Brouter Support

Brouter Support
Version 1.0

Novell Documentation Part Number:
107-000049-001

9 June 1993

Disclaimer

Novell®, Inc. makes no representations or warranties with respect to the contents or use of this manual, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any NetWare software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of NetWare software, at any time, without any obligation to notify any person or entity of such changes.

Trademarks

Novell, Inc. has made every effort to supply trademark information about company names, products, and services mentioned in this book. Trademarks indicated below were derived from various sources.

Novell Trademarks

NetWare and Novell are registered trademarks of Novell, Inc.
Internetwork Packet Exchange (IPX), ODI, Open Data-Link Interface, LSL, Link Support Layer, MLID, Multiple Link-Interface Driver, MLI, Multiple Link Interface, MPI, Multiple Protocol Interface, MSM, Media Support Module, TSM, Topology Support Module, HSM, Hardware Support Module, RX-Net, NE1000, NE2000, NE/2, NE2-32, and NTR2000 are trademarks of Novell, Inc.

Other Trademarks

ARCnet is a registered trademark of Datapoint Corporation.
Apple, AppleTalk and LocalTalk are registered trademarks of Apple Computer, Inc.
COMPAQ is a registered trademark of Compaq Computer Corporation.
DEC and LAT are trademarks of Digital Equipment Corporation.
Gateway and G/Net are registered trademarks of Gateway Communications, Inc.
IBM, OS/2, and PS/2 are registered trademarks of International Business Machines Corporation.
Intel is a registered trademark of Intel Corporation.
MS-DOS is a registered trademark of Microsoft Corporation.
Proteon and ProNET are registered trademarks of Proteon, Inc.
Xerox is a registered trademark of Xerox Corporation.
XNS is a trademark of Xerox Corporation.

Table of Contents

Brouter Support Specifications

Brouter Test Procedures

Index

Chapter 1

Brouter Support

Introduction:

Source Route Bridging is a mechanism that allows the source of traffic to dynamically discover routes and determine which one to use when sending data to a particular destination. The server can function as a server, router, or bridge. It requires the routing information field in source routed Token-Ring frames to be inspected for a match between the Token-Ring and bridge numbers in that field and those assigned to the bridge and its attached rings.

Depending on your network configuration and the route selection algorithm used, several routes can be discovered for each destination. The source examines the available routes and then determines the best route to use when sending data.

The Source Route Bridge NLM (BRIDGE.NLM) allows Token-Ring networks to be linked together. This NLM allows segmentation of network traffic to reduce the load on any one segment and also provides parallel bridging, which is a fault-tolerant technique of determining alternate routes for data when a bridge fails.

NetWare's Source Routing can be used for any protocol that supports Source Route Bridging (on Token-Ring networks), including IPX, TCP/IP, AppleTalk, NetBIOS, and OSI. When combined with WAN connectivity support (as in MultiProtocol Router Plus), it also enables the bridging of geographically distant LAN's across X.25 networks of leased-line synchronous links over PPP.

The Source Route Bridge NLM serves a variety of purposes:

- It connects separate Token-Ring networks and allows you to surpass the single-network limitation of 72, 92, or 260 stations (depending on the cable type).

- It allows you to add multiple source-routed interfaces to an existing NetWare v3.1x file server in a source route bridged Token-Ring environment; thus reducing the load on existing bridges by creating a parallel bridge, improving the performance of NetWare servers by attaching them directly to multiple rings, and reducing bridge hop counts for users.
- It allows protocols that cannot be routed (such as NetBIOS and SNA) to be forwarded.
- Optimizes internet capacity by localizing traffic on LAN segments.
- MultiProtocol Router Plus (includes WAN connectivity), connects geographically distant Token-Ring networks.
- Extends the physical reach of networks beyond the limits of each LAN segment (packets can be sent through a maximum of thirteen bridges).
- Increases network security and available bandwidth by filtering with source ring number, hop count, and protocol ID.

The typical NetWare v3.11 file server in a bridged Token-Ring environment (without the Source Route Bridge NLM), can only be attached to one ring.

Although NetWare v3.11 can provide source routing, it supports only one interface. This limitation causes users on the outer rings to go through several bridge hops before they can access the server.

The Source Route Bridge NLM however, supports two or more Token-Ring interfaces by,

- Allowing a parallel bridge to be created and adding redundancy to the network.
- Increasing the server performance by distributing traffic across multiple network interfaces and reducing hop counts between users and the server.

