loaded first. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

NETX-1.00-7: In order to load the **<module3>.VLM**, one of the following VLMs must be loaded: **<module1>.VLM**, **<module2>.VLM**.

**Explanation** The **<module3>.VLM** requires that **<module1>.VLM** or **<module2>.VLM** be loaded before **<module3>.VLM** can run effectively. Either the current configuration has **<module3>.VLM** loading before **<module1>.VLM** or **<module2>.VLM**, or **<module1>.VLM** or **<module2>.VLM** did not load successfully.

**Action** Make sure that **<module1>.VLM** or **<module2>.VLM** loads successfully before **<module3>.VLM**.

NETX-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NETX-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NETX-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

NETX-1.00-57: DOS is only configured for <number> drives, NETX.VLM requires 26 drives for full functionality. The NETX.VLM file will load with partial support. Add LASTDRIVE=Z to the CONFIG.SYS file and reboot the workstation; then load the NETX.VLM file.

**Explanation** This message is only a warning. For full NetWare shell compatibility, you must set the "Lastdrive" parameter in the CONFIG.SYS file to Z. Otherwise, problems will result in mapping drives, etc.

**Action** Add "LASTDRIVE=Z" to the CONFIG.SYS file; then reboot the workstation before loading the DOS Requester.

NETX-1.00-58: The PRINT.VLM file has not been loaded. The NETX.VLM file will load successfully without print services. To enable printing services, load the PRINT.VLM file before loading the NETX.VLM file.

**Explanation** This message is only a warning. For full NetWare shell compatibility, the PRINT.VLM file must be loaded. Otherwise, problems will arise when using pre-NetWare v4.0 print utilities.

**Action** If you want print functionality, load the PRINT.VLM before loading NETX.VLM.

NMR-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

NMR-1.00-5: **<Module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

Explanation The **<module2>**.VLM requires that the **<module1>**.VLM be loaded first. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

Action Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

NMR-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

Explanation The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded first. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

Action Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

NMR-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NMR-1.00-43: There is a missing or invalid ON/OFF value for '<*parameter*>' on line <*number*> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NMR-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

NWDRV-3.00-00: The NetWare VLM is not loaded or is not configured correctly. All network functions will be disabled. Reconfigure the VLM and restart Windows.

**Explanation** Without the NetWare DOS Requester (VLM.EXE and .VLM files), the workstation cannot communicate with NetWare servers. The NetWare Windows driver (NETWARE.DRV) depends on the VLM in order to provide network support for Windows.

**Action** Exit Windows. Go to the directory where the VLM.EXE is located and type "VLM." For more information about the NetWare VLM, See "Virtual Loadable Modules (VLMs)" on page 74.

NWDRV-3.00-10: The VNETWARE.386 file has not been loaded. All network functions will be disabled. To use NetWare, install VNETWARE.386 and restart Windows.

**Explanation** The NetWare Windows driver (NETWARE.DRV) depends on the VNETWARE.386 file to run Windows in enhanced mode.

**Action** Exit Windows. Copy VNETWARE.386 to the Windows subdirectory (for example, C:\WINDOWS). VNETWARE.386 should be on the *WSWIN\_1* diskette. (VNETWARE.386 is located on CD-ROM under CLIENT\DOSWIN\WSWIN\_1.) Also make sure that the "[386Enh]" section of the SYSTEM.INI file has the following line:



**NETWORK=\*VNETBIOS, VNETWARE.386, VIPX.386**

Then restart Windows.

NWDRV-3.00-20: There was a problem loading or executing the NetWare Directory Services support libraries. All NetWare Directory Services functions are now disabled.

Explanation To access NetWare Directory Services, the Windows driver (NETWARE.DRV) depends on the following .DLL files:

**NWNET . DLL**

**NWLOCALE . DLL**

**NWCCALLS . DLL**

Action Exit Windows. Copy these .DLL files to the Windows subdirectory (for example, C:\WINDOWS). The .DLL files are located on the *WSWIN\_1* diskette. (The .DLL files are located on CD-ROM under CLIENT\DOSWIN\WSWIN\_1.) Restart Windows.

