

## ODI Contents

This help file contains a list of all Help topics available for ODI. You can use the scroll bar to see the entries that are not currently visible in the Help window. For information on how to use Help, press F1 or choose How to Use Help from the Help menu.



[Overview](#)



[How to](#)  
[Use](#)



[Troubleshooting](#)



[Help](#)

## **Help**

[Contents](#)

[About ODI...](#)

## Overview

**Note:** This appendix is intended ONLY for users that are using the Open Data-Link Interface (ODI) for Windows 3.1 or Windows for Workgroups without Workgroups networking.

This appendix applies to users who are using the Open Data-Link Interface (ODI) for Windows 3.1 or Windows for Workgroups without Workgroups networking.

The Open Data-Link interface (ODI) allows multiple network protocols to be used concurrently on a LAN adapter in a workstation on the network and is an alternative to NDIS. The components of ODI are:

- **Network drivers:** Multiple Link Interface Drivers (MLIDs) are device drivers that handle the sending and receiving of packets to and from a physical or logical LAN media.
- **Link Support Layer (LSL):** Novell's LSL handles the communication between protocol stacks and the network drivers (MLIDs)
- **Protocol Stacks:** Network Layer protocol stacks transmit and receive data over a logical or physical network.

Custom verifies that your system has LSL.COM running during setup time, and if so, automatically installs the version of the stack that supports ODI. After ODI is installed you can access a Novell Netware server, for example, as well as use Chameleon in Windows.

## **How to Use**

These topics describe how to use ODI.

[Setting ODI Manually](#)

[Switching from NDIS to ODI](#)

[Switching from ODINSUP to ODI](#)

[Sample NET.CFG File](#)

[Additional Notes](#)

[ODI Terminology](#)

## Setting ODI Manually

Follow this section if you are performing a *first time* installation.

- Follow steps 1 - 10 if *you do not have* ODI installed on your system and you are installing ChameleonNFS (new installation) for the first time.
- Follow steps 7 - 10 if *you have* ODI on your system and are installing ChameleonNFS (new installation) for the first time.

Once your system is configured for ODI drivers, the ChameleonNFS installation will load the driver NMODI.COM in the AUTOEXEC.BAT file. This driver allows ChameleonNFS to communicate with the ODI drivers.

See also

[Setting Up Your ODI Environment](#)

## Setting Up Your ODI Environment

The files mentioned in the following steps are contained on the Novell ODI diskette:

- LSL.COM
- LAN Driver for network board (for example, 3C503.COM)
- protocol stack file (for example, IPXODI.COM)

**Note:** Use the latest LSL.COM, IPXODI.COM, and NETX files from Novell before setting up your ODI environment.

1. At the DOS prompt, create a directory named C:\ODI.
2. Copy the LSL.COM, driver (3C503.COM for example), protocol stack files (IPXODI.COM), and Netware shell (NETX) to this directory.

3. Append the following lines to the end of the AUTOEXEC.BAT file:

```
C:\ODI\LSL.COM  
C:\ODI\<LAN driver> (for example, 3C503)
```

Optionally, you can add the following lines if you want to have the Novell IPX protocol stack:

```
C:\ODI\IPXODI.COM  
C:\ODI\NETX
```

4. Create a NET.CFG file and place it in the C:\ODI directory. Refer to the sample file provided at the end of this appendix.

A NET.CFG file is required to run *both* NetWare and ChameleonNFS concurrently.

**Caution:** You need to know which NET.CFG file is used by the ODI environment (LSL). We suggest that you have only one NET.CFG file on your disk.

To ensure smooth establishment of an ODI environment, refer to your network adapter vendor's ODI documentation.

5. Reboot your system.
6. If you are using Novell Netware and loaded Novell ODI drivers, then log onto your server using the appropriate account and password, and verify to see if you are connected to the server by using the WHOAMI command.