Bridge Routing and the HSM

In order for BRIDGE.NLM to utilize current and future Hardware Support Module (HSM) and board hardware enhancements, the following interface specification between the BRIDGE.NLM and the HSM is to be used.

The BRIDGE.NLM uses the driver management IOCTL (I/O Control) to pass Brouter Requests to the HSM. This is the same IOCTL used for HUB Management. See the Control Procedures section of HSM Procedures (Chapter 5) of the Developer's Guide for more information.

ODI Server Driver Table 1 describes the general format of the management ECB used for Brouter requests. When the HSM's driver management IOCTL is called, the ESI register points to this ECB.

Table 1
Brouter Request ECB (General Format)

Field Name	Data Size	Comments
Link	4 bytes	unused by HSM
BLink	4 bytes	unused by HSM
Status	2 bytes	unused by HSM
ESRAddress	4 bytes	unused by HSM
LogicalID	2 bytes	unused by HSM
ProtocolID = BROUTE	6 bytes	ProtocolID is BROUTE for all Brouter Requests.
BrouterRequest	2 bytes	Use one of the values from Table 2.
Additional Fields	Size depends on the request being made.	These fields are dependant on the value of the BrouterRequest.

The Protocol ID is BROUTE for the Brouter Support Request. The Brouter Request follows immediately after the Protocol ID field in the ECB. The ECB fields prior to the Protocol ID field (16 bytes total) are not used and do not need to be initialized.

The BrouterRequest code determines the action to be taken by the HSM. The Brouter support request types are BrouterSupportRequest, SelectTBridging, SelectSRBridging, SelectPromiscuousMode and UpdateFilterAddressTable. The Additional Fields are explained later in this chapter.

The BRIDGE.NLM first sends the Brouter Support Request to the HSM. If the HSM returns a BrouterStatus (explained in tables 3 and 4) of zero or if an error occurs, Brouter Enhancements are not supported and no additional requests will be made. The HSM will return its status in the EAX register.

If Brouter enhancements are supported, a successfully executed request returns a zero. A non-zero reply means that the request failed or that the board failed to select the mode type requested.

A table of the Brouter Requests is shown below:

Table 2
Brouter Request Options

Value	Description
0 - BrouterSupportRequest	See BrouterSupportRequest ECB table (Table 3) below for more information.
1 - SelectTBridging	See Table 5 below for more information.
2 - SelectSRBridging	See Table 6 below for more information.
3 - SelectPromiscuousMode	See Table 7 below for more information.
4 - UpdateAddressFilterTable	See Table 8 below for more information.

Brouter Request Options 1 through 4 have bit-wise defined values. See tables 5 through 8 for bit value descriptions.

Append Table 3 in place of the Additional Fields in Table 1 when the BrouterRequest field value is equal to zero.

Table 3
Brouter Support Request ECB

BrouterRequest = 0	Data Size	Comments
Filter table size	4 bytes	This field only applies if Transparent Bridging is supported

The BrouterSupportRequest is used by BRIDGE.NLM to find out what Brouter enhancements, if any, are supported by the HSM. This will always be the first request that BRIDGE.NLM issues.

The HSM returns the supported Brouter options in the Brouter-Status bits. The BrouterStatus bits are defined in Table 4.

Note

Bits 31-28 are used for Transparent Bridging, while bits 23-15 are used for Source Route (SR) bridging. Bit 0 is the low order bit.

Table 4
Brouter Status Bits

Bit Number(s)	Value	Parameter Description
31	0	Transparent Promiscuous mode not supported
	1	Transparent Promiscuous mode supported
30	0	HSM cannot handle Source Routing data the same as transparent data.
	1	HSM can forward and filter Source Routing data just like transparent data (requires Bit 31 to be set).
29-28	00	Transparent filter/ forward capability not supported
	01	Transparent filtering supported (filter when both the Destination Address and Source Address are in the filtering table).
	10	Transparent filtering supported (filter when just the Destination Address is in the filtering table).

Table 4
Brouter Status Bits

Bit Numbers (continued)	Value	Parameter Description
23	0	Source Route bridging not supported (bits 15-22 are set to 0)
	1	Source Route bridging support
22	0	HSM supports fixed bridge number size (4 bits).
	1	HSM supports variable Source Route bridge number partition size (from 1-14). Makes sense only when Bits 19-16 are defined.
21	0	HSM cannot filter STE frames
	1	HSM can (optionally) filter STE frames.
20	0	HSM does not support multiple BridgeNumbers.
	1	HSM does support multiple Bridge Numbers (only makes sense with > 2-Way bridging, where each RingOut can have a different BridgeNumber).