NWDRV-3.00-30: There was a problem loading the Unicode tables. All NetWare Directory Services functions are now disabled.

**Explanation** Unicode tables support international languages. They are located in the .001 files (for example, 437\_UNI.001). In the DOS, Windows, and OS/2 environments, the program searches the following areas for the Unicode files:

- ◆ The current directory (for DOS and OS/2 environments) or the Windows directory.
- ◆ The load directory (the directory containing the current utility's .EXE file). Normally this is your PUBLIC directory.
- ◆ The NLS directory that is located within the load directory (the directory containing the current utility's .EXE file). If the Unicode files are not found there, the program searches the NLS directory that is a sibling of the load directory (that is, at the same level as the load directory). Note: The NWLanguage environment variable does not apply here.
- ◆ The PATH and DPATH (for OS/2 only) environment variables. These environment variables may be found either in the user's login script or in the AUTOEXEC.BAT file.
- ◆ The following four files must be in one of these locations of the user's search drive, or this error message is displayed:

```
<code page>_UNI.<country id>  
UNI_<code page>.<country id>  
UNI_MON.<country id>  
UNI_COL.<country id>
```

**Action** While this error can be caused by insufficient memory, insufficient rights, or even file corruption, it usually results from the files not being found in the correct path. The best action to take is to copy the .001 files from the *WSWIN\_1* diskette to your Windows subdirectory or any other directory that is in your PATH environment setting. Make sure that you have selected a language for Windows that is supported by NetWare. Do this by choosing the "International" icon from the "Control Panel."

NWDRV-3.00-50: The temporary drive could not be mapped. Contact the network administrator.

**Explanation** The NetWare Windows driver (NETWARE.DRV) allocates temporary drives on the server to perform some functions.

Action Contact the network administrator to make sure that NetWare is running and the server is operating correctly.

NWP-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

NWP-1.00-5: **<module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

Explanation The **<module2>**.VLM requires that the **<module1>**.VLM be loaded. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

Action Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

NWP-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

Explanation The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

Action Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

NWP-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NWP-1.00-43: There is a missing or invalid ON/OFF value for '<*parameter*>' on line <*number*> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NWP-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

NWP-1.00-59: The SECURITY.VLM file has not been loaded. The NWP.VLM file will load successfully without NCP signature support. Load the SECURITY.VLM file before loading the NWP.VLM file or add SIGNATURE LEVEL=0 to the NET.CFG file; then load NWP.VLM.

**Explanation** This message is only a warning. The DOS Requester will function properly without NCP authentication. For NCP authentication to operate properly, the SECURITY.VLM module must be loaded before NWP.VLM. Either the NWP.VLM is being loaded before SECURITY.VLM or SECURITY.VLM did not load successfully.

**Action** If you want NCP authentication, make sure that the SECURITY.VLM is configured to load before NWP.VLM and that it loads successfully before attempting to load NWP.VLM. If you do not want NCP authentication, add "SIGNATURE LEVEL=0" to the "NetWare DOS Requester" section of the NET.CFG file.

REDIR-1.00-1: The message file <name> is invalid. The program cannot be loaded.

**Explanation** The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

**Action** Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

REDIR-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

**Explanation** The <module2>.VLM requires that the <module1>.VLM be loaded. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

**Action** Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

REDIR-1.00-7: In order to load the **<module3>.VLM**, one of the following VLMs must be loaded: **<module1>.VLM**, **<module2>.VLM**.

**Explanation** The **<module3>.VLM** requires that **<module1>.VLM** or **<module2>.VLM** be loaded. Either the current configuration has **<module3>.VLM** loading before **<module1>.VLM** or **<module2>.VLM**, or **<module1>.VLM** or **<module2>.VLM** did not load successfully.