## Running Custom Application

7. Start Windows.
8. Install ChameleonNFS (refer to your Quick Install Card).

You can tell when ODI-based ChameleonNFS is installed on your system by viewing the Setup menu on the Custom application. If the Hardware option is grayed out, then ODI is installed on your system and has been detected by the installation program.

When you are finished setting up Custom, select the Save option to save your configuration.

9. Custom automatically detects the location of the NET.CFG file and then modifies it.

If Custom cannot detect the NET.CFG file, a prompt appears asking you to enter its path. If you know the file's location, then enter its path and choose the OK button.

If you do not know the location of the NET.CFG file, then choose the Cancel button. Custom will automatically copy NetManage's sample NET.CFG file and place it at the root directory of drive C. Custom will also open a Window Help file that will give you information about NET.CFG.

**Note:** The NET.CFG file that NetManage supplies is only a *sample* file. Because user environments are configured differently, this sample file may or may not work with your configuration. There is an

ODI online help file available that contains detailed information about the sample NET.CFG file.  
You can choose to view this help file when prompted for it.

10. Reboot your system.

## Switching from NDIS to ODI

This section applies to users who are switching from an NDIS-based interface to an ODI-based interface. This section describes, step-by-step how to:

- upgrade your ChameleonNFS to version 4.6
- modify your configuration files in order to switch from an NDIS environment to an ODI environment

Follow these steps:

1. Start Windows in the Enhanced mode.
2. Install ChameleonNFS (upgrade installation) according to the instructions on your Quick Install Card.
3. After completing installation, reboot your system and do the following:
  - Start Windows and then choose the Network icon from the Control Panel icon.
  - Select NetManage ChameleonNFS (all versions) from the list of networks and choose the Remove... button. The NEWT icon will disappear from your screen.

4. From the CONFIG.SYS file, delete the lines:

```
DEVICE=C:\NETMANAG\PROTMAN.DOS /I:C:\NETMANAG
DEVICE=C:\NETMANAG\ELNKII.DOS      (or your card's driver)
DEVICE=C:\NETMANAG\NETMANAG.DOS
```

5. Delete this line from the AUTOEXEC.BAT file:

```
C:\NETMANAG\NETBIND
```

6. At the DOS prompt, create a directory named C:\ODI.
7. Copy the LSL.COM, Multiple Link Interface Driver (MLID) 3C503.COM for example), protocol stack files (IPXODI.COM), and Netware shell (NETX) to this directory.
8. Append the following lines to the end of the AUTOEXEC.BAT file:

```
C:\ODI\LSL.COM
C:\ODI\<LAN driver> (for example, 3C503.COM)
```

Optionally, you can add the following lines if you want to have the Novell IPX protocol stack:

```
C:\ODI\IPXODI.COM
C:\ODI\NETX
```

9. Create a NET.CFG file and place it in the C:\ODI directory. Refer to the sample file provided at the end of this chapter.

A NET.CFG file is required to run *both* NetWare and ChameleonNFS concurrently.

**Note:** Make sure you add C:\ODI to your path statement in your AUTOEXEC.BAT file.

To ensure smooth establishment of an ODI environment, refer to your network adapter vendor's ODI documentation.

10. Reboot your system.
11. Start Windows from the Program Manager, and select the Run entry from the File menu and enter the following:

```
\NETMANAG\CUSTOM.EXE -o
```
12. Custom automatically detects the location of the NET.CFG file and then modifies it.

If Custom cannot detect the NET.CFG file a prompt appears asking you to enter its path. If you know the file's location, then enter its path and choose the OK button.

If you do not know the location of the NET.CFG file, then choose the Cancel button. Custom will automatically copy NetManage's sample NET.CFG file and place it at the root directory of drive C.



**Note:** The NET.CFG file that NetManage supplies is only a *sample* file. Because user environments are configured differently, this sample file may or may not work with your configuration. There is an ODI online help file available that contains detailed information about the sample NET.CFG file. You can choose to view this help file when prompted for it.