Table 4
Brouter Status Bits

Bit Numbers (continued)	Value	Parameter Description
19-16	0000	Source Routing filtering not supported.
	0001	2-Way Source Routing supported (single RingOut).
	0010	3-Way Source Routing support (2 RingOut's).
	xxxx	N-Way Source Routing support. A value of 15 (xxxx = 1111) means that any number of RingOut's up to a maximum of 120 is supported.
15	0	HSM supports maximum Route Descriptor limit of 8 (IBM).
	1	HSM supports maximum Route Descriptor limit of 14 (IEEE).
14-0		These bits are reserved by Novell and should not be modified by the HSM.

Note The HSM should support transparent filtering when both the Destination Address and Source Address are in the filtering table.

Note If the HSM supports transparent filtering, the next 4 bytes following the BrouterStatus must contain the maximum number of addresses contained in the filtering table. Preferably, the filtering table will be larger than 2000 entries.

Note The BRIDGE.NLM always repeats Source Route filtering, thus an HSM that filters only on RingIn and BridgeNumber, may claim to support N-Way Source Routing (bits 19 through 16 = 1111), with a greater impact on performance.

SelectTBridging.

When the Transparent Bridge Indication field in a BrouterRequest is non-zero, the HSM must start Transparent Bridging. When the Indication field is zero, Transparent Bridging must end.

Transparent mode on the board reduces the intervention of the host processor by determining if frames should be passed on or filtered out. This is done by comparing the destination address of each frame with learned source addresses.

This mode is selected only when the HSM reports that it can support Transparent Promiscuous mode. The HSM needs to go into Transparent promiscuous mode. When applicable, it needs to initialize its Transparent filter table, and be ready to receive UpdateAddressFilterTable requests, as well as Select Promiscuous mode requests.

If the driver cannot support Transparent promiscuous mode, the Bridge will try to use the default receive monitor (RMON) Promiscuous mode.

Append Table 5 in place of the Additional Fields in Table 1 when the BrouterRequest field value equals one.

Table 5
SelectTBridging Request ECB

BrouterRequest = 1	Data Size	Comments
Transparent Bridging Indication	4 bytes	0 - Transparent Bridging must end. Non Zero - HSM must start Transparent Bridging.

SelectSRBridging

This mode will only be selected when the HSM reports that it can, at a minimum, support 2-way SR bridging. It will not present the HSM with any requests that it (the HSM) cannot support (e.g. the HSM will not be given more than a single RingOut/BridgeNumber unless it can perform more than 2-way bridging.

Append Table 6 in place of the Additional Fields in Table 1 when the BrouterRequest field value equals two.

Table 6
Non-Zero Source Routing Indication ECB

BrouterRequest = 2	Data Size	Comments
Source Route Indication	4 bytes	When the Source Route Indication is zero, the HSM must end the Source Route bridging. When it is non-zero the HSM must start Source Route bridging.
Partition Size	1 byte	The Partition Size is the size of the bridge number field in the Route Designator. Allowed values are from 1 to 14 bits. If the HSM does not support variable partition sizes, the field value is 4.
FilterSTE	1 byte	FilterSTE indicates that the HSM should filter out Spanning Tree Explorer frames. The HSM is not required to filter STE frames. The bridge routing software (BRIDGE.NLM) can perform this function.

Table 6
Non-Zero Source Routing Indication ECB

BrouterRequest = 2 (continued)	Data Size	Comments
RoutingInfoSize	1 byte	The RoutingInfoSize can be set to a value between 2 (minimum) and 30 (maximum) which represents the length in Bytes in the Routing Information Field. Since the BRIDGE.NLM software also verifies this the HSM may elect to ignore this field.
Reserved	1 byte	
RingIn Number	2 bytes	Sets the RingIn Number that the HSM uses when handling Source Routing packets.
RingOut Number	2 bytes	Sets the RingOut Numbers that the HSM uses when handling Source Routing packets.
Bridge Number	2 bytes	Sets the Bridge Numbers that the HSM uses when handling Source Routing packets. The last Bridge Number must be terminated by 0xffff.
Ring-Out/Bridge Number Pair	2 bytes	When N-Way (more than 2-way) bridging is supported, the RingOut/Bridge Number pair fields are repeated N times in the Brouter request. The value for N when using N-way bridging is determined by the Brouter Status bits 16 through 19 in Table 4. After the last Ring-Out/Bridge Number pair the sequence is teminated by hex FFFF in the Ring-Out number position.