**Action** Make sure that **<module1>.VLM** or **<module2>.VLM** loads successfully before **<module3>.VLM**.

REDIR-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

REDIR-1.00-43: There is a missing or invalid ON/OFF value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

REDIR-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

RSA-1.00-1: The message file **<name>** is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

RSA-1.00-5: **<Module1>**.VLM is not loaded. The **<module2>**.VLM file cannot be loaded before **<module1>**.VLM. Load the **<module1>**.VLM file first then try to load the **<module2>**.VLM file.

Explanation The **<module2>**.VLM requires that the **<module1>**.VLM be loaded first. Either the current configuration has **<module1>**.VLM and **<module2>**.VLM loading out of order, or **<module1>**.VLM did not load successfully.

Action Make sure that **<module1>**.VLM is configured to load before **<module2>**.VLM. To do this, change the load order of the "VLM=" parameter in the NET.CFG file.

RSA-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

Explanation The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded first. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

Action Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

RSA-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

RSA-1.00-43: There is a missing or invalid ON/OFF value for '<*parameter*>' on line <*number*> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

RSA-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

SECURITY-1.00-1: The message file <*name*> is invalid. The program cannot be loaded.

**Explanation** The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

**Action** Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.



SECURITY-1.00-5: <**Module1**>.VLM is not loaded. The <**module2**>.VLM file cannot be loaded before <**module1**>.VLM. Load the <**module1**>.VLM file first then try to load the <**module2**>.VLM file.

- Explanation The <*module2*>.VLM requires that the <*module1*>.VLM be loaded first. Either the current configuration has <*module1*>.VLM and <*module2*>.VLM loading out of order, or <*module1*>.VLM did not load successfully.
- Action Make sure that <*module1*>.VLM is configured to load before <*module2*>.VLM. To do this, change the load order of the “VLM=” parameter in the NET.CFG file.

SECURITY-1.00-7: In order to load the <**module3**>.VLM, one of the following VLMs must be loaded: <**module1**>.VLM, <**module2**>.VLM.

- Explanation The <*module3*>.VLM requires that <*module1*>.VLM or <*module2*>.VLM be loaded first. Either the current configuration has <*module3*>.VLM loading before <*module1*>.VLM or <*module2*>.VLM, or <*module1*>.VLM or <*module2*>.VLM did not load successfully.
- Action Make sure that <*module1*>.VLM or <*module2*>.VLM loads successfully before <*module3*>.VLM.

SECURITY-1.00-42: There is a missing or invalid value for ‘<**parameter**>’ on line <**number**> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

- Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.
- Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

SECURITY-1.00-43: There is a missing or invalid ON/OFF value for ‘<**parameter**>’ on line <**number**> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

- Explanation The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.
- Action No action is required at this point. However, to avoid this message the next time VLM.EXE is loaded, delete or correct the line specified by the error message.

SECURITY-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

- Explanation A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.
- Action Check and correct the parameter specified in the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

SECURITY-1.00-67: The SECURITY.VLM file must be loaded before any other NetWare Protocol Module. The SECURITY.VLM file will not be loaded. Load the SECURITY.VLM file before loading any NetWare Protocol Modules.

- Explanation This message is only a warning. The DOS Requester will function properly without NCP authentication. The SECURITY.VLM file must be loaded before any NetWare Protocol Modules (BIND.VLM or NDS.VLM) to function properly.
- Action If you want authentication, load the SECURITY.VLM after TRAN.VLM and before any NetWare Protocol Module (BIND.VLM or NDS.VLM); then set the “Signature Level=” parameter in the NET.CFG file. If you do not want NCP authentication, set “SIGNATURE LEVEL=0” in the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

TRAN-1.00-1: The message file <*name*> is invalid. The program cannot be loaded.

- Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.
- Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

TRAN-1.00-5: <*Module1*>.VLM is not loaded. The <*module2*>.VLM file cannot be loaded before <*module1*>.VLM. Load the <*module1*>.VLM file first then try to load the <*module2*>.VLM file.

- Explanation The <*module2*>.VLM requires that the <*module1*>.VLM be loaded first. Either the current configuration has <*module1*>.VLM and <*module2*>.VLM loading out of order, or <*module1*>.VLM did not load successfully.
- Action Make sure that <*module1*>.VLM is configured to load before <*module2*>.VLM. To do this, change the load order of the “VLM=” parameter in the NET.CFG file.