13. Close Custom, reboot your system and do the following:

- Start Windows and choose the Network icon from the Control Panel icon.
- Choose the Add>> button. The Dialog box expands to include a list of available networks.
- Select NetManage ChameleonNFS (all versions) from the list and choose the Install... button. This network is added to the list of Installed Networks.
- Choose the OK button. The NEWT icon will appear on your screen.

## Switching from ODINSUP to ODI

This section applies to users who are switching from an ODINSUP-based interface to an ODI-based interface.

**Note:** There is no need to keep the ODINSUP configuration with native ODI support. You may want to eliminate some extra layers, thus releasing more memory and increasing reliability.

1. Start Windows in the enhanced mode.
2. Install ChameleonNFS (upgrade installation) according to the instructions on your Quick Install Card.
3. When the installation is complete, delete these lines from the AUTOEXEC.BAT file:

```
ODINSUP  
NETBIND
```

4. From the CONFIG.SYS file, delete the lines:

```
DEVICE=C:\NETMANAG\PROTMAN.DOS /I:C:\NETMANAG  
DEVICE=C:\NETMANAG\NETMANAG.DOS
```

5. From the Program Manager, select the Run option from the File menu and enter the following:

```
C:\NETMANAG\CUSTOM.EXE -o
```

6. Reboot your system.

```
# #####
# net.cfg
# NetManage, Inc.      10/23/95
#
# This net.cfg file is a working example and tested
# with NetManage TCPIP protocol stack (nmodi/newt) and # Novell IPX protocol stack
# (ipxodi/netx)simultaneously.
#
# In case a problem occurs:
#
# 1. Make sure there's no other net.cfg on your hard      # disk.
#
# 2. In order for nmodi.com to load/run you need at      # least the following entries in the
appropriate LINK      # DRIVER section of NET.CFG.
#   Ethernet Adapters:
#       frame ethernet_ii
#       Protocol IP      800      ETHERNET_II
#       Protocol ARP      806      ETHERNET_II
#       Protocol RARP      8035      ETHERNET_II
#   TokenRing Adapters:
#       Frame TOKEN-RING_SNAP
#       Protocol IP      800      TOKEN-RING_SNAP
#       Protocol ARP      806      TOKEN-RING_SNAP
#       Protocol RARP      8035      TOKEN-RING_SNAP
#
# Please note that all ethernet sections are similar      # as well as all token-ring sections.
#####
# 3COM EtherLink III Adapter
Link Driver 3C509
    Frame ETHERNET_802.3
    Frame ETHERNET_II
    Protocol IPX      0      ETHERNET_802.3
    Protocol IPX      8137      ETHERNET_II
    Protocol IP      800      ETHERNET_II
    Protocol ARP      806      ETHERNET_II
    Protocol RARP      8035      ETHERNET_II
# INTEL Ether Express Adapter
Link Driver EXP160DI
    Frame ETHERNET_802.3
    Frame ETHERNET_II
    Protocol IPX      0      ETHERNET_802.3
    Protocol IPX      8137      ETHERNET_II
    Protocol IP      800      ETHERNET_II
    Protocol ARP      806      ETHERNET_II
    Protocol RARP      8035      ETHERNET_II
#IBM Token Ring Adapter
Link Driver TOKEN
    Max Frame Size 1552
    Frame TOKEN-RING
    Frame TOKEN-RING_SNAP
    Protocol IPX      E0      TOKEN-RING
    Protocol IPX      8137      TOKEN-RING_SNAP
    Protocol IP      800      TOKEN-RING_SNAP
    Protocol ARP      806      TOKEN-RING_SNAP
    Protocol RARP      8035      TOKEN-RING_SNAP
#IBM LAN Support Program
Link Driver LANSUP
    Max Frame Size 1552
    Frame TOKEN-RING
```

```

Frame TOKEN-RING_SNAP
Protocol IPX      E0      TOKEN-RING
Protocol IPX      8137    TOKEN-RING_SNAP
Protocol IP       800     TOKEN-RING_SNAP
Protocol ARP      806     TOKEN-RING_SNAP
Protocol RARP     8035    TOKEN-RING_SNAP

#OLICOM Tken Ring Adapter
Link Driver OCTOK16
Max Frame Size 1552
Frame TOKEN-RING
Frame TOKEN-RING_SNAP
Protocol IPX      E0      TOKEN-RING
Protocol IPX      8137    TOKEN-RING_SNAP
Protocol IP       800     TOKEN-RING_SNAP
Protocol ARP      806     TOKEN-RING_SNAP
Protocol RARP     8035    TOKEN-RING_SNAP

#Madge Token Ring
Link Driver MADGEODI
Max Frame Size 1552
Frame TOKEN-RING
Frame TOKEN-RING_SNAP
Protocol IPX      E0      TOKEN-RING
Protocol IPX      8137    TOKEN-RING_SNAP
Protocol IP       800     TOKEN-RING_SNAP
Protocol ARP      806     TOKEN-RING_SNAP
Protocol RARP     8035    TOKEN-RING_SNAP

# eof # # # # # # # # # # # # # # # # # # # # # # # # # # # #
```

