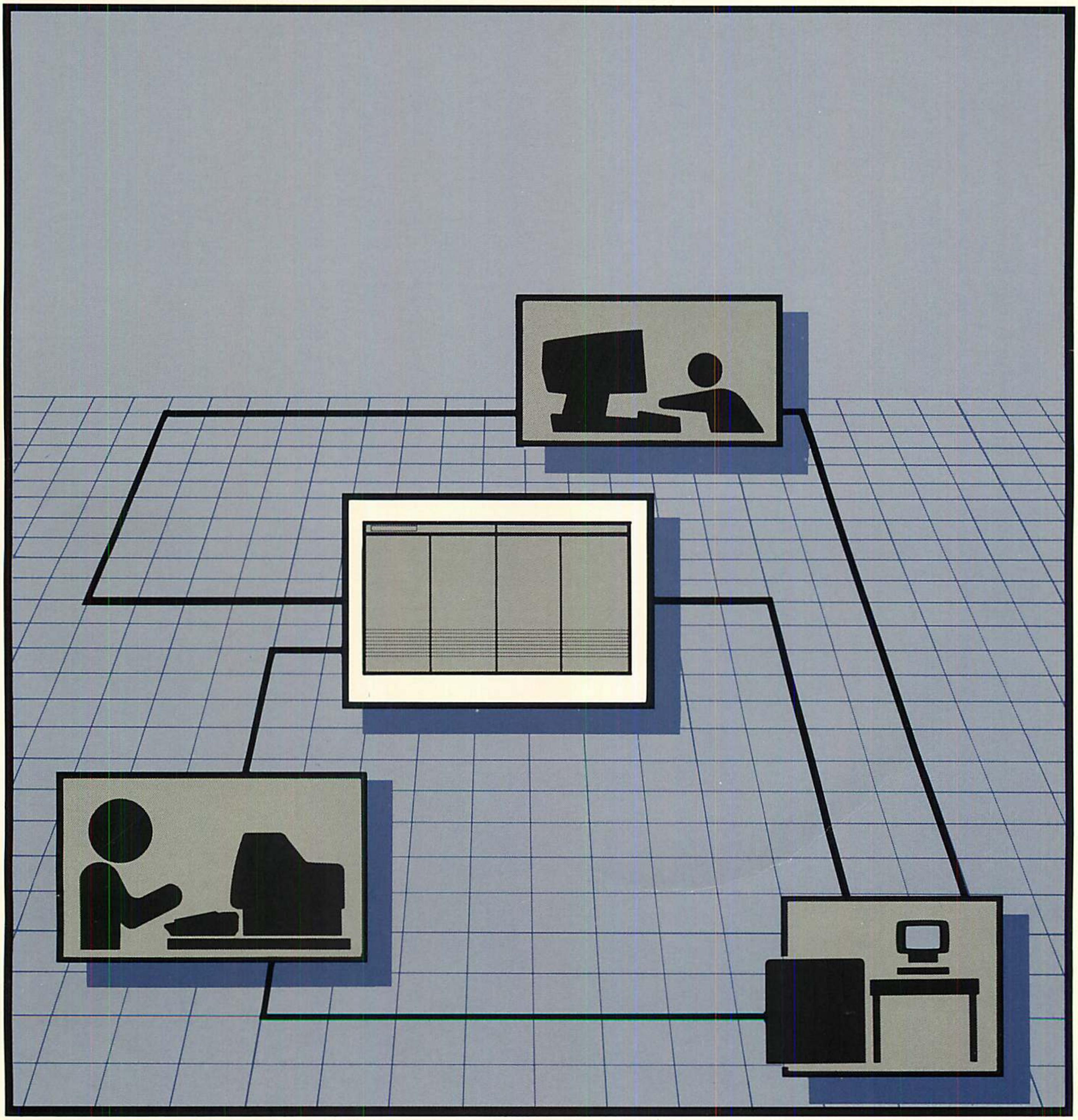
HP AdvanceNet

HEWLETT PACKARD

NS3000/V

Network Manager Reference Manual
Volume II



HP AdvanceNet

NS3000/V
Network Manager Reference Manual

Volume II

HEWLETT PACKARD
19420 Homestead Road, Cupertino, CA 95014

Part No. 32344-90012
U0189

Printed in U.S.A. MAY 1987
Update 4, JAN 1989

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

PRINTING HISTORY

New editions are complete revisions of the manual. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The dates on the title page change only when a new edition or a new update is published. No information is incorporated into a reprinting unless it appears as a prior update; the edition does not change when an update is incorporated.

The software code printed alongside the date indicates the version level of the software product at the time the manual or update was issued. Many product updates and fixes do not require manual changes and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one to one correspondence between product updates and manual updates.

First Edition	JUN 1985	32344A.00.00
Update 1	NOV 1985	32344A.00.00
Update 2	JULY 1986	32344A.00.00
Update 3	NOV 1986	32344A.00.02
Second Edition	MAY 1987	32344A.00.04
Update 1	JULY 1987	32344A.00.05
Update 2	JULY 1988	32344A.00.08
Update 3	OCT 1988	32344A.00.09
Update 4	JAN 1989	32344A.00.09

PREFACE

HP Network Services for MPE-V/E Based Systems (NS3000/V) is the HP data communications product that enables your HP 3000 to communicate with other HP computer systems as part of a distributed network. These systems can be other HP 3000s, HP 9000s, HP 1000s and PCs. Networks that operate over NS3000/V links can be interconnected to form a catenet, or internetwork. There are several network link products for NS3000/V, as described below.

The following link products connect computers on a local area network using the IEEE 802.3 networking standard:

- ThinLAN/3000 Link (includes ThickLAN option for thick coaxial cable).
- StarLAN/3000 Link.
- StarLAN 10 3000/V Link

The term IEEE 802.3 links is used to designate information that applies to all the IEEE 802.3 links (ThinLAN/3000 Link, StarLAN/3000 Link and StarLAN 10 3000/V Link).

The ThinLAN/3000 Link, including the ThickLAN option, can connect HP 3000s with HP 1000s and HP 9000s. With MPE-V release V delta 5 or later, the IEEE 802.3 links can be configured to support Ethernet* traffic concurrently with IEEE 802.3 traffic over the LAN. For example using third-party ARPA services, an HP 3000 can communicate with ARPA services on an Ethernet node on the LAN.

Two other link products enable you to establish remote connections to HP 3000s, as well as local connections, using point-to-point networking technology:

- Asynchronous SERIAL Network Link for MPE-V/E based systems (Asynchronous 3000/V Link).
- NS Point-to-Point Network Link for MPE-V/E based systems (NS Point-to-Point 3000/V Link).

Finally, the following link product allows HP 3000s to connect to public or private packet switching networks (PSNs) using NS3000/V.

- NS X.25 Network Link for MPE-V/E based systems (NS X.25 3000/V Link).

Intended Audience of this Manual

This manual is intended for those with a good deal of knowledge in data communications. Also required is knowledge of the MPE operating system at the system supervisor level, and a familiarity with the SYSDUMP dialogue, resource management and console commands.

*Ethernet is a registered trademark of Xerox Corporation.

Related Publications

Refer to Volume I of this *NS3000/V Network Manager Reference Manual* for a list of additional HP publications concerned with NS3000/V and other HP networking products.

Organization of the Manual

This manual is divided into two volumes. Volume I introduces NS3000/V and describes network architecture, management, design, and configuration.

This volume, Volume II, describes tasks performed after initial network configuration. It contains the following sections:

Section 1, Commands, describes the MPE commands for NS3000/V link products.

Section 2, Software and Line Verification, describes both the utilities available for software verification and the line tests used to check that a node is communicating correctly with a network.

Section 3, Logging and Tracing, describes the NMS Trace/Log File Analyzer (NMDUMP).

Section 4, Changing the Network, describes how to change the network topology.

Appendix A, Network Management with OpenView, summarizes the functions of the network management command NCSCONTROL which is used with the OpenView NS Monitor Applications products.

This manual also contains a glossary.

CONVENTIONS USED IN THIS MANUAL

NOTATION **DESCRIPTION**

nonitalics

Words in syntax statements which are not in italics must be entered exactly as shown. Punctuation characters other than brackets, braces and ellipses must also be entered exactly as shown. For example:

EXIT;

italics

Words in syntax statements which are in italics denote a parameter which must be replaced by a user-supplied variable. For example:

CLOSE *filename*

[ ]

An element inside brackets in a syntax statement is optional. Several elements stacked inside brackets means the user may select any one or none of these elements. For example:

$\begin{bmatrix}A\\B\end{bmatrix}$ User *may* select A or B or neither.

{ }

When several elements are stacked within braces in a syntax statement, the user must select one of those elements. For example:

$\begin{Bmatrix}A\\B\\C\end{Bmatrix}$ User *must* select A or B or C.

...

A horizontal ellipsis in a syntax statement indicates that a previous element may be repeated. For example:

[,*itemname*]...;

In addition, vertical and horizontal ellipses may be used in examples to indicate that portions of the example have been omitted.

A shaded delimiter preceding a parameter in a syntax statement indicates that the delimiter *must* be supplied whenever (a) that parameter is included or (b) that parameter is omitted and any *other* parameter which follows is included. For example:

itema[,*itemb*][,*itemc*]

means that the following are allowed:

itema
itema,*itemb*
itema,*itemb*,*itemc*
itema,,*itemc*

CONVENTIONS (continued)

Δ

When necessary for clarity, the symbol Δ may be used in a syntax statement to indicate a required blank or an exact number of blanks. For example:

SET[(*modifier*)]Δ(*variable*);

underlining

When necessary for clarity in an example, user input may be underlined. For example:

NEW NAME? ALPHA

Brackets, braces or ellipses appearing in syntax or format statements which must be entered as shown will be underlined. For example:

LET *var*[[*subscript*]] = *value*

Output and input/output parameters are underlined. A notation in the description of each parameter distinguishes input/output from output parameters. For example:

CREATE (*parm1*,*parm2*,*flags*,*error*)

shading

Shading represents inverse video on the terminal's screen. In addition, it is used to emphasize key portions of an example.

▭

The symbol ▭ may be used to indicate a key on the terminal's keyboard. For example, RETURN indicates the carriage return key.

CONTROL *char*

Control characters are indicated by CONTROL followed by the character. For example, CONTROL Y means the user presses the control key and the character Y simultaneously.

CONTENTS

Section 1
COMMANDS

Section 2
SOFTWARE AND LINE VERIFICATION

CONTENTS (continued)

Section 3
LOG AND TRACE FILES

Section 4
CHANGING A NETWORK

Appendix A
Network Management with OpenView

COMMANDS

SECTION 1

This section describes the NS3000/V network commands for the NS3000/V services and associated links. The commands are listed and described below in Table 1-1. The capabilities required by the user, if any, are noted for each command. The commands are presented in alphabetical order.

TABLE 1-1. MPE COMMANDS

Command	Description
DSCONTROL	Initiates or terminates the DS Compatible Network Links. Provides various functions to control their operation: CANCEL, TRACE, MON, COMP, and RETRY. Must be executed from the console, unless distributed with the ALLOW or ASSOCIATE commands. If distributed, user must have CS (allows access to Communication Services software) and ND (allows access to nonsharable I/O devices) capabilities.
LINKCONTROL TRACE	Activates or deactivates link level tracing. User must have NM (Node Manager) capability.
NETCONTROL	Initiates, terminates, and controls the operation of the Network Transport. User must have NM capability.
NETCONTROL ADDLINK	Dynamically adds a configured network link to the active network configuration. User must have NM capability.
NETCONTROL DELLINK	Dynamically deletes a configured network link from the active network configuration. User must have NM capability.
NETCONTROL MON	Enables or disables internal monitoring for the Network Transport. Requires a cold dump; used only for troubleshooting and only on the advice of your HP Representative. User must have NM capability.
NETCONTROL START	Initiates the Network Transport functional entities. User must have NM capability.
NETCONTROL STATUS	Displays the status of the Network Transport functional entities. User must have NM capability.

TABLE 1-1. MPE COMMANDS,CON'T.

Command	Description
NETCONTROL STOP	Terminates the Network Transport functional entities. Can terminate the Network Services. User must have NM capability.
NETCONTROL TRACE	Enables or disables message tracing for a specified network transport functional entity. User must have NM capability.
NETCONTROL UPDATE	Dynamically updates selected network transport configuration parameters for an active network interface. User must have NM capability.
NETCONTROL VERSION	Displays the version of the software modules of the Network Transport. User must have NM capability.
NSCONTROL	Initiates, terminates and controls the operation of the Network Services. User must have NM capability.
NSCONTROL ABORT	Immediately terminates the Network Services. User must have NM capability.
NSCONTROL AUTOLOGON	Enables or disables the autologon feature for the NFT, RFA, and RPM remote network services. User must have NM capability.
NSCONTROL LOADKEYS	Loads the Network Services command keywords. Used for localization. User must have NM capability.
NSCONTROL LOG	Enables or disables detailed logging (configured as CLAS0004 of SUB0006) for the Network Services. User must have NM capability.
NSCONTROL SERVER	Alters the characteristics of the Network Services server processes. User must have NM capability.
NSCONTROL START	Initiates the Network Services. User must have NM capability.
NSCONTROL STATUS	Displays status information about the Network Services. User must have NM capability.
NSCONTROL STOP	Allows existing users to continue with current task, but prevents initiation of any new tasks or new users for the Network Services. User must have NM capability.
NSCONTROL VERSION	Displays the version of the software modules of the Network Services. User must have NM capability.

TABLE 1-1. MPE COMMANDS, CON'T.

Command	Description
RESUMENMLOG	Resumes logging after a recoverable error. User must have NM capability.
SHOWCOM	Displays status and error information for communications links. Executable only from the console unless distributed with ALLOW or ASSOCIATE. User capability not required.
SHOWNMLOG	Displays the identification number and available space of the log file. User must have NM capability.
SWITCHNMLOG	Closes the current log file and creates and opens a new one. User must have NM capability.

EXPLANATION OF COMMAND EXAMPLES

Many of the examples of command use included in this section use an IEEE 802.3 local area network as an example network. As a result, the network interface name (*niName*) shown in these examples is LAN1, signifying a local area network that has been given that name. Any configured network interface name can be used, as long as the network's network interface type is one that is allowed for the particular command function being used.

DSCONTROL

Enables and disables operational and troubleshooting functions on each DS Compatible Link.

Syntax

```
:DSCONTROL dsdevice;function[;function]...
```

where the parameter *function* has the following options:

```
[;CANCEL]

[ ;OPEN  [,MASTER] [,[SPEED=]speed] ]
[ ;SHUT  [,SLAVE ]                  ]

[ ;TRACE [,ON  [,ALL][,mask][,numentries][,WRAP][,filename] ] ]
[        [,OFF                                              ] ]

[ ;MON     [,DS] ]
[ ;MONOFF  [,CS] ]

[ ;COMP   ]
[ ;NOCOMP ]

[ ;RETRY={DEFAULT} ]
[        {count  } ]
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	NO
	Programmatically?	YES
Breakable?		YES
Capabilities?		NM

Parameters

dsdevice

(Required parameter.) The logical device number or the device class name of the DS Compatible communications device (IODSO or IODSX). On your system's I/O configuration listing, the device is back referenced by a pound sign (#) to a previously defined INP.

CANCEL

Applicable only to X.21. Cancels all queued outgoing call requests. Sends an abort request to the communications device. Validates that the device is X.21 related and that there is a queued request. CANCEL is executed before any other DSCONTROL parameter.

OPEN

(Required parameter.) Makes the line available for remote communication. The DS Point-to-Point Network Links (IODSO) are enabled but no activity is initiated on the communications link. For the DS X.25 Network Link (IODSX), a communications link with the Packet Switched Network (PSN) is established

MASTER

Limits INP line activity for the DS Compatible Links to outgoing requests only. No incoming sessions are allowed.

SLAVE

Limits INP line activity for the DS Compatible Links to incoming requests only; no outgoing activity is allowed.

Default: Both MASTER and SLAVE processing are allowed.

SPEED=*lineSpeed*

Transmission rate in characters per second (Bit Rate/8). This parameter is effective only if your system generation for the line (SYSDUMP) selected SPEED CHANGEABLE. Refer to the Discussion for details about when to specify SPEED=*linespeed*.

Remember, both ends of the line must operate at the same speed.

Default: System configuration values

SHUT

Initiates an orderly line shutdown. Refer to Discussion for details about the line closing procedure.

COMP

Sets data compression as the default mode of operation for all line users. The line need not be open to use COMP.

NOCOMP

Sets uncompressed data as the default mode of operation for all line users. The line need not be open to use NOCOMP.

RETRY=[DEFAULT
count]

Changes the communications error retry count to the specified value. The retry counter controls the number of times the system attempts to send or receive a message across the communications link.

DEFAULT

Specifies a limit of 15 retries after a line error occurs.

count Can be any value within the range of 0 to 255.

Default: 15

TRACE,ON

Activates the CS trace facility to provide a record of all INP communications activity. Trace parameters are positional as shown in the syntax diagram. The line must already be open or the OPEN keyword must also be included to open the line.

ALL generates trace records for all CS intrinsic calls. If ALL is not specified, then trace records are written only when an intrinsic call completes with a transmission error, in which case, the word ERROR appears on the trace listing.

mask indicates the type of activities to be traced, as follows (PCMP entries are generated automatically):

%000, or omitted, means use the driver default mask (%037, so all entries except PSTN and INP interconnect entries are generated)

%001 = generate PSTX entries

%002 = generate PSCT entries

%004 = generate PRTX entries

%010 = generate PRCT entries

%020 = generate POPR and PEDT entries

%040 = generate PSTN entries

%100 = generate IP interconnect entries

numEntries is a decimal integer for the maximum number of trace entries in a trace record. It cannot be greater than 248. The value actually used by the trace facility will be the largest integer multiple of eight that is not greater than the number you enter. For an INP, the value may not exceed 24. (If the value requested for an INP is greater than 24, a warning message will be printed and the maximum default of 24 will be used.) If *numEntries* is set to zero or omitted, there will be a maximum of 24 trace entries per trace record for the INP. It is not possible to change the value of *numEntries* once a trace file has been built. If the value you choose is inadequate, you will have to purge the file and rebuild it, or let CS/3000 rebuild it.

WRAP causes trace entries that overflow the trace record area (greater than *numEntries*) to overlay the prior trace entries. If WRAP is omitted, overflow trace entries are discarded, and NOWRAP appears on the trace listing. (This parameter does not affect the EOF marker of the file.)

If WRAP is specified then entries are deposited in a trace record in a circular pattern. For example, with a maximum of 35 trace entries per trace record, trace entries beyond the 35th will overlay the first, second, third (and so on) trace entries in the record. When this happens, the overlaid trace entries will be missing from the listing; a warning message will appear in the listing stating that the entries are missing.

fileName names the file the user wants the trace information to be written to. If no name is supplied, CS/3000 will create a file named DSTRCnnn, where nnn is the right-justified ldev number of the DS Compatible device. For example, if the IODS0 ldev is 51, the trace file name is DSTRC051. If a trace file with the same name exists it will be purged, and a new trace file will be created.

NOTE

When using a DS X.25 Network Link, there is not enough space left on the INP to trace using the default number of entries, 24. The user must specify *numEntries* = 16.

TRACE,OFF

Deactivates the CS trace facility, so that no records are kept of INP line actions, states, and events. Also closes the trace file.

```
MON [ ,DS ]
    [ ,CS ]
```

Activates an internal communication monitoring activity that records interrupts and other events to a core-resident table maintained by MPE-V. This information can only be accessed by a subsequent cold dump of the HP 3000 system. The line must be open for the use of MON. Used only for system troubleshooting.

MON Requests monitoring of all levels of activity.

MON,DS Requests monitoring at the DS/3000 level of internal software operation.

MON,CS Requests monitoring at the Communication System (CS/3000) level of internal software operation.

Default: No monitoring

MOFF

Deactivates internal monitoring. Line must be open for the use of MOFF.

Discussion

The DSCONTROL command can be issued from the master console or it can be distributed with the ALLOW or ASSOCIATE commands. If the command is distributed, the user requires CS and ND capability.

Only one DS Compatible communications device can be active (OPEN) on a controller at any given time. Once opened (with the DSCONTROL command), a communications link can be shared by *multiple* NS3000/V users. It cannot, however, be shared by users of other communications subsystems supported by your system (for example, MRJE). Thus, you must SHUT the DS Compatible communications device before the controller can be opened for use by another subsystem.

Before issuing a DSCONTROL command, use the SHOWDEV command to check whether a communications link is already established. The display provided by SHOWDEV will indicate available, AVAIL, or unavailable, UNAVAIL, for the specified ldev of the communications driver, IODS0 or IODSX. If the device is available for use that means the line is already established. However, if the ldev for the INP port is in use by another datacomm subsystem, it will be shown as unavailable on the SHOWDEV display.

If a DS Compatible device class includes more than one device, the functions specified in the DSCONTROL command apply to all devices in that class. If you have not been ALLOWed to use this command, you can only control those devices in the device class with which you have been ASSOCIATEd (if any).

If you include more than one function in a DSCONTROL command, each function (with its subparameter list) must be separated by a semicolon. A function that duplicates or conflicts with a previous function overrides that function. Functions can appear in any order but are executed in the following order:

1. CANCEL
2. OPEN/SHUT
3. TRACE
4. MON/MOFF
5. COMP/NOCOMP

The default name of the trace file is:

```
DSTRCxxx.PUB.SYS
```

where *xxx* is the logical device number of the *dsdevice*.

If no trace file exists when you turn on the trace facility and you do not specify *numentries*, the system creates a file to hold 24 entries in each record. If you are using a DS X.25 Network Link, however, you must specify *numEntries* =16, because there is not enough space on the INP for 24.

Specify SPEED=*lineSpeed* if yours is a European installation with modems running at half speed, or if the line is hardwired and you want to override the configured default. It may be also be necessary to include this parameter if your node is communicating with a node using DS, which is either an HP 3000, Series II or III on MPE-IV or earlier, or an HP 1000 that is using HSI controllers, and the length of cables used for HSI communications has been changed since the system was configured. The value of *linespeed* is as follows:

HSI speed	250000 (cable lengths less than 1000 ft.)
	125000 (cable lengths greater than 1000 ft.)

INP or SSLC speed	250, 300, 600, or 1200
INP only speed	2400 or 7200

The SPEED= keyword may be omitted from a DSCONTROL command. For example, the following two commands have exactly the same effect:

```
:DSCONTROL 60;OPEN,MASTER,SPEED=250000

:DSCONTROL 60;OPEN,MASTER,250000
```

Remember, both ends of the line must operate at the same speed.

For DS Point-to-Point Links, which use the BSC protocol, the SHUT parameter initiates an orderly line closing procedure. If no sessions or applications are using the line when you shut it, line disconnection occurs immediately. If any user (including applications) has the line open, the line remains connected until all sessions and applications CLOSE the line, or until those accessing the line terminate or are aborted. Once CLOSE is issued by the console operator, no new users may access the line until the operator reopens it. When using the capability of the DS X.25 Network Link to connect to a PSN, the SHUT parameter disconnects the line immediately, even if there are current users on the line.

Occasionally you may not be able to SHUT a DS Point-to-Point line. This could happen, for example, if an NS3000/V user forgot to issue a DSLINE *xxx*;CLOSE command and still has a local session. It could also happen if a remote session is hung. In such a situation, you can immediately terminate all activity across the line by issuing an ABORTIO *xxx* command (where *xxx* is the logical device number of the *dsdevice*). Use the SHOWDEV command to verify that the device is available. You may need to issue multiple ABORTIO commands before the device becomes available. Following the successful use of the ABORTIO command, a second DSCONTROL *xxx*;SHUT command will complete successfully.

Example

To open DS X.25 line number 55, thereby making it available for use by NS3000/V users, enter:

```
:DSCONTROL 55;OPEN
```

To permit the local HP 3000 to process only master (outgoing) requests on DS Compatible line number 55, enter:

```
:DSCONTROL 55;OPEN,MASTER
```

To activate the CS Trace facility for DS Compatible line 55 (the line is already open), enter:

```
:DSCONTROL 55;TRACE,ON,ALL
```

To open DS X.25 line 55 and activate CS Trace with a maximum of 250 entries in a trace record, enter:

```
:DSCONTROL 55;OPEN;TRACE,ON,,,250
```

To open the line named REMSYS and establish compression as a default and enable internal monitoring, enter: :DSCONTROL REMSYS;OPEN;COMP;MON

LINKCONTROL

Activates or deactivates link level tracing on the specified communications line. The line must already be initialized.

Syntax

```
LINKCONTROL linkName;TRACE=ON[,ALL][,mask][,numEntries][,WRAP][,fileName]
```

```
LINKCONTROL linkName;TRACE=OFF
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	YES
	Programmatically?	YES
Breakable?		NO
Capabilities?		NM

Parameters

linkName

The configured name of an active data communications line. This name must be the same as that specified in the Link Configuration screen of NMMGR. Refer to Volume I, Section 7 for information on link configuration.

TRACE=ON

Activates link level tracing. Only one active trace per link is allowed. If a trace is already active, it will be closed and another trace activated.

The trace file activated by this command *does not* use the link trace file values or filename configured through NMMGR.

ALL

Generates trace records for all intrinsic calls. If ALL is not specified then trace records are written only when an intrinsic call completes with a transmission error, in which case the word ERROR appears on the trace listing.

mask

An octal number indicating the type of trace entries to be generated. Value is entered in the format %*nnn* Combine one or more of these values to generate the entry types shown

- %001 = Protocol Send Text entries (PSTX).
- %002 = Protocol Send Control entries (PSCT).
- %004 = Protocol Receive Text entries (PRTX).
- %010 = Protocol Receive Control entries (PRCT).
- %020 = Protocol Operator entries (POPR) and Protocol Editor entries (PEDT).
- %040 = Protocol State Transition entries (PSTN).
- %100 = IP interconnect entries.

Default: %37

numEntries

The maximum number of entries in a trace file record. Must be a multiple of eight that is less than or equal to 24. If the value specified is not a multiple of eight, NMS converts it to the next highest multiple of eight. Trace entries are deposited in a record in a circular manner.

Default: 24

WRAP

Specifies that if the trace record is full for a given CS intrinsic, previous entries are overlayed. Its absence indicates that succeeding entries will be flushed. This parameter does not affect the EOF marker of the trace file. HP recommends that this parameter be omitted.

fileName

Trace output will be sent to a file, built by CSTRACE with the specified file name. If a file name is not specified, the default destination will be LINKT*nnn*, where *nnn* is the ldev of the CS device. If a trace file of the same name exists, it will be purged and a new trace file will be created.

Default: LINKT*nnn*

TRACE=OFF

Deactivates link level tracing enabled by LINKCONTROL *or* enabled in the Link Configuration of the configuration file for the specified link name.

Discussion

The communications device of an NS3000/V link must have been activated before you can issue the LINKCONTROL TRACE=ON command. Activating the link means activating the Network Transport and the network interface for either the LANIC, INP or ATP (depending on the link). Refer to the NETCONTROL command.

There are two methods of activating link level tracing for NS links. You can activate link level tracing at any time after link initiation by using the LINKCONTROL command. You can also activate link level tracing during link initiation by using NMMGR to specify link trace parameters, in the Link Configuration for that link.

The first time a link is activated any configured link trace parameters are read and then retained. Thereafter, for the life of the system, these same link trace parameters will be used for that link whenever it is reactivated.

The LINKCONTROL TRACE=OFF command may be used to deactivate link trace, regardless of which way it was started. Link trace will automatically be deactivated whenever the link is terminated.

Trace entries are written to the trace file in a circular fashion; that is, when the maximum disc space for the trace file is reached, new entries will overwrite the oldest records in the file. This occurs whether or not the WRAP parameter was specified. (Note that the WRAP parameter applies to the entries in each record, not the storage of records in the log file.)

Link traces are CS traces. They can be formatted with the CSDUMP formatting program.

Text Reference

Refer to the *LAN/3000 Diagnostic and Troubleshooting Guide* and the *Fundamental Data Communications Handbook* for a detailed description of the types of trace entries, a discussion of link level tracing, and information on how to format trace files.

NETCONTROL

Command used to initialize, terminate, and control the operation of the Network Transport.