Note If the HSM supports multiple RingOut connections but cannot handle multiple BridgeNumbers, set all inserted Bridge Number fields to the same value.

If the HSM supports more RingOut/Bridge Number pairs than the BRIDGE.NLM, the BRIDGE.NLM may pass fewer pairs than the number the HSM can support.

The HSM must also be able to handle multiple SelectSRBridging requests, because of Spanning Tree changes or network reconfiguration.

Select Promiscuous Mode.

This request will only be given to the HSM after a SelectTBridging request. The actions expected from the HSM based on the 4 types are (note that none of the mode selections affect the filter table):

0) No promiscuous mode. Turn off promiscuous mode forwarding.

1) Start Transparent promiscuous mode. Start or restart promiscuous mode for non Source Routed frames.

2) Source Route promiscuous mode indicates that the HSM needs to treat source routed frames just like transparent frames, and subject the destination and source addresses to the filter table, just like it would with transparent data, with the exception that the routing information indication bit in the source address is assumed to be zero (for table lookup purposes).

Non source routed frames need to be filtered, with the exception of locally destined frames.

3) Source Route/Transparent promiscuous mode is a combination of modes 1 and 2.

Append Table 7 in place of the Additional Fields in Table 1 when the BrouterRequest field value equals three.

Table 7
Select Promiscuous Mode ECB

BrouterRequest = 3	Data Size	Comments
Type	1 byte	The type field indicates the following: 0 - No promiscuous mode requested 1 - Transparent promiscuous mode 2 - SR (Source Route) promiscuous mode 3 - SRT promiscuous mode (SourceRouteTransparent)

Note

Promiscuous Mode types 2 and 3 will not be requested when the SelectSRBridging is activated. This means that the HSM has reported that it can do at least 2-Way Source Route bridging.

UpdateAddressFilterTable

Append Table 8 in place of the Additional Fields in Table 1 when the BrouterRequest field value equals four.

Table 8
Update Address Filter Table ECB

BrouterRequest = 4	Data Size	Comments
Action	1 byte	The Action field indicates the following: 0 - Clear filter table (delete all addresses) 1 - Add given addresses to the filter table 2 - Delete given addresses from the filter table
Number of Addresses	1 byte	This byte contains the number of addresses that follow (used for Add and Delete only). The action done on the 'Number of Addresses' field depends on the value selected in the Action field.
Addresses (in Canonical form)	6-bytes	Each address is 6 bytes long. There can be from 1 to 255 of these addresses.

All addresses will be given in Canonical form (bytes sent in L - H bit order on the wire). See Appendix G in the ODI Server Document for more information.

The pseudocode for DriverManagement is on the next page.

Driver Management Pseudocode for Brouter Support

On Entry	
ESI	Pointer to the management ECB containing the Brouter request
EBP	Pointer to adapter data space
EBX	Pointer to frame data space

On Return	
EAX	00000000h = Success; ECB relinquished. FFFFFF88 = No such handle; Protocol ID not supported.

Description

The MSM checks the Protocol ID field in the ECB. If the first byte is an ASCII letter greater than 40h, it is a valid management Protocol ID. The MSM will then pass the ECB to the HSM's Driver Management routine if one is available (indicated by the Drive Management pointer field in the DriverParameter block.