## Troubleshooting

- o Make sure your system is running the latest version LSL and MLID. Usually manufacturers of network interface cards have the latest version of available MLID.

Or, you can download them from Compuserve (GO NOVLIB), or from FTP.NOVELL.COM using FTP.

- o In case the NMODI driver fails to load, check the following:

- a) Make sure there is no other NET.CFG on your hard disk.
- b) In order for nmodi.com to load/run you need the following entries in the appropriate LINK DRIVER section of NET.CFG.

Ethernet Adapters:

```
frame ethernet_ii
Protocol IP      800    ETHERNET_II
Protocol ARP     806    ETHERNET_II
Protocol RARP    8035   ETHERNET_II
```

TokenRing Adapters:

```
Frame TOKEN-RING_SNAP
Protocol IP      800    TOKEN-RING_SNAP
Protocol ARP     806    TOKEN-RING_SNAP
Protocol RARP    8035   TOKEN-RING_SNAP
```

- o Please note that all Ethernet Link Driver sections are the same as well as all token-ring sections, independent of a particular MLID.

If your network media is Token-Ring and you have more than one ring, you may want to put the following line in your AUTOEXEC.BAT file before NMODI:

```
C:\ODI\ROUTE BOARD=2
```

- o The board parameter insures that ROUTE.COM is bound to logical board TOKEN-RING\_SNAP, which handles TCP/IP traffic. Refer to the Novell IPX documentation for more information about ROUTE.COM

## **Additional Notes**

This section describes some of the more advanced NET.CFG parameters and syntax of those parameters.

The NET.CFG file is a configuration file that contains section headings and options that deviate from the established defaults of the ODIF1ProjectGlossary48 software. If you changed the default hardware settings on the LAN adapter or if you are using multiple protocols, you need a NET.CFG file. You may also need to see the documentation specific to your protocol for additional NET.CFG information. Use any DOS text editor to create the file. Specify only options that will change from the defaults.

See also

[NET.CFG File Syntax and Conventions](#)

[Link Support - NET.CFG Section](#)

[IPX Protocol IDs](#)

## NET.CFG File Syntax and Conventions

Main section headings must be left-justified and are not case sensitive. The heading must precede the options you want to include in that section. Options are not case sensitive and \*must\* be preceded by a tab or hard spaces. Precede comments with a semicolon (;). End each line with a hard return. Write all numbers in decimal notation except where noted otherwise. The following are common main section headings in the NET.CFG file.

### Link Driver - NET.CFG keyword

There are several NET.CFG file options available to LAN drivers. See the documentation supplied with your network adapter for the specific settings that can be used.