Syntax

```
NETCONTROL {function} [;function]...
           {entity  } [;entity  ]
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	NO
	Programmatically?	YES
Breakable?		YES
Capabilities?		NM

Parameters

function One or more of the functions defined for NETCONTROL. These are:

ADDLINK Dynamically adds a configured network link to the active network configuration. Refer to NETCONTROL ADDLINK for more information.

DELLINK Dynamically deletes a configured network link from the active network configuration. Refer to NETCONTROL DELLINK for more information.

MON Controls internal monitoring for the Network Transport. Refer to NETCONTROL MON for more information.

START Initializes the Network Transport. Refer to NETCONTROL START for more information.

STATUS Provides information about the configuration of the Network Transport. Refer to NETCONTROL STATUS for more information.

STOP — Terminates the Network Transport. Refer to NETCONTROL STOP for more information.

TRACE — Controls tracing for the Network Transport. Refer to NETCONTROL TRACE for more information.

UPDATE — Dynamically updates selected network transport configuration parameters for an active network interface. Refer to NETCONTROL UPDATE for more information.

VERSION — Provides a listing of the version numbers for the software modules of the Network Transport; essentially a subset of the listing provided by the NMMAINT utility described in Section 2 . Refer to NETCONTROL VERSION for more information.

Only one of each type of function (START, TRACEON, MONON, etc.) is allowed on a command line. For example, the command:

```
:NETCONTROL TRACEON=HDM;START;TRACEON=HD;NET=LAN1
```

is not allowed. Moreover, START and STOP, TRACEON and TRACEOFF, and MONON and MONOFF cannot both be issued on the same command line.

entity — One or more of the entities defined for NETCONTROL. The keywords for these entities are shown in Figure 1-1.

NET — Allows you to specify an entity that consists of a network interface and the protocol modules configured for that network interface. You use the network interface name configured in NMMGR to specify the group entity. This group entity can be used at initialization or termination.

GATE — Allows you to specify the name of a configured gateway half.

NI — Allows you to specify a particular network interface module for a function to act on.

PROT — Allows you to specify a particular protocol module for a function to act on. Refer to Figure 1-1 for an overview of the protocols available.

Refer to the discussion below for more information. Each function is executed against one or more entities. For information on how the entities are affected by a particular function, refer to the command page for that function.

CONF — Allows you to specify the configuration file to be used to initialize the Network Transport.

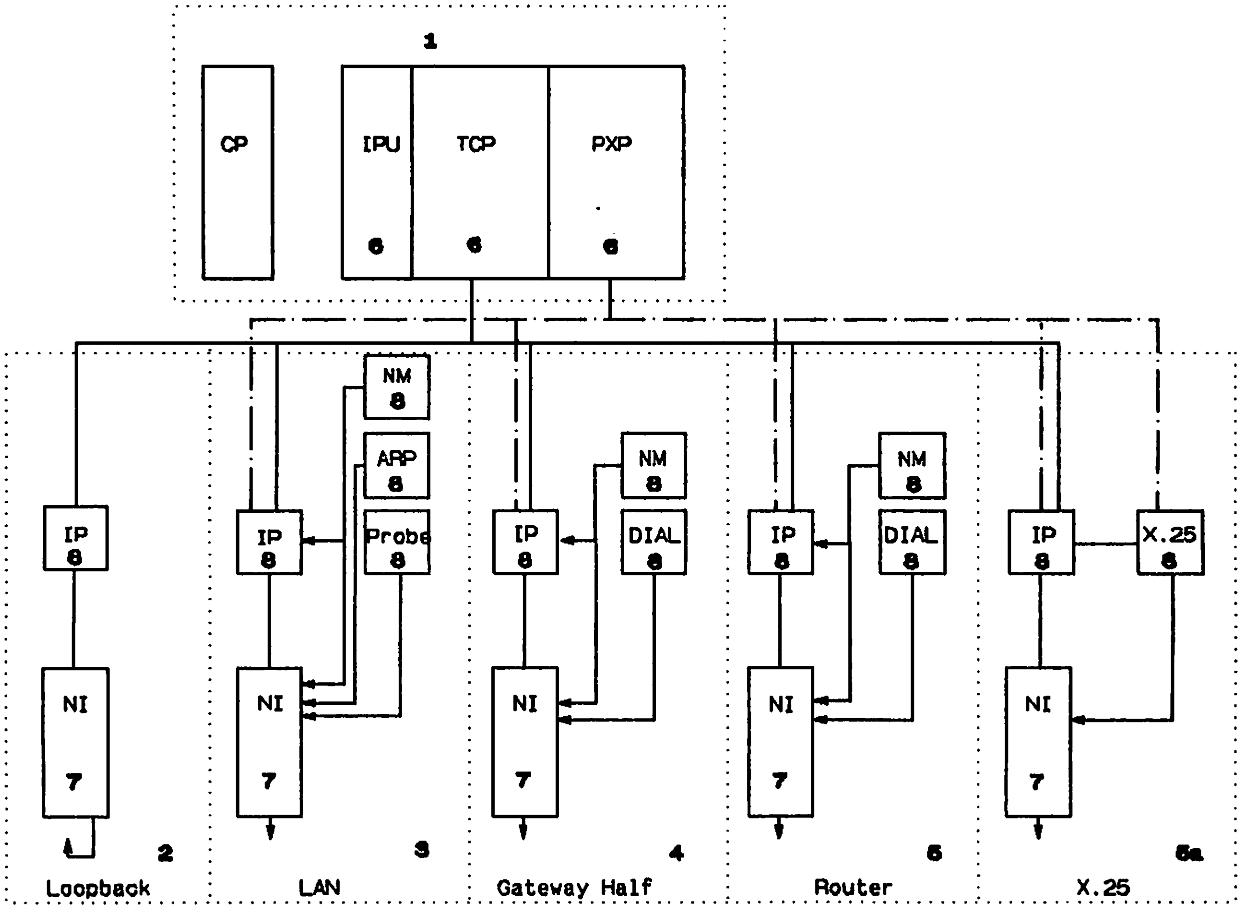
Discussion

The NETCONTROL command is composed of functions (START, STOP, STATUS, etc.) to be executed against one or more entities. These entities are defined as particular modules or groups of modules of the Network Transport, as configured with NMMGR, basically the network interfaces, protocols, and the configuration file. Refer to Volume I, Section 8, Network Transport Configuration, for more information.

The entities of NETCONTROL are shown in Figure 1-1. Notice that the first five entities are composed of groups of modules. For example, let us look at the third entity, labelled 3. This entity, NET=*niName* where *niName* is the configured LAN NI name, combines the network interface (NI) configured for any IEEE 802.3 link and the protocols configured for that NI which can include IP, Probe, ARP and NM.

The remaining three entities, numbers 6-8 in Figure 1-1, allow exact specification of one and only one module of the Network Transport. This is especially useful when troubleshooting. Refer to NETCONTROL STATUS, NETCONTROL TRACE, and NETCONTROL MON for more information.

NETWORK TRANSPORT MODULES



NETCONTROL ENTITIES

1 General Transport: Control Process (CP) and General Protocols (IPU, TCP, PXP)

2 NET=*niName* (Loopback - includes NI and IP)

3 NET=*niName* (LAN - includes IP, NI, Probe, and ARP)

4 GATE=*gateName* (Gateway Half - includes IP, NI and Dial)

5 NET=*niName* (Router - includes IP, NI and Dial)

5A NET=niName (X.25 - includes IP, NI and X.25)

6 PROT=*gprot* (one of IPU, TCP, or PXP)

7 NI=*niName* (Loopback, LAN gateway half, X.25 or router)

8 NI=*niName*;PROT=*niProt* (IP only for loopback; IP, Probe, ARP, or NM for LAN; IP Dial or NM for gateway half; IP, Dial, or NM for router; IP or X.25 for X.25)

Figure 1-1. The NETCONTROL Entities Defined

NETCONTROL ADDLINK

Dynamically adds a configured network link to the active network configuration.

Syntax

```
NETCONTROL {GATE=gateName};ADDLINK=linkName[;function]
           {NET=niName   }
```

Parameters

GATE=*gateName*

In place of the NET entity, specifies the name of a configured gateway half.

NET=*niName*

Specifies the network interface. Enter any valid network interface name, as configured with NMMGR.

ADDLINK=*linkName*

Specifies the name of the link to be dynamically added to the active network configuration. *linkName* must be a valid NI Link name configured in the Link Configuration screen (refer to Figure 7-1) and corresponding to the Link name supplied in the Network Interface Links screen.

function

You may want to issue one or more of the NETCONTROL functions on the same command line with ADDLINK. Only one of each type of function is allowed on a command line. STOP is permissible, but functionless, since it deallocates the tables associated with the network. Also, TRACEON and TRACEOFF, MONON and MONOFF cannot both be issued on the same command line. Moreover, be aware that the functions STATUS, TRACE, and MON act only on the IP protocol when the network interface is specified with the NET keyword. However, the PROT keyword can be used to specify the general protocol for the function to act on. If a network interface or general protocol is not specified, the functions act on the Control Process. For these reasons, although other functions can be issued on the same command line with START, it is not recommended. Refer to the command pages describing NETCONTROL MON, NETCONTROL STATUS, NETCONTROL START, NETCONTROL STOP, NETCONTROL TRACE, and NETCONTROL VERSION for more information.

Discussion

The ADDLINK parameter adds a configured link to the active network configuration without having to first bring down and then restart the network transport. The link being added can be a link to a newly configured node, a link being shared with another node, or a link being restarted after failure due to link errors. Specifying the name of a link that has not been configured or specifying the name of a link that is already active will result in no change to the active configuration but will produce error messages at the console.

NETCONTROL DELLINK

Dynamically deletes a configured network link from the active network configuration.

Syntax

```
NETCONTROL {GATE=gateName};DELLINK=linkName[;function]
           {NET=niName   }
```

Parameters

GATE=*gateName*

In place of the NET entity, specifies the name of a configured gateway half.

NET=*niName*

Specifies the network interface. Enter any valid network interface name, as configured with NMMGR.

DELLINK=*linkName*

Specifies the name of the link to be dynamically deleted from the active network configuration. *linkName* must be a valid NI Link name configured in the Link Configuration screen (refer to Figure 7-1) and corresponding to the Link name supplied in the Network Interface Links screen.

function

You may want to issue one or more of the NETCONTROL functions on the same command line with DELLINK. Only one of each type of function is allowed on a command line. STOP is permissible, but functionless, since it deallocates the tables associated with the network. Also, TRACEON and TRACEOFF, MONON and MONOFF cannot both be issued on the same command line. Moreover, be aware that the functions STATUS, TRACE, and MON act only on the IP protocol when the network interface is specified with the NET keyword. However, the PROT keyword can be used to specify the general protocol for the function to act on. If a network interface or general protocol is not specified, the functions act on the Control Process. For these reasons, although other functions can be issued on the same command line with START, it is not recommended. Refer to the command pages describing NETCONTROL MON, NETCONTROL STATUS, NETCONTROL START, NETCONTROL STOP, NETCONTROL TRACE, and NETCONTROL VERSION for more information.

Discussion

The DELLINK parameter deletes a configured link from the active network configuration without having to first bring down the entire network interface. This command is particularly useful for bringing down a hardware device for cabling and modem changes, for making a device unusable for security reasons, or for allowing a device to be shared with other subsystems.

The link being deleted must be an existing configured link. Specifying the name of a link that has not been configured or specifying the name of a link that is already deleted will result in no change to the active configuration but will produce error messages at the console.

NETCONTROL MON

Enables or disables internal monitoring.

Syntax

```
NETCONTROL ⎡NI=niName[;PROT=niProt];⎤ ⎧MONON ⎫
           ⎣PROT=gProt;             ⎦ ⎩MONOFF⎭
```

Parameters

NI=*niName*

Specifies that a network interface is the pertinent entity for each specified function to act on, unless modified by the PROT keyword. Enter any valid network interface name, as configured with NMMGR.

PROT=$\left\{\begin{matrix} niProt \\ gProt \end{matrix}\right\}$

Specifies that a protocol is the pertinent entity for each specified function to act on. Enter the name of the protocol, as follows:

niProt — The name of a network interface protocol; must be one of PROBE, IP, ARP or NM for the LAN NI, must be IP only for the Loopback NI, must be one of X25 or IP for the X.25 NI, must be one of DIAL, IP or NM for the Router and Gateway Half Link NIs. For a function to operate on a network interface protocol, the name of the network interface must also be specified, using the syntax NI=*niName*;PROT=*niProt*.

gprot — The name of a general protocol; must be one of TCP, PXP, or IPU.

MONON

Activates internal monitoring. The general transport must be operational, or START must be issued on the same command line with MONON, for this function to take affect. This function, when used in combination with the network interface and protocol keywords (NI and PROT) allows you to selectively activate monitoring for a particular protocol or network interface for a designated network. The default is to monitor the CP.

MONOFF

Deactivates internal monitoring for the specified entity. The monitor must be operational for this function to take affect. This function, when used in combination with the network interface and protocol keywords (NI and PROT) allows you to selectively deactivate monitoring for a particular protocol or network interface for a designated network. If you do not specify, the default is to deactivate monitoring for the CP. If monitoring for other entities is on, it continues until specifically deactivated.

NETCONTROL START

Initiates the Network Transport, including the general protocols and the network interfaces. Initiates individual networks on a running transport. Allows you to specify which configuration file to use for initiation.

Syntax

```
NETCONTROL START[;CONF=filename] [;GATE=gateName] [;function]...
                                 [;NET=niName   ]
```

Parameters

START

This function, if issued when the general transport is not active, initializes the general transport. The NET keyword, used to designate a network interface, when combined with the function START, causes all the configured protocols and associated modules of the specified network interface to be initiated. Notice that each configured network interface must be initiated by a separate NETCONTROL command. The CONF keyword, used to designate the configuration file to use for initiation, can *only* be specified with the instance of START which initiates the general transport.

CONF=*filename*

Specifies the file name of the configuration file which the Network Transport will use during operation. Must be a valid MPE-V file name, a maximum of eight characters with the first character alphabetic. The group and account cannot be specified. The file must be of type NCONF, created or updated with NMMGR, and must reside in the NET group of the SYS account. The configuration file may only be specified as part of a NETCONTROL command that includes the START function, and only at initiation. The configuration file cannot be changed while the Network Transport is active. If the configuration file is not specified at initiation, the Network Transport uses the configuration file NSCONF.NET.SYS if it exists.

GATE=*gateName*

Specifies the name of a gateway half to be initiated by START. Enter any valid gateway half, as configured with NMMGR. All configured protocols and links associated with the specified gateway half will be initiated. However, if any other functions are included with the START function, they act only on the IP protocol of the specified gateway half.

NET=*niName*

Specifies the network interface to be initiated by START. Enter any valid network interface name, as configured with NMMGR. All configured protocols and links associated with the specified network interface will be initiated. However, if any other functions are included with the START function, they act only on the IP protocol of the specified network interface.

function

You may want to issue one or more of the NETCONTROL functions on the same command line with START. Only one of each type of function is allowed on a command line. Also, TRACEON and TRACEOFF, MONON and MONOFF cannot both be issued on the same command line, and STOP is not allowed with START. Moreover, be aware that the functions STATUS, TRACE, and MON act only on the IP protocol when the network interface is specified with the NET keyword. However, the PROT keyword can be used to specify the general protocol for the function to act on. If a network interface or general protocol is not specified, the functions act on the Control Process. For these reasons, although other functions can be issued on the same command line with START, it is not recommended. Refer to the command pages describing NETCONTROL MON, NETCONTROL STATUS, NETCONTROL TRACE, and NETCONTROL VERSION for more information.

Discussion

If the NET keyword is included in the same command line, the functions (TRACE, STATUS, and MON) will only affect the IP protocol of the specified network interface. If NET is not included, the functions act only on the Control Process. The function VERSION is not affected by the NET keyword. Providing the general transport and network interfaces are operational, the NETCONTROL functions can also be used individually with additional keywords that allow you to specify the protocol or network interface you want the function to act on.

Refer to the appropriate command for each function for more information. The first example is of a typical node initiation sequence:

Example 1

```
:NETCONTROL START;NET=LAN1
:NETCONTROL START;NET=LOOP
:NSCONTROL START;SERVER=ALL,10,40
:DSCONTROL DSMODEM;OPEN
:DSCONTROL DSX25;OPEN
:DSCONTROL DSDEV;OPEN
```

In Example 1, the node has one IEEE 802.3 Link (LAN1), requiring a NETCONTROL START to be issued for each configured network interface (NET=*niName*). Once both network interfaces (and related entities) of the Network Transport have been successfully initiated, as indicated by the lack of error messages, any other related subsystems installed on the node can be initiated. This node, as is typically the case, has NS3000/V Services installed. It also has three DS Compatible Links, consisting of a DS Point-to-Point Modem Link, configured as device class name DSMODEM, a DS X.25 Network Link, configured as node name DSX25 in the NETCONF data base, and a DS Point-to-Point Hardwired Link, configured as device class name DSDEV. To initialize the Network Services subsystem and the DS Compatible Links, an NSCONTROL START is issued after the initiation of the Network Transport, followed by a DSCONTROL command for each of the DS Compatible Links. Refer to the NSCONTROL and DSCONTROL command pages in this section for more information.

Be aware that to successfully initialize a node, the commands must be issued in the order specified: first all required NETCONTROL commands, then either the NETCONTROL commands or the DSCONTROL commands.

The first example provided an overview of initializing a node, showing where NETCONTROL fits into the process.

The next five examples examine the START function and how it affects the entities defined for initialization. Refer to Figure 1-1 and the discussion of entities under NETCONTROL. As will be shown in the examples, the keywords included with the START function and the entities affected determine which events occur at initialization. To understand this relationship, it is helpful to see the events that occur when the Network Transport is initialized. These examples show logging messages for subsystem ID SUB0003, CLAS0004 and CLAS0005, logged to the console (the NMMGR default).

Example 2

```
:NETCONTROL STATUS
Transport Not Active. (NETXPORTWARN 0001)
ENCOUNTERED ONE OR MORE WARNINGS WHILE PROCESSING COMMAND (CIERR 4437)

:NETCONTROL START
** NETXPORT Control Process; Transport start
- Loc: 50; CLAS0004; Parm= %000027; PIN: 0
** NETXPORT TCP SIP; General protocol start
- Loc: 10; CLAS0004; Parm= %000000; Port ID: %000200 %005016
** NETXPORT PXP SIP; General protocol start
- Loc: 5; CLAS0004; Parm= %000000; Port ID: %000200 %005064
** NETXPORT IP Update; General protocol start
- Loc: 3; CLAS0004; Parm= %000000; Port ID: %000200 %005132
```

This example shows the events associated with the START function at initiation. As indicated in the status report, the general transport is not active. Therefore, the first events of initiation are to initialize the Control Process (CP) and the general protocols. Compare the displayed events to the defined entities of Figure 1-1. The events displayed in this example create the general transport. The START function always creates the general transport of the Network Transport at initialization, if it does not already exist, before acting on any of the other entities.

In Example 2, notice that the configuration file was not specified with the optional CONF keyword. A configuration file is needed for any NETCONTROL START command issued when the general transport is not active. This is because the information in the configuration file is used to initialize the general transport. If you do not specify a configuration file on the first START, as was done in Example 2, the default configuration file, NSCONF.NET.SYS, is used if it exists. All subsequent NETCONTROL commands use the configuration file used to initialize the general transport. For this reason, the configuration file can only be specified in the first NETCONTROL START command, where *first* means that the general transport is not active. If the general transport is active when you try to initiate or change the configuration file, you get the errors shown in Example 3.

Example 3

```
:NETCONTROL START
            ^

Already Started. (NETXPORTERR 4045)
ENCOUNTERED ONE OR MORE ERRORS WHILE PROCESSING COMMAND. (CIERR 4436)

:NETCONTROL START;CONF=LANCONF
Transport Is Active, Configuration File Not Allowed (NETXPORTERR 4022)
ENCOUNTERED ONE OR MORE ERRORS WHILE PROCESSING COMMAND. (CIERR 4436)
```

The initiation events, as shown in Example 2, are always executed for the first NETCONTROL START command, whether or not a network interface is specified.

However, once the general transport is initialized, subsequent NETCONTROL START commands do not change the modules of the general transport. Examples 4 through 6 show what happens when a network interface is specified with the START function.

In Example 4, the IEEE 802.3 LAN NI, configured as LAN1, is started on the first NETCONTROL START command. Notice that the initiation events to initialize the general transport are immediately followed by the start of the LAN NI with its associated protocols, Internet Protocol (IP) and Probe. The final event is the initial Probe request which sends address information to all nodes on the LAN. Note that Ethernet is not enabled here since the address resolution protocol (ARP) was not started. Compare the displayed events to the defined entities of Figure 1-1. The events displayed create the general transport and the LAN NI entities.

Example 4

```
:NETCONTROL START;NET=LAN1
** NETXPORT Control Process; Transport start
- Loc: 50; Class: 4; Parm= %000027; PIN: 0
** NETXPORT TCP SIP; General protocol start
- Loc: 10; Class: 4; Parm= %000000; Port ID: %000200 %005016
** NETXPORT PXP SIP; General protocol start
- Loc: 5; Class: 4; Parm= %000000; Port ID: %000200 %005064
** NETXPORT IP Update; General protocol start
- Loc: 3; Class: 4; Parm= %000000; Port ID: %000200 %005132
** NETXPORT LAN NI; Network interface start
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000211 %001742
** NETXPORT IP; Protocol start
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000211 %002010
** NETXPORT Probe; Protocol start
- Loc: 35; Class: 4; Parm= %000000; Port ID: %000211 %002124
** NETXPORT Probe; Probe request
- Loc: 38; Class: 5; Parm= %000000; Port ID: %000211 %002124
```

Example 5 shows the initiation events for the combination of the START function and the Loopback network interface. For this example, the Loopback NI is configured as LOOP and the general transport is active.

Example 5

```
:NETCONTROL START;NET=LOOP
** NETXPORT Loopback NI; Network interface start
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000232 %000600
** NETXPORT IP; Protocol start
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000232 %000646
```

Notice that only the Loopback NI and its associated protocol, Internet Protocol (IP), are started; there was a previously issued NETCONTROL START command. The general transport is already active. Compare the displayed events to the defined entities of Figure 1-1. The events displayed create the Loopback NI entity.

Starting the IEEE 802.3 LAN NI, configured as LAN1, when the general transport is already active, gives you the following:

Example 6

```
:NETCONTROL NET=LAN1;START
** NETXPORT LAN NI; Network interface start
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000211 %001742
** NETXPORT IP; Protocol start
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000211 %002010
** NETXPORT Probe; Protocol start
- Loc: 35; Class: 4; Parm= %000000; Port ID: %000211 %002124
** NETXPORT Probe; Probe request
- Loc: 38; Class: 5; Parm= %000000; Port ID: %000211 %002124
```

Notice that only the LAN NI and its associated protocols, Internet Protocol (IP) and Probe, are started. The general transport is already active. Ethernet is not enabled since the address resolution protocol (ARP) was not started. Compare the displayed events to the defined entities of Figure 1-1. The events displayed create the LAN NI entity.

NETCONTROL STATUS

Displays status information, including the actual configuration for the entity specified.

Syntax

```
NETCONTROL ⎡NI=niName[;PROT=niProt];⎤ STATUS[=ALL]
           ⎣PROT=gProt;             ⎦
```

Parameters

NI=*niName*

Specifies that a network interface is the pertinent entity for each specified function to act on, unless modified by the PROT keyword. Enter any valid network interface name, as configured with NMMGR.

PROT={*niProt* / *gProt*}

Specifies that a protocol is the pertinent entity for each specified function to act on. Enter the name of the protocol, as follows:

niProt — The name of a network interface protocol; must be one of PROBE, IP, ARP or NM for the LAN NI, mmst be IP only for Loopback NI, must be one of X25 or IP for the X.25 NI, must be one of DIAL, IP, or NM for the Router and Gateway Half Link NIs. For a function to operate on a network interface protocol, the name of the network interface must also be specified, using the syntax NI=*niName*;PROT=*niProt*.

gprot — The name of a general protocol; must be one of TCP, PXP, or IPU.

STATUS[=ALL]

Displays status information. The general transport must be operational, or START must be issued in the same command line with STATUS, for this function to take affect. This function, when used in combination with the network interface and protocol keywords (NI and PROT) allows you to selectively display status information for a particular protocol or network interface. If you do not specify, the default is to display the CP status. If qualified with the ALL keyword, additional information used for diagnostic purposes is displayed.

Discussion

The STATUS parameter of the NETCONTROL command gives information such as starting time, configuration file, etc.

Example

```
:NETCONTROL STATUS

GENERAL TRANSPORT STATUS : WED, DEC 24, 1986, 10:38 AM

TRANSPORT STARTED        : WED, DEC 24, 1986, 12:43 AM

FLAGS                    : %021000
MAX NETWORK INTERFACES   : 8
PATH DESCRIPTORS         : INIT 0 MAX 100
MAX NODE NAMES           : 35
MAX INBOUND DESTINATIONS: 35

HOME NETWORK             :
CONFIGURATION FILE       : NSCONF.NET.SYS
TRACE MASK               :

NODE NAME                : TIGGER.DCL.IND
```

```
:NETCONTROL NI=LAN1;STATUS


NETWORK INTERFACE REPORT : WED, DEC 24, 1986, 10:45 AM

NETWORK INTERFACE STARTED : WED, DEC 24, 1986, 12:44 AM
FLAGS                     : %000000
NI PROTOCOLS              : CURRENT 2 MAXIMUM 2
MAPPING TABLE SIZE        : 100
OUTBOUND BUFFERS          : SIZE 1500 NUMBER 240
INBOUND BUFFERS           : SIZE 1564 NUMBER 240
NETWORK INTERFACE TYPE    : LAN
NAME                      : LAN1
TRACE MASK                :
IDLE DEVICE TIMEOUT       : 0

DEVICE INFORMATION        :

DEVICE                    : LAN504 (# 0)
DEVICE TYPE               : LAN
LDEV                      : 504
LINK BUFFER SIZE          : 1564
IDLE DEVICE TIMEOUT       : 0
```

NETCONTROL STATUS

```
:NETCONTROL NI=LAN1;PROT=IP;STATUS

NETWORK INTERFACE PROTOCOL STATUS : WED, DEC 24, 1986, 10:51 AM

PROTOCOL STARTED : WED, DEC 24, 1986, 12:44 AM

PROTOCOL NAME    : IP

PROTOCOL ID      : %002400
PROTOCOL FLAGS   : %000000
TRACE MASK       :

NETWORK NAME     : LAN1
ADDRESS          : C 192.006.020 002
```

Many of the items displayed are determined at configuration time. In this example, the NETCONTROL START command is used to initialize the network. Since the CONF keyword is not specified at initiation, the Network Transport uses the configuration file named NSCONF.NET.SYS. You can also check the status for each network interface and protocol, as shown.

NETCONTROL STOP

Terminates the Network Transport and the network interfaces.

Syntax

```
NETCONTROL [GATE=gateName;] STOP
           [NET=niName;   ]
```

Parameters

GATE=*gateName* — Specifies the name of a gateway half to be terminated by STOP. Enter any valid gateway half, as configured with NMMGR. All configured protocols and links associated with the specified gateway half will be terminated.

NET=*niName* — Specifies the network interface to be terminated by STOP. Enter any valid network interface name, as configured with NMMGR. All configured protocols for the specified network interface will be terminated.

STOP — This function can be used to terminate all entities of the Network Transport, or to selectively terminate the network interface specified with the NET keyword, or the gateway half specified with the GATE keyword. Refer to Figure 1-1 for more information.

Discussion

If STOP is issued without the NET keyword, all entities of the Network Transport are terminated. The order of termination is any and all active network interface entities, followed by the general protocols and then the control process. If STOP is combined with the keyword NET, only the specified network interface is terminated. If STOP is combined with the keyword GATE, only the specified gateway half is terminated.

The first example shows a typical node termination sequence:

Example 1

```
:NSCONTROL STOP
:NETCONTROL STOP
```

In Example 1, the node has an NS3000/V Link. NETCONTROL STOP terminates all active entities of the Network Transport. The NSCONTROL command prevents users or programs from accessing any network services. Refer to the NSCONTROL command page in this section for a complete description of NSCONTROL.

The first example provides an overview of terminating a node, showing where NETCONTROL fits into the process. The next examples examine the STOP function and how it affects the entities defined for termination. As will be shown in the examples, the keywords included with the STOP function determine which entities are affected. To understand this relationship, it is helpful to see the events that occur when the Network Transport is terminated. These examples show logging messages for subsystem ID SUB0003, class CLAS0004, logged to the console (the NMMGR default).

Example 2

```
:NETCONTROL STOP
** NETXPORT IP; Protocol stop
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000211 %002010
** NETXPORT Probe; Protocol stop
- Loc: 35; Class: 4; Parm= %000000; Port ID: %000211 %002124
** NETXPORT LAN NI; Network interface stop
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000211 %001742
** NETXPORT IP; Protocol stop
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000216 %000600
** NETXPORT Loopback NI; Network interface stop
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000216 %000646
** NETXPORT PXP SIP; General protocol stop
- Loc: 5; Class: 4; Parm= %000000; Port ID: %000200 %005064
** NETXPORT IP Update; General protocol stop
- Loc: 3; Class: 4; Parm= %000000; Port ID: %000200 %005132
** NETXPORT TCP SIP; General protocol stop
- Loc: 10; Class: 4; Parm= %000000; Port ID: %000200 %005016
** NETXPORT Control Process; Transport stop
- Loc: 50; Class: 4; Parm= %000027; PIN: 0
```

Example 2 shows the events associated with the STOP function at termination. The general transport and both network interfaces were active when the command was issued. Notice that these events are basically to terminate the general transport and both NI entities. The control process closes the configuration file before it terminates. As shown, the STOP function always terminates the network interfaces first. Once all active network interfaces are terminated, the general transport is terminated.

The next example shows what happens if the general transport and both network interfaces are active, and the user specifies the Loopback NI. Notice that the STOP function acts only on the Loopback NI entity. The general transport is still active.

Example 3

```
:NETCONTROL NET=LOOP;STOP
** NETXPORT IP; Protocol stop
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000216 %000600
** NETXPORT Loopback NI; Network interface stop
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000216 %000646
```

In Example 4, only the general transport and the IEEE 802.3 LAN NI is active. The STOP function terminates the LAN NI entity. The general transport is still active. Note that Ethernet is not enabled here since the address resolution protocol was not active.

Example 4

```
:NETCONTROL STOP;NET=LAN1
** NETXPORT IP; Protocol stop
- Loc: 51; Class: 4; Parm= %000000; Port ID: %000211 %002010
** NETXPORT Probe; Protocol stop
- Loc: 35; Class: 4; Parm= %000000; Port ID: %000211 %002124
** NETXPORT LAN NI; Network interface stop
- Loc: 78; Class: 4; Parm= %000000; Port ID: %000211 %001742
```

NETCONTROL TRACE

Enables or disables message tracing for the specified entity.

Syntax

```
            {NI=niName [;PROT=niProt]}
NETCONTROL  {PROT=gprot              };{TRACEON=type [,options]}
            {GATE=gateName           } {TRACEOFF               }
            {NET=niName              }
```

where the parameter *option* has the following options:

```
TRACEON=type[[,DISC][,fileName][,recSize][,fileSize]]
```

Parameters

NI=*niName*

Specifies that a network interface is the pertinent entity for each specified function to act on, unless modified by PROT. Enter any valid network interface name, as configured with NMMGR.

PROT={*niProt* / *gProt*}

Specifies that a protocol is the pertinent entity for each specified function to act on. Enter the name of the protocol, as follows:

niProt — The name of a network interface protocol; must be one of PROBE, IP, ARP or NM for the LAN NI, must be IP only for Loopback NI, must be one of X25 or IP for the X.25 NI, must be one of DIAL, IP, or NM for the Router and Gateway Half Link NIs. For a function to operate on a network interface protocol, the name of the network interface must also be specified, using the syntax NI=*niName*;PROT=*niProt*.

gprot — The name of a general transport protocol; must be one of TCP, PXP, or IPU.

GATE=*gateName*

In place of the NET entity, specifies the name of a configured gateway half.

NET=*niName*

Specifies the network interface. Enter any valid network interface name, as configured with NMMGR.

TRACEON

Activates tracing. The general transport must be operational, or START must be issued in the same command line with TRACEON, for this function to take affect. This function will affect only the entity specified by the network interface or protocol keywords (NI and PROT). This allows you to selectively activate tracing for a particular protocol or network interface. If you do not specify, the default is to trace the Control Process (CP). The parameters on the TRACEON function allow you to specify the following:

=*type*

(Required parameter) Specifies type of trace for the designated network interface or protocol. The *type* field is made up of one or more of the following key letters, concatenated, and entered in any order:

M - Trace Messages
H - Trace Header Data
D - Trace Data
S - Trace State Transitions
B - Trace Buffers
N - Trace Nodal Management Events
The most useful information consists of MHD.

DISC

(Optional parameter) Trace information is to be written to a disc file, specified by *fileName*. You use NMDUMP to format the file, as described in Section 3.

fileName

(Optional parameter) File name to which trace information will be written. The default filename is NMTC*xxxxx*.PUB.SYS where *xxxxx* is the number of the most recent file, incremented in numerical order.

recSize

(Optional parameter) Logical record size of file for which trace information is to be written.

Default: 128W

fileSize

(Optional parameter) Maximum number of logical records of the trace file.

Default: 1024 records

TRACEOFF

Deactivates tracing. The general transport must be active and a previous TRACEON issued on the entity specified for this function to take affect. This function affects the entity specified with the network interface and protocol keywords (NI and PROT). This allows you to selectively deactivate tracing for a particular protocol or network interface. If you do not specify, the default is to deactivate the Control Process.

NETCONTROL UPDATE

Dynamically updates selected network transport parameters and configuration information.

Syntax