TRAN-1.00-7: In order to load the **<module3>**.VLM, one of the following VLMs must be loaded: **<module1>**.VLM, **<module2>**.VLM.

**Explanation** The **<module3>**.VLM requires that **<module1>**.VLM or **<module2>**.VLM be loaded first. Either the current configuration has **<module3>**.VLM loading before **<module1>**.VLM or **<module2>**.VLM, or **<module1>**.VLM or **<module2>**.VLM did not load successfully.

**Action** Make sure that **<module1>**.VLM or **<module2>**.VLM loads successfully before **<module3>**.VLM.

TRAN-1.00-42: There is a missing or invalid value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

TRAN-1.00-43: There is a missing or invalid ON/OFF value for '**<parameter>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

**Explanation** The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

TRAN-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

**Explanation** A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See "NetWare DOS Requester Parameters" on page 116 for more information about NET.CFG parameters.

VLM-1.00-1: The message file <*name*> is invalid. The program cannot be loaded.

Explanation The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action Either update the .MSG file with a valid copy or delete the file. The DOS Requester will use the default messages that are bound to the binary files.

VLM-1.00-8: The VLM.EXE file is already loaded. VLM.EXE cannot be reloaded. If you want to load VLM.EXE with a different configuration, unload VLM.EXE first with the /U parameter and then try loading the VLM.EXE file.

Explanation The VLM.EXE file has already been loaded into memory. There cannot be two copies of the VLM.EXE file loaded in memory at one time.

Action If the VLM.EXE file is being reloaded in an attempt to reconfigure the VLMs already loaded, you must first unload VLM.EXE using the "/U" parameter; then reload VLM.EXE.

VLM-1.00-9: The VLM.EXE file is testing the VLMs.

Explanation This message is used to show the progress the VLM.EXE file is making in configuring the VLMs to be loaded. Because this process can take a few seconds, the progress is displayed using a period (.) to indicate each VLM being tested.

Action No action is required; this is simply an information message.

VLM-1.00-10: The conventional memory block for the EMS stack cannot be resized. The VLM.EXE file cannot be loaded. DOS memory is probably corrupt. Reboot your computer and then try to load the VLM.EXE again.

Explanation This error indicates that the DOS memory chain has been corrupted.

Action It is unsafe to continue operating the computer. The best solution is to reboot the computer; then try to load the VLM.EXE.

VLM-1.00-11: There is insufficient memory to load the VLMs. The VLM.EXE file cannot be loaded. Reconfigure the VLMs to be loaded in the NET.CFG file and then try to load the VLM.EXE file.

Explanation The VLM.EXE file has determined the amount of memory required by the VLMs to be loaded, and there is not enough memory to load the VLMs. The VLM.EXE file can use expanded memory (EMS), extended memory (XMS),

or conventional memory. This message means that there is not enough of any one of these to support the VLMs to be loaded.

**Action** Check the amount of memory available (including expanded memory and extended memory). Some memory managers provide dynamic memory pools by converting extended memory to expanded memory or vice versa. Other memory managers require the user to configure the amount and type of memory you want. In either case, make sure there is enough memory of one type to support the VLMs to be loaded or reduce the number of VLMs to be loaded, either by using the “VLM=” parameter in the NET.CFG file or by renaming the VLMs that you do not want.

#### VLM-1.00-15: An invalid command line parameter was specified.

**Explanation** The VLM.EXE file was loaded with an invalid command line parameter.

**Action** Run VLM.EXE with the “/?” parameter (for example: VLM/?) to display the valid parameters; then load the VLM.EXE file with a valid command line parameter.

VLM-1.00-16: The VLM.EXE file <*version*> is currently loaded.

Explanation This message is displayed during the VLM.EXE file diagnostics display. (The diagnostics display is accessed by running VLM.EXE with the “/D” parameter.)

Action No action is required; this is simply an information message.