Driver Management Pseudocode

```
IF ProtocolID is not valid
        RETURN  FFFFFF88h

IF ProtocolID = BROUTE
        IF Brouter Request = 0 (Brouter Support Request)
                Fill in BrouterStatus field
                Return 0
        ELSE IF Brouter Request = 1 (SelectTBridging)
                IF Transparent Bridging Indication = Non-Zero
                        Begin Transparent Bridging
                        Return 0
                ELSE
                        End Transparent Bridging
                        Return 0
        ELSE IF Brouter Request = 2 (SelectSRBridging)
                IF SourceRouteIndication = Non-Zero
                        Begin SourceRouteBridging with indicated
                          parameters (PartitionSize, FilterSTE,
```

```
                                    RoutingInfoSize, RingInNumber,
                                    RingOutNumbers, BridgeNumbers)
                                Return 0
        ELSE
                End SourceRouteBridging
                Return 0

        ELSE IF BrouterRequest = 3 (Select Promiscous Mode)
                IF Type = 0
                        End Promiscous Mode
                        Return 0
                ELSE IF Type = 1
                        Begin Transparent Promiscuous mode
                        Return 0
                ELSE IF Type = 2
                        Start Source Routing Promiscuous mode
                        Return 0
                ELSE IF Type = 3
                        Begin Source Route Transparent Promiscuous
                        mode
                        Return 0
                ELSE
                        Return ERROR

        ELSE IF Brouter Request = 4 (UpdateAddressFilterTable)
                IF Action = 0
                        Clear filter table
                        Return 0
                ELSE IF Action = 1
                        Add given addresses to table
                        Return 0
                ELSE IF Action = 2
                        Delete given addresses from the table
                        Return 0
                ELSE
                        Return ERROR
        ELSE
                Return ERROR

ENDIF (ProtocolID = BROUTE)
```

Chapter 2

Brouter Test Procedures

Introduction:

Requirements for this test are BRTEST.NLM, two Token-Ring LAN boards, one cable for each LAN board and two MSAUs (Multi-Station Access Units). The setup for Configuration 1 is shown in Figure 1 and setup for Configuration 2 is shown in Figure 2.

The main purpose of the Brouter test (BRTEST.NLM) is to verify correct operation of HSM's that support Source Route Bridging enhancements. BRTEST.NLM tests the LAN driver's ability to process the following parameters in the bridge routing software (BRIDGE.NLM):

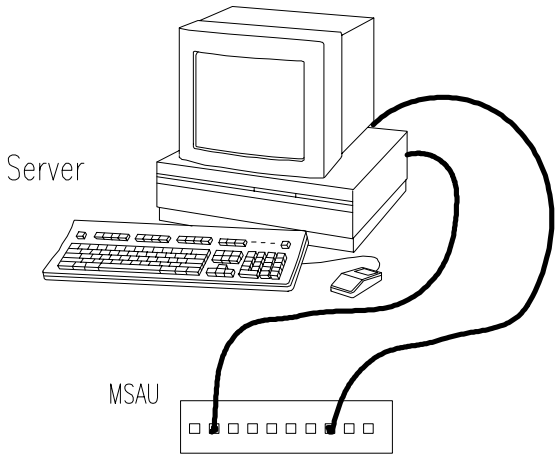
- ARE (All Route Explorer frames)
- STE (Spanning Tree Explorer)
- SRF (Specifically Routed Frames)
- Multiple Bridge
- Multiple Ring-Outs
- Variable Partition Size
- Unicast, Multicast and Broadcast frames
- Promiscuous Mode tests

Configuration 1

Specification tests cover Brouter Request handling and Source Route bridging functionality performed by using a single NetWare v3.1x server.

The test program (BRTEST.NLM) acts as a protocol stack, which is loaded and bound to the driver for each logical board. The test program sends a variety of raw-send test packets out one board and monitors the response of the board configured for Source Route bridging to verify correct driver operation. (See Figure 1).

Figure 1
Brouter Request and Source Route Bridging Test Configuration.



Setup

The server is configured with two Token-Ring interface boards. Both boards are connected to the same ring (MSAU). One board transmits 'raw' data, while the other board acts as a Source Routing Bridge that receives the transmitted raw data.

A summary of test software needed and test duration is shown below.

NetWork Configuration	**(See Figure 1)**
Test Software:	LSLENH.NLM, PATCHMAN.NLM, TOKENTSM.NLM, MSM31X.NLM, CLIB.NLM, STREAMS.NLM, BRTEST.NLM.
Test Objective:	To verify the correct operation of HSM's that support Source Route Bridging enhancements.
Pass/Fail Criteria:	Must pass BRTEST.NLM test without failures.
Test Duration:	15 minutes or less depending on supported parameters.

Running the Test

To start the test

1. Set up the hardware as shown in Figure 1.
2. Make sure the NetWare v3.1x server is functioning properly.
3. Load LSLENH
4. Load the LAN driver and configure both boards for the TOKEN and TOKEN-RING_SNAP frame types.
5. Load BRTEST.NLM.
6. Insert a blank formatted floppy disk in Drive A: when prompted.