```
NETCONTROL {GATE=gateName};UPDATE={configArea}[;function]
           {NET=niName  }
```

Parameters

GATE=*gateName*

In place of the NET entity, specifies the name of a configured gateway half.

NET=*niName*

Specifies the network interface. Enter any valid network interface name, as configured with NMMGR.

UPDATE= {*configArea*}

Specifies which configuration areas will be dynamically updated. If ALL is specified, or if no area is specified, all configuration areas will be updated if changes have been made.

Configuration area (*configArea*) is one of the following:

INTERNET

The network will be updated to reflect the current information in the internet routing tables, which corresponds to the configuration subtree NETXPORT.NI.*niName*.INTERNET. These tables contain the information describing the gateways for all directly connected networks and gateway halves, as well as all networks the gateways can reach.

MAPPING

The information configured in the Router Reachable Nodes Screen will be updated. Information for Router NIs will be overlayed based on matching IP/Device mapping records. This provides the ability to change routes as well as add new reachable nodes.

NETDIR

The information configured in the Network Directory (file NSDIR) for the specified network interface will be loaded into the appropriate mapping table. Refer to Section 14 for details of how to modify nodes/paths in the Network Directory.

X25

The information configured in the X.25 Path Table for SVCs and/or the information configured in the X.25 Local User Group Table will be updated. This command function provides the ability to add new SVC destinations and to add a node to the L.U.G. (Local User Group) table.

ALL — Any of the possible areas will be updated if changes have been made with NMMGR.

function — You may want to issue one or more of the NETCONTROL functions on the same command line with UPDATE. Only one of each type of function is allowed on a command line. STOP is permissible, but functionless, since it deallocates the tables associated with the network. Also, TRACEON and TRACEOFF, MONON and MONOFF cannot both be issued on the same command line. Moreover, be aware that the functions STATUS, TRACE, MON, act only on the IP protocol when the network interface is specified with the NET keyword. However, the PROT keyword can be used to specify the general protocol for the function to act on. If a network interface or general protocol is not specified, the functions act on the Control Process. For these reasons, although other functions can be issued on the same command line with START, it is not recommended. Refer to the command pages describing NETCONTROL MON, NETCONTROL STATUS, NETCONTROL START, NETCONTROL STOP, NETCONTROL TRACE and NETCONTROL VERSION for more information.

Discussion

The UPDATE parameter to the NETCONTROL command dynamically updates the network with configuration changes already made through NMMGR. In this way, configuration changes become active without having to first take down and then restart the network.

UPDATE's four functions - INTERNET, NETDIR, X25, and MAPPING - localize which subgroup of configuration data will be updated. UPDATE can be entered at any time, and with any permissible combination of the other NETCONTROL functions (START, STOP, STATUS, TRACEON, TRACEOFF, MONON, MONOFF, DEBUG, VERSION).

Certain UPDATE functions are valid only with particular types of network interfaces. The following table shows the correspondence between entities, interface types, and UPDATE functions:

VALID UPDATE FUNCTIONS

Entity	Interface Type	Function
NET	LAN802.3	INTERNET or NETDIR
NET	ROUTER	INTERNET or MAPPING
GATE	GATEHALF	INTERNET
NET	X.25	INTERNET, NETDIR or X25

NOTE

Dynamic updating can circumvent some of the normal safeguards against exceeding mapping and routing table sizes. If this happens, relevant error messages will be logged to the console (i.e, Maximum Configuration Space Exceeded). For INTERNET, it is possible for the IP Routing Tables to accept some of the changes and discard the remainder because of a lack of table space. In such a case, it is advisable to either restore the configuration to its state before the latest changes, or bring down the transport and increase the Max Networks and Max Gateways fields of the IPU configuration to permit the changes. For MAPPING, overflowing the Mapping and Routing Tables will prevent any further updates from having any effect, and error messages will again be logged to the console. This condition can only be removed by shutting down the network, increasing the relevant parameters (i.e., the maximum number of nodes) and then restarting it.

NETCONTROL VERSION

Displays the version numbers for the Network Transport software modules.

Syntax

```
NETCONTROL VERSION[=MOD]
```

Parameters

VERSION[=MOD]

Displays the overall version of the Network Transport. If qualified with the MOD keyword, displays the version of each of the software modules of the Network Transport and the overall version. The general transport must be operational, or START must be issued in the same command line with VERSION, for this function to take effect.

Discussion

At times, you may want to check on various aspects of the network. The VERSION function of the NETCONTROL command allows you to check the version numbers of the Network Transport modules to ensure that they are compatible and up-to-date. The VERSION function is a utility function, and can be executed with any valid NETCONTROL command. It is not affected by specifying particular entities with the network interface and protocol keywords (NET, NI and PROT). Refer to Figure 1-1 for more information on NETCONTROL entities.

Example 1

```
:NETCONTROL VERSION

Network Transport Overall Version : A.00.00
```

To look at the version numbers of the individual modules, you specify the MOD keyword. You see the display shown in Example 2. This example is a sample only, and the modules and version numbers may not be the same as on your installed system.

Example 2

```
:NETCONTROL VERSION=MOD

Network Transport 32343A module versions:

Program file:  NETCP.NET.SYS              Version:  A0106057
Program file:  NETSERVE.NET.SYS           Version:  A0106000
Program file:  SOCKREG.NET.SYS            Version:  A0106003
Catalog file:  NETMSG.NET.SYS             Version:  A0106009
Catalog file:  SOCKCAT.NET.SYS            Version:  A0106000
Program file:  STUD.NET.SYS               Version:  A0106006
Program file:  NODESTAT.NET.SYS           Version:  A0106000
SL procedure:  NET'SM4'VERS               Version:  A0106000
SL procedure:  NET'UI'VERS                Version:  A0106009
SL procedure:  NET'SL'VERS                Version:  A0106019
SL procedure:  NET'NI'VERS                Version:  A0106030
SL procedure:  NET'PROBE'VERS             Version:  A0106012
SL procedure:  NET'ARP'VERS               Version:  A0106000
SL procedure:  NET'DIAL'VERS              Version:  A0106016
SL procedure:  NET'TCP0'VERS              Version:  A0106021
SL procedure:  NET'TCP1'VERS              Version:  A0106021
SL procedure:  NET'PXP0'VERS              Version:  A0106010
SL procedure:  NET'PXP1'VERS              Version:  A0106010
SL procedure:  NET'IP'VERS                Version:  A0106038
SL procedure:  NET'IPU'VERS               Version:  A0106005
SL procedure:  NET'PD'VERS                Version:  A0106015
SL procedure:  NET'X25'VERS               Version:  A0106006
SL procedure:  SOCKIOVERS                 Version:  A0106063
SL procedure:  SOCKACCESSVERS             Version:  A0106063
SL procedure:  SOCKMISC1VERS              Version:  A0106063
SL procedure:  SUBSYS3FMTVERS             Version:  A0106001
SL procedure:  SUBSYS5FMTVERS             Version:  A0106001
SL procedure:  NET'VCSWS'VERS             Version:  A0106000
SL procedure:  NET'PAP'VERS               Version:  A0106010

Network Transport 32343A overall version = A.01.06
```

NSCONTROL

Initiates, terminates, and controls the Network Services subsystem of NS3000/V.

Syntax

```
NSCONTROL function[;function]...
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	YES
	Programmatically?	YES
Breakable?		NO
Capabilities?		NM

Parameters

function

One or more of the functions defined for NSCONTROL. These are:

START[=*services*] Enables the Network Services for use over NS3000/V links. The first START creates the Network Services control process, called DSDAD. The optional service list (*services*) allows you to select which of the services are enabled for local or remote use. Refer to the NSCONTROL START command for more information and a list of the core services available.

STOP[=*services*] Terminates the Network Services. The optional service list (*services*) allows you to select which services are terminated for local or remote use. When all services are stopped, the DSDAD process terminates. Refer to the NSCONTROL STOP command for more information.

ABORT Immediately terminates all Network Services servers and services -- only used in special or abnormal situations. Refer to the NSCONTROL ABORT command for more information.

AUTOLOGON Enables or disables the automatic logon feature available with the NFT, RFA, and RPM services. With autologon disabled, remote users must first create a remote session with the REMOTE HELLO command prior to using these services. Refer to the NSCONTROL AUTOLOGON command for more information.

LOADKEYS — Loads the Network Services command keywords from the ASCAT.NET.SYS message catalog -- only used if the catalog is modified, such as for localization. Refer to the NSCONTROL LOADKEYS command for more information.

LOG — Enables or disables NMS logging of Network Services detailed events, configured as SUB0006, CLAS0004 in the NMCONFIG.PUB.SYS configuration file. Detailed events are used only for troubleshooting and are normally disabled. Refer to the NSCONTROL LOG command for more information.

SERVER — Controls the minimum and maximum number of servers available for the Network Services. A server is a system program that is run for each user; the server types are identified by their program names, DSSERVER, NFT, LOOPBACK, NSSTATUS and PADSVR. The correct minimum number of servers can speed up the processing of service requests, such as DSCOPY and REMOTE HELLO; specifying the maximum number of servers puts a limit on the amount of system resources used for the Network Services. Refer to the NSCONTROL SERVER command for more information.

STATUS — Displays a summary of information about the users, services, and servers of the Network Services. Refer to the NSCONTROL STATUS command for more information.

TRACE — Enables or disables tracing for PAD services. Refer to the NSCONTROL TRACE command for more information.

VERSION — Displays the version identification number for each software module and the overall subsystem version number of the Network Services. Refer to the NSCONTROL VERSION command for more information.

The NSCONTROL functions can be combined on the same command line. In fact, multiples of the same function can be combined on the same command line.

Discussion

NS3000/V Network Services is composed of user services, each ot which performs a specific task. These services are VT, PAD, Reverse VT, NFT, RPM, RFA, PTOP, RDBA, LOOPBACK and NSSTAT. Refer to the *NS3000/V User/Programmer Reference Manual* for details on Network Services.

NOTE

DS Services provides the same services except RPM and Reverse VT and is used over DS Compatible Links. They are started with the DSCONTROL command, described earlier in this chapter.

To function, Network Services require Network Interprocess Communication (NetIPC), the user interface included with NS3000/V links. NetIPC is used extensively by the Network Services when processing service requests and is available for use in customer applications. It is not a service in the same sense as VT or RFA since it consists of a set of intrinsics and associated code in the System SL. NetIPC does not need to be initiated; it is always available because its intrinsics can always be called. NetIPC intrinsics are described in the *NetIPC3000/V Programmer's Reference Manual.*

The NETCONTROL START command must be issued before NSCONTROL START. This is because the NETCONTROL command controls the Network Transport subsystem, which must be initiated before the Network Services or any NetIPC application can successfully execute. NetIPC depends on the Network Transport to identify sockets and exchange messages. Refer to the NETCONTROL START command, also described in this section.

NSCONTROL ABORT

Immediately terminates all the servers and services of the Network Services.

Syntax

```
NSCONTROL ABORT
```

Parameters

ABORT

Immediately terminates all servers and services -- only used in special or abnormal situations.

Discussion

There are two NSCONTROL functions that can be used to terminate the Network Services. The STOP function is the normal way to terminate the Network Services. The ABORT function should *only* be used in special or abnormal situations. It immediately terminates all the services and all the server processes. Anyone using a service finds their task (REMOTE, DSCOPY, etc) immediately terminated. In contrast, the STOP function allows existing users to continue using the services until they finish their tasks and prevents any new users from using the services. Refer to the NSCONTROL STOP command.

The two functions can be used in sequence to be sure that all Network Services are terminated. Special situations where this may be appropriate include when the system is being prepared for software installation, or when the system needs to be taken down for maintenance. Abnormal situations can occur when an application has been incorrectly implemented. If the session cannot be terminated by any other method, use NSCONTROL ABORT to terminate all Network Services. This will clear any problems. The sequence to use prior to issuing the NSCONTROL ABORT command is shown in Example 1; an abnormal situation is described in Example 2.

Example 1

Issue a message to all users to stop using the Network Services. Use whatever method is appropriate for your installation. Then use the following sequence of commands to terminate the Network Services:

`:NSCONTROL STOP`

Prevents any users or programs from accessing Network Services.

`:NSCONTROL STATUS=USERS`

Checks that all users of the Network Services are finished.

```
NO CURRENT NETWORK SERVICE USERS
```

`:NSCONTROL ABORT`

Terminates all users, services, and server processes.

Example 2

If a remote session has been terminated by the user but still shows as active on a SHOWJOB display, use ABORTJOB to terminate the session.

In the unlikely event that ABORTJOB doesn't work, use NSCONTROL ABORT. Be sure to follow the sequence shown in Example 1 before issuing the NSCONTROL ABORT command.

NSCONTROL AUTOLOGON

Enables or disables the autologon feature of certain NS3000/V services.

Syntax

```
                      ┌┌ON ┐┌,ALL                          ┐┐
NSCONTROL  AUTOLOGON= ││   ││                              ││...[;function]...
                      └└OFF┘└,service[,service] ...        ┘┘
```

Parameters

AUTOLOGON — Disables or enables autologon for certain NS3000/V services.

- ON — Enables autologon for an NS3000/V service.
- OFF — Disables autologon for an NS3000/V service.
- ALL — Alters the autologon state for the NFT, NFTL, RFA, RFAL, RPM, and RPML services.

The *services* which allow autologon are:

- NFT — Changes autologon capability for the NFT service.
- RFA — Changes autologon capability for the RFA service.
- RPM — Changes autologon capability for the RPM service.

Defaults: ON and ALL

function — You may want to issue one or more of the NSCONTROL functions on the same command line with AUTOLOGON. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

NSCONTROL AUTOLOGON allows the user the ability to disable and re-enable autologon for the NS3000/V services supporting this feature. Autologon is enabled at NS3000/V startup. NSCONTROL AUTOLOGON must be executed after the NSCONTROL START command. When the NS3000/V services are stopped, the autologon option resets to the default.

Disabling autologon gives the system more security by allowing logon UDC security to work correctly. Remote users must first establish a remote session with the REMOTE HELLO command before using an NS3000/V service. With autologon enabled, a remote user could bypass a security UDC and complete an NS3000/V service before the UDC would abort it.

NSCONTROL AUTOLOGON

Disabling autologon on a node requires all remote users to first establish a remote session using the REMOTE HELLO command before using the service. Incoming requests attempting to use the autologon feature will fail, since a remote session cannot be established automatically.

It is recommended that users with security logon UDCs disable autologon for all services to preserve the security of the system from remote users.

NSCONTROL LOADKEYS

Loads the Network Services command keywords.

Syntax

```
NSCONTROL LOADKEYS[;function]...
```

Parameters

LOADKEYS

Loads the Network Services command keywords from the ASCAT.NET.SYS catalog -- only used if the catalog is modified, such as for localization purposes.

function

You may want to issue one or more of the NSCONTROL functions on the same command line with LOADKEYS. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

The LOADKEYS functions is *only* used to switch between pre-prepared ASCAT.NET.SYS catalogs. When the node is initiated, the Network Services command keywords are automatically loaded into an extra data segment from the ASCAT.NET.SYS catalog. This is done to ensure fast access to the command keywords during command parsing. However, it might be useful to have commands in the appropriate language of the installation. If so, the LOADKEYS function is used to reload the alternate catalog into the extra data segment without having to coolstart the system. Make a copy and a listing of the catalog before switching catalogs.

NSCONTROL LOADKEYS

Example

`:HELLO MANAGER.SYS,NET`	Logon to the NET group in the SYS account
`:RENAME ASCAT,ASCATOLD`	Rename the old catalog.
`:RENAME ASCATINEW, ASCAT`	Subtitute the new catalog for the old
`:NSCONTROL LOADKEYS`	Reloads the catalog.

NOTE

If an NSCONTROL command reports CIERR 5077, follow the above example to restore the old ASCAT catalog and contact your HP representative for assistance.

NSCONTROL LOG

Enables or disables detailed event logging for the Network Services.

Syntax

```
                         ⎡⎧,ALL      ⎫          ⎤
                         ⎢⎪,RPM      ⎪⎡,LOW ⎤   ⎥
NSCONTROL LOG=⎧ON ⎫      ⎢⎨,ENV      ⎬⎣,HIGH⎦   ⎥...[;function]...
              ⎩OFF⎭      ⎢⎪,DSDAD    ⎪          ⎥
                         ⎢⎪,DSSERVER ⎪          ⎥
                         ⎣⎩,PADSVR   ⎭          ⎦
```

Parameters

LOG

Enables or disables NMS logging of Network Services detailed events, configured as SUB0006, CLAS0004 in the NMCONFIG.PUB.SYS configuration file. Detailed events are only used for troubleshooting and normally disabled.

ON — Enables detailed logging of the specified Network Service modules.

OFF — Disables detailed logging of the specified Network Service modules.

Defaults: ALL and LOW

For each Network Services software module, two levels of event logging are provided. These are HIGH, which logs all events, and LOW, the default, which logs a subset of the events, as specified below.

ALL	LOW	Logs LOW events for all modules.
	HIGH	Logs HIGH events for all modules.
RPM	LOW	Logs RPMCREATE and RPMKILL requests.
	HIGH	Same as LOW.
ENV	LOW	Logs environment information from DSLINE and REMOTE HELLO commands.
	HIGH	Same as LOW, plus environment table locking and use counts.
DSDAD	LOW	Logs creation and deletion of sockets, ports, and server processes.
	HIGH	Same as LOW, plus all received service requests

		and internal messages between DSDAD and server processes.
DSSERVER	LOW	Logs internal initialization messages between DSDAD and user processes.
	HIGH	Same as LOW, plus all received messages from other processes.
PADSVR	LOW	Logs internal errors, resource errors, IPC errors and detailed events.
	HIGH	Same as LOW.

function You may want to issue one or more of the NSCONTROL functions (or multiple instances of the LOG function) on the same command line with LOG. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

One of the log classes defined for the Network Services is detailed event logging, which records normal Network Services events. Detailed event logging is enabled and disabled *only* with the LOG function of the NSCONTROL command, unlike most log classes, which are enabled at system startup as part of the NMS initialization. During normal operation, the Network Services detailed event logging is disabled in order to avoid the overhead of frequent logging. Typically, detailed event logging is only enabled to investigate a specific action or series of events if required for troubleshooting.

When detailed event logging is enabled with the LOG function, whether the log messages generated by detailed logging are directed to the console or to a log file is determined by the configuration of the CLAS0004 class of the SUB0006 subsystem id of the NMMGR Logging Configuration, as described in Section 7. The log file is the recommended destination for detailed logging. Logging detailed events to the system console is not recommended, since the log messages tend to clutter the console screen. This configuration, as is true for all Network Services logging classes, must be specified in the configuration file NMCONFIG.PUB.SYS.

The log messages for all log classes, including detailed logging, are directed to the log file created by NMS, NMLG*xxxx*. Once a log file is filled, NMS creates a new log file. Because of the large number of log messages generated for detailed logging (when enabled), the log file fills quickly. In order to examine the correct log file, use the SWITCHNMLOG command to free the log file immediately following the action to be examined. The log file can then be formatted for examination using the NMS Log/Trace Formatter, NMDUMP, described in Section 3 of this volume.

Example

```
    :NSCONTROL LOG=ON,ENV,LOW;LOG=ON,DSDAD,HIGH
```

Logs the environment information from DSLINE and REMOTE HELLO commands, and the service requests received by the DSDAD process. This might be used if the Network Manager wants to monitor usage of the Network Services. The destination for CLAS0004 of SUB0006 in the NMMGR Logging Configuration (described in Section 14 of Volume I) should be the NM log file, not the system console.

NSCONTROL SERVER

Alters the characteristics of the Network Services server processes.

Syntax

```
                  {serverName}
NSCONTROL SERVER= {          }[,minservers][,maxservers][;function]...
                  {ALL       }
```

Parameters

SERVER

Dynamically alters the minimum or maximum number of servers.

serverName

The servers that control the network services are:

NFT — The specified options apply to the server that controls NFT.

DSSERVER — The specified options apply to the server that controls VT, Reverse VT, RFA, RDBA, PTOP, and RPM.

NSSTATUS — The specified options apply to the server that controls NSSTAT (and NSTATL) services.

LOOPBACK — The specified options apply to the server used by the LOOPBACK services.

PADSVR — The specified options apply to the server used by the PAD and PADL services.

ALL — The specified options apply to all servers (NFT, DSSERVER, LOOPBACK, NSSTATUS and PADSVR).

Default: ALL

minservers — The minimum number of servers which will be in existence at all times. This includes active and reserved servers. These servers are created immediately, then kept in reserve until a service request is received. Once the service request is completed, the server is returned to reserve status. If necessary, additional servers are created immediately to fit the new minimum specified by the NSCONTROL SERVER command.

Default: 0

The total number of reserved servers, NFT, PADSVR and DSSERVER, is limited to half of the number of processes configured for the PCB table entries through SYSDUMP. If you specify a number larger than this limit, you receive an error message that states how many reserved servers are allowed.

maxservers The maximum number of servers. If necessary, reserved servers will be terminated to fit the new maximum. However, a server that is in use is not terminated until returned to the reserved server pool.

Default: 32,767

function You may want to issue one or more of the NSCONTROL functions (or multiple instances of the SERVER function) on the same command line with SERVER. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

The number of server processes is controlled with the SERVER function. The maximum number of servers limits how many processes of each server type can be in existence at any time. If the servers are at the maximum limit and a new service request (such as a DSCOPY or REMOTE HELLO) is received the request will be rejected. By setting a maximum limit, the node manager controls the amount of process resources available for NS3000/V.

Because the creation and initialization of a server takes time, using reserved servers decreases the set up time for a service request. A reserved server is created ahead of time and is held in reserve until a service request is received. The minimum number of servers controls the number of reserved processes for each type of server. The number set for the minimum does not limit the number of concurrent users of the Network Services. If there are more concurrent users than the minimum number of servers specified, new users can use the Network Services, but there is a delay while the additional servers are created.

There is no simple formula for determining how many precreated servers to specify. Since each precreated server consumes one set of process resources, one PCB table entry and one stack data segment (approximately 30K bytes of virtual memory), the number chosen must be a tradeoff between using system resources and allowing fast service response. The network manager needs to estimate, on the average, the number of concurrent users of each type of server. This number is used for the minimum number of servers of each type. Since the DSSERVER process is used by several services, and some of these services (VT in particular) are active for a long time, it makes sense to preallocate a larger number of DSSERVER servers than NFT, LOOPBACK, NSSTATUS or PADSVR servers.

An alternative to precreating server processes that gives some improvement in performance, yet is less expensive in system resources such as virtual memory and DST table entries, is to allocate the program files NFT.NET.SYS and DSSERVER.NET.SYS, LOOPBACK.NET.SYS, PADSVR.NET.SYS and NSSTATUS.NET.SYS. Allocating the program file DSCOPY.PUB.SYS as well provides an improvement in performance for the DS Services used over the DS Compatible Links. This alternative is most advantageous for DSSERVER, where the allocation of the program file is a significant portion of the set up time. The NFT server must read keywords and messages from the NFTCAT2 catalog as well as allocate the program file when the server is created, so the performance gain is not as great as for DSSERVER.

Creating reserved servers or using the allocation alternative means that the program file is in use, just as when a program is run. However, no * is printed next to the filename in the LISTF display -- this is also true for a program in use with the RUN command. Since the program file is in use, it cannot be purged, replaced, or backed up. Before any software installation, when the program files are replaced or backed up, check that the program files are not allocated and that there are no reserved servers.

Example 1

```
:NSCONTROL SERVER=DSSERVER,5,10
```

Sets the minimum number of DSSERVER processes to five and the maximum to ten. Five reserved DSSERVER processes are created immediately -- available for future service requests. The minimum number of servers, which includes both reserved and active servers, is restricted to five. When an active server is returned to the reserved pool, if there are already five reserved servers, the extra server is terminated. The maximum limit means that if there are ten DSSERVER processes active, any new service requests will be rejected.

Example 2

```
:NSCONTROL SERVER=ALL,10
```

There are 10 server processes created for NFT, 10 for DSSERVER, 10 for PADSVR, 10 for LOOPBACK, and 10 for NSSTATUS when this command is executed. Later, when users issue service requests (such as DSCOPY and REMOTE HELLO) they do not have to wait for the servers to be created. The maximum number of servers is the default, 32,767.

Example 3

```
:ALLOCATE DSSERVER.NET.SYS
:NSCONTROL SERVER=NFT,2,10;SERVER=DSSERVER,,10
:NSCONTROL STATUS=SERVERS

SERVER      MIN    MAX   FEATURES

PADSVR       0   32767    TRACE
NSSTATUS     0   32767
LOOPBACK     0   32767
HDSPNS       0   32767
PDSERVER     0   32767
NFT          2      10
DSSERVER     0      10

JOB/SESSION   PIN   STATUS     SERVICE

#S49           76   RESERVED   NFT
#S51          158   RESERVED   NFT
```

In this example, the Network Manager has chosen to allocate the program file used for the DSSERVER servers and to establish two reserved servers for NFT. To limit the system resources available, the maximum number of servers is set to 10 for both server types. In this way, performance is improved with a minimum amount of system resources used. Notice that the SERVER function can be repeated; multiple instances of NSCONTROL functions are allowed on the same command line. (HDSPNS and PDSERVER are servers used for Resource Sharing Products from Hewlett Packard's Office Systems Divisions.)

NSCONTROL START

Enables the Network Services subsystem of NS3000/V.

Syntax

```
NSCONTROL START[=service [,service]...][,NET=niName [,niName]...]
               [;function]...
```

Parameters

START[=*services*]

Enables the Network Services (VT, Reverse VT, PAD, NFT, RFA, RDBA, PTOP, RPM, LOOPBACK, and NSSTAT). The first START creates the Network Services control process, called DSDAD. The optional service list (*services*) allows you to select which of the services are enabled for local or remote use. The *niName* must be specified in the NET= parameter for each X.25 network on which you want to initiate PAD services.

Defaults: If you omit the optional service list but include the NET= parameter with *niNames*, all the NS services (for local and remote use) will be started, including the PAD and PADL services on the networks specified in the NET= parameter. If you specify only NSCONTROL START, all the NS services except PAD and PADL will be started.

The *services* which allow users on remote nodes to use resources on the local node are:

VT
Allows remote users to log onto the local node using the REMOTE HELLO command.

VTR
Allows remote users to access local terminals using the Reverse VT service.

PAD
Allows remote users to use a public or private PAD to establish a session on the local node.

NFT
Allows remote users to transfer files to or from the local node using the DSCOPY command and intrinsic.

RFA
Allows remote users to access files on the local node, using the RFA and RDBA services.

PTOP
Allows remote users to create and communicate with PTOP slave processes on the local node. The VT service must also be started.

RPM
Allows remote users to create and kill processes on the local node using the Remote Process Management service.

NSSTAT — Allows remote users to use the NSSTATUS intrinsic to retrieve network services information from the local node.

LOOPBACK — Allows remote users to use the loopback diagnostic server on the local node.

The *services* which allow users on the local node to use resources on remote nodes are:

VTL — Allows local users to log onto remote nodes using the REMOTE HELLO command.

VTRL — Allows local users to access terminals on remote nodes using the Reverse VT service.

PADL — Allows local users to programmatically access terminals or printers attached to a remote HP 2334A.

NFTL — Allows local users to transfer files to or from remote nodes using the DSCOPY command and intrinsics.

RFAL — Allows local users to open and access files and databases on remote nodes, using the RFA and RDBA services.

PTOPL — Allows local users to create and communicate with PTOP slave processes on remote nodes. The VTL service must also be started.

RPML — Allows local users to create and kill processes on the local and remote nodes using the RPM service.

NSSTATL — Allows local users to use the NSSTATUS intrinsic to retrieve network services information from the local and remote nodes.

function — You may want to issue one or more of the NSCONTROL functions on the same command line with START. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

You use the service list if you wish to select which services to start, and whether local or remote users are allowed to use the services. To allow remote users to use VT, PAD, NFT, RFA/RDBA, PTOP, NSSTAT, LOOPBACK, RPM, on your local node, you must START the appropriate remote services. Additionally, if you wish to allow local users to use VT, PAD, NFT, RFA/RDBA, PTOP, RPM, and NSSTAT to remote nodes, you must START the appropriate local services.

The NETCONTROL START command must be issued before the NSCONTROL START command. This is because the Network Services subsystem of NS3000/V depends on the Network Transport subsystem. Refer to the NETCONTROL START command for more information.

The NSCONTROL command is used to control the operation of the Network Services subsystem of NS/3000. The DS Services subsystem (and the DS Compatible Links) are controlled with the DSCONTROL command, described in this section, entirely independent of the NSCONTROL command.

If NS3000/V is installed on your node, and a user issues a DSCOPY request when an NSCONTROL START has not been issued, but a DSCONTROL command has been issued, then the DS Compatible version of NFT is invoked and a warning is issued before processing the request. The request may be delayed by this extra processing.

Example 1

```
:NETCONTROL START;NET=LAN1
:NSCONTROL START
```

Example 1 shows the command sequence necessary to start the Network Services. Enter the NETCONTROL START command to initiate the Network Transport before the NSCONTROL START command, as shown in the example. Issuing the NSCONTROL START creates the DSDAD process and starts all the user services.

To successfully initialize a node, the commands must be issued in the order specified. At least one of the required NETCONTROL START commands must be issued first, before the NSCONTROL START command.

Example 2

```
:NETCONTROL START;NET=X25NET1
:NSCONTROL START=VTL,VTRL,PADL,NFTL,RFAL,PTOPL,RPML,NSSTATL,NET=X25NET1
:NSCONTROL STATUS=SERVICES

SERVICE    TYPE      SERVER     DESCRIPTION

PADL       LOCAL     PADSVR     OUTGOING PAD SERVICE
PAD        REMOTE    PADSVR     INCOMING PAD SERVICE
NSSTATL    LOCAL     NSSTATUS   OUTGOING NSSTATUS SERVICE
NSSTAT     REMOTE    NSSTATUS   INCOMING NSSTATUS SERVICE
HDSPNS     REMOTE    HDSPNS
PDS        REMOTE    PDSERVER
LOOPBACK   REMOTE    LOOPBACK   INCOMING LOOPBACK SERVICE
RPML       LOCAL     DSSERVER   OUTGOING REMOTE PROCESS MANAGEMENT
RPM        REMOTE    DSSERVER   INCOMING REMOTE PROCESS MANAGEMENT
PTOPL      LOCAL     DSSERVER   OUTGOING PROGRAM-TO-PROGRAM COMMUNICATION
PTOP       REMOTE    DSSERVER   INCOMING PROGRAM-TO-PROGRAM COMMUNICATION
RFAL       LOCAL     DSSERVER   OUTGOING REMOTE FILE ACCESS
RFA        REMOTE    DSSERVER   INCOMING REMOTE FILE ACCESS
NFTL       LOCAL     NFT        OUTGOING NETWORK FILE TRANSFER
NFT        REMOTE    NFT        INCOMING NETWORK FILE TRANSFER
VTRL       LOCAL     DSSERVER   OUTGOING REVERSE VIRTUAL TERMINAL
VTR        REMOTE    DSSERVER   INCOMING REVERSE VIRTUAL TERMINAL
VTL        LOCAL     DSSERVER   OUTGOING VIRTUAL TERMINAL
VT         REMOTE    DSSERVER   INCOMING VIRTUAL TERMINAL
```

```
SERVICE     STARTED   FEATURES

PADL          YES
PAD           NO
NSSTATL       YES
NSSTAT        NO
HDSPNS        NO
PDS           NO
LOOPBACK      NO
RPML          YES
RPM           NO        AUTOLOGON
PTOPL         YES
PTOP          NO
RFAL          YES
RFA           NO        AUTOLOGON
NFTL          YES
NFT           NO        AUTOLOGON
VTRL          YES
VTR           NO
VTL           YES
VT            NO
```

For security reasons, the network manager of this node has decided to restrict the Network Services to outgoing only. The command shown in Example 2 enables users on the local node to use resources on remote nodes. The reverse is not true. Users on remote nodes are not allowed to logon or use any of the services on the local node. The status display shows all the local services enabled and all the remote services disabled.

Example 3

```
:NSCONTROL START

TRANSPORT NOT INITIALIZED (DSERR 644)
INVALID CONTROL OPTION (CIERR 5062)
```

The Network Transport must be initialized before the NSCONTROL START command is issued. If not, the error messages shown in Example 3 are displayed.

NSCONTROL STATUS

Displays information about the Network Services.

Syntax

```
                   ┌=USERS    ┐
NSCONTROL STATUS   │=SERVICES │ [;function]...
                   │=SERVERS  │
                   └=ALL      ┘
```

Parameters

STATUS

Displays information about the Network Services. Can be used to check if the Network Services were successfully initiated, or to check on the current status using the following parameters:

USERS — Displays the sessions on the node that are associated with the Network Services.

SERVICES — Displays information about the services.

SERVERS — Displays information about the servers.

ALL — Displays information about all three of the above.

The STATUS function can be qualified with one parameter, or with a list of parameters separated by commas. If STATUS is not qualified with any parameters, the default is to display all information about the users, servers and services.

Default: ALL

function

You may want to issue one or more of the NSCONTROL functions (or multiple instances of the SERVER function) on the same command line with SERVER. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

This function displays information on those local sessions that were created by a DSLINE and REMOTE HELLO and on those remote sessions that were created by a REMOTE HELLO. The STATUS display does not list information on either local sessions that are using DSCOPY without a REMOTE HELLO or temporary remote sessions created by NFT RFA, or RPM. It also does not display information on sessions using the DS Services over the DS Compatible Links.

The following examples show the information provided by the STATUS function of the NSCONTROL command.