VLM-1.00-17: The VLM.EXE file is not currently loaded.

Explanation This message is displayed during the VLM.EXE file diagnostics display. (The diagnostics display is accessed by running VLM.EXE with the “/D” parameter.)

Action No action is required; this is simply an information message.

VLM-1.00-30: A task switcher has been detected in memory. The VLM.EXE file cannot be loaded under a task switcher. Exit the task switcher; then try again.

Explanation The DOS Requester cannot function properly when loaded after a task switcher. A task switcher has been loaded. (Task switchers include DR DOS TaskMax, MS DOS 5.0 DOSSHELL, and MS Windows 3.1 in standard mode.)

Action Exit the task switcher before attempting to load the DOS Requester.

VLM-1.00-31: Network error on server <*name*>. Check network cabling or server status.

Explanation This is a critical communications error between the workstation and the NetWare server. This error may also appear as “General failure on device NETWORK.” A communications error can be caused by either a hardware or software failure.

Action Check the hardware and make sure that the network cable is connected properly and that the NetWare server is up and running properly. If the condition persists, report the problem to your NetWare support engineer.

VLM-1.00-32: The VLM.EXE file is not loaded. VLM.EXE cannot be unloaded.

Explanation An attempt was made to unload the VLM.EXE file using the “/U” parameter.

Action The action that caused this message to occur is invalid, so no further action is required.

VLM-1.00-33: The loaded VLM.EXE file has a different version. VLM.EXE cannot be unloaded. Make sure the VLM.EXE file you are using has the same version number and then try to unload VLM.EXE.

- Explanation An attempt was made to unload the VLM.EXE file using the “/U” parameter.
- Action Use the same version of the VLM.EXE file to try the unload as was previously loaded in memory; otherwise, the VLM.EXE file that is loaded in memory can be removed only by rebooting the computer.

VLM-1.00-34: The loaded VLM.EXE file indicates it is unsafe to execute an unload for VLM number <**number**>. VLM.EXE will not be unloaded. Unload all memory resident programs (TSRs) that were loaded after the VLM.EXE file and then try to unload VLM.EXE.

- Explanation An attempt was made to unload the VLM.EXE file using the “/U” parameter. The loaded VLM.EXE file refused to unload. This usually is caused by interrupts being used after the VLM.EXE file was loaded. Another possible cause is that a VLM is in an unsafe state to unload.
- Action Unload all the TSR (terminate-and-stay-resident) programs that were loaded after the VLM.EXE file; then try to unload VLM.EXE. Make sure all loaded VLMs are in a safe state for unloading.

VLM-1.00-36: The VLM.EXE file cannot use expanded memory (EMS). VLM.EXE will use an alternate memory scheme.

- Explanation An attempt was made to load the VLM.EXE file in expanded memory (EMS) using the “/ME” parameter. Either no LIM EMS 4.0 expanded memory manager is present or insufficient expanded memory is available to load the VLMs. The VLM.EXE file will attempt to use an alternate type of memory, either extended memory (XMS) or conventional memory, before failing to load.
- Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, either make sure an expanded memory manager is loaded with sufficient memory to support the VLMs you want, or do not specify the “/ME” parameter when loading VLM.EXE.

VLM-1.00-37: The VLM.EXE file is using expanded memory (EMS).

- Explanation This message indicates that the VLM.EXE file is loading the VLMs into expanded memory (EMS).
- Action No action is required; this is simply an information message.

VLM-1.00-38: The VLM.EXE file cannot use extended memory (XMS). VLM.EXE will use an alternate memory scheme.

**Explanation** An attempt was made to load the VLM.EXE file in extended memory (XMS) using the “/MX” parameter. Either no XMS 2.0 extended memory manager is present or insufficient extended memory is available to load the VLMs. The VLM.EXE file will attempt to use an alternate type of memory, either expanded memory (EMS) or conventional memory, before failing to load.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, either make sure an extended memory manager is loaded with sufficient memory to support the VLMs you want, or do not specify the “/MX” parameter when loading VLM.EXE.