To use the "Link Driver drivename" heading, replace "drivename" with the name of the driver you are using; the name is typically the LAN driver's filename. Replace drivename with one of the following driver names:

- 3C501 (for 3COM EtherLink)
- 3C503 (for 3Com EtherLink Series II)
- 3C505 (for 3COM EtherLink Plus)
- 3C523 (for 3Com EtherLink/MC)
- 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)
- 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 drivename, place 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 drivename" heading for each network board.

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

number        A decimal number

hex            A hexadecimal number

The following chart lists the options defined by the DOS ODI software. Other options may be available for the LAN driver or protocol stack you are using. Refer to their documentation for more information. Main section headings are in white print and flush with the left margin. Options are listed under each heading and indented.

## Link Support - NET.CFG Section

`BUFFERS communication_number [size]`

This option configures the number and size of receive buffers that will be maintained by the LSLF1ProjectGlossary44. 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. The total buffer space must fit into approximately 59 KB (number times size).

Default: 1130

`MEMPOOL number[k]`

Some protocols use this option to configure the size of the memory pool buffers that the LSL will maintain. The k notation means multiply by 1024. Refer to the protocol documentation for settings.

`PROTOCOL - NET.CFG keyword`

The `PROTOCOL` keyword can be used in the `NET.CFG` file to define the protocol ID's and frame formats that each protocol type uses.



## **IPX Protocol IDs**

The IPX protocol IDs and associations are shown below:

Ethernet Protocol ID's:

PROTOCOL IPX	0	ETHERNET_802.3
--------------	---	----------------

PROTOCOL IPX	8137	ETHERNET_II
--------------	------	-------------

## ODI Terminology

Token Ring Protocol ID's:

PROTOCOL IPX        E0 TOKEN-RING

By putting one of these entries in the network adapter LINK DRIVER section of NET.CFG, you specifically tell IPX what frame format to use and what the protocol ID is.

### TCPIP Protocol IDs

Ethernet TCPIP Protocol ID's

PROTOCOL IP        800        ETHERNET\_II

PROTOCOL ARP       806        ETHERNET\_II

PROTOCOL RARP      8035    ETHERNET\_II

### Token Ring TCPIP Protocol ID's

PROTOCOL IP        800        TOKEN-RING\_SNAP

PROTOCOL ARP       806        TOKEN-RING\_SNAP

PROTOCOL RARP      8035    TOKEN-RING\_SNAP

By putting these entries in the network adapter LINK DRIVER section of NET.CFG, you specifically tell all the TCPIP protocols what frame format to use and what the protocol ID's should be.

### NMODI.COM - NetManage ODI Driver

The NMODI driver is the interface that NEWT (the IP protocol stack) uses to talk to the ODI drivers. It must be loaded AFTER the adapter MLID driver. If it is loaded before the MLID adapter it will not know what kind of network there is, what frame format (board number) to use etc. This driver must also be loaded before Windows is loaded. The best place to load it is right after all the other ODI drivers are done loading. A working configuration with the drivers loaded from AUTOEXEC would look like this:

```
@ECHO OFF
C:
CD\NETWARE (or whatever directory your ODI drivers are in)
LSL
NE2000
IPXODI
NETX
NMODI
```

Note that NMODI can be loaded at any point after the NE2000 driver (or what ever adapter driver you are using) but must be loaded before Windows loads. For NMODI to load properly, the correct FRAME FORMAT statements must be in the NET.CFG file. Normally, CUSTOM modifies the NET.CFG to include the correct statements.

### Frame Format - Advanced Help

FRAME FORMATS are defined in the NET.CFG file. The TCPIP protocol \*MUST\* use a frame format of ETHERNET\_II on ethernet networks and TOKEN-RING\_SNAP on token ring networks. By default, IPX on ethernet uses a proprietary frame format called Ethernet\_802.3 (Novell Ethernet) and on token ring IPX by default uses the TOKEN-RING frame format. TCPIP WILL NOT work over these frame formats at all.