NSCONTROL STATUS

Example 1

```
:NSCONTROL STATUS=SERVICES

SERVICE    TYPE     SERVER     DESCRIPTION

PADL       LOCAL    PADSVR     OUTGOING PAD SERVICE
PAD        REMOTE   PADSVR     INCOMING PAD SERVICE
NSSTATL    LOCAL    NSSTATUS   OUTGOING NSSTATUS SERVICE
NSSTAT     REMOTE   NSSTATUS   INCOMING NSSTATUS SERVICE
HDSPNS     REMOTE   HDSPNS
PDS        REMOTE   PDSERVER
LOOPBACK   REMOTE   LOOPBACK   INCOMING LOOPBACK SERVICE
RPML       LOCAL    DSSERVER   OUTGOING REMOTE PROCESS MANAGEMENT
RPM        REMOTE   DSSERVER   INCOMING REMOTE PROCESS MANAGEMENT
PTOPL      LOCAL    DSSERVER   OUTGOING PROGRAM-TO-PROGRAM COMMUNICATION
PTOP       REMOTE   DSSERVER   INCOMING PROGRAM-TO-PROGRAM COMMUNICATION
RFAL       LOCAL    DSSERVER   OUTGOING REMOTE FILE ACCESS
RFA        REMOTE   DSSERVER   INCOMING REMOTE FILE ACCESS
NFTL       LOCAL    NFT        OUTGOING NETWORK FILE TRANSFER
NFT        REMOTE   NFT        INCOMING NETWORK FILE TRANSFER
VTRL       LOCAL    DSSERVER   OUTGOING REVERSE VIRTUAL TERMINAL
VTR        REMOTE   DSSERVER   INCOMING REVERSE VIRTUAL TERMINAL
VTL        LOCAL    DSSERVER   OUTGOING VIRTUAL TERMINAL
VT         REMOTE   DSSERVER   INCOMING VIRTUAL TERMINAL

SERVICE    STARTED  FEATURES

PADL         YES
PAD          YES
NSSTATL      YES
NSSTAT       YES
HDSPNS       YES
PDS          YES
LOOPBACK     YES
RPML         YES
RPM          YES      AUTOLOGON
PTOPL        YES
PTOP         YES
RFAL         YES
RFA          YES      AUTOLOGON
NFTL         YES
NFT          YES      AUTOLOGON
VTRL         YES
VTR          YES
VTL          YES
VT           YES
```

Shows the status of the Network Services. Local means the service gives local users access to remote resources; remote means the service gives remote users access to local resources. Server indicates the type of server used for the service. For this example, all the services were started as indicated by YES in the STARTED column of the display. A NO in that column would indicate that the service was not started. The STATUS display can be used to verify whether each individual service is started or not, and whether it is available for local or remote use. This is helpful when using the optional *services* list of the

NSCONTROL functions, START and STOP. The AUTOLOGON feature can be used for the RPM, RFA and NFT services. See the NSCONTROL AUTOLOGON command earlier in this section for details.

The HDSPNS and PDS services are for the Resource Sharing Products from Hewlett Packard's Office Systems Division.

Example 2

```
:NSCONTROL STATUS=SERVERS

SERVER      MIN     MAX    FEATURES

PADSVR        0   32767    TRACE
NSSTATUS      0   32767
LOOPBACK      0   32767
HDSPNS        0   32767
PDSERVER      0   32767
NFT           0   32767
DSSERVER      0   32767

JOB/SESSION   PIN   STATUS    SERVICE

#S49           76   ACTIVE    NFT
#S51          158   ACTIVE    VT
DSDAD         126   ACTIVE    VT
#S8            81   ACTIVE    PTOP
```

Shows the status of the servers. Here the minimum number of each server is 0 and the maximum is 32,767, the defaults. There is one NFT server in existence and three DSSERVER processes in existence. Two of the DSSERVER processes are being used for the VT service, and one DSSERVER process is being used for the PTOP service.

The TRACE feature can be used with PADSVR for the PAD and PADL services. See the NSCONTROL TRACE command later in this section for more details.

The servers HDSPNS and PDSERVER are servers used for Resource Sharing Products from Hewlett Packard's Office Systems Division.

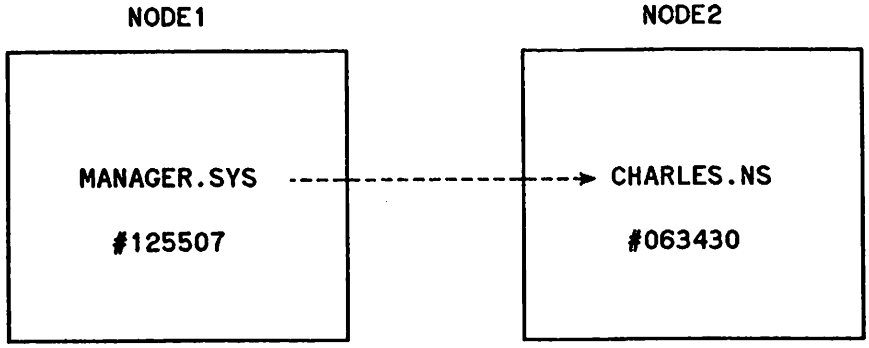


Figure 9-3. Representation of Example 3

In the third example, assume that a user has entered the following commands on NODE1:

Example 3

```
:HELLO MANAGER.SYS
:DSLINE NODE2
:REMOTE HELLO CHARLES.NS
```

The result on NODE1 is:

```
:NSCONTROL STATUS=USERS

JOBNUM  SESSION  TYPE         USER.ACCOUNT
           ID     SERVICES      NODENAME

#S2     #125507  LOCAL        MANAGER.SYS
        #063430   VT            NODE2.DOMAIN.ORGANIZATION
```

and on NODE2:

```
:NSCONTROL STATUS=USERS

JOBNUM  SESSION  TYPE         USER.ACCOUNT
           ID     SERVICES           NODENAME

#S3     #063430  REMOTE       CHARLES.NS
        #125507   ORIGIN        NODE1.DOMAIN.ORGANIZATION
```

The display on NODE2 shows the remote session for CHARLES.NS from the REMOTE HELLO on NODE1. As illustrated in Figure 3, the session IDs can be used to match up the local and remote sessions. The local session on NODE1, with id #125507, is the origin of the remote session on NODE2, with ID #063430.

Example 4

```
:NSCONTROL STATUS=USERS,SERVICES

NO CURRENT NETWORK SERVICE USERS

NO NETWORK SERVICES ARE CURRENTLY ACTIVE
```

Shows the users of Network Services on the node and the status of all the services. In this example, an NSCONTROL START has not been issued, so the services are not started and there are no Network Services users.

NSCONTROL STOP

Terminates Network Services subsystem.

Syntax

```
NSCONTROL STOP[=service [,service]...][,NET=niName [,niName]...]
               [;function]...
```

Parameters

STOP=*services*

Terminates the network services subsystem, and prevents users from using the services in the list. Existing users of the services can continue until they finish. The optional service list (*services*) allows you to select which of the services are disabled for local or remote use. When all Network Services are stopped, the DSDAD process will terminate.

Defaults: If you omit the optional service list but include the NET= parameter with *niNames*, all the NS services (for local and remote use) will be stopped, including the PAD and PADL services on the networks specified in the NET= parameter. If you specify only NSCONTROL STOP all the NS services, including PAD and PADL on active networks, will be stopped.

The *services* list is the same as for the START function, except that the specified services are stopped, not started. Specifying the following services prevents users on remote nodes from using resources on the local node:

VT — Prevents remote users from logging onto the local node using the REMOTE HELLO command.

VTR — Prevents remote users from accessing local terminals using the Reverse VT service.

PAD — Prevents remote users from using a public or private PAD to establish a session on the local node.

NFT — Prevents remote users from transfering files to or from the local node using the DSCOPY command and intrinsic.

RFA — Prevents remote users from accessing files on the local node.

PTOP — Prevents remote users from creating and communicating with PTOP slave processes on the local node.

RPM — Prevents remote users from creating and killing processes on the local node using the Remote Process Management service.

LOOPBACK	Allows remote users to use the loopback diagnostic server on the local node.
NSSTAT	Allows remote users to use the NSSTATUS intrinsic to retrieve network services information from the local node.

Specifying the following *services* prevents users on the local node from using resources on remote nodes:

VTL	Prevents local users from logging onto remote nodes using the REMOTE HELLO command.
VTRL	Prevents local users from accessing terminals on remote nodes using the Reverse VT service.
PADL	Prevents local users from programmatically accessing terminals or printers attached to a remote HP 2334A.
NFTL	Prevents local users from transferring files to or from remote nodes using the DSCOPY command and intrinsics.
RFAL	Prevents local users from opening and accessing files and databases on remote nodes using the RFA and RDBA services.
PTOPL	Prevents local users from creating and communicating with PTOP slave processes on remote nodes.
RPML	Prevents local users from creating and killing processes on the local and remote nodes using the RPM service.
NSSTATL	Allows local users to use the NSSTATUS intrinsic to retrieve network services information from the local and remote nodes.

function You may want to issue one or more of the NSCONTROL functions on the same command line with STOP. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

The NSCONTROL command can be used to terminate the Network Services using one of two methods. The STOP function is the normal way to shut down the Network Services. It allows existing users to continue using the services until they finish their tasks, but prevents any new users from using the services. The ABORT function should *only* be used in abnormal situations. It immediately terminates all the services and all the server processes. Anyone using a service will find their task (REMOTE, DSCOPY, etc) immediately terminated.

Example 1 shows NSCONTROL STOP without the service list. All Network Services are stopped. Any active servers are allowed to continue until finished with the current task, at which point they are terminated. No new service requests are accepted. When all the servers and services are stopped, the DSDAD process terminates. Use the STATUS function to check that all users and servers are finished before issuing the NETCONTROL STOP command.

Example 1

```
:NSCONTROL STOP
:NSCONTROL STATUS=USERS,SERVERS

NO NETWORK SERVERS ARE CURRENTLY ACTIVE OR RESERVED

:NETCONTROL STOP
```

In this example, all users were finished. However, if there were still active users, then issuing the NETCONTROL STOP command would be equivalent to issuing an NSCONTROL ABORT. It "cleans up" all services and transport activity, which means it immediately terminates the Network Services. It is important to check the status of the services before issuing the NETCONTROL STOP command.

Example 2

```
:NSCONTROL STOP=VT,VTR,PAD,NET=X25NET1,X25NET2
```

Stops the VT and Reverse VT services, and stops the PAD services on the X.25 networks named X25NET1 and X25NET2. This prevents remote users from logging on to the local node using REMOTE HELLO and from opening local terminals using Reverse VT. This also prevents remote users on the X25NET1 and X25NET2 networks from using a PAD to establish a session on the local node. If there are any other started services, they remain available.

NSCONTROL TRACE

Invokes tracing for PAD services.

Syntax

```
                   ⎧ON ⎫ ⎧,PADSVR⎫
NSCONTROL TRACE=   ⎨   ⎬ ⎨       ⎬ [,file] [,recs] [,maxdata]
                   ⎩OFF⎭ ⎩,pin # ⎭
```

Parameters

ON

PADSVR — A symbolic name for the PADSUP server. Invokes tracing for all the PAD services that are currently active or will be subsequently started through the NSCONTROL START command.

pin # — A process identification number of the PADSUP server process that can be obtained through the NSCONTROL STATUS command. Invokes tracing for a PAD service associated with the *pin #* specified in the command.

OFF

PADSVR — Deactivates tracing for all active PAD services.

pin # — Deactivates tracing for a PAD service associated with this *pin #*.

file — The name of a new or existing MPE file in which the trace is to be stored. If this parameter is omitted, the trace information is sent to a default file named TRxxxxxx, where TR is followed by either the six characters PADSVR (if the tracing was invoked by specifying the PADSVR option) or by the *pin #* (if the tracing was invoked by specifying the *pin #* option). This file will always reside in the PUB group of the SYS account.

recs — The number of records allocated to a new trace file.

Default: 1024.

maxdata — The maximum amount of data to be traced on an individual send or receive request.

Range: 0 to 8000 bytes

Default: 2000 bytes

NSCONTROL VERSION

Displays the version numbers for the Network Services software modules and the overall subsystem version.

Syntax

```
NSCONTROL VERSION[=MOD][;function]...
```

Parameters

VERSION[=MOD]

Displays the overall version of the Network Services. If qualified with the MOD keyword, displays the version of each of the software modules of the Network Services and the overall version.

function

You may want to issue one or more of the NSCONTROL functions (or multiple instances of the VERSION function) on the same command line with VERSION. Refer to the NSCONTROL command page for a description of the functions available.

Discussion

The software modules of all HP products have a version identification number which includes the version, update, and fix level of the software module. The VERSION function of the NSCONTROL command allows you to check the version numbers of the Network Services software modules to ensure that they are compatible and up-to-date. The display is the same as that for NMMAINT, described in Section 2, except that only the Network Services subsystem is displayed.

Example 1

```
:NSCONTROL VERSION

Network Services Overall Version : A.00.08
```

If you want to look at the version numbers of the individual modules, you would specify the MOD keyword. You would see a display like the one shown in Example 2. This example is a sample only and the modules and version numbers may not be the same as on your installed system.

Example 2

```
:NSCONTROL VERSION=MOD

Network Services individual module versions:

Program file:   DSDAD.NET.SYS              Version:   A0008005
SL procedure:   ASCXVERS                   Version:   A0008007
SL procedure:   ASBUFVERS                  Version:   A0008000
SL procedure:   ASENVVERS                  Version:   A0008000
SL procedure:   DSUTILVERS                 Version:   A0008001
SL procedure:   SUBSYS6FMTVERS             Version:   A0008001
Catalog file:   ASCAT.NET.SYS              Version:   A0008003
SL procedure:   VTSVERS1                   Version:   A0008001
SL procedure:   VTSVERS2                   Version:   A0008001
Program file:   IOVTERMO.PUB.SYS           Version:   A0008001
Program file:   LOOPBACK.NET.SYS           Version:   A0008000
Program file:   NSSTATUS.NET.SYS           Version:   A0008000
Program file:   DSSERVER.NET.SYS           Version:   A0008000
SL procedure:   ASRFAVERS                  Version:   A0008000
SL procedure:   ASPTOPVERS                 Version:   A0008000
SL procedure:   ASRPMVERS                  Version:   A0008000
Program file:   NFT.NET.SYS                Version:   A0008000
Program file:   RPMDAD.PUB.SYS             Version:   A0008000
Program file:   IOPADTRM.PUB.SYS           Version:   A0008011
Program file:   IOPADLP.PUB.SYS            Version:   A0008011
Program file:   PADSVR.NET.SYS             Version:   A0008011

Network Services overall subsystem version:  A.00.08
```

RESUMENMLOG

Resumes logging after a recoverable error.

Syntax

```
RESUMENMLOG
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	YES
	Programmatically?	YES
Breakable?		NO
Capabilities?		NM

Discussion

RESUMENMLOG causes the resumption of logging to the NM disc log file upon the correction of a recoverable I/O error.

Assume that the system is on line, NM logging is enabled, and a recoverable error occurs on NMLG file number 104. The error is corrected and the RESUMENMLOG command is entered. The following message is then displayed on the system console:

```
NMLG FILE NUMBER nnnn. NM LOGGING RESUMED

NMLG FILE NUMBER nnnn ON
```

Refer to the *NS3000/V Error Message and Recovery Manual* for more information on recoverable errors.

SHOWCOM

Displays status information about a communications device.

Syntax

```
SHOWCOM ldev [;ERROR ]...
             [;RESET ]
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	YES
	Programmatically?	YES
Breakable?		NO
Capabilities?		Executable Only At Console*

* unless distributed to users with the ALLOW or ASSOCIATE commands

Parameters

ldev — The logical device number of a communication system device. As described in Section 4 of Volume I, the logical device number is assigned during system configuration and can be found on the I/O listing for the system.

ERROR — A request for the full status list. If not specified, an abbreviated list is displayed.

RESET — A request to reset all status information to zero after it has been displayed.

Discussion

The status information generated by this command can be used to determine communication line activity and quality. When the SHOWCOM command is invoked on an open line, the statistics displayed are the values accumulated since the last opening of the line, unless the display was reset, in which case the statistics shown are accumulated since the last reset of the display. The status information obtained at the close of the line reflects the statistics generated over the last open/close sequence, unless the display was reset during that period. Use of the LAN Node Diagnostic Remote Node Test (described in the

LAN/3000 Diagnostic and Troubleshooting Guide) resets the SHOWCOM status information to zero after operation.

Example 1 shows the abbreviated status list display of SHOWCOM. In this example, the display is for a LANIC configured as logical device 50.

Example 1

```
:SHOWCOM 50

                        LDN - 50
MESSAGES SENT         15246        MESSAGES RECVD 14273
            LAST RECOVERABLE ERROR        0
            LAST IRRECOVERABLE ERROR      0
            LINE IS CONNECTED
```

The abbreviated status display for the INP is the same as that of the LANIC. The abbreviated list includes:

- LDN. Logical device number.
- MESSAGES SENT and MESSAGES RECEIVED. Refers to the number of text blocks transmitted or received. Each text block is sent with either a Block Check Character (BCC) or a Cyclic Redundancy Check (CRC) to ensure error-free transmission.
- LAST RECOVERABLE ERROR and LAST IRRECOVERABLE ERROR. A recoverable error is one from which the protocol can recover and allow the logical link to remain established. An example would be a BCC/CRC Error or a Response Timeout. An irrecoverable error, in general, is one from which the protocol cannot recover and which results in the logical link being severed. Such an error does not necessarily cause the line to be closed. The display shows the CS Error Code for the last recoverable and the irrecoverable error, if any.
- Line state:
    - CONNECTED means initialized and in use,
    - CLOSED means not initialized,
    - DISCONNECTED means still initializing or an irrecoverable error has occurred.
    - UNDEFINED means a software or hardware error has occurred.

However, the information provided in the full status display differs for the two types of devices, as shown in the following examples.

Example 2 shows the the full status display for a LANIC, again configured as logical device 50.

Example 2

```
:SHOWCOM 50;ERROR

    TRANSMIT             LDN - 50              RECEIVE
MESSAGES SENT           15376          MESSAGES RECVD 14429
COLLISIONS              0              BCC/CRC ERRORS 0
EXC COLLISION ERRS      0              BUFF OVERFLOWS 0
UNDERRUNS               0              OVERRUNS       0
CLR TO SEND LOSSES      0              LENGTH ERRORS  0
            # OF RECOVERABLE ERRORS     0
            LAST RECOVERABLE ERROR      0
            # OF IRRECOVERABLE ERRORS   0
            LAST IRRECOVERABLE ERROR    0
            LINE IS CONNECTED
```

Example 3 shows the the full status display for a INP, here configured as logical device 18.

Example 3

```
:SHOWCOM 18;ERROR

    TRANSMIT             LDN - 18              RECEIVE
MESSAGES SENT           4232           MESSAGES RECVD 4992
RETRANSMISSIONS         0              BCC/CRC ERRORS 0
RESPONSE TIMEOUTS       145            RECV TIMEOUTS  349
UNDERRUNS               0              OVERRUNS       0
CLR TO SEND LOSSES      0              CARRIER LOSSES 0
            # OF RECOVERABLE ERRORS     145
            LAST RECOVERABLE ERROR      7
            # OF IRRECOVERABLE ERRORS   0
            LAST IRRECOVERABLE ERROR    0
            LINE IS CONNECTED
```

The full status list, in addition to the items listed above, includes:

- # OF RECOVERABLE ERRORS and # OF IRRECOVERABLE ERRORS. The total number of errors which have occurred.

- **Transmit fields**

  - COLLISIONS (LANIC only). As part of the CSMA/CD protocol, the line is monitored. A collision is detected if the line is in use or if two nodes transmit simultaneously.

  - EXC COLLISION ERRS (LANIC only). The IEEE 802.3 standard specifies up to 16 attempts to transmit a data packet. If all 16 attempts result in a collision error, the number of excessive collisions is incremented by one.

  - UNDERRUNS. The LANIC transmits data onto the line at an extremely high rate, which is one word every 1.6 microseconds. This field shows the number of times, if any, that data could not be transmitted onto the line at the required rate. This field is not meaningful for the INP.

- CLR TO SEND LOSSES. This field is only valid for communications lines configured as full-duplex. It refers to the fact that after the INP raises Request to Send (RTS) at initiation and the modem raises Clear to Send (CTS), both RTS and CTS should be raised as long as the line is open. If, for any reason, the modem drops CTS while the INP has RTS raised, the number of CTS losses is incremented by one.

- RETRANSMISSIONS (INP only). For DS or NS Point-to-Point Links, this is the number of times a NAK was sent, which happens if the BCC/CRC field did not check on the remote side.

- RESPONSE TIMEOUTS (INP only). The number of times an enquiry (ENQ) was sent without receiving an acknowledgment (ACK) within the allotted time period.

- **Receive fields**

  - BCC/CRC ERRORS. Counts the number of times that the text received from the remote did not pass the checksum bit error check.

  - BUFF OVERFLOWS (LANIC only). The number of times data was received but could not be accepted because no buffers were available in the inbound buffer pool. The number of buffers in this pool is configured with NMMGR. Refer to Section 7 of Volume I (Link Configuration).

  - OVERRUNS. The LANIC transmits data to the HP 3000 at an extremely high rate, which is one word every 1.6 microseconds. This field shows the number of times, if any, that data was received but could not be passed to the HP 3000 at the required rate. This field is not meaningful for the INP.

  - LENGTH ERRORS. (LANIC only). This is the number of frames in which the IEEE 802.3 length field in the frame did not match the number of bytes actually contained in the frame.

  - RECV TIMEOUTS (INP only). The number of times the receive timer expired without any transmission being received from the remote. The receive timeout is configured with SYSDUMP. Refer to Section 4 of Volume I (System Configuration).

  - CARRIER LOSSES (INP only). This field is only valid for communications lines configured as full-duplex. It refers to the fact that after the modem raises Carrier Detect (CD), meaning that it detected the carrier signal from the transmitting modem, that CD should remain on as long as the line is open. If the modem drops CD due to closing the line or a possible transmission line interruption, then the number of carrier losses is incremented by one.

Text Reference

Refer to the *LAN/3000 Diagnostic and Troubleshooting Guide* for a detailed explanation of the SHOWCOM display for the LANIC. The CS error codes are specified in the *Fundamental Data Communications Handbook* (5957-4634).

Displays the number and available space of the log file.

Syntax

```
SHOWNMLOG
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	YES
	Programmatically?	YES
Breakable?		NO
Capabilities?		NM

Discussion

SHOWNMLOG displays the number of the current NMLG file and the percentage of available file space currently used.

The information appears in the following format:

```
NMLG FILE NUMBER nnnn IS mm% FULL
```

where *nnnn* is the NMLG file number and *mm* is the percentage of file space used.

If network logging is disabled due to an irrecoverable error, NMS displays the following message explaining the cause. The manager will have to do a warm or cool start to bring up the system again.

```
NMLG FILE NUMBER nnnn ERROR #nn.  NM LOGGING STOPPED.(NMCNERR 36)
```

If network logging is enabled but currently suspended due to a recoverable error, NMS displays the following messages explaining the cause. Once the error is corrected, the manager can then issue the RESUMENMLOG command explained in this section.

```
NMLG FILE NUMBER nnnn IS mm% FULL

NMLG FILE NUMBER nnnn ERROR #mm. NMLOGGING SUSPENDED.(NMCNERR 38)
```

SWITCHNMLOG

Closes the current log file and creates and opens a new one.

Syntax

```
SWITCHNMLOG
```

Use

Available	In Session?	YES
	In Job?	YES
	In Break?	YES
	Programmatically?	YES
Breakable?		NO
Capabilities?		NM

Discussion

SWITCHNMLOG closes the current NMLG file and creates and opens a new one. When SWITCHNMLOG is entered, NMS displays the message:

```
NMLG FILE NUMBER nnnn IS mm% FULL

NMLG FILE NUMBER pppp ON
```

where *nnnn* is the previous NMLG FILE number, *mm* is the percentage of file space used, and *pppp* is the newly opened file numbered one more than the last file number.

If network logging is disabled due to an irrecoverable error when SWITCHNMLOG is entered, NMS displays the following message explaining the cause. The system will need to be brought back up with a warm or cool start.

```
NMLG FILE NUMBER nnnn ERROR #nn. NM LOGGING STOPPED.  (NMCNERR 36)
```

If network logging is enabled but currently suspended due to a recoverable error, NMS displays the following message explaining the cause. When the problem is corrected, the manager can issue the RESUMENMLOG command.

```
NMLG FILE NUMBER nnnn ERROR #nn. NM LOGGING SUSPENDED.  (NMCNERR 38)
```

SOFTWARE AND LINE VERIFICATION

SECTION 2

This chapter is divided into two sections: the first part describes software verification, and the latter part describes line verification.

- You can use the following three utilities to check that all software modules of NS3000/V services and link products are current and compatible:
  - NMMAINT
  - CSLIST
  - DSLIST
- You can use the following five line verification tests to determine if a node is correctly communicating with a network:
  - IPC TEST
  - XPT TEST
  - NSLOGON
  - Loopback Initiator
  - QUICKVAL
  - NODESTAT

NOTE

If your network uses any types of computers other than HP 3000s (for example, HP 9000s), be sure to read the *NS Cross-System NFT Reference Manual.* This chapter pertains to HP 3000-to-HP 3000 communication. This means, for example, that tests described for two HP 3000 nodes are not applicable for HP 3000-to-HP 9000 communication. However, the single-node tests described in this section do apply for the HP 3000 even when it is linked to non-HP 3000 computers.

NOTE

For nodes on NS X.25 3000/V networks, also refer to the *NS 3000/V X.25 Link Guide* for additional software tests.

SOFTWARE VERIFICATION

Each data communications product consists of a variety of software modules. Each software module has an individual version number. The software modules of all HP data communications products use a standard version stamp. This stamp has the format:

vuuffiii

where:

v = the version number of the software. This corresponds to a major revision or a version for a new or revised system environment.

u = the update level of the software. This corresponds to a significant revision in product functionality.

f = the fix level of the software. This corresponds to a new, supported revision of the software, and may map directly to a MIT.

i = the internal fix level of the software. This is for differentiating special releases of software that do not correspond to a normal MIT cycle. This field should not be checked under normal circumstances to determine the compatibility of a product.

A subsystem is a grouping of software modules. The software modules within each subsystem usually have a common or similar function. NS3000/V is grouped into the following subsystems:

- Network Services
- Network Transport
- Node Management Services
- Link Support Services.
- Node Management Configurator
- Communication Services (CS/3000)

See Volume I, Section 3 of this manual set for more information on NS3000/V subsystems.

The following three utilities check that all the software modules that make up NS3000/V services and its links are current and compatible.

- NMMAINT. Displays version numbers of NS3000/V network services and link products, as well as HP SNA software products.
- CSLIST. The CSLIST utility lists the software version numbers of the link and diagnostic software known as CS/3000. CS/3000 contains the intrinsics that manage the transfer of data between communications devices. CS/3000 also contains modules for the INP and LANIC download files.
- DSLIST. The DSLIST utility lists the software module version numbers of the DS services and DS compatible links.

Some of the software modules belong to more than one subsystem and are displayed by more than one utility. Because data communications products consist of multiple subsystems, you may need to use all three utilities to check the software modules of one product.

NMMAINT

The NM Maintenance Utility (NMMAINT) is a utility program supplied with NS3000/V (services and links). NMMAINT is used to display the individual and overall version numbers for the software modules of NS3000/V, SNA IMF, and SNA NRJE Network Services, as well as the SNA and NS3000/V network link products.

Each software module within a subsystem has its own version ID number. If the version, update, and fix levels of these modules do not match, the subsystem will not work correctly. NMMAINT can be used to determine if your software installation is valid. *The information provided by NMMAINT must be included in any Service Request submitted to HP.* Refer to the *NS3000/V Error Message and Recovery Manual* for information about submitting Service Requests (SRs).

To run NMMAINT, issue the command:

```
:RUN NMMAINT.PUB.SYS
```

NMMAINT responds with the following:

```
NMS Maintenance Utility 32099-11018A.01.02  (C) Hewlett Packard Co. 1985
```

NMMAINT then lists the version identification numbers for each software module and information for each subsystem. As shown in the example below, the NMMAINT utility displays version information for the subsystems of the products actually installed on your system. The Node Management Services, Link Services, and Network Transport subsystems are displayed if an NS3000/V link product is installed. The Network Services subsystem is displayed if the NS3000/V services product is installed. The SNA Transport, NRJE, and IMF subsystems are displayed if the appropriate HP-to-IBM data communications products are installed on your system. The example shows a system with NS3000/V services and a IEEE802.3 link installed. The IPCVERSION module is port software. This is not part of the NetworkIPC user service, nor does it form a subsystem, but its individual version ID number is displayed by NMMAINT for your information.

Version ID numbers include version, update, fix levels, and an internal fix level in the format *vuuffiii*. For NMVERS00 in the example below, the version ID number is A0103023. A is the version level, the next two digits represent the update level, and the next two digits are the fix level. The remaining numbers, 023, show the internal fix level, which is used only within Hewlett-Packard. In the following examples, the software version numbers are not intended to be the same as the version numbers of your software.

Example

```
:RUN NMMAINT.PUB.SYS

 NMS Maintenance Utility 32099-11018A.01.03   (C) Hewlett Packard Co. 1986

 Subsystem version ID's


Node Management Services 32098-11018 module versions:

SL procedure:   NMVERS00                    Version:  A0103023
SL procedure:   NMVERSCSL                   Version:  A0103023
SL procedure:   NMVERS01                    Version:  A0103000
SL procedure:   NMLOGSLVERS                 Version:  A0103010
SL procedure:   NMLOGDATAVERS               Version:  A0103014
SL procedure:   NMVERS04                    Version:  A0103007
SL procedure:   NMVERS05                    Version:  A0103000
SL procedure:   BFMVERS                     Version:  A0103X13
Program file:   NMMAINT.PUB.SYS             Version:  A0103005
Program file:   NMFILE.PUB.SYS              Version:  A0103006
Program file:   NMLOGMON.PUB.SYS            Version:  A0103016
Program file:   NMDUMP.PUB.SYS              Version:  A0103P41
Catalog file:   NMCAT.PUB.SYS               Version:  A01030P9

Node Management Services 32098-20010 overall version = A.01.03

Network Transport 32343A module versions:

Program file:   NETCP.NET.SYS               Version:  A0100000
Program file:   NETSERVE.NET.SYS            Version:  A0100000
Program file:   SOCKREG.NET.SYS             Version:  A0100003
Catalog file:   NETMSG.NET.SYS              Version:  A0100009
Program file:   STUD.NET.SYS                Version:  A0100006
SL procedure:   NET'SM4'VERS                Version:  A0100000
SL procedure:   NET'UI'VERS                 Version:  A0100009
SL procedure:   NET'SL'VERS                 Version:  A0100019
SL procedure:   NET'NI'VERS                 Version:  A0100030
SL procedure:   NET'PROBE'VERS              Version:  A0100012
SL procedure:   NET'DIAL'VERS               Version:  A0100016
SL procedure:   NET'TCP0'VERS               Version:  A0100021
SL procedure:   NET'TCP1'VERS               Version:  A0100021
SL procedure:   NET'PXP0'VERS               Version:  A0100010
SL procedure:   NET'PXP1'VERS               Version:  A0100010
SL procedure:   NET'IP'VERS                 Version:  A0100038
SL procedure:   NET'IPU'VERS                Version:  A0100005
SL procedure:   NET'PD'VERS                 Version:  A0100015
SL procedure:   NET'X25'VERS                Version:  A0100006
SL procedure:   SOCKIOVERS                  Version:  A0100063
SL procedure:   SOCKACCESSVERS              Version:  A0100063
SL procedure:   SOCKMISC1VERS               Version:  A0100063
SL procedure:   SUBSYS3FMTVERS              Version:  A0100001
SL procedure:   SUBSYS5FMTVERS              Version:  A0100001

Network Transport 32343A overall version = A.01.00
Network Services individual module versions:

Program file:   DSDAD.NET.SYS               Version:   A0008005
SL procedure:   ASCXVERS                    Version:   A0008007
SL procedure:   ASBUFVERS                   Version:   A0008000
SL procedure:   ASENVVERS                   Version:   A0008000
SL procedure:   DSUTILVERS                  Version:   A0008001
SL procedure:   SUBSYS6FMTVERS              Version:   A0008001
Catalog file:   ASCAT.NET.SYS               Version:   A0008003
SL procedure:   VTSVERS1                    Version:   A0008001
SL procedure:   VTSVERS2                    Version:   A0008001
Program file:   IOVTERM0.PUB.SYS            Version:   A0008001
Program file:   LOOPBACK.NET.SYS            Version:   A0008000
Program file:   NSSTATUS.NET.SYS            Version:   A0008000
Program file:   DSSERVER.NET.SYS            Version:   A0008000
SL procedure:   ASRFAVERS                   Version:   A0008000
SL procedure:   ASPTOPVERS                  Version:   A0008000
SL procedure:   ASRPMVERS                   Version:   A0008000
Program file:   NFT.NET.SYS                 Version:   A0008000
Program file:   RPMDAD.PUB.SYS              Version:   A0008000
Program file:   IOPADTRM.PUB.SYS            Version:   A0008011
Program file:   IOPADLP.PUB.SYS             Version:   A0008011
Program file:   PADSVR.NET.SYS              Version:   A0008011

Network Services overall subsystem version:   A.00.08


Link Support Services 32098-20011  module versions:

SL procedure:  TRAN'VERS0                  Version:  B0208A03
SL procedure:  TRAN'VERS1                  Version:  B0208A03
SL procedure:  SUBSYS8FMTVERS              Version:  B0208019
SL procedure:  LINKVERS                    Version:  B0208033
Program file:  LINKMGR.PUB.SYS             Version:  B0208040
Program file:  PCMANAGE.PUB.SYS            Version:  B0208003

Link Support Services 32098-20011  overall version = B.02.08

SL procedure:  IPCVERSION                  Version:  d0101000

Node Management Configurator 32098-20012 module versions:

SL procedure:  NMCVERS                     Version:  A0200000
SL procedure:  NMVERS06                    Version:  A0200000
Program file:  NMMGR.PUB.SYS               Version:  A0200000
Program file:  NMMGRVER.PUB.SYS            Version:  A0200000
V+ forms file: NMMGRF.PUB.SYS              Version:  A0200000
Catalog file:  NMMGRCAT.PUB.SYS            Version:  A0200000
Catalog file:  NMMGRHLP.PUB.SYS            Version:  A0200000
NM conf file:  NMSAMP1.PUB.SYS             Version:  A0200000
NM conf file:  NMAUX1                      Version:  A0200000
SL procedure:  NETDIRVERS                  Version:  A0200000
Program file:  NMSIG.PUB.SYS               Version:  A0200000
```

```
   Node Management Configurator 32098-20012 overall version = A.02.00
```

The first group of numbers in the above example are the version ID numbers of the modules of the Node Management Services subsystem, part of an NS3000/V link. Notice that the first five characters of the version for each module listed in this group are the same. This means that all the software modules in the subsystem match. This must be true for all the modules of a given subsystem. If a subsystem module is invalid, the following error message is printed:

```
Program file:  NMMAINT.PUB.SYS              ** MODULE ERROR **
   ONE OR MORE SUBSYSTEM MODULES ARE INVALID.  (NMERR 105)
```

This message indicates that the modules of the subsystem are not compatible.

Because the module version ID numbers match, NMMAINT displays the overall subsystem version number for the Node Management Services; for the above example, it is A.01.03. The rest of the subsystems are handled in a similar fashion.

NMMAINT also checks that all the modules that belong with a particular subsystem are present. If a module is missing, NMMAINT displays the name of the module with the following error message in place of the version number.