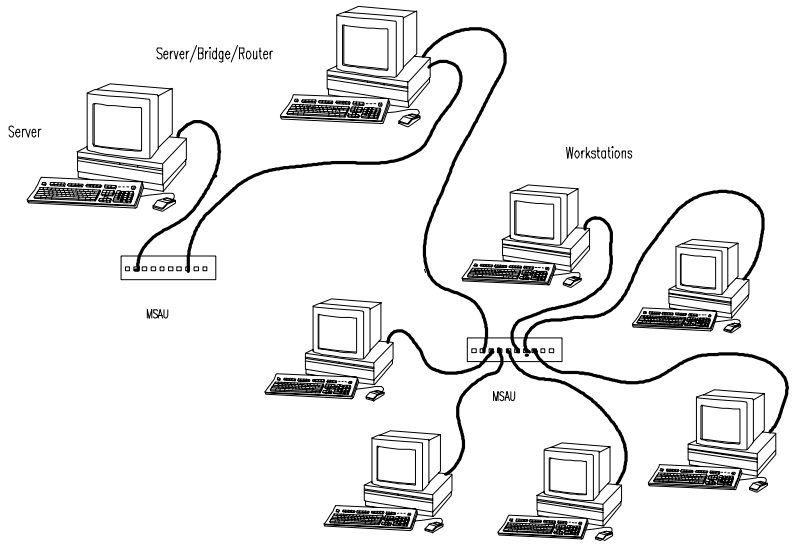
The user will be prompted to insert a floppy disk in drive A: when the test starts so the test results can be recorded. The results may be printed after the test completes. The test runs to completion unless an error occurs or until the test discovers an enhancement that is not supported. A message will display about the test status. If an error occurs or if the test finds a parameter that is not supported the test will stop and wait for a response from the keyboard before continuing. After a key is pressed the test continues execution and checks for other supported enhancements. A results summary will be displayed on the screen when the test is finished running.

Configuration 2

The configuration for the second test consists of two NetWare 3.1x servers, 6 workstations and 2 MSAUs. The first server should be set up as a standard server. The second server (the bridge) should be configured with two Token-Ring interface boards, each connected to a separate ring. The drivers under test on the second server should be loaded with the Novell MultiProtocol Router (v2.1) software and configured for Source Route Bridging.

This configuration is set up as shown in Figure 2. This test ensures correct Source Route bridging in a normal operating environment.

Figure 2
Source Route Bridging System Test



Setup

A non-routing 3.1x server configured as a source routing end station (using ROUTE.NLM) is attached to the first MSAU. A MultiProtocol Router is set up with one side of the bridge (the second server) connected to the second MSAU along with 6 source routing enabled workstations (using ROUTE.COM). The other side of the bridge is connected to the first MSAU.

A subset of currently used driver (MLID) tests (Read/Write and Endurance) are then performed on this configuration.

NetWork Configuration	**(See Figure 2)**
Test Software:	ROUTE.COM, ROUTE.NLM, LANTEST.EXE, NOVELL MULTIPROTOCOL ROUTER.
Test Objective:	Verify correct operation of HSM's with Source Route Bridging enhancements.
Pass/Fail Criteria:	Must pass Read/Write and Endurance test without failures.
Test Duration:	1 hour (minimum) for LANTEST Read/Write test. 12 hours (minimum) for LANTEST Endurance test.

Running the Test

To start this test,

1. Set up the hardware as shown in Figure 2.
2. Make sure that ROUTE.NLM is loaded (on the non-routing server) and that the server is functioning properly.

   ROUTE.NLM needs to be loaded and bound for both Token-Ring and Token-Ring_Snap frame types.
3. Load the driver and configure it for Token-Ring and Token-Ring_Snap then bind it to IPX.
4. Make sure the Server/Bridge/Router has the MultiProtocol Router software loaded on it and is functioning properly. (See NetWare MultiProtocol Router Installation Guide).
5. Boot the workstations and login to the server.
6. Run LANTEST.EXE on each workstation
   - Run the Read/Write test for a minimum of 1 hour.
   - Run the Endurance test for a minimum of 12 hours.

The tests must run without errors for the time specified.

Appendix A

Brouter Support Test Results

If your driver includes Brouter Support please attach this form to the ODI Server Driver Test Results Form.

Configuration 1-Stand Alone Test . . . . . Passed [ ] Failed [ ]
Configuration 2-System Test . . . . . Passed [ ] Failed [ ]

Index

P

R

S

T