The Chameleon installation program (CUSTOM) will search the path to find the NET.CFG file and if it doesn't find it, it will prompt you for the actual location of NET.CFG or where you would want to put a new

one. If one exists, CUSTOM processes NET.CFG and looks for the "Link Driver" section. After it finds that, it looks to see if the correct frame format statements are in place and if not, it adds them.

In the following example NET.CFG file, the ethernet and token ring adapter entries are show as they were in NET.CFG \*before\* CUSTOM makes any modifications.

```
#-----  
Link Driver NE2000  
    INT 5  
    PORT 300  
Link Driver 3c509  
    INT 5  
Link Driver TOKEN
```

```
#-----
```

This is what this part of the NET.CFG file looks like \*after\* CUSTOM has modified the file:

```
#-----
```

```
Link Driver NE2000  
    INT 5  
    PORT 300  
    Frame Ethernet_802.3  
    Frame Ethernet_II  
Link Driver 3c509  
    Frame Ethernet_802.3  
    Frame Ethernet_II  
Link Driver TOKEN  
    Frame TOKEN-RING  
    Frame TOKEN-RING_SNAP
```

```
#-----
```

As you can see, the required TCPIP frame format entries were automatically added (ethernet\_II or token-ring\_snap) by CUSTOM. When the network adapter driver loads (NE2000.COM, 3C509.COM or TOKEN.COM), it looks at NET.CFG to see what frame formats to use. In the above ethernet example, both ethernet\_802.3 and ethernet\_ii are seen by the driver and it then loads support for both of them. The adapter driver also treats each frame format as a completely independent and separate "virtual network adapter" and protocols such as IPX and TCPIP can then use the correct frame format or "virtual network adapter".

In the ethernet world, the following frame formats are supported by "most" ODI network adapter drivers (note: some adapter drivers do not support all frame formats. Contact your adapter vendor for an updated driver if necessary):

Frame format	Typical protocols supported
--------------	-----------------------------

=====

ethernet_802.3	IPX (often the default frame format for IPX)
ethernet_802.2	IPX, DLC (mainframe communications)
ethernet_ii	IPX, TCPIP, AppleTalk Phase I
ethernet_snap	IPX, AppleTalk Phase II

In the token ring world, the following frame formats are supported by "most" ODI network adapter drivers (note: some adapter drivers do not support all frame formats. Contact your adapter vendor for an updated driver if necessary):

Frame format	Typical protocols supported
--------------	-----------------------------

=====

token-ring	IPX (the default frame format for IPX)
token-ring_snap	IPX, TCPIP, AppleTalk Phase II

Each protocol can often use one or more frame formats that will carry the packet across the network. The Novell IPX protocol can use *\*ANY\** of the above mentioned frame formats on both ethernet and token ring. The trick here is to make sure that IPX on the file server is setup in *\*exactly\** the same way as IPX on the workstation. If the file server has IPX riding on ETHERNET\_SNAP and the workstation has IPX riding on ETHERNET\_802.3, the two will never connect and the workstation will get the error "File Server not found" when NETX.EXE is loaded. Exactly the same holds true for token ring.

The actual order of these frame format lines in NET.CFG can be extremely important. As it happens, when IPXODI is loaded (the ODI version of the IPX protocol stack), it will try and "bind" or use the first frame format that is present under the "Link Driver xxx" statement. For example, in the following example section, there is no frame format statement:

```
#-----
Link Driver NE2000
#-----
```

When IPXODI is loaded, IPX uses the ETHERNET\_802.3 frame format by default (refer to the default rules explained above). After CUSTOM is done with NET.CFG, it looks like this:

```
#-----
Link Driver NE2000
    Frame Ethernet_802.3
    Frame Etherent_II
#-----
```

In this case, when IPXODI is loaded, IPX uses the ETHERNET\_802.3 frame format again but it uses that frame format because it was specified in the NET.CFG file. If this section were to have the frame format statements reversed, it would look like this:

```
#-----
Link Driver NE2000
    Frame Etherent_II
    Frame Ethernet_802.3
#-----
```

In this case, when IPXODI is loaded, IPX would use the ETHERNET\_II frame format because it is the \*first\* frame format to be specified in the NET.CFG configuration file. If the fileserver didn't support IPX on ETHERNET\_II, the workstation would never connect to the server.