```
SL procedure:  NMVERS01                     **  REQ'D MODULE MISSING **
   ONE OR MORE REQUIRED SUBSYSTEM MODULES ARE MISSING.    (NMERR 104)
```

If an optional module is not present, NMMAINT displays a message similiar to the following:

```
   Program file:  NMDUMP.PUB.SYS  **NOT INSTALLED**
```

If the modules were correct when installed, only unusual circumstances such as a reload, a disc problem, or a system failure would result in missing or invalid modules. Restore a known valid version of the modules in error. For more troubleshooting information, refer to the *NS3000/V Error Message and Recovery Manual.*

Question marks displayed for the overall version number indicate that the fix levels of the individual modules do not match. Remember that the internal fix level, represented by the last three numbers of the version ID, does not need to match between modules for the software to be compatible. Fix numbers are requested in Service Requests for HP to use when troubleshooting.

As each subsystem is displayed, NMMAINT checks that all the modules are present and compatible. However, NMMAINT does not perform any cross-subsystem version verification. When a system has HP-to-IBM products as well as HP-to-HP products installed, the Node Management Services, Link Services and the port software are used by both types of data communications products. Therefore, it is important to check that the version numbers of these common subsystems and port software module are correct. It is possible for the HP-to-IBM products to use previous versions of the common software that are not compatible with the HP-to-HP products. For more information, refer to the *NS3000/V Error Message and Recovery Manual.*

NMMAINT displays information on only the subsystems for the products installed on your system. In the example above, the SNA Link, SNA NRJE, and SNA IMF products were not installed, so their subsystems were not displayed.

CSLIST

The Communications Systems (CS/3000) subsystem consists of the software modules used for link management and diagnostics. CS is used by all HP network link products, including NS3000/V links and DS Compatible Links. The CSLIST utility lists the version numbers for the software modules of the CS subsystem. CSLIST also provides detailed information on the INP and LANIC download files on your system. *The information provided by CSLIST must be included in any Service Request submitted to HP.* Refer to the *NS3000/V Error Message and Recovery Manual* for information about submitting Service Requests (SRs).

Example 1 shows how to run CSLIST.

Example 1

```
:RUN CSLIST.PUB.SYS

HP30131A.55.25 CSLIST/3000  SUN, MAR 17, 1984, 9:05 AM
(C) HEWLETT-PACKARD CO. 1980

THIS ROUTINE HAS TWO MAJOR FUNCTIONS - ONE ASSOCIATED WITH
THE CS MODULES AND ONE ASSOCIATED WITH THE DOWNLOAD FILES.

THE FIRST PORTION REPORTS THE CS MODULES INSTALLED
ON THE SYSTEM.
NOTINSTD INDICATES THE MODULE HAS NOT BEEN INSTALLED ON THE SYSTEM.

CSLIST ALSO ALLOWS THE USER TO OBTAIN INFORMATION
CONCERNING THE HP-STANDARD DOWNLOAD FILES.
THIS INFORMATION INCLUDES PROTOCOL TYPE, BOARD TYPE,
COMPILE DATE, AND VERSION NUMBER INFORMATION.

DO YOU WANT A COMPLETE LISTING OF INSTALLED VUFS? YES

DO YOU WANT THE DOWNLOAD FILE INFORMATION?  NO

SHOULD OUTPUT BE DIRECTED TO THE LP? YES
```

After the the RUN command for CSLIST is issued, CSLIST displays a header and a description, followed by a series of prompts. In Example 1, the user requests a complete listing of the CS module version numbers, without the download file information, and specifies that the output be directed to the lineprinter (device LP).

Example 2 shows a typical listing of CS modules produced by CSLIST.

Example 2

```
HP30131v.uu.ff CSLIST/3000  SUN, MAR 17, 1985, 9:05 AM
(C) HEWLETT-PACKARD CO. 1980

COMPLETE LISTING OF INSTALLED VUFS NOW BEING PRODUCED.
OUTPUT GOING TO DEVICE LP.

COMSYS1     INSTALLED VUF IS v.uu.ff
COMSYS2     INSTALLED VUF IS v.uu.ff
COMSYS3     INSTALLED VUF IS v.uu.ff
COMSYS4     INSTALLED VUF IS v.uu.ff
COMSYS5     INSTALLED VUF IS v.uu.ff
CSUTILITY   INSTALLED VUF IS v.uu.ff
CSDUMMY     INSTALLED VUF IS v.uu.ff
CSDUMP      INSTALLED VUF IS v.uu.ff
TRACPROG    INSTALLED VUF IS v.uu.ff
IOINPO      INSTALLED VUF IS v.uu.ff
DSM         INSTALLED VUF IS v.uu.ff
INPDPAN     INSTALLED VUF IS v.uu.ff
NETCONF     INSTALLED VUF IS v.uu.ff
CSLIST      INSTALLED VUF IS v.uu.ff
IOLANO      INSTALLED VUF IS v.uu.ff
LANDPAN     INSTALLED VUF IS v.uu.ff
LANDIAG     INSTALLED VUF IS v.uu.ff
```

In Example 1, the download file information was not selected. Because CSLIST lists information on the download files for all HP 3000 products installed on your system, requesting the download file information may produce a lengthy listing. The alternative, recommended if you want to check a specific download file, is to use CSLIST with an entrypoint. The entrypoint, either LAN or INP, is appended to the RUN command. This is shown in Example 3. When you use CSLIST with an entrypoint, CSLIST displays an abbreviated description and prompts you for the download file.

Example 3

```
:RUN CSLIST.PUB.SYS,LAN


CSLIST ALLOWS THE USER TO OBTAIN INFORMATION
CONCERNING USER-SPECIFIED DOWNLOAD FILES.  THIS
INFORMATION INCLUDES PROTOCOL TYPE, BOARD TYPE,
COMPILE DATE, AND VERSION NUMBER INFORMATION.

SHOULD OUTPUT BE DIRECTED TO THE LP?

DOWNLOAD FILE NAME =CSDLAN1.PUB.SYS

LAN DOWNLOAD FILE = CSDLAN1.PUB.SYS
LAST MODIFIED WED, NOV 21, 1984,  9:41 AM
DRIVER OPTIONS = %000000
DATE CODE = B.00.014.077

DOWNLOAD FILE NAME =CSDNLPB2.PUB.SYS

DOWNLOADFILE= CSDNLPB2.PUB.SYS           PROTOCOL TYPE=  NS/X.25/LAPB
BOARD TYPE=  INP 20B  COMPILE DATE= WED, JUL 4, 1984,  6:26 PM
      IC VERSION = 01.02
PROTOCOL VERSION = 01.04
   TRACE VERSION = 02.06
   RAMCP VERSION = 05.04

DOWNLOAD FILE NAME =CSDNBSC2.PUB.SYS

DOWNLOADFILE= CSDNBSC2.PUB.SYS            PROTOCOL TYPE= NS/BSC
BOARD TYPE=  INP 20B  COMPILE DATE= THU, OCT 25, 1984,  1:56 PM
      IC VERSION = 01.02
PROTOCOL VERSION = 01.11
   TRACE VERSION = 02.06
   RAMCP VERSION = 05.05

DOWNLOAD FILE NAME =

END OF PROGRAM
```

Three download files are specified in Example 3--one for an IEEE 802.3 link using a LANIC, one for a Point-to-Point Link (INP), and one for an NS X.25 Link (INP). Notice that CSLIST provides information on all three download files, even though two are for the INP.

DSLIST

For the modules of the DS Services subsystem of NS3000/V, and for DS Compatible Links, use the DSLIST program installed in the PUB group of the SYS account to obtain versions of the software modules.

In the DSLIST display, shown in the example below, notice that most of the modules are grouped under a product number heading. The utilities that apply to all DS Compatible Links are listed under the Common Modules heading.

All modules shown, except those for DSN/X.25, should be displayed on your system. The DSN/DS HP32189B modules are used by DS Point-to-Point Links and by the DS Services subsystem of NS3000/V. The DSN/X.25 HP32191B modules are installed with the DS X.25 Network Link and appear only if a DS X.25 Link is installed on your system. The CS HP30131A modules show a subset of the display provided by CSLIST. The COMSYS module is an overall version number for the CS modules. The NETCONF module is the utility used for the network configuration of the DS X.25 Network Link, described in the *DSN/X.25 for the HP 3000 Reference Manual.*

It is essential that all the DS software modules installed on the system have the same version identification; all the DS X.25 software modules (if installed) have the same version, as well as all the common modules and CS modules, in order to ensure successful operation. The information provided by DSLIST should be included in any SR submitted. For information on submitting an SR, refer to the *NS3000/V Error Message and Recovery Manual.*

Example

```
:RUN DSLIST.PUB.SYS

HEWLETT PACKARD 32189v.uu.ff DSLIST/3000 SUN, MAR 17, 1985,  7:07 PM

DSN/DS HP32189B:
   MODULE     VERSION
SL DSSEGS     v.uu.ff, INTERNAL FIX xxx
SL DSRTECALL  v.uu.ff, INTERNAL FIX xxx
   DSMON      v.uu.ff, INTERNAL FIX xxx
   DSTEST     v.uu.ff, INTERNAL FIX xxx
   DS2026     v.uu.ff, INTERNAL FIX xxx
   DS2026CN   v.uu.ff, INTERNAL FIX xxx
   DSCOPY     v.uu.ff, INTERNAL FIX xxx
   IODSO      v.uu.ff, INTERNAL FIX xxx
   IODSTRMO   v.uu.ff, INTERNAL FIX xxx
   IODSTRMX   v.uu.ff, INTERNAL FIX xxx

DSN/X.25 HP32191B:
   MODULE     VERSION
   DSMONX     v.uu.ff, INTERNAL FIX xxx
   IODSX      v.uu.ff, INTERNAL FIX xxx
   IOPADO     v.uu.ff, INTERNAL FIX xxx
   IOPAD1     v.uu.ff, INTERNAL FIX xxx
```

(continued on next page)

(DSLIST example, continued)

```
COMMON MODULES:
   MODULE      VERSION
SL DSIOM       v.uu.ff, INTERNAL FIX xxx
   DSDUMP      v.uu.ff, INTERNAL FIX xxx
   DSLIST      v.uu.ff, INTERNAL FIX xxx

CS SUBSYSTEM HP30131A:
   MODULE      VERSION
SL COMSYS      v.uu.ff, INTERNAL FIX xxx
   NETCONF     v.uu.ff, INTERNAL FIX xxx

END OF PROGRAM
:
```

LINE VERIFICATION

Perform the troubleshooting tests in this section only after you have completed the problem-resolution steps in the *NS3000/V Error Message and Recovery Manual*. Other steps recommended in the *Error Message and Recovery Manual* include: 1) issuing the SHOWCOM command, 2) inspecting the output, 3) checking the log files for useful information, and 4) investigating the link products.

The line tests IPC, XPT, QuickVal, and NSLOGON are provided to verify the operation of NS3000/V services and link products. The XPT, IPC, and NSLOGON line tests are interactive and use the NetIPC intrinsics to check if the Network Transport is working correctly. QuickVal, which runs in batch mode, checks the Network Services, the Network Transport and NetIPC. Run the QuickVal test to check out newly installed NS3000/V services software. NODESTAT is an interactive tool that provides status information about a network node and is described later in this section.

Configuring Files

When performing software line verification, proceed as follows:

1. During configuration, be sure log classes 1 through 4 are all enabled.

2. Also during configuration, enable the TCP checksum.

3. When using the QuickVal test, be sure to use the INFO=*nodename* parameter as shown in the subsection "USING QUICKVAL" later on in this section.

Test Sequence

As specified in the *NS3000/V Error Message and Recovery Manual*, you already should have eliminated the possibility of a hardware problem by running:

- the LAN Node Diagnostic (LANDIAG) test if your node uses a LANIC board, or

- the INP Diagnostic Support Monitor test (DSM) if your node uses an INP board, or

- the TERMDSM test if your node uses an ATP.

You used LANDIAG 1-16 to test the IEEE 802.3 Link components, and LANDIAG 17 to test the connection between nodes.

The line tests can be run on one node (software loopback) and between nodes. First, start clean--shut down and restart the Network Transport and Network Services. Issue the command SHOWCOM *ldev*; RESET to set the values of the status display to zero.

Now, you are ready to run the first line test, XPT. By running this test in software loopback, you see if the local Network Transport is functioning properly. If it is not functioning properly, you have isolated the problem and need not proceed with the line tests. However, if the local transport is working, run

QuickVal in software loopback to make sure the services are working for the local node. Then, go on to the two-node XPT line test.

The two-node XPT line test checks the remote transport and both the local and remote links. If this test reveals an error, refer to "Interpreting the Tests" in this section, and perform any necessary hardware troubleshooting.

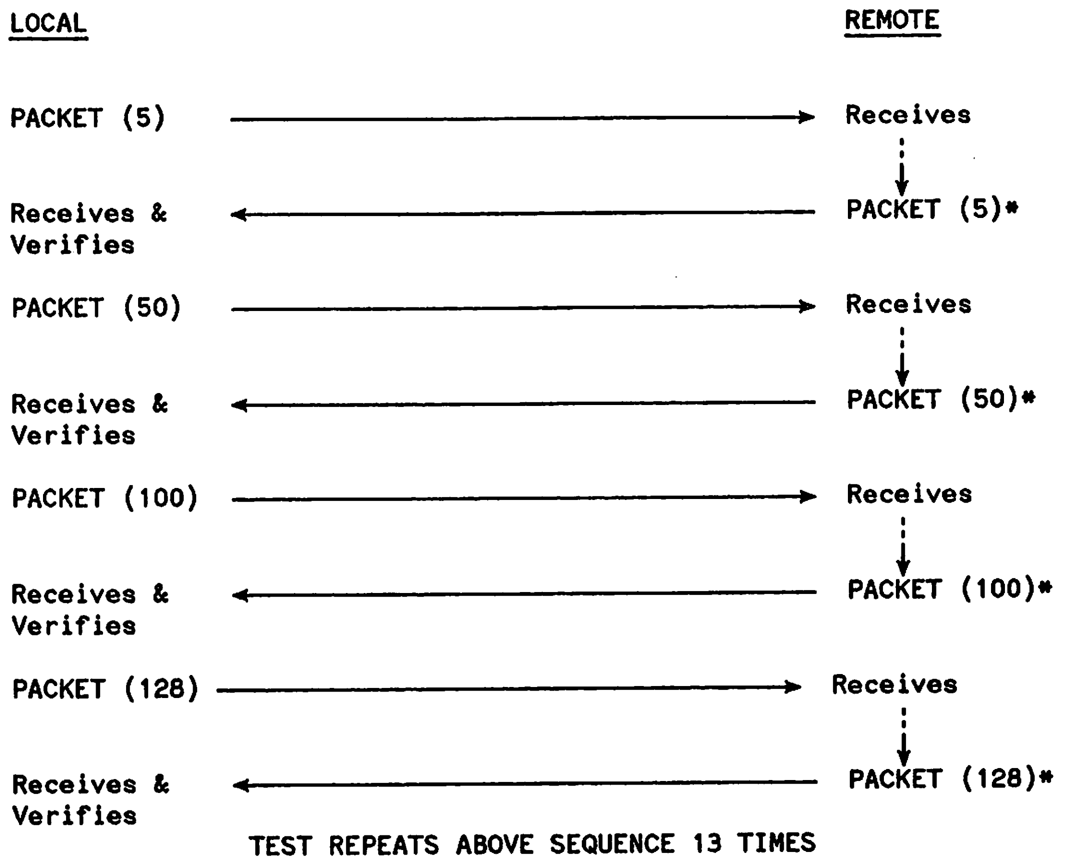
If both the software loopback and two-node XPT line tests reveal no errors, run the IPC line test between nodes. (Running this test in software loopback is not necessary because you already checked the transport.) The two-node IPC line test, in addition to checking the Network Transport and both links, uses Network Services, specifically RPM. This means that if this test shows that an item is not working, the problem most likely is with Network Services, probably RPM.

To check all Network Services, run QuickVal between nodes. Even if the IPC two-node line test reveals no errors, run the QuickVal test in case an error exists in a Network Service other than RPM. QuickVal tests the services by executing the intrinsics and commands of each service. If more than one service is not working, it could mean that DSDAD and its associated modules are not functioning properly. DSDAD is the control process for all Network Services.

THE IPC and XPT TESTS

The IPC and XPT line tests are used to verify that the Network Transport is operating correctly. These line tests use the NetIPC intrinsics to establish connections and exchange data. They can be run in software loopback or over a network between two nodes. The IPC line test uses the Network Services, specifically RPM, to initiate the remote process. When running the XPT line test, the local and remote processes must be initiated separately because the XPT line test does not use Network Services.

Checking the Network Transport consists of testing the Transmission Control Protocol (TCP) and the Packet Exchange Protocol (PXP). PXP is tested by sending 52 packets to a remote program and receiving the packets back, slightly modified. Packet sizes alternate between 5, 50, 100, and 128 bytes. Refer to Figure 2-1.



* = PACKET modified before being returned.

Figure 2-1. Testing PXP

TCP is tested with three types of data transmission The first consists of sending packets from the local program to the remote, the second from the remote to the local, and the third from the local to the remote and back to the local. A total of 52 packets are sent, with sizes alternating between 5, 50, 500, and 5000 bytes. The 5,000-byte packets are fragmented by TCP to their correct link sizes, which are configurable in the configuration file. The packets are reassembled by TCP when received. Refer to Figure 2-2.

The packets used in these tests consist of a sequence of ASCII characters between '!' and "z". In the tests of TCP, each side verifies when it receives a packet that the characters are correct. In the test of PXP, only the originator verifies that the characters in each packet, when returned, are correct.

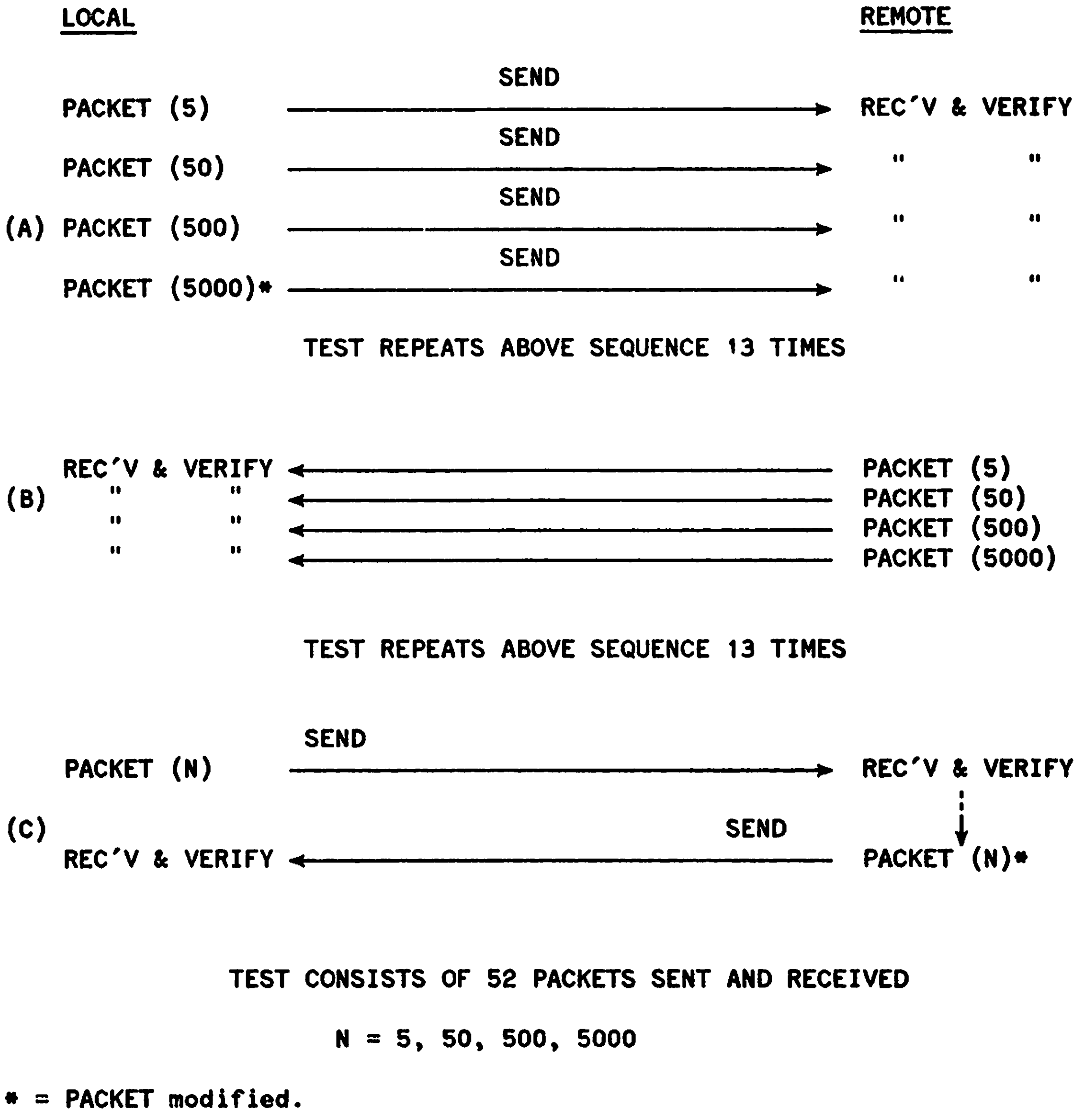


Figure 2-2. Testing TCP

Initialize the Network

You must initialize the network before running the line tests. In all cases the Network Transport and Network Services must be terminated before being initiated again. This is accomplished with the NETCONTROL STOP and NSCONTROL STOP commands. For command syntax, refer to Section 1 in Volume II of this manual set. Software loopback is initiated by issuing the commands shown in Example 1 at the console. The network interface name (*niName*) in Example 1 would be a loopback network interface name.

Example 1

```
:NETCONTROL NET=niName; START
:NSCONTROL START
```

The NSCONTROL START command is not required when performing the XPT line test, since this test does not use Network Services.

To perform a two-node test, issue the commands shown in Example 2 at the console of each node. These commands initiate the network. The network interface name (*niName*) in Example 2 would be any network interface name that is not a loopback name.

Example 2

```
LOCAL     :NETCONTROL NET=niName; START
          :NSCONTROL START

REMOTE    :NETCONTROL NET=niName; START
          :NSCONTROL START
```

Again, NSCONTROL START is not required for the XPT line test.

How the Tests Work

The IPC and XPT line-test programs reside in the NET.SYS group and account. Do not run more than one test at a time. For the IPC line test, the local controlling program is IPCLOCAL and the remote program is IPCREMOT. The remote process is initiated with the RPMCREATE intrinsic.

Refer to Example 3 for running the IPC line test.

Example 3

```
:run ipclocal.net.sys

IPCLOCAL   A.00.01   (C) HEWLETT-PACKARD CO. 1985

Local node name: (default is NODE1) RETURN
Default nodename "NODE1" is used
Local socket name (default is LSOCK1): RETURN
Default socket name "LSOCK1" is used
Remote socket name (default is RSOCK1): RETURN
Default remote socket name "RSOCK1" is used
Turn on individual packet messages? (Y/N; default is N) RETURN

Remote node name (default is NODE2; loop gives loopback):LOOP
Remote logon (user.acct[,group]):  MANAGER.SYS
Remote password (if any) in form userpass,acctpass,grppass
```

Example 3 shows the IPC test prompts. You must respond to the prompts with correct nodenames. If you are running the IPC test in software loopback, respond to the REMOTE NODE NAME prompt by typing LOOP.

For the local and remote socket names, any name can be entered. The defaults are LSOCK1 and RSOCK1, respectively. You must know a valid remote logon and any remote passwords. When you enter a remote password, it is not echoed on your screen. However, it must be typed in exactly as shown in Example 3. Commas are needed before and after the account to separate it from the user and group.

The local controlling program for the XPT line test is XPTLOCAL and the remote program is XPTREMOT. The XPT test is run by first entering RUN XPTLOCAL.NET.SYS at the local node and answering the prompts, followed by logging on and issuing the command RUN XPTREMOT.NET.SYS at the remote node. If you do not run the XPTLOCAL program first or if you give an incorrect socket name to XPTREMOT, you receive Program Error #18. You also receive the NetIPC error SOCKET TIMEOUT at the remote console. If only XPTLOCAL is run, it loops endlessly looking for the remote program. BREAK can be issued to terminate the test. See Examples 4A and 4B for the prompts given by the XPT line test.

Example 4A

```
:RUN XPTLOCAL.NET.SYS

XPTLOCAL   A.01.00   (C) HEWLETT-PACKARD CO. 1986

Local node name: (default is NODE1) RETURN
Local socket name (default is LSOCK1): RETURN
Remote socket name (default is RSOCK1): RETURN
Turn on individual packet messages? (Y/N; default is N) RETURN

REMOTE NODE NAME (default is NODE2; loop gives loopback): LOOP

NOW RUN THE REMOTE PART OF THE TEST: XPTREMOT.NET.SYS
ON THE REMOTE NODE
```

Example 4B

```
:RUN XPTREMOT.NET.SYS

XPTREMOT   A.00.00   (C) HEWLETT-PACKARD CO. 1985
THE LOCAL PROGRAM, XPTLOCAL.NET.SYS, MUST BE RUN FIRST.
START XPTLOCAL ON THE LOCAL NODE AND THEN RUN THIS PROGRAM.