VLM-1.00-39: The VLM.EXE file is using extended memory (XMS).

**Explanation** This message indicates that the VLM.EXE file is loading the VLMs into extended memory (XMS).

**Action** No action is required; this is simply an information message.

VLM-1.00-40: The VLM.EXE file cannot use conventional memory. VLM.EXE will use an alternate memory scheme.

**Explanation** An attempt was made to load the VLM.EXE file in conventional memory using the “/MC” parameter. Insufficient conventional memory is available to load the VLMs. The VLM.EXE file will attempt to use an alternate type of memory, either expanded memory (EMS) or extended memory (XMS), before failing to load.

**Action** No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, either make sure there is sufficient conventional memory to support the VLMs you want, or do not specify the “/MC” parameter when loading VLM.EXE.

VLM-1.00-41: The VLM.EXE file is using conventional memory.

**Explanation** This message indicates that the VLM.EXE file is loading the VLMs into conventional memory.

**Action** No action is required; this is simply an information message.



VLM-1.00-42: There is a missing or invalid value for '*<parameter>*' on line *<number>* of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

VLM-1.00-43: There is a missing or invalid ON/OFF value for '*<parameter>*' on line *<number>* of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

VLM-1.00-44: There is an invalid string length specified for '*<parameter>*' on line *<number>* of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation The configuration file contains a parameter that is defined to use a string of a specified minimum or maximum length, but a string has been entered that does not fit the specified limits. This invalid line will be ignored by the configuration process.

Action No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

VLM-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

**Action** Check and correct the parameter specified in the NET.CFG file. See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

VLM-1.00-46: A duplicate VLM ID was found during VLM load test. The file will not be loaded. check the VLM= and USE DEFAULTS= parameter specified in the NET.CFG file before continuing.

**Explanation** The VLMs for the DOS Requester have been improperly configured to include two VLMs with the same VLM ID. This can be caused by either including a VLM twice in the “VLM=” list, or by a VLM that has attempted to reuse a VLM ID assigned to a different VLM.

**Action** Check the NET.CFG file for duplicates in the VLM list. Make sure that the “Use Defaults=” parameter is set to “OFF.” See “NetWare DOS Requester Parameters” on page 116 for more information about NET.CFG parameters.

VLM-1.00-47: A file server could not be found. Check the network cabling and the server's status before continuing.

**Explanation** No NetWare server was found when attempting to build an initial connection. This could be caused by one of the following:

- ◆ An improperly configured network board
- ◆ Improperly configured IPX
- ◆ A disconnected network cable
- ◆ An attempt to load the DOS Requester before a NetWare server has been initialized.

**Action** Try one or more of the following:

- ◆ Make sure that the network board is properly configured by checking the network board's settings and the NET.CFG file settings. See "Link Driver Parameters" on page 92 for more information about NET.CFG parameters.
- ◆ Make sure that IPX is configured to be bound properly to the right LAN driver.
- ◆ Check the network cabling.
- ◆ Make sure there is a NetWare server up and running before attempting to load the DOS Requester.

**Note:** If the server has not been loaded properly, attempting to access an invalid drive from the workstation will cause the system to attempt to build an initial connection again.

VLM-1.00-48: The preferred connection could not be established. Check the PREFERRED SERVER and PREFERRED TREE statement specified in the NET.CFG file. Also check the server's status before continuing.

**Explanation** An attempt was made to connect to a specific Directory tree or NetWare server. This attempt failed. The server or tree is not running at this time, or (in the preferred server case) the server may be unknown by the initial server's bindery.

**Action** Make sure the spelling of the preferred server or tree is correct. Check the server/tree status to see that it is up and running. Finally, check to see if any

restrictions have been made at the NetWare server to isolate the server from your network.

VLM-1.00-60: There is an unrecognized entry '**<entry>**' on line **<number>** of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

- |             |  |
|-------------|--|
| Explanation | The configuration file contains a parameter under a header where it does not belong. This invalid line will be ignored by the configuration process.                     |
| Action      | No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message. |