To solve this problem, the NET.CFG keyword "PROTOCOL" is used to tell the protocol which frame format to use. It is shown below for both ethernet and token ring:

#-----

Link Driver 3c509

Frame Ethernet\_802.3

Frame Ethernet\_II

Protocol IPX E0 Ethernet\_802.3

Link Driver TOKEN

Frame TOKEN-RING

Frame TOKEN-RING\_SNAP

Protocol IPX 0 TOKEN-RING

#-----

When this keyword is used, there can never be any confusion about which frame format IPX will use.

### **Troubleshooting ODI Driver Installation**

Troubleshooting ODI problems would start with identifying the drivers installed on the PC, verifying that the drivers load ok and then checking the Netmanage Drivers and protocol stack.

### **Basic Troubleshooting**

1) The first step in basic troubleshooting ODI problems would be to verify that you have the proper drivers installed on your PC. The following drivers should be installed on your system and appear in your AUTOEXEC.BAT file:

LSL.COM

MLID Driver

IPXODI (optional)

NETX (optional)

2) The next step is to watch your PC as it boots up and verify that these drivers load ok. If you see errors indicating that the drivers aren't found, you may want to check the location of the files and make sure that AUTOEXEC.BAT references the proper location of them.

3) If the drivers load ok, verify that the NMODI.COM driver loads before Windows starts and after the ODI drivers have loaded. This file is \*required\* to be loaded so that the NetManage protocol drivers can be used.

4) Quick Check: A quick check for seeing if all the ODI drivers are loaded correctly would be as follows:

- Start the Chameleon PING application
- Double click on the NEWT icon at the bottom of the screen and look to see if the "PHYSICAL ADDRESS" field is filled in with a 12 digit address.

This is the physical MAC address of your network adapter and can only be seen if all of the ODI drivers

load properly. If this entry is blank, there is very likely a problem with how the ODI drivers loaded or how NMODI.COM was loaded. See the Tips and Tricks section on detecting ODI drivers before installing Chameleon.

## Advanced Troubleshooting

This section defines some advanced troubleshooting techniques.

If you get the message ~A File Server could not be found,~ check to see if:

- The network board is seated properly.
- The cable is connected properly. (Run COMCHECK to verify the connections.)
- All cabling is terminated properly.
- The workstation node address conflicts with any other node address.

If you want information about the drivers that are loaded, type a question mark after the driver name.

For example

```
NE2000 ? <Enter>
```

or

```
LSL.COM ? <Enter>
```

or

```
IPXODI ? <Enter>
```

Check to see if the correct Frame Format parameters are in the NET.CFG file for your network adapter in the LINK DRIVER section. Normally, IPX on ethernet networks uses the ETHERNET\_802.3 frame format unless otherwise specified. On token ring networks, by default IPX uses the TOKEN-RING frame format.

## Tips and Tricks for Troubleshooting

### Detecting ODI Drivers During Installation

You can tell when Custom has detected an ODI driver in the system by looking at the "SETUP/HARDWARE" option in Custom. This option should be greyed out and not selectable. If the HARDWARE option \*is not\* greyed out, then check to see if the LSL and adapter MLID driver are loaded from DOS.