Socket name (MUST BE THE SAME AS THE REMOTE SOCKET NAME IN LOCAL PROGRAM)
Default is RSOCK1: RETURN
Default socket name "RSOCK1" is used
```

As Examples 4A and 4B show, you are asked for the same information (see Example 3) as in the IPC line test, with the exception of the prompts for a remote logon and password. The reason you are not prompted for these items is that the XPT test does not use RPM. This is why you must logon and run the XPTREMOT program.

For both line tests, leave packet messages OFF. They should be turned on only to check the progress of the tests if a problem is detected. If packet messages are ON, do not display them at the console because so many other messages are displayed at the console that it would be difficult to pinpoint the packet messages.

Interpreting the Tests

The progress of the IPC and XPT line tests is shown by messages printed at the user terminal indicating the packets sent, received, and verified. Important messages that indicate either an error or the success of a key part of a line test appear in inverse video at the console or user terminal. The final message at the console indicates if all parts of the line test succeeded.

Example 5 shows the messages printed at the console for an IPC software loopback test. These messages, except for the first two, are also generated by the XPT loopback test. The first two messages in Example 5 are RPM-related. Only the IPC line test uses RPM.

Example 5

```
17:21/#S3/36/FROM/MANAGER.SYS/LOCAL: REMOTE PROCESS CREATED

17:21/#S3/37/FROM/MANAGER.SYS/REMOTE BEGUN
17:21/#S3/37/FROM/MANAGER.SYS/REMOTE: PXP STARTED

17:21/#S3/36/FROM/MANAGER.SYS/LOCAL: SENDING PXP MESSAGES

17:21/#S3/37/FROM/MANAGER.SYS/REMOTE: PXP TEST COMPLETED SUCCESSFULLY
17:21/#S3/37/FROM/MANAGER.SYS/REMOTE: TCP BEGUN

17:21/#S3/36/FROM/MANAGER.SYS/LOCAL: PXP TEST SUCCESSFUL
17:21/#S3/36/FROM/MANAGER.SYS/LOCAL: CONNECTION BEGUN

17:23/#S3/36/FROM/MANAGER.SYS/LOCAL: SEND TO REMOTE SUCCEEDED

17:23/#S3/37/FROM/MANAGER.SYS/REMOTE: RECEIVE FROM LOCAL SUCCESSFUL

17:24/#S3/36/FROM/MANAGER.SYS/LOCAL: RECEIVE FROM REMOTE SUCCESSFUL

17:24/#S3/37/FROM/MANAGER.SYS/REMOTE: SEND TO LOCAL SUCCESSFUL

17:26/#S3/37/FROM/MANAGER.SYS/REMOTE: RECEIVE AND SEND BACK SUCCESSFUL
17:26/#S3/36/FROM/MANAGER.SYS/LOCAL: SEND AND RECEIVE SUCCEEDED

17:26/#S3/37/FROM/MANAGER.SYS/REMOTE: TCP TEST SUCCESSFUL
17:26/#S3/37/FROM/MANAGER.SYS/REMOTE: ALL TESTS SUCCEEDED

17:26/#S3/36/FROM/MANAGER.SYS/LOCAL: TCP TEST SUCCESSFUL
17:26/#S3/36/FROM/MANAGER.SYS/LOCAL: ALL TESTS COMPLETED SUCCESSFULLY
```

Because Example 5 uses software loopback, the local and remote messages are displayed at the same console. The IPCREMOT process generates the remote messages. Example 5 shows the PXP test and three TCP tests beginning and completing successfully. Messages regarding connection initiations are also shown. Refer to Figures 2-1 and 2-2 for more information about the PXP and TCP tests.

Examples 6A and 6B show the console messages printed for the IPC two-node test. In this case, the local and remote messages are displayed at separate consoles. Again, the XPT test prints the same messages as the IPC test, except for the two RPM-related messages (the first message in Examples 6A and 6B). Example 6A shows the messages printed at the local console, and Example 6B shows the messages printed at the remote console. Note that console logging for the network transport is not enabled in this example.

Example 6A

```
14:06/#S2/47/FROM/MANAGER.SYS/LOCAL: REMOTE PROCESS CREATED
14:07/#S2/47/FROM/MANAGER.SYS/LOCAL: SENDING PXP MESSAGES
14:07/#S2/47/FROM/MANAGER.SYS/LOCAL: PXP TEST SUCCESSFUL
14:07/#S2/47/FROM/MANAGER.SYS/LOCAL: CONNECTION BEGUN
14:07/#S2/47/FROM/MANAGER.SYS/LOCAL: SEND TO REMOTE SUCCEEDED
14:08/#S2/47/FROM/MANAGER.SYS/LOCAL: RECEIVE FROM REMOTE SUCCESSFUL
14:10/#S2/47/FROM/MANAGER.SYS/LOCAL: SEND AND RECEIVE SUCCEEDED
14:10/#S2/47/FROM/MANAGER.SYS/LOCAL: TCP TEST SUCCESSFUL
14:10/#S2/47/FROM/MANAGER.SYS/LOCAL: ALL TESTS COMPLETED SUCCESSFULLY
```

Example 6B

```
14:06/#S7/54/FROM/MANAGER.SYS/REMOTE BEGUN
14:06/#S7/54/FROM/MANAGER.SYS/REMOTE: PXP STARTED
14:07/#S7/54/FROM/MANAGER.SYS/REMOTE: PXP TEST COMPLETED SUCCESSFULLY
14:07/#S7/54/FROM/MANAGER.SYS/REMOTE: TCP BEGUN
14:07/#S7/54/FROM/MANAGER.SYS/REMOTE: RECEIVE FROM LOCAL SUCCESSFUL
14:08/#S7/54/FROM/MANAGER.SYS/REMOTE: SEND TO LOCAL SUCCESSFUL
14:10/#S7/54/FROM/MANAGER.SYS/REMOTE: RECEIVE AND SEND BACK SUCCESSFUL
14:10/#S7/54/FROM/MANAGER.SYS/REMOTE: TCP TEST SUCCESSFUL
14:10/#S7/54/FROM/MANAGER.SYS/REMOTE: ALL TESTS SUCCEEDED
```

If an error occurs, it is indicated at the console. Important information about error types and parameters appears at the console and user terminal. Some errors allow the tests to continue, so the original error may scroll off the terminal.

Four types of errors that may occur are 1) RPM errors, 2) packet-verification errors, 3) receive failures, and 4) socket-creation failures. Since XPT does not use RPM, RPM errors can occur only in an IPC line test. The other error types can occur in either test.

RPM errors most likely are user errors. Look up error numbers in the *NS3000/V Error Message and Recovery Manual.* Most RPM errors are due to a network problem. Make sure logons, passwords and nodenames are correct. (For the XPT test, errors in nodenames cause the XPTLOCAL program to abort with a NetIPC error. Also, an error in the socket name in XPTREMOT causes XPTREMOT to abort with a NetIPC error, in which case XPTLOCAL loops looking for the remote process. XPTREMOT can be run again with the correct socket name and the program will continue.)

Packet verification errors, if any occur, can be seen when packet messages are on.

Receive failure errors may be timing related. An error in one test should not affect the outcome of another test. If a receive failure error occurs, obtain as much information as possible from the screen. Also, find the first SOCKERR number, and read the message. As for errors in socket-creation or other areas vital to NetIPC, these errors cause a test to termninate.

To interpret line test errors, refer to "Software Line Test Errors" in the *NS3000/V Error Message and Recovery Manual.*

Examples 7A and 7B show the local and remote messages, respectively, that are printed when an IPC line test is run over two nodes with packet messages off. Except for the REMOTE PROCESS CREATED message, which is generated by RPM, XPT prints the same messages.

Example 7A

```
LOCAL         *** REMOTE PROCESS CREATED ***
LOCAL         *** LOCAL SOCKET CREATED ***
LOCAL         IPCSHUTDOWN COMPLETED
LOCAL         *** PXP TEST COMPLETED ***
LOCAL         LOCAL IPCRECVCN SUCCEEDED ***
 SEND FROM LOCAL TO REMOTE SUCCESSFUL
LOCAL RECEIVED MESSAGES SUCCESSFULLY
SEND AND RECEIVE AT LOCAL SUCCEEDED
LOCAL         IPCSHUTDOWN CD COMPLETED
LOCAL         IPCSHUTDOWN SD COMPLETED
LOCAL         ****** TCP TEST COMPLETED ******
```

Example 7B

```
REPLY SOCKET ESTABLISHED
REMOTE        IPCSHUTDOWN COMPLETED
REMOTE        ***PXP TEST COMPLETED***
REMOTE      ***REMOTE SOCKET CREATED***
REMOTE      ***IPCCONNECT REQUEST INITIATED***
REMOTE      ***CONNECTION ESTABLISHED***
REMOTE      ***CONNECTION IS UP***
RECEIVE FROM LOCAL SUCCESSFUL
SEND FROM REMOTE TO LOCAL SUCCESSFUL
REMOTE ******TCP TEST SUCCESSFUL*****
```

It is possible that one session, local or remote, may abort during an IPC or XPT line test. If this occurs, abort the other session by using the ABORTJOB command. No further information can be obtained from the test.

Packet Tracing

The PARM option is available for the IPC and XPT line tests. Adding parameter 32 enables packet tracing for both the local and remote processes. For the IPC line test, trace files are: TRACEL1, TRACEL2, TRACEL3 and TRACEL4 for the local PXP and TCP tests, respectively, and TRACER1, TRACER2, TRACER3 and TRACER4 for the remote tests. The PARM parameter can be set by adding ;PARM=32 to the run command:

```
:RUN IPCLOCAL.NET.SYS; PARM=32
```

Use this only on the advice of your Hewlett-Packard representative. The XPT tests have the same trace-file names. Add the PARM option to only the XPTLOCAL run command. Volume II, Section 3 of this manual set describes how to use NMDUMP to format trace files.

Individual Packet Messages

Example 8A shows the local packet messages for the IPC line test with individual packet messages on. These messages appear at the local user terminal. Except for the REMOTE PROCESS CREATED message, the XPT test prints the same messages. Turn packet messages on only in problem cases, when you want to see exactly what happens for each packet.

Example 8B shows the remote messages that appear at the remote user terminal for the XPT test when packet messages are on. Remote messages may also appear for the IPC test, interspersed with the local messages at the local terminal. To see the IPC remote messages, the remote process must be created in an already existing remote session. To do this, use the current *user.account,group* for the remote logon in a software loopback test or do a REMOTE HELLO to a *user.account,group* in the current session. Then, run the IPCLOCAL test and specify the existing remote session as the remote logon.

Example 8A

```
IPCLOCAL    A.00.00    (C) HEWLETT-PACKARD CO. 1985


LOCAL          *** REMOTE PROCESS CREATED ***
LOCAL          *** LOCAL SOCKET CREATED ***
5 bytes sent, Packet (1)
5 bytes received, Packet (1)
Data verified, Packet (1)
50 bytes sent, Packet (2)
50 bytes received, Packet (2)
Data verified, Packet (2)
100 bytes sent, Packet (3)
100 bytes received, Packet (3)
Data verified, Packet (3)
128 bytes sent, Packet (4)
128 bytes received, Packet (4)
Data verified, Packet (4)
           .
           .
           .
(Test sequence repeats for 52 packets)
5 bytes sent, Packet (49)
5 bytes received, Packet (49)
Data verified, Packet (49)
50 bytes sent, Packet (50)
50 bytes received, Packet (50)
Data verified, Packet (50)
100 bytes sent, Packet (51)
100 bytes received, Packet (51)
Data verified, Packet (51)
128 bytes sent, Packet (52)
128 bytes received, Packet (52)
Data verified, Packet (52)
LOCAL           IPCSHUTDOWN COMPLETED
LOCAL          *** PXP TEST COMPLETED ***
LOCAL          *** LOCAL IPCRECVCN SUCCEEDED ***
LOCAL             5 bytes of data sent (1)
LOCAL            50 bytes of data sent (2)
LOCAL           500 bytes of data sent (3)
LOCAL          5000 bytes of data sent (4)
           .
           .
           .
(Test sequence repeats for 52 packets)
```

Example 8A (Continued)

```
LOCAL          5 bytes of data sent (49)
LOCAL         50 bytes of data sent (50)
LOCAL        500 bytes of data sent (51)
LOCAL       5000 bytes of data sent (52)
 SEND FROM LOCAL TO REMOTE SUCCESSFUL
LOCAL     5 bytes of data received (1)
Data verified, Packet #1
LOCAL     50 bytes of data received (2)
Data verified, Packet #2
LOCAL     500 bytes of data received (3)
Data verified, Packet #3
LOCAL     5000 bytes of data received (4)
Data verified, Packet #4
          .
          .
          .
```

(Test Sequence Repeats for 52 packets)

```
LOCAL          5 bytes of data received (49)
Data verified, Packet #49
LOCAL          50 bytes of data received (50)
Data verified, Packet #50
LOCAL          500 bytes of data received (51)
Data verified, Packet #51
LOCAL          5000 bytes of data received (52)
Data verified, Packet #52
 LOCAL RECEIVED MESSAGES SUCCESSFULLY
LOCAL          5 bytes sent, Packet #1
LOCAL          5 bytes of data received (1)
Data verified, Packet #1
LOCAL          50 bytes sent, Packet #2
LOCAL          50 bytes of data received (2)
Data verified, Packet #2
LOCAL          500 bytes sent, Packet #3
LOCAL          500 bytes of data received (3)
Data verified, Packet #3
LOCAL          5000 bytes sent, Packet #4
LOCAL          5000 bytes of data received (4)
Data verified, Packet #4
          .
          .
          .
```

(Test sequence repeats for 52 packets)

Example 8A (Continued)

```
LOCAL          5 bytes sent, Packet #49
LOCAL          5 bytes of data received (49)
Data verified, Packet #49
LOCAL          50 bytes sent, Packet #50
LOCAL          50 bytes of data received (50)
Data verified, Packet #50
LOCAL          500 bytes sent, Packet #51
LOCAL          500 bytes of data received (51)
Data verified, Packet #51
LOCAL          5000 bytes sent, Packet #52
LOCAL          5000 bytes of data received (52)
Data verified, Packet #52
SEND AND RECEIVE AT LOCAL SUCCEEDED
LOCAL          IPCSHUTDOWN CD COMPLETED
LOCAL          IPCSHUTDOWN SD COMPLETED
LOCAL         ****** TCP TEST COMPLETED ******
```

Example 8B

```
IPCREMOT   A.00.00   (C) HEWLETT-PACKARD CO. 1985
REPLY SOCKET ESTABLISHED
5 bytes received, pxp id #157446004 (1)
5 bytes sent, pxp id #157446004 (1)
50 bytes received, pxp id #158036223 (2)
50 bytes sent, pxp id #158036223 (2)
100 bytes received, pxp id #158626440 (3)
100 bytes sent, pxp id #158626440 (3)
128 bytes received, pxp id #159216749 (4)
128 bytes sent, pxp id #159216749 (4)
          .
          .
          .
```

(Test sequence repeats for 52 packets)

```
5 bytes received, pxp id #161008052 (49)
5 bytes sent, pxp id #161008052 (49)
50 bytes received, pxp id #157469490 (50)
50 bytes sent, pxp id #157469490 (50)
100 bytes received, pxp id #158059707 (51)
100 bytes sent, pxp id #158059707 (51)
128 bytes received, pxp id #158650018 (52)
128 bytes sent, pxp id #158650018 (52)
REMOTE      IPCSHUTDOWN COMPLETED
REMOTE      ***PXP TEST COMPLETED***
REMOTE      ***REMOTE SOCKET CREATED***
REMOTE      ***IPCCONNECT REQUEST INITIATED***
REMOTE      ***CONNECTION ESTABLISHED***
REMOTE      ***CONNECTION IS UP***
REMOTE      5 bytes of data received (1)
Data verified
REMOTE      50 bytes of data received (2)
Data verified
REMOTE      500 bytes of data received (3)
Data verified
REMOTE      5000 bytes of data received (4)
Data verified
          .
          .
          .
```

(Test sequence repeats for 52 packets)

Example 8B (Continued)

```
REMOTE      5 bytes of data received (49)
Data verified
REMOTE      50 bytes of data received (50)
Data verified
REMOTE      500 bytes of data received (51)
Data verified
REMOTE      5000 bytes of data received (52)
Data verified
RECEIVE FROM LOCAL SUCCESSFUL
REMOTE      5 bytes of data sent (1)
REMOTE      50 bytes of data sent (2)
REMOTE      500 bytes of data sent (3)
REMOTE      5000 bytes of data sent (4)
            .
            .
            .
```

(Test sequence repeats for 52 packets)

```
REMOTE      5 bytes of data sent (49)
REMOTE      50 bytes of data sent (50)
REMOTE      500 bytes of data sent (51)
REMOTE      5000 bytes of data sent (52)
SEND FROM REMOTE TO LOCAL SUCCESSFUL
REMOTE      5 bytes of data received (1)
Data verified
5 bytes sent, (1)
REMOTE      50 bytes of data received (2)
Data verified
50 bytes sent, (2)
REMOTE      500 bytes of data received (3)
Data verified
500 bytes sent, (3)
REMOTE      5000 bytes of data received (4)
Data verified
5000 bytes sent, (4)
            .
            .
            .
```

(Test sequence repeats for 52 packets)

Example 8B (Continued)

```
REMOTE      5 bytes of data received (49)
Data verified
5 bytes sent, (49)
REMOTE      50 bytes of data received (50)
Data verified
50 bytes sent, (50)
REMOTE      500 bytes of data received (51)
Data verified
500 bytes sent, (51)
REMOTE      5000 bytes of data received (52)
Data verified
5000 bytes sent, (52)
REMOTE      ******TCP TEST SUCCESSFUL******
```

NSLOGON

NSLOGON is a tool that can be used to verify that the Network Transport is operating correctly. The main purpose of this tool is to quickly verify the connectivity between a node and all other nodes in the network or catenet. This will save the task of having to do a DSLINE and REMOTE to every possible destination node when the network is first started, a new node is added, or any major change is made to the configuration file.

NSLOGON uses the NetIPC intrinsics to establish a connection to a well-known server on a remote node. Therefore, both the Network Transport and the Network Services must be started on all nodes before using this tool. NSLOGON can either contact all or some of the nodes listed in the network directory, or you can enter the nodenames yourself. There doesn't have to be a directory on the system.

How to Use NSLOGON

The files neccessary for running NSLOGON reside in the NET.SYS group and account. The two files, NSLOGON and LOGONCMD, must be present. You also must have Node Manager (NM), Process Handling (PH), and Privileged Mode (PM) capabilities in order to execute the program.

To run the program, enter the following:

```
:RUN NSLOGON.NET.SYS
```

You now have three different choices on how you want the program to operate. The first prompt you will get is the following:

```
Do you wish to logon to ALL nodes in the directory?(YES):
```

If you answer YES, Y, or RETURN to this question, a list of all the nodenames in the network directory is made. The program will attempt to contact every node in that list one at a time.

If you answer NO or N to that question, then the next prompt you will get will be this:

```
Now you have a choice of either entering each nodename
yourself, or allowing me to prompt you with the nodenames
from the directory.
Would you like me to supply the nodenames?(YES):
```

If you answer YES, Y, or RETURN to this question, a list of all the nodenames in the network directory will also be made. However, this time you will be prompted with the nodename before an attempt is made to contact it. You may then respond with YES, Y, or RETURN and the program will try to reach that node, or you may respond with NO or N and the program will promt you with the next nodename in the list. This choice is helpful if there are a lot of nodenames in the network directory that you know are unreachable.

If you answered YES to either of the above questions, in order for the program to build a list of the nodenames in the network directory, the file NSDIR.NET.SYS must be present on the system the program is running on. If you want to run this program from a LAN node that does not have a Network directory, you can use RFA to access the directory on the PROXY SERVER/gateway node on a LAN that has a network directory. For example, after you have remotely logged on to the gateway node, enter the following, substituting the nodename of the gateway node for *GatewayNodename*:

```
:FILE NSDIR.NET.SYS = NSDIR.NET.SYS:GatewayNodename
```

If you answered NO to both of the above questions, the following will appear:

```
  Please enter destination nodename or CR to quit.
 >
```

This prompt will allow you to enter the nodenames yourself The program will continue prompting you for input until a RETURN is entered. If the destination nodename has the same DOMAIN.ORGANIZATION as the local node, the nodename you enter does not have to be fully qualified.

Sample Output

The following is an example output.

```
  SAKURA.DCL.IND                                        Successful
  MAYTAG.DCL.IND                                        Successful
  TIGGER.DCL.IND                                        Successful
  CECIL.DCL.IND                                         Successful
  ERNIE.DCL.IND                                         *** FAILED ***
  CONNECTION FAILURE DETECTED.  (SOCKERR 67)
  SCHRODER.DCL.IND                                      Successful

  END OF PROGRAM
```

Errors

In the above example, when the program tried to reach ERNIE.DCL.IND, an error occurred. When the program is unable to establish a connection between the node it is running on and the destination node, it will print out a message that it failed and a NetIPC error message. This error message will usually be all that is needed to determine any problems that there may be. However, there are a few messages that will not provide enough information to determine the problem. For these cases, you should turn on all Transport logging classes, and refer to the *NS3000/V Error Message and Recovery Manual* for any error messages that are logged to the console. Below is a list of the more common error messages and their causes.

NOTE

You should have already eliminated the possibility of a hardware problem by running the LANDIAG, DSM, or TERMDSM tests. Also it is assumed that the Network Transport, all Network Interfaces, and the Network Services have been started on the local, remote and all intermediate nodes.

```
CONNECTION FAILURE DETECTED. (SOCKERR 67)
```

If you receive this message while running NSLOGON, it means that the local node sent a packet to the remote node, but the local node never received any packets back from the remote node. This usually indicates that there is a problem on either the destination node or one of the intermediate nodes. There is probably incorrect routing information in the internet or mapping portions in one of the configuration files. For example, the wrong device or gateway is perhaps being used to reach a particular destination. You would also get this error message if the routing information on the local node was incorrect.

If a local configuration file is in error, it ıs much easier to verify because you have easy access to the configuration file. It is moe difficult to determine if a problem is on a remote node, especially if remote nodes are far away. The first thing to verify is that all directly connected nodes (nodes on the same network) can communicate with each other. If all Transport logging classes have been turned on, then an error message should appear on the console of the node where the packet is being discarded.

```
PATH DESCRIPTOR OR PATH DESCRIPTOR
EXTENSIONS UNAVAILABLE (SOCKERR 16)
```

If you receive this error message (or `DESTINATION UNAVAILABLE. SOCKERR 116`) while running NSLOGON, there is probably something wrong with the local configuration file. The problem can either be in the internet or mapping portions of the configuration file or in the network directory. Usually this message indicates that there is an incorrect IP Address in one of these areas. If all logging has been turned on then a Path Resolve error appears on the console. The meaning of the message can be determined by refering to the *NS3000/V Error Message and Recovery Manual.*

LOOPBACK INITIATOR PROGRAM

A node containing the loopback initiator program and the loopback server (collectively called the loopback service) may be used to verify that a layer 5 (NetIPC) connection can be established between two nodes and that data can successfully be transferred between these nodes. The loopback service should not be confused with loopback performed at the network interface level, which is described in Section 12 of Volume I. The loopback service described in this section usually is performed between two nodes and tests layer 5 and below (the layers below the network services).

The loopback initiator is a program that establishes a connection to a loopback server on a specified node. Under user control, it sends data to the server, receives the data back from the server, and verifies that the data has not been corrupted.

The loopback service generally should be run if a network transport problem is suspected. To perform the service, the network services of the remote node must be started.

The loopback service can also be used to gather performance data on layer 5 connections. By using the initiator to send a number of frames of data to a server, you can gather information on the amount of time it takes to send a frame to the server and receive the frame back from the server. This information includes the average, minimum, and maximum round trip time. Upon request, the accumulated information can be displayed in the form of a histogram.

Whenever the loopback initiator detects that a frame has been corrupted in a transfer, the original frame and the received frame will be displayed in ASCII for you to examine. Errors that are encountered with NetIPC calls from the loopback initiator also will be displayed, but the frames of data will not. If five errors occur, the loopback initiator assumes that the connection is unstable; it will shut down the connection to the server, and terminate the loopback service.

To use the loopback initiator, run the program LOOPINIT.NET.SYS. You will be prompted for options and parameters before the transfer(s) begin. After all data transfers are complete, the loopback initiator will ask if additional performance data (histogram) is wanted.

Three examples of running the loopback initiator program are shown and described below. Comments have been added in parentheses.

Example Of Error-Free Run

```
:RUN LOOPINIT.NET.SYS

Enter destination node name :nodename

Specify frame text (y/n)?n (Chose default--text generated internally)
Length of frame in bytes (1..1450)?1000 (An arbitrary selection)

Number of frames to send (1..9999)?250 (An arbitrary selection)

Encountered 0 error(s) in  250 frame(s).

ROUND TRIP DELAY:
   Minimum time =   83 milliseconds
   Maximum time  =  346 milliseconds
   Average time  =  101 milliseconds

Display histogram (y/n)?y

32 +
   -
28 +
   -
24 +
   -
20 +                   *  * *
   -                   * ** *
16 +                   * ** **
   -                  ** ** **
12 +                 *** *****
   -                 *********
 8 +                 ********** *  *
   -               ************ ** *   *
 4 +               *****************   **
   -              ************************  *             *
   +----+----+----+----+----+----+----+----+----+----+----+----+----+
   70   75   80   85   90   95  100  105  110  115  120  125  130  135
                               (milliseconds)

Continue (y/n)?n

END OF PROGRAM
:
```

The vertical axis in the histogram shows the number of frames sent, while the horizontal axis shows milliseconds elapsed. For example, at 90 milliseconds, the height of the asterisks indicates that 20 frames completed their round trips in 90 milliseconds. The histogram is especially helpful when gathering performance data on layer 5 connections.

Example Of Error Detection

```
:RUN LOOPINIT.NET.SYS

Enter destination node name :nodename

Specify frame text (y/n)?y (Selected to enter own text)
Use // to terminate input.
Test sentence number 1.
Test sentence number 2.
Test sentence number 3.
//

Number of frames to send (1..9999)?5 (An arbitrary selection)

Error with frame number    2
Data corruption error:
```

```
Frame sent (Length =  69):
Test sentence number 1.Test sentence number 2.Test sentence number 3.

Frame received (Length =  69):
(&%t sentence number 1.Test sen67&ce number 2.Test sentence number 3.

Encountered 1 error(s) in    5 frame(s).

ROUND TRIP DELAY:
   Minimum time  =   55 milliseconds
   Maximum time  =   76 milliseconds
   Average time  =   64 milliseconds

Display histogram (y/n)?n
Continue (y/n)?n

END OF PROGRAM
:
```

The data corruption error tells you that a problem exists somewhere in the lower five layers (NetIPC and below). Refer to the *NS3000/V Error Message and Recovery Manual* for troubleshooting information.

Example Of NetIPC Call Error

```
:RUN LOOPINIT.NET.SYS

Enter destination node name :nodename

Specify frame text (y/n)?y (Selected to enter own text)
Use // to terminate input.
Test sentence number 1.
Test sentence number 2.
Test sentence number 3.
//

Number of frames to send (1..9999)?5 (An arbitrary selection)

Error with frame number    1
CONNECTION FAILURE DETECTED.   (SOCKERR 67)
Encountered 1 error(s) in    5 frame(s).
ROUND TRIP DELAY:
   Minimum time  =   49 milliseconds
   Maximum time  =   80 milliseconds
   Average time  =   65 milliseconds
Display histogram (y/n)?n
Continue (y/n)?n

END OF PROGRAM
:
```

For more information about a socket error, refer to the *NS3000/V Error Message and Recovery Manual.* Once you have fixed a socket error, you can try again to run the loopback initiator program.

USING QUICKVAL

This is an example of running QuickVal in loopback mode. The messages printed at the console are shown.

```
:RUN QVAL.NET;INFO="NODE6"
NS/3000 QUICK VALIDATION [HP32344A0000000] (C) HEWLETT-PACKARD CO. 1985
ABOUT TO STREAM QVALJOB.NET.SYS
 #J828
11:12/#J828/72/LOGON FOR: QVAL,MANAGER.SYS,NET ON LDEV #10
END OF PROGRAM
:
11:12/#J828/72/FROM/MANAGER.SYS/NETWORK QUICK VALIDATION TEST.
11:12/#J828/72/FROM/MANAGER.SYS/BEGIN BUILDING FILES FOR NFT TEST.
11:12/#S1303/78/LOGON FOR: MANAGER.SYS,NET ON LDEV #60
11:12/#J828/72/FROM/MANAGER.SYS/WAIT FOR MESSAGE SHOWING SUCCESSFUL COMPLETION OF
QUICK VAL!
11:12/#J828/72/FROM/MANAGER.SYS/IF YOU DON'T GET ONE, STUDY SPOOL FILE FOR THIS
JOB CAREFULLY.
11:12/#J828/72/FROM/MANAGER.SYS/BEGIN NFT TEST.
11:13/#J828/72/FROM/MANAGER.SYS/CHECK LOCAL TRANSFER
11:13/#J828/72/FROM/MANAGER.SYS/CHECK REMOTE INTERCHANGE MODE TRANSFER
11:13/#J828/72/FROM/MANAGER.SYS/CHECK REMOTE TRANSPARENT MODE TRANSFER
11:13/#J828/72/FROM/MANAGER.SYS/NFT TEST COMPLETED.
11:13/#J828/72/FROM/MANAGER.SYS/BEGIN RPM TEST
11:14/#J828/86/PRPM0002 CREATED.
11:14/#J828/72/FROM/MANAGER.SYS/RPM TEST COMPLETED.
11:14/#S1303/78/LOGOFF ON LDEV #60
11:14/#S1304/90/LOGON FOR: MANAGER.SYS,NET ON LDEV #60
11:14/#J828/72/FROM/MANAGER.SYS/BEGIN RFA & PTOP TEST
11:14/#J828/72/FROM/MANAGER.SYS/RFA AND PTOP TEST COMPLETED.
11:14/#J828/72/FROM/MANAGER.SYS/BEGIN DSLINE, REMOTE HELLO TEST.
11:14/#S1304/90/LOGOFF ON LDEV #60
11:14/#S1305/97/LOGON FOR: MANAGER.SYS,NET ON LDEV #60
11:14/#S1305/97/LOGOFF ON LDEV #60
11:14/#J828/72/FROM/MANAGER.SYS/DSLINE, REMOTE HELLO TEST COMPLETED.
11:14/#J828/72/FROM/MANAGER.SYS/QUICK VALIDATION TEST COMPLETED.
11:14/#J828/72/LOGOFF ON LDEV #10
```

If any of the tests do not complete, a highlighted message is displayed on the console. For example:

```
11:14/#J828/72/FROM/MANAGER.SYS/REMOTE RPMKILL FAILED
```

This is an example of the job stream produced by QuickVal.

```
:JOB QVAL,MANAGER.SYS,NET
PRIORITY = CS; HIPRI; TIME = UNLIMITED SECONDS
JOB NUMBER = #J828
FRI, MAR  8, 1985, 11:12 AM
HP3000 / MPE V  G.XX.XX (BASE G.XX.XX).
*** DAVIS NODE 6, IND NETWORK TEST CENTER, CUPERTINO, CA BL/GR
:
:COMMENT +-------------------------------------------------------------+
:COMMENT + Job for a quick validation of NS/3000.                      +
:COMMENT +                                                             +
:COMMENT + (These comments are contained in QVALTEMP.NET.SYS and      +
:COMMENT +  QVALJOB.NET.SYS.)                                          +
:COMMENT +                                                             +
:COMMENT +  Requirements:                                              +
:COMMENT +   (1) Program files: BLDFILE.NET.SYS, CMPFILE.NET.SYS       +
:COMMENT +                      QVRPM.NET.SYS, DSTEST.PUB.SYS          +
:COMMENT +                      PRPM0002.NET.SYS, QVAL.NET.SYS         +
:COMMENT +   (2) Accounts     : MANAGER.SYS,NET (local and remote)     +
:COMMENT +                      (no passwords)                         +
:COMMENT +                                                             +
:COMMENT +  Execution:  :Run QVAL.NET.SYS;info="nodename"              +
:COMMENT +              where nodename is your destination node name.  +
:COMMENT +              This program will read a "template" file       +
:COMMENT +              called QVALTEMP.NET.SYS and substitute the     +
:COMMENT +              given node name for a control character        +
:COMMENT +              sequence.  The resulting file is saved to      +
:COMMENT +              QVALJOB.NET.SYS.  The program then streams     +
:COMMENT +              this job, which constitutes the actual quick   +
:COMMENT +              validation test.                               +
:COMMENT +                                                             +
:COMMENT +  Description:                                               +
:COMMENT +   This job executes 3 DSCOPY transfers:                     +
:COMMENT +    (1) Local transfer with a fixed binary file (medium size)+
:COMMENT +    (2) Remote transfer using an ENVID in interchange mode   +
:COMMENT +        with a fixed ascii file (small file, large records)  +
:COMMENT +    (3) Remote transfer using a logon spec in transparent    +
:COMMENT +        mode with a variable ascii file (large file, small   +
:COMMENT +        records)                                             +
:COMMENT +                                                             +
:COMMENT +   The job then executes QVRPM.NET.SYS test RPM.  One local  +
:COMMENT +    create and one remote create are done.  The process      +
:COMMENT +    created is PRPM0002.NET.SYS.                             +
:COMMENT +                                                             +
:COMMENT +   The job then executes a DSLINE command and a Remote Hello +
:COMMENT +   with an environment id specified.                         +
:COMMENT +   DSTEST is then executed to check RFA and PTOP.            +
:COMMENT +   Then a Remote Hello - DSLINE combination is tested.       +
```

QuickVal Job Stream (Continued)

```
:COMMENT +                                                             +
:COMMENT +   ****************  CAUTION !!!!! **************            +
:COMMENT +   1.  Do not run more than one instance of this test        +
:COMMENT +       (or any other Quick Validation test)                  +
:COMMENT +       at a time.  They will ALL fail!!!                     +
:COMMENT +                                                             +
:COMMENT +    FAFILE,FAFILE1,FAFILE2;  FBFILE,FBFILE1;  VAFILE,VAFILE1 +
:COMMENT +------------------------------------------------------------+
```

QuickVal Test

```
:
:TELLOP NETWORK QUICK VALIDATION TEST.
:TELLOP BEGIN BUILDING FILES FOR NFT TEST.
:
:PURGE FBFILE
:PURGE FBFILE1
       ^
FILE FBFILE1.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:PURGE FAFILE
:PURGE FAFILE1
       ^
FILE FAFILE1.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:PURGE FAFILE2
       ^
FILE FAFILE2.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:PURGE VAFILE
:PURGE VAFILE1
       ^
FILE VAFILE1.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:PURGE VAFILE2
       ^
FILE VAFILE2.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:
:RUN BLDFILE

 ENTER FILE CHARACTERISTICS
 FILENAME, [TYPE(S/K/R/M/C),
            FILE SIZE,           #RECORDS-TO-FILL,
            RECORD SIZE (BYTES), BLOCK FACTOR,
            RECORD TYPE(F/V/U),  DATA TYPE(A/B),
            #USER LABELS,        FILE CODE]
FBFILE,S,101,100,200,,F,B,25

END OF PROGRAM
:
:RUN BLDFILE

 ENTER FILE CHARACTERISTICS
 FILENAME, [TYPE(S/K/R/M/C),
            FILE SIZE,           #RECORDS-TO-FILL,
            RECORD SIZE (BYTES), BLOCK FACTOR,
            RECORD TYPE(F/V/U),  DATA TYPE(A/B),
            #USER LABELS,        FILE CODE]
FAFILE,S,10,9,10000,,F,A

END OF PROGRAM
:
```

QuickVal Test (Continued)

```
:RUN BLDFILE

 ENTER FILE CHARACTERISTICS
 FILENAME, [TYPE(S/K/R/M/C),
            FILE SIZE,           #RECORDS-TO-FILL,
            RECORD SIZE (BYTES), BLOCK FACTOR,
            RECORD TYPE(F/V/U),  DATA TYPE(A/B),
            #USER LABELS,        FILE CODE]
VAFILE,S,10000,10000,9,,V,A

END OF PROGRAM
:
:DSLINE NFTENV=NODE6
ENVIRONMENT 1: NFTENV.ADS.TOPO02=NODE6.ADS.TOPO02
:REMOTE HELLO MANAGER.SYS,NET
HP3000 / MPE V  G.XX.XX (BASE G.XX.XX).  FRI, MAR  8, 1985, 11:12 AM
*** DAVIS NODE 6, IND NETWORK TEST CENTER, CUPERTINO, CA BL/GR
:
:CONTINUE
:REMOTE PURGE FAFILE1
        ^
FILE FAFILE1.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:
:CONTINUE
:REMOTE PURGE VAFILE1
        ^
FILE VAFILE1.NET.SYS NOT FOUND, NO PURGE DONE. (CIWARN 383)
:
:TELLOP WAIT FOR MESSAGE SHOWING SUCCESSFUL COMPLETION OF QUICK VAL!
:TELLOP IF YOU DON'T GET ONE, STUDY SPOOL FILE FOR THIS JOB CAREFULLY.
```

NFT Test

```
:TELLOP BEGIN NFT TEST.
:CONTINUE
:DSCOPY FBFILE; FBFILE1
  SOURCE FILE: FBFILE.NET.SYS
  TARGET FILE: FBFILE1.NET.SYS
  100 LOGICAL RECORD(S) TRANSFERRED.

END OF SUBSYSTEM
:
:CONTINUE
:DSCOPY
NETWORK FILE TRANSFER [HP32344A0000003] (C) HEWLETT-PACKARD CO. 1985

DSCOPY + INT

DSCOPY FAFILE TO FAFILE1 :NFTENV
  SOURCE FILE: FAFILE.NET.SYS
  TARGET FILE: FAFILE1.NET.SYS
  9 LOGICAL RECORD(S) TRANSFERRED.

DSCOPY FAFILE1 :NFTENV TO FAFILE2
  SOURCE FILE: FAFILE1.NET.SYS
  TARGET FILE: FAFILE2.NET.SYS
  9 LOGICAL RECORD(S) TRANSFERRED.

DSCOPY + CLEAR

DSCOPY VAFILE TO VAFILE1 [MANAGER.SYS,NET],NODE6
  SOURCE FILE: VAFILE.NET.SYS
  TARGET FILE: VAFILE1.NET.SYS
  10000 LOGICAL RECORD(S) TRANSFERRED.

DSCOPY VAFILE1 [MANAGER.SYS,NET],NODE6 TO VAFILE2
  SOURCE FILE: VAFILE1.NET.SYS
  TARGET FILE: VAFILE2.NET.SYS
  10000 LOGICAL RECORD(S) TRANSFERRED.

DSCOPY //

END OF SUBSYSTEM
:
:TELLOP CHECK LOCAL TRANSFER
:FILE COMPR1=FBFILE
:FILE COMPR2=FBFILE1
:RUN CMPFILE

NUMBER OF LABELS COMPARED: 25
NUMBER OF RECORDS COMPARED: 100
FILES COMPARED OK

END OF PROGRAM
```

NFT Test (Continued)

```
:IF JCW > WARN THEN
*** EXPRESSION FALSE: COMMANDS IGNORED UNTIL MATCHING ENDIF/ELSE
: TELLOP NFT LOCAL TRANSFER FAILED
:ELSE
*** RESUME EXECUTION OF COMMANDS
: PURGE FBFILE1
: PURGE FBFILE
:ENDIF
:
:TELLOP CHECK REMOTE INTERCHANGE MODE TRANSFER
:FILE COMPR1=FAFILE
:FILE COMPR2=FAFILE2
:RUN CMPFILE;PARM=1

NUMBER OF RECORDS COMPARED: 9
FILES COMPARED OK

END OF PROGRAM
:IF JCW > WARN THEN
*** EXPRESSION FALSE: COMMANDS IGNORED UNTIL MATCHING ENDIF/ELSE
: TELLOP NFT INTERCHANGE MODE TRANSFER FAILED (REMOTE SESSION)
:ELSE
*** RESUME EXECUTION OF COMMANDS
: REMOTE PURGE FAFILE1
: PURGE FAFILE2
: PURGE FAFILE
:ENDIF
:
:TELLOP CHECK REMOTE TRANSPARENT MODE TRANSFER
:FILE COMPR1=VAFILE
:FILE COMPR2=VAFILE2
:RUN CMPFILE

NUMBER OF RECORDS COMPARED: 10000
FILES COMPARED OK

END OF PROGRAM
:IF JCW > WARN THEN
*** EXPRESSION FALSE: COMMANDS IGNORED UNTIL MATCHING ENDIF/ELSE
: TELLOP NFT REMOTE TRANSPARENT MODE TRANSFER FAILED (NFT LOGON)
:ELSE
*** RESUME EXECUTION OF COMMANDS
: PURGE VAFILE2
: REMOTE PURGE VAFILE1
: PURGE VAFILE
:ENDIF
:
:TELLOP NFT TEST COMPLETED.
```

RPM Test

```
:TELLOP BEGIN RPM TEST
:
:RUN QVRPM.NET.SYS;INFO="NODE6";PARM=5
QVRPM [HP32344A0000000] (C) Hewlett-Packard Co. 1985
Non-zero results indicate an RPM error.
About to attempt local RPM Create:
Local RPM Create result =           0
About to attempt local RPM Kill:
Local RPM Kill result =           0
About to attempt remote RPM Create:
Remote RPM Create result =           0
About to attempt remote RPM Kill:
Remote RPM Kill result =           0
END OF PROGRAM
:
:TELLOP RPM TEST COMPLETED.
:
```

ENVID Test

```
:DSLINE @; CLOSE
REMOTE SESSION ABORTED.
ENVIRONMENT 1: NFTENV.ADS.TOP002=NODE6.ADS.TOP002
:REMOTE :YS1=NODE6  HELLO MANAGER.SYS,NET
HP3000 / MPE V  G.XX.XX (BASE G.XX.XX).  FRI, MAR  8, 1985, 11:14 AM
*** DAVIS NODE 6, IND NETWORK TEST CENTER, CUPERTINO, CA BL/GR
ENVIRONMENT 1: YS1.ADS.TOP002=NODE6.ADS.TOP002
:REMOTE :YS1 SHOWME
USER: #S1304,MANAGER.SYS,NET      (NOT IN BREAK)
MPE VERSION: HP32033G.XX.XX.  (BASE G.XX.XX).
CURRENT: FRI, MAR  8, 1985, 11:14 AM
LOGON:   FRI, MAR  8, 1985, 11:14 AM
CPU SECONDS: 1          CONNECT MINUTES: 1
$STDIN LDEV: 60          $STDLIST LDEV: 60
*** DAVIS NODE 6, IND NETWORK TEST CENTER, CUPERTINO, CA BL/GR
```

RFA and PTOP Test

```
:TELLOP BEGIN RFA and PTOP TEST
:FILE REMOTE; DEV=YS1#DISC
:RUN DSTEST.PUB.SYS,DIAG

HEWLETT PACKARD 32189B.51.03 DSTEST/3000 FRI, MAR  8, 1985, 11:14 AM
.RFA OR PTOP? RFA
.REMOTE COMPUTER? 3000
.NUMBER OF PASSES? 100
.PATTERN?
.BLOCK SIZE?
512 WORD REMOTE RECS WRITTEN/READ: 100 ,SECS: 10.354 ,AVE: .103
.CONTINUE(Y/N)? Y
.RFA OR PTOP? PTOP
.DSLINE? YS1
.NUMBER OF PASSES? 100
.PATTERN?
.BLOCK SIZE?
END OF REMOTE PROGRAM.
512 WORD PROG TO PROG WRITES DONE: 100 ,SECS: 7.471 ,AVE: .074
.CONTINUE(Y/N)? N

END OF PROGRAM
:
:TELLOP RFA and PTOP TEST COMPLETED.
:
```

DSLINE and REMOTE HELLO Test

```
:TELLOP BEGIN DSLINE, REMOTE HELLO TEST.
:COMMENT SS.3.12
:DSLINE; CLOSE
REMOTE SESSION ABORTED.
ENVIRONMENT 1: YS1.ADS.TOP002=NODE6.ADS.TOP002
:COMMENT SS.5.1
:REMOTE HELLO MANAGER.SYS,NET; DSLINE= YS1= NODE6
HP3000 / MPE V  G.XX.XX (BASE G.XX.XX).  FRI, MAR  8, 1985, 11:14 AM
*** DAVIS NODE 6, IND NETWORK TEST CENTER, CUPERTINO, CA BL/GR
ENVIRONMENT 1: YS1.ADS.TOP002=NODE6.ADS.TOP002
:REMOTE :YS1 SHOWME
USER: #S1305,MANAGER.SYS,NET      (NOT IN BREAK)
MPE VERSION: HP32033G.XX.XX.  (BASE G.XX.XX).
CURRENT: FRI, MAR  8, 1985, 11:14 AM
LOGON:   FRI, MAR  8, 1985, 11:14 AM
CPU SECONDS: 1         CONNECT MINUTES: 1
$STDIN LDEV: 60         $STDLIST LDEV: 60
*** DAVIS NODE 6, IND NETWORK TEST CENTER, CUPERTINO, CA BL/GR
:REMOTE :YS1 BYE
CPU=1. CONNECT=1. FRI, MAR  8, 1985, 11:14 AM
:TELLOP DSLINE, REMOTE HELLO TEST COMPLETED.
:TELLOP NETWORK QUICK VALIDATION TEST COMPLETED.
:
:EOJ
CPU SEC. = 42.  ELAPSED MIN. = 3.  FRI, MAR  8, 1985, 11:14 AM
```

NODESTAT

NODESTAT is an interactive menu-driven tool that provides information about a network node's status. You must have PM, NA, NM and IA capabilities to use NODESTAT.

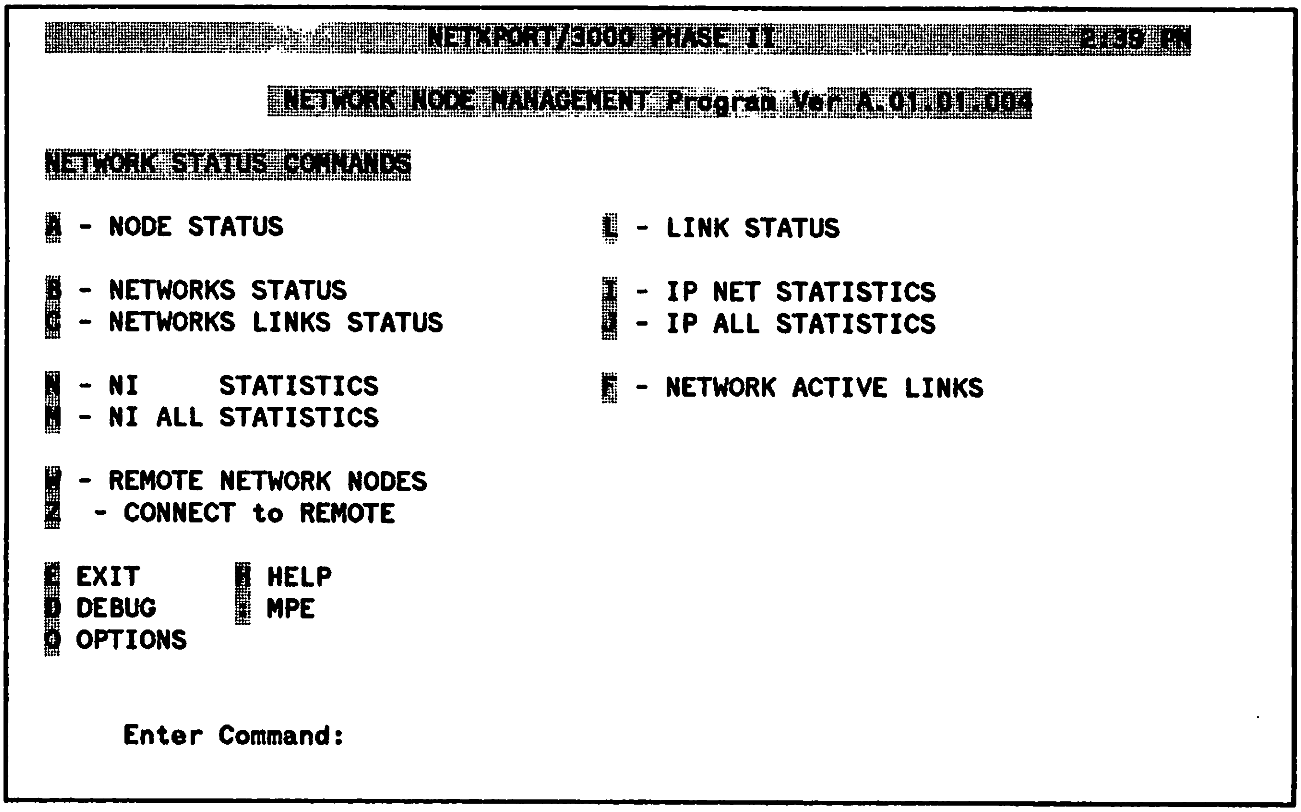
NOTE

Do not run NODESTAT while the network is being started.

To run NODESTAT, enter:

```
:RUN NODESTAT.NET.SYS
```

The following main menu screen is displayed:



```
                    NETXPORT/3000 PHASE II                      2:39 PM

          NETWORK NODE MANAGEMENT Program Ver A.01.01.004

NETWORK STATUS COMMANDS

A - NODE STATUS                      L - LINK STATUS

B - NETWORKS STATUS                  I - IP NET STATISTICS
C - NETWORKS LINKS STATUS            J - IP ALL STATISTICS

N - NI     STATISTICS                F - NETWORK ACTIVE LINKS
M - NI ALL STATISTICS

W - REMOTE NETWORK NODES
Z  - CONNECT to REMOTE

E EXIT      H HELP
D DEBUG     : MPE
O OPTIONS


     Enter Command:
```

Figure 2-3. Main Menu of Nodestat Service

On the command line you can enter a letter corresponding to the statistics/status you want to examine. The other options provided are:

E — EXIT, to exit from the NODESTAT program.

D — DEBUG, allows you access to the MPE-Debug facility.

H — HELP, provides help information.

: — MPE, allows you to use MPE commands from NODESTAT (commands executable during break.)

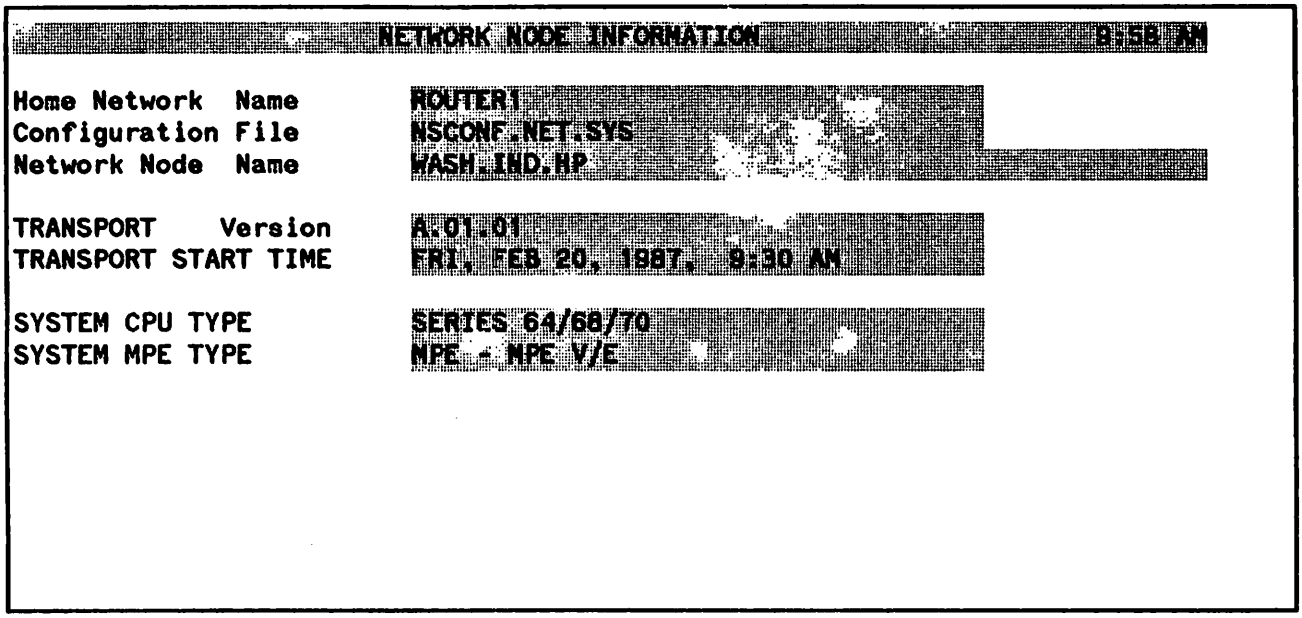
O — OPTIONS, provides NODESTAT program options. Allows you to specify a time interval to update the statistics displayed when you press RETURN.

Any of the commands listed on the main menu can be executed from any other screen. Pressing RETURN updates the displayed statistics. Pressing the space bar from a screen returns you to the main menu screen.

The following figures provide sample displays of each of the options available from the NODESTAT main menu screen. User input (if any) is also described.

A: Node Status

The following information is displayed for the node:

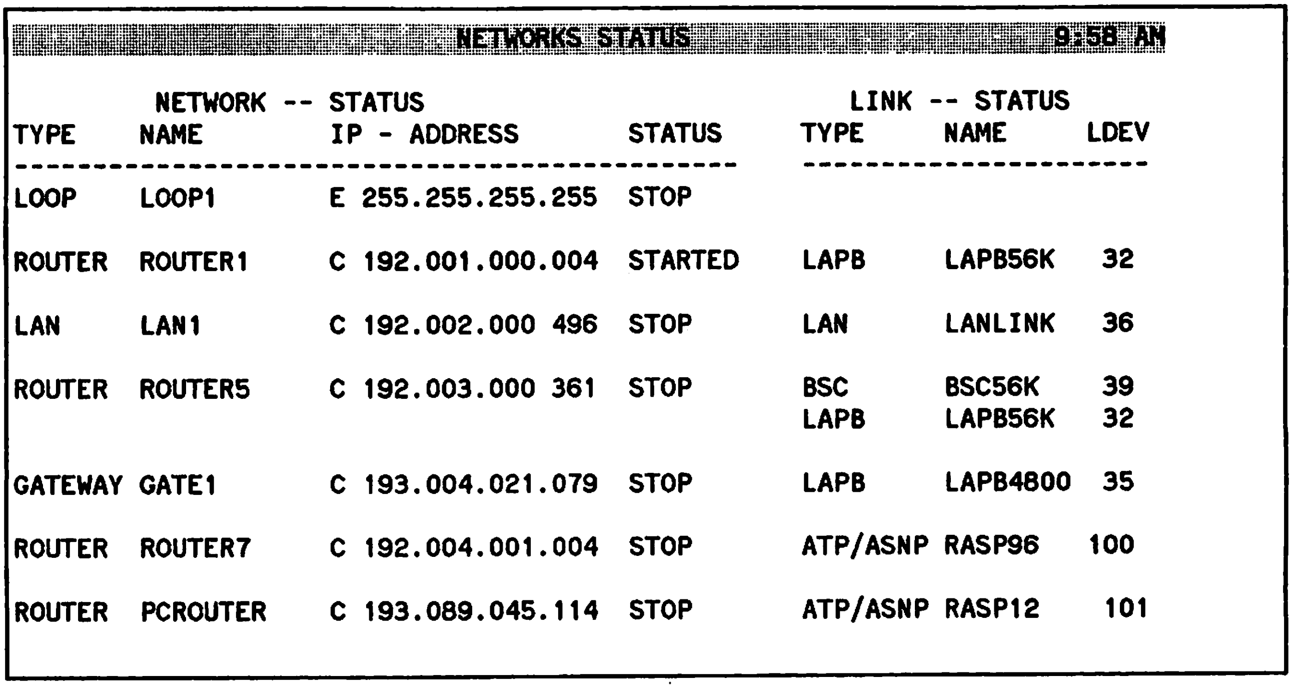


```
                          NETWORK NODE INFORMATION                        8:58 AM

Home Network  Name        ROUTER1
Configuration File        NSCONF.NET.SYS
Network Node  Name        WASH.IND.HP

TRANSPORT    Version      A.01.01
TRANSPORT START TIME      FRI, FEB 20, 1987,  8:30 AM

SYSTEM CPU TYPE           SERIES 64/68/70
SYSTEM MPE TYPE           MPE - MPE V/E
```

Figure 2-4. Network Node Information Screen

B: Networks Status

The status of the configured network from the active network configuration file is displayed.

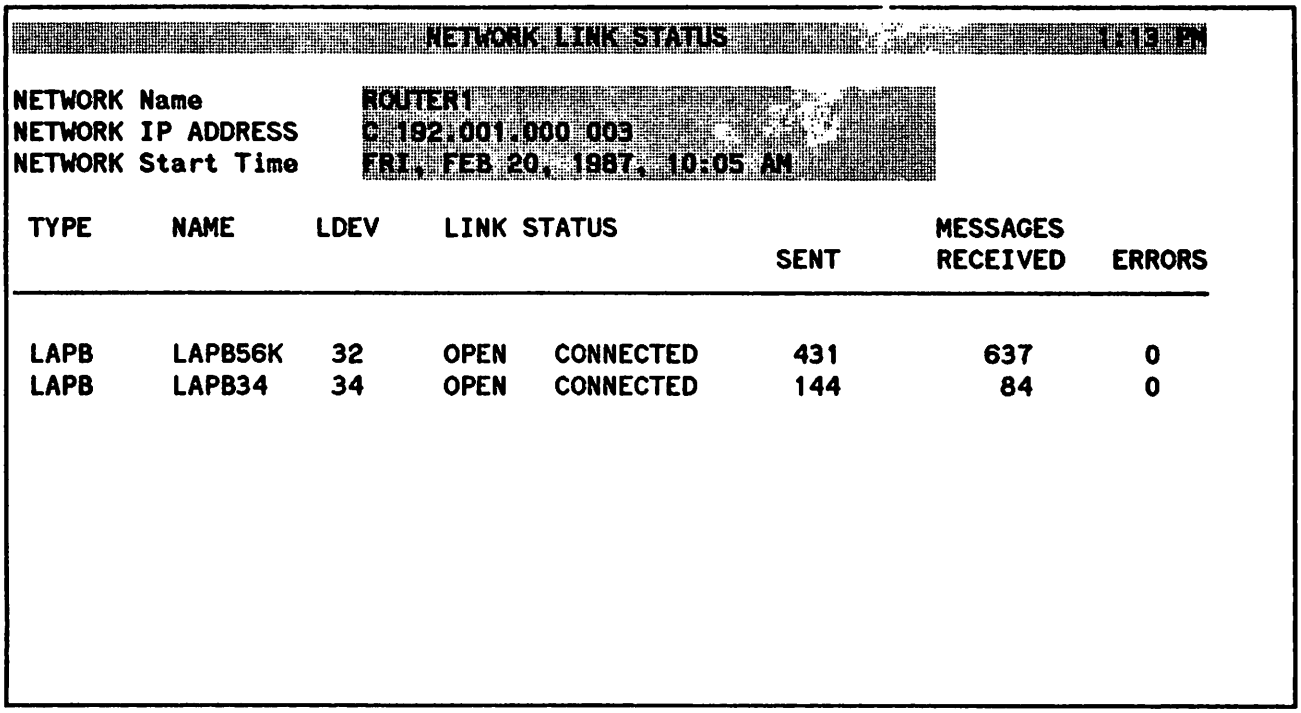


```
                           NETWORKS STATUS                        9:58 AM

         NETWORK -- STATUS                               LINK -- STATUS
TYPE     NAME      IP - ADDRESS        STATUS        TYPE       NAME      LDEV
----------------------------------------------       ----------------------
LOOP     LOOP1     E 255.255.255.255   STOP

ROUTER   ROUTER1   C 192.001.000.004   STARTED       LAPB       LAPB56K    32

LAN      LAN1      C 192.002.000 496   STOP          LAN        LANLINK    36

ROUTER   ROUTER5   C 192.003.000 361   STOP          BSC        BSC56K     39
                                                     LAPB       LAPB56K    32

GATEWAY  GATE1     C 193.004.021.079   STOP          LAPB       LAPB4800   35

ROUTER   ROUTER7   C 192.004.001.004   STOP          ATP/ASNP   RASP96    100

ROUTER   PCROUTER  C 193.089.045.114   STOP          ATP/ASNP   RASP12     101
```

Figure 2-5. Networks Status Screen

C: Network Links Status

At the prompt, enter a valid network name to obtain the following displayed information.



```
                    NETWORK LINK STATUS                           1:13 PM

NETWORK Name          ROUTER1
NETWORK IP ADDRESS    C 192.001.000 003
NETWORK Start Time    FRI, FEB 20, 1987, 10:05 AM

 TYPE     NAME     LDEV    LINK STATUS                      MESSAGES
                                                 SENT       RECEIVED    ERRORS
------------------------------------------------------------------------------

 LAPB     LAPB56K   32     OPEN   CONNECTED      431           637         0
 LAPB     LAPB34    34     OPEN   CONNECTED      144            84         0
```

Figure 2-6. Network Link Status Screen

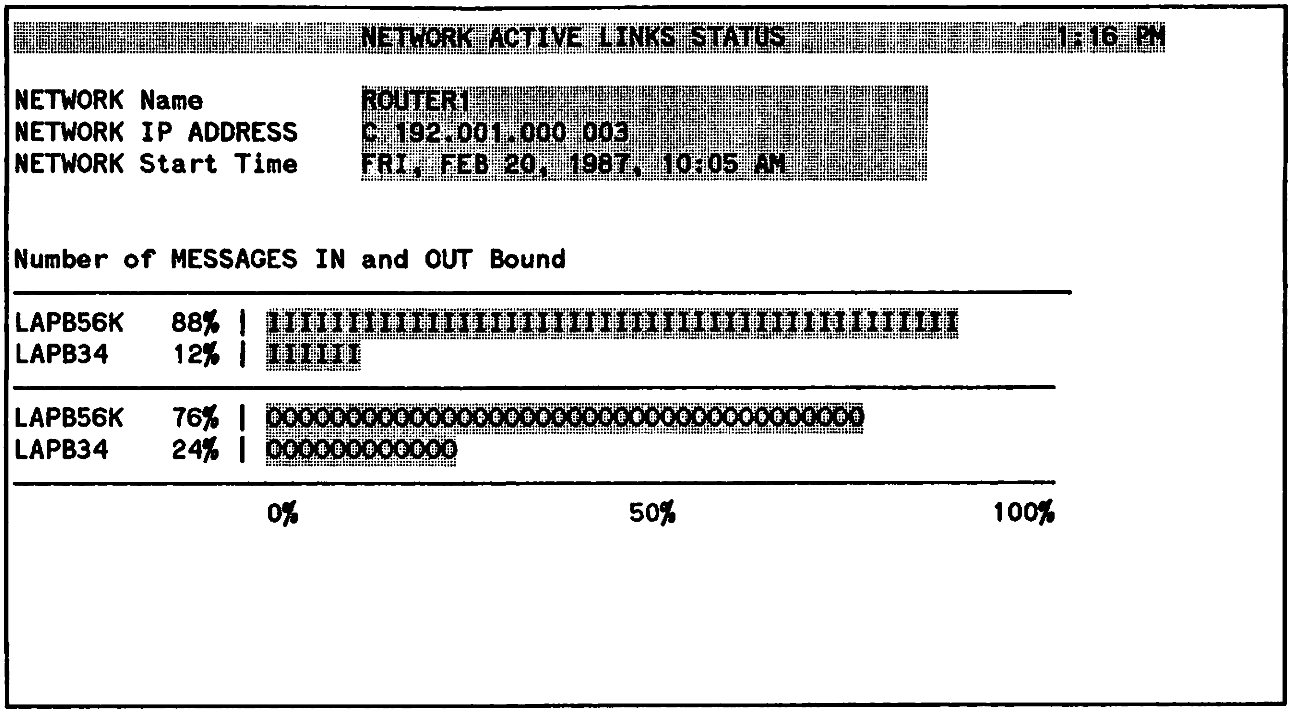
SENT — Number of messages sent as displayed with SHOWCOM.

MESSAGES RECEIVED — Number of messages received as displayed with SHOWCOM.

ERRORS — Number of recoverable errors.

F: Network Active Links

At the prompt, enter a network name to obtain the following displayed information.