### Location of ODI Drivers and NET.CFG

The actual location of NET.CFG can be important in getting all of the drivers to work. To be safe and to avoid problems, it is a good idea to place ALL the ODI related driver files in their own directory (eg C:\NETWARE or C:\ODI). When they get loaded, the batch file that loads them should change to that directory BEFORE loading the drivers. When LSL.COM loads, suffice to say that it can play funny games if NET.CFG is not in the same directory that it gets started from. The safe way to load the ODI drivers in AUTOEXEC.BAT would look like this (assume all ODI drivers and NET.CFG are in

```
C:\NETWARE) :
```

```
CD \ODI
```

```
LSL
```

```
NE2000 (or TOKEN or whatever MLID driver you have)
```

```
IPXODI
```

```
NETX
```

One mistake that can lead to problems would be to load these files without changing to the \ODI directory. Eg:

```
CD\  
C:\ODI\LSL  
C:\ODI\NE2000  
C:\ODI\IPXODI  
C:\ODI\NETX
```

This can lead to problems if the NET.CFG file is in the root directory or in some other location.

### **AUTOEXEC.BAT**

This file is read by your PC when it boots up. Commands that are placed in this file are automatically executed.

### **COMCHECK**

This utility comes with the Novell installation disks and is used to test communications on the network adapter.

### **DMA**

Short for Direct Memory Address setting. Check your adapter documentation to see what options you can use.

### **ETHERNET\_802.3**

Novell's Proprietary frame format on ethernet. This is usually the default frame format used by IPX.

### **ETHERNET\_II**

This is an alternative frame format that can be used for IPX. It is also REQUIRED to be used for TCPIP.

### **FORMATs**

#### **Frame Format**

Is used to specify the frame type that a given protocol uses. See the NET.CFG section for a full description.

#### **I/O port**

Input/Output Port Setting. Check your adapter's documentation to see what the options for this setting are.

#### **INT**

Most network adapters use an interrupt. This setting must not conflict with other adapters that are installed in the computer. Good choices usually include INT 5 or INT 2.

#### **Interrupt**

Most network adapters use an interrupt. This setting must not conflict with other adapters that are installed in the computer. Good choices usually include INT 5 or INT 2.

#### **IPX**

Novell's proprietary protocol used in Novell workstation/server communications.

#### **IPXODI**

This is the ODI IPX protocol stack driver used to access Novell servers and peripherals.

#### **LINK DRIVER**

The section of NET.CFG that defines specific parameters for the network adapter. See the NET.CFG section for more information

**LSL**

Link Support Layer Driver (LSL.COM). This is the first file that gets loaded to support the ODI drivers. It is the glue that binds all the network drivers together and enables the workstation to communicate over several protocols.

**MLID**

Multiple Link Interface Driver - This is a hardware specific network adapter driver such as NE2000.COM, 3C509.COM, TOKEN.COM, etc.

**NETX**

This is the Network Shell. It provides all of the Novell Network functionality such as mapping drives and capturing printers.

**Node Address**

Every computer on a local area network has a unique physical address that is associated with the network adapter installed in that system.

**ODI**

Open Datalink Interface Specification

**PORT**

Input/Output Port Setting. Check your adapter's documentation to see what the options for this setting are.

**SLOT**

Slot refers to the place where network adapter boards go inside of the computer.



## Contents

### Using Help

#### To choose a Help topic:

- Mouse      Point to the underlined topic you want to view and click the mouse button. When the pointer is over an item you can choose, the pointer changes to a hand icon.
- Keyboard    Press Tab to move the highlight to the underlined topic you want to view, and then press Enter.

#### To exit Help:

- Mouse      1: Point to the File menu and click the mouse button.  
              2: Click the Exit command.
- Keyboard    1: Press Alt  
              2: Type the letter F  
              3: Type the letter X

## About ODI

Choose the About ODI... command from the Help menu for information about this program.

Choose the Copy button to copy the application version information into the Windows Clipboard. You can then paste this information into any application.

Choose the OK button to continue.