```
                      NETWORK ACTIVE LINKS STATUS                    1:16 PM

NETWORK Name          ROUTER1
NETWORK IP ADDRESS    C 192.001.000 003
NETWORK Start Time    FRI, FEB 20, 1987, 10:05 AM


Number of MESSAGES IN and OUT Bound
------------------------------------------------------------------------------
LAPB56K   88% | IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
LAPB34    12% | IIIIII
------------------------------------------------------------------------------
LAPB56K   76% | OOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO
LAPB34    24% | OOOOOOOOOOOO
------------------------------------------------------------------------------
                0%                       50%                       100%
```

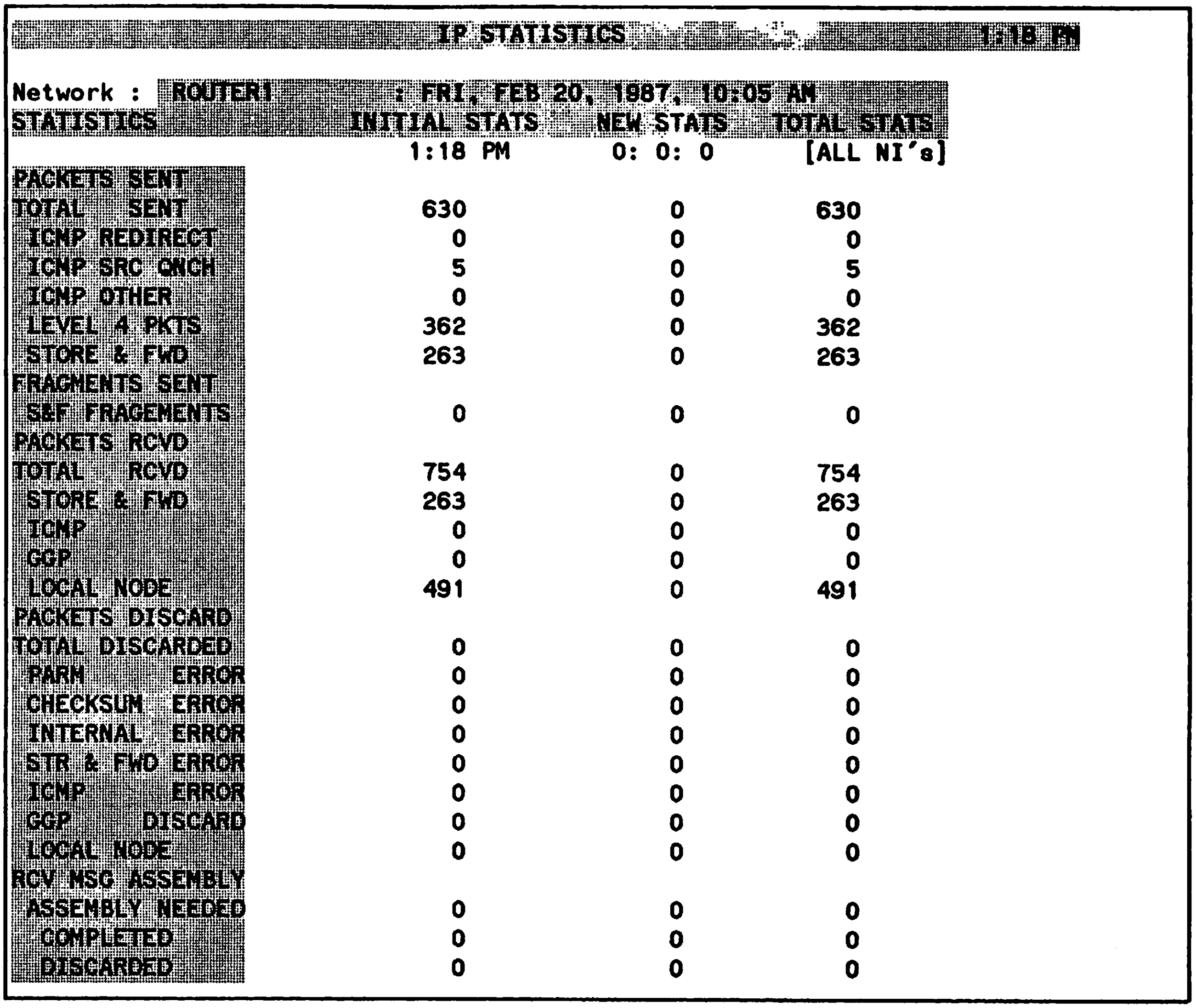
Figure 2-7. Network Active Links Status Screen

This screen shows the percentage of activity of each link, based on the total activity of the network.

```
I  Indicates the percentage of the total inbound messages
   for the network.

O  Indicates the percentage of the total outbound messages
   for the network.
```

I: IP Net Statistics

At the prompt, enter a valid network interface (NI) name to obtain the following information.



```
                              IP STATISTICS                                 1:18 PM

Network :  ROUTER1          : FRI, FEB 20, 1987, 10:05 AM
STATISTICS               INITIAL STATS    NEW STATS    TOTAL STATS
                            1:18 PM        0: 0: 0      [ALL NI's]
PACKETS SENT
TOTAL    SENT                 630             0            630
 ICMP REDIRECT                  0             0              0
 ICMP SRC QNCH                  5             0              5
 ICMP OTHER                     0             0              0
 LEVEL 4 PKTS                 362             0            362
 STORE & FWD                  263             0            263
FRAGMENTS SENT
 S&F FRAGEMENTS                 0             0              0
PACKETS RCVD
TOTAL    RCVD                 754             0            754
 STORE & FWD                  263             0            263
 ICMP                           0             0              0
 GGP                            0             0              0
 LOCAL NODE                   491             0            491
PACKETS DISCARD
TOTAL DISCARDED                 0             0              0
 PARM      ERROR                0             0              0
 CHECKSUM  ERROR                0             0              0
 INTERNAL  ERROR                0             0              0
 STR & FWD ERROR                0             0              0
 ICMP      ERROR                0             0              0
 GGP     DISCARD                0             0              0
 LOCAL NODE                     0             0              0
RCV MSG ASSEMBLY
 ASSEMBLY NEEDED                0             0              0
  COMPLETED                     0             0              0
  DISCARDED                     0             0              0
```

Figure 2-8. IP Statistics Screen

INITIAL STATS — Provides statistics for the specified NI.

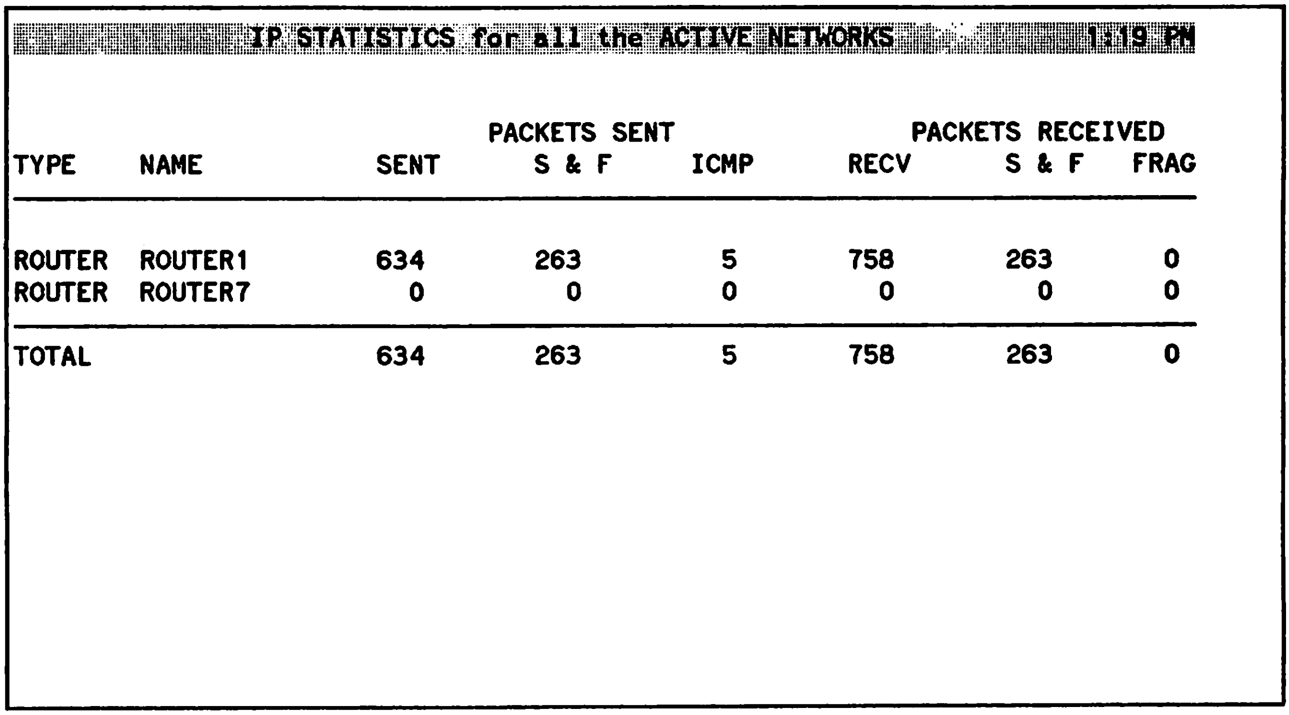
NEW STATS — Indicates the change in statistics since the function was called.

TOTAL STATS — Provides statistics totals for all NIs.

The time under NEW STATS rolls over after 24 hours.

J: IP All Statistics

The IP statistics screen provides a summary of IP statistics for all active networks.

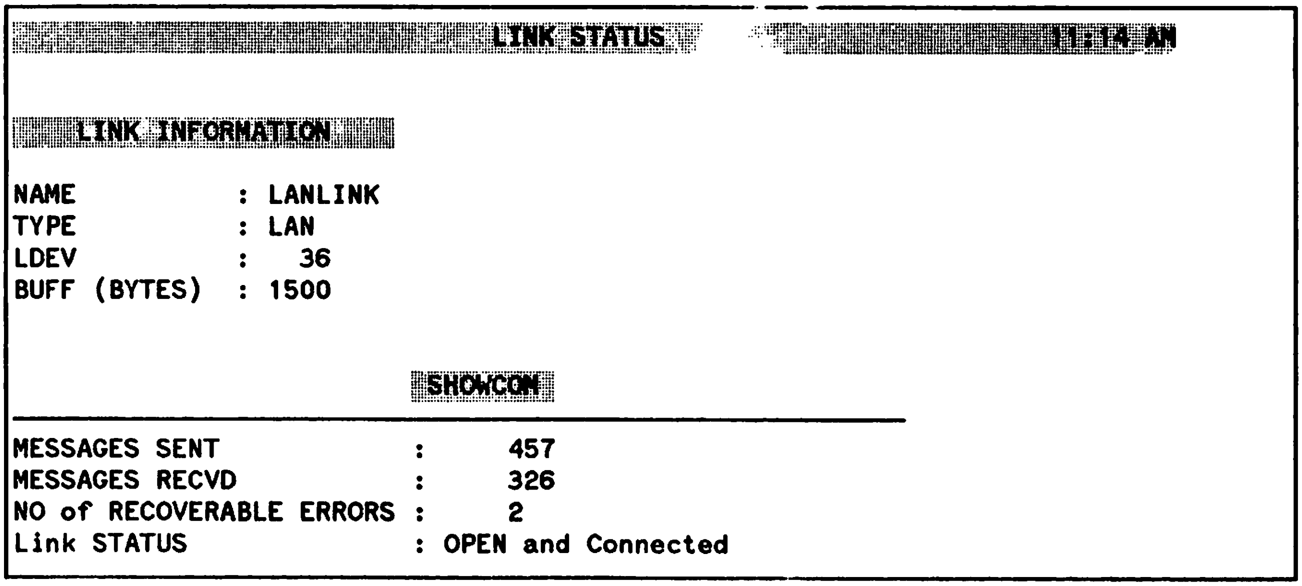


```
          IP STATISTICS for all the ACTIVE NETWORKS                 1:19 PM


                                   PACKETS SENT                 PACKETS RECEIVED
TYPE    NAME              SENT        S & F      ICMP       RECV        S & F    FRAG
_____________________________________________________________________________________

ROUTER  ROUTER1            634         263          5        758         263        0
ROUTER  ROUTER7              0           0          0          0           0        0
_____________________________________________________________________________________
TOTAL                      634         263          5        758         263        0
```

Figure 2-9. IP Statistics for all the Active Networks Screen

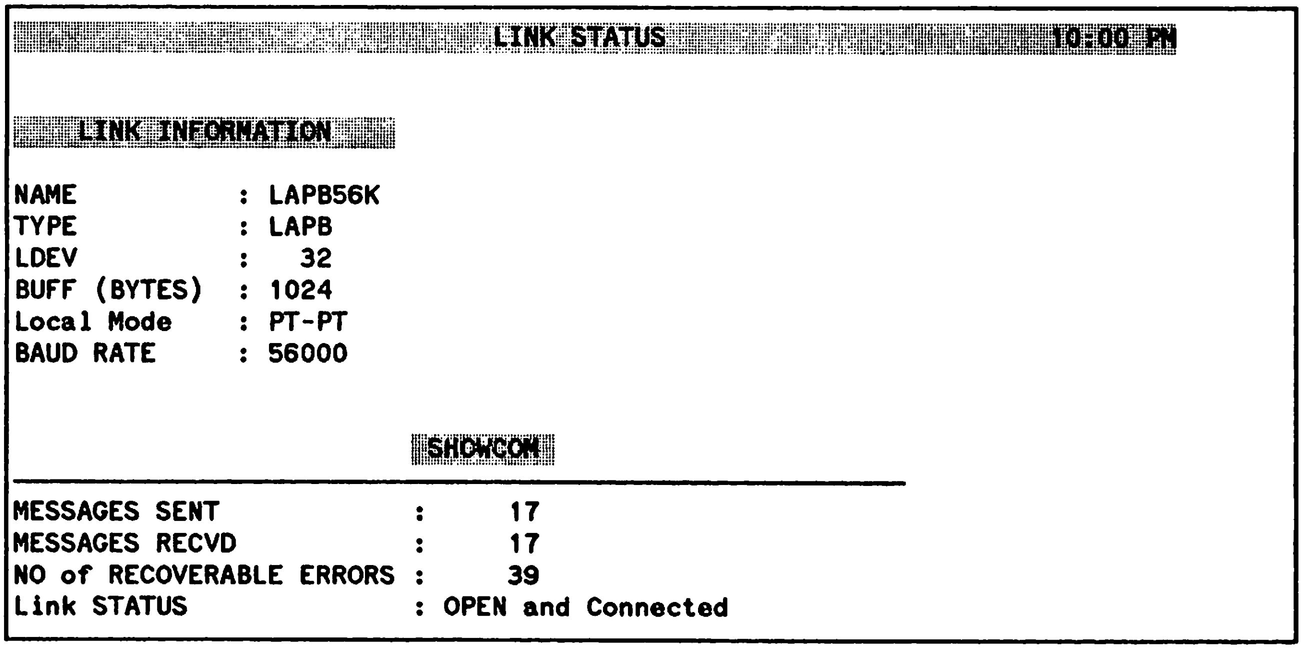
L: Link Status

At the prompt, enter a valid link name. Information displayed on the link status screen is specific to the type of link. Figures 2-10, 2-11, 2-12 and 2-13 show examples for IEEE 802.3, LAP-B, BSC and ASNP links in that order.



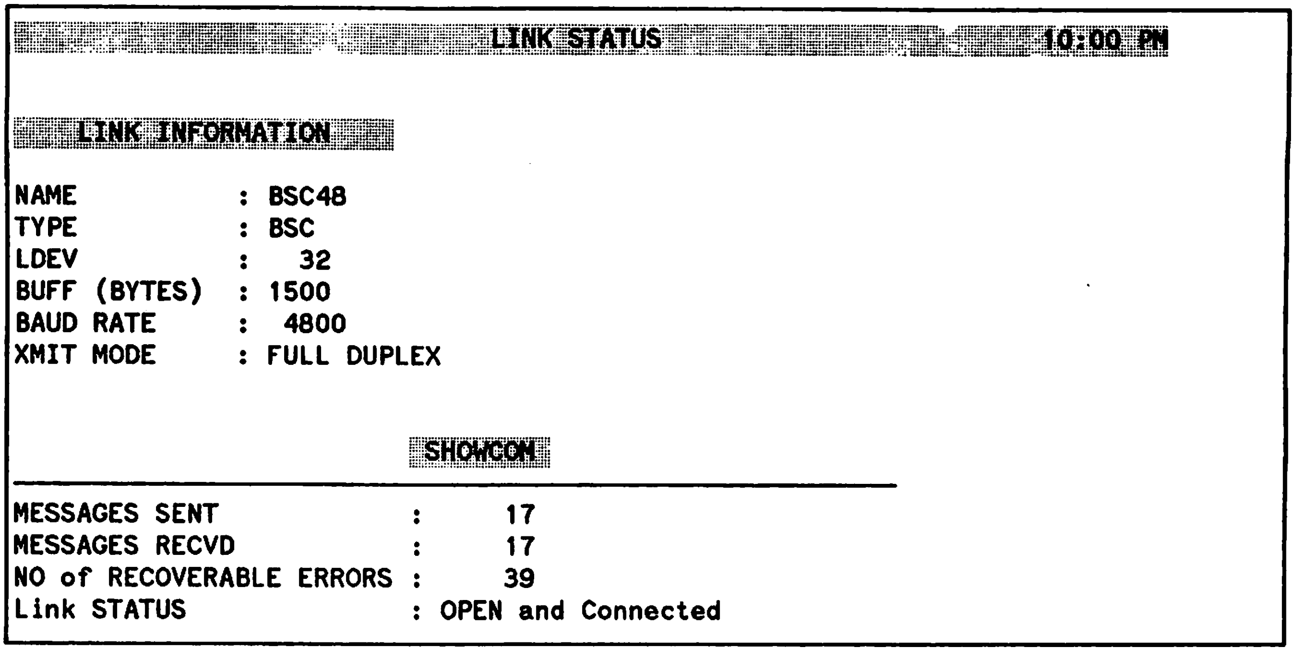
```
                                  LINK STATUS                        11:14 AM


LINK INFORMATION

NAME            : LANLINK
TYPE            : LAN
LDEV            :   36
BUFF (BYTES)    : 1500


                                   SHOWCOM
-----------------------------------------------------------
MESSAGES SENT               :      457
MESSAGES RECVD              :      326
NO of RECOVERABLE ERRORS    :      2
Link STATUS                 : OPEN and Connected
```

Figure 2-10. IEEE 802.3 Link Status Screen



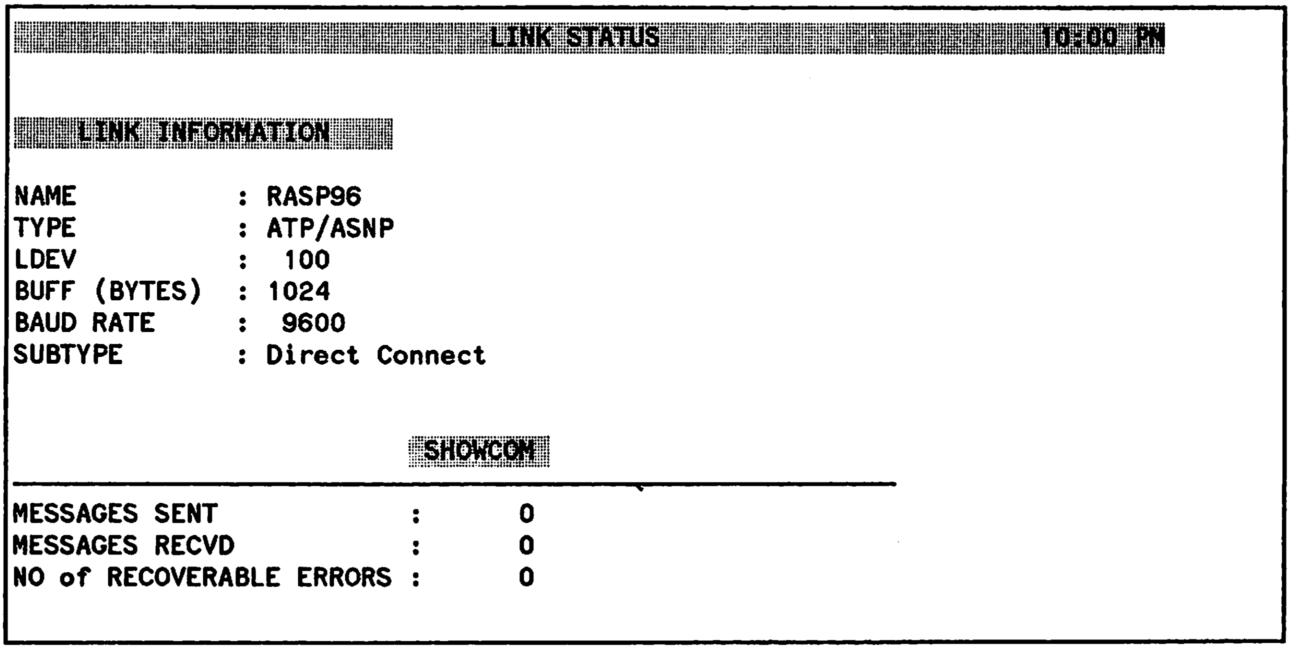
```
                                  LINK STATUS                        10:00 PM


LINK INFORMATION

NAME            : LAPB56K
TYPE            : LAPB
LDEV            :   32
BUFF (BYTES)    : 1024
Local Mode      : PT-PT
BAUD RATE       : 56000


                                   SHOWCOM
-----------------------------------------------------------
MESSAGES SENT               :      17
MESSAGES RECVD              :      17
NO of RECOVERABLE ERRORS    :      39
Link STATUS                 : OPEN and Connected
```

Figure 2-11. LAP-B Link Status Screen



```
                              LINK STATUS                       10:00 PM

 LINK INFORMATION

NAME          : BSC48
TYPE          : BSC
LDEV          :   32
BUFF (BYTES)  : 1500
BAUD RATE     :  4800
XMIT MODE     : FULL DUPLEX


                           SHOWCOM
________________________________________________________
MESSAGES SENT              :      17
MESSAGES RECVD             :      17
NO of RECOVERABLE ERRORS   :      39
Link STATUS                : OPEN and Connected
```

Figure 2-12. BSC Link Status Screen

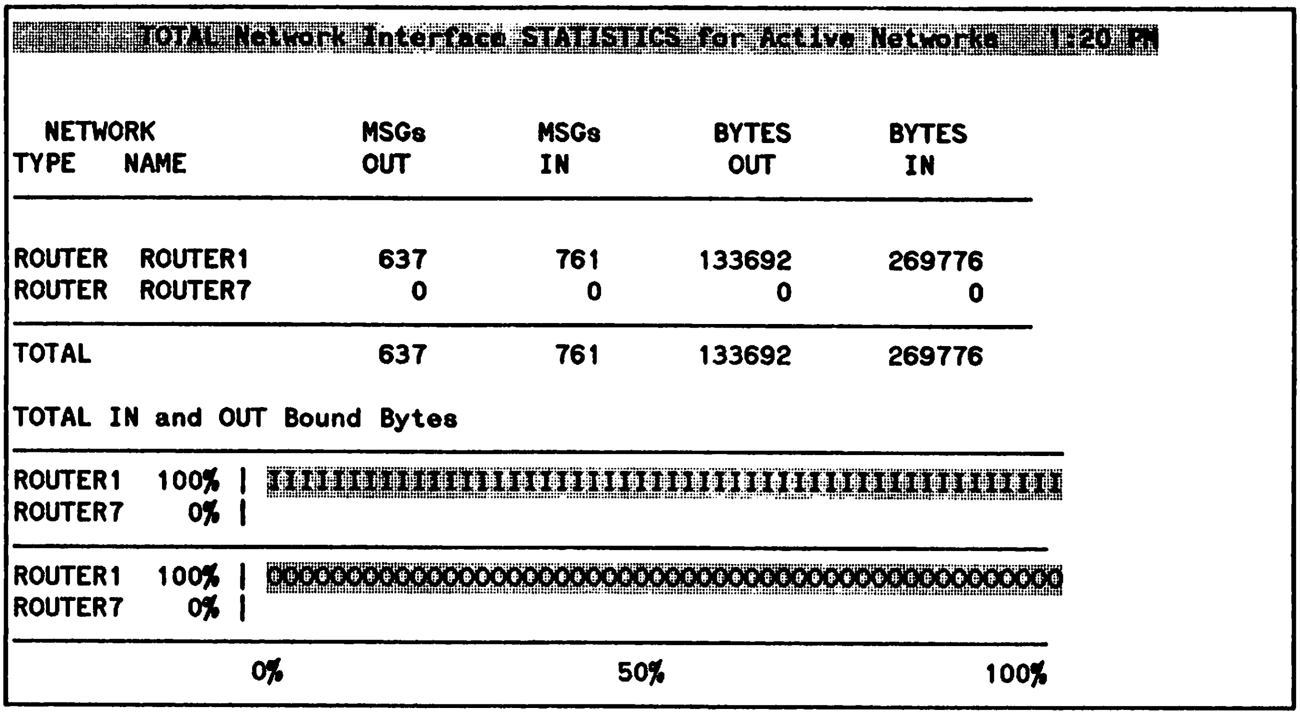


```
                              LINK STATUS                       10:00 PM

 LINK INFORMATION

NAME          : RASP96
TYPE          : ATP/ASNP
LDEV          :  100
BUFF (BYTES)  : 1024
BAUD RATE     :  9600
SUBTYPE       : Direct Connect


                           SHOWCOM
________________________________________________________
MESSAGES SENT              :       0
MESSAGES RECVD             :       0
NO of RECOVERABLE ERRORS   :       0
```

Figure 2-13. ASNP Link Status Screen

M: NI All Statistics

The total statistics screen provides statistics for all NI for the active network.



```
    TOTAL Network Interface STATISTICS for Active Networks   1:20 PM


  NETWORK                     MSGs       MSGs        BYTES       BYTES
TYPE    NAME                  OUT        IN          OUT          IN
----------------------------------------------------------------------

ROUTER   ROUTER1               637        761       133692      269776
ROUTER   ROUTER7                 0          0            0           0
----------------------------------------------------------------------
TOTAL                          637        761       133692      269776

TOTAL IN and OUT Bound Bytes
------------------------------------------------------------------------
ROUTER1   100% | IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
ROUTER7     0% |
------------------------------------------------------------------------
ROUTER1   100% | OOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO
ROUTER7     0% |
------------------------------------------------------------------------
                 0%                      50%                      100%
```

Figure 2-14. Total Statistics Screen

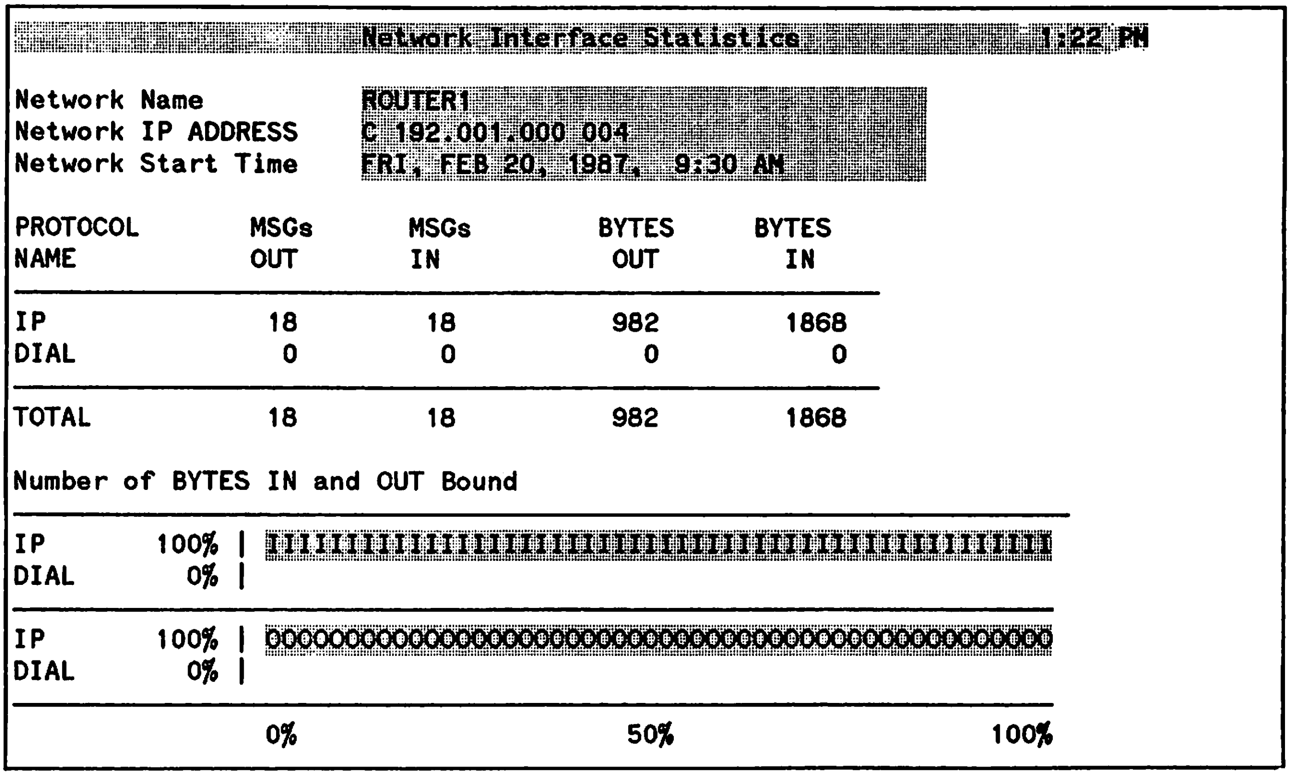
Total IN and OUT Bound Bytes shows the percentage of activity of each network based on the total activity of the node.

I Indicates the percentage of the total inbound bytes for the node.

O Indicates the percentage of the total outbound bytes for the node.

N: NI Statistics

At the prompt, enter a valid network name to obtain network interface statistics.



```
                              Network Interface Statistics                1:22 PM

Network Name             ROUTER1
Network IP ADDRESS       C 192.001.000 004
Network Start Time       FRI, FEB 20, 1987,  9:30 AM

PROTOCOL            MSGs       MSGs        BYTES       BYTES
NAME                OUT        IN          OUT         IN
------------------------------------------------------------
IP                    18         18          982        1868
DIAL                   0          0            0           0
------------------------------------------------------------
TOTAL                 18         18          982        1868

Number of BYTES IN and OUT Bound
------------------------------------------------------------------------
IP        100% | IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
DIAL        0% |
------------------------------------------------------------------------
IP        100% | OOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO
DIAL        0% |
------------------------------------------------------------------------
                 0%                      50%                      100%
```

Figure 2-15. Network Interface Statistics Screen

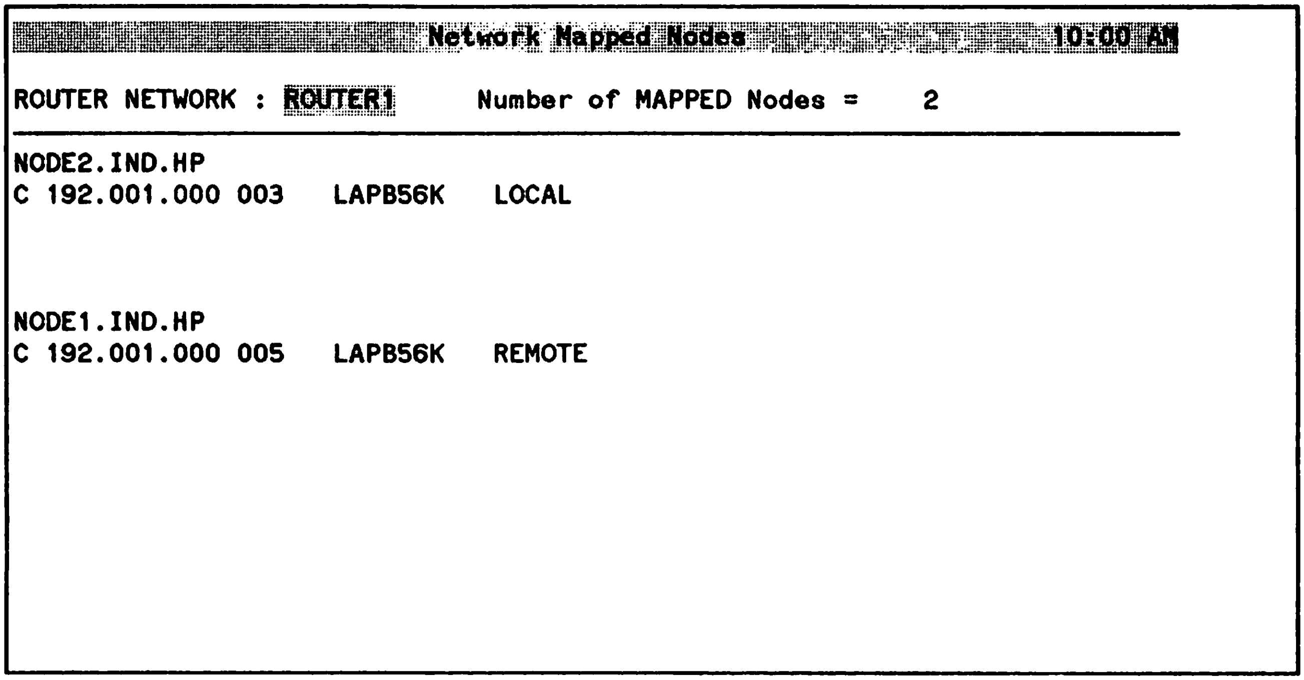
This screen shows the percentage of activity of the specified protocol based on the total activity of the network.

I Indicates the percentage of the total inbound bytes for the network.

O Indicates the percentage of the total outbound bytes for the network.

W: Remote Network Nodes

Enter a valid LAN or ROUTER type network name to obtain the following network statistics. For a LAN (IEEE 802.3 network), only remote nodes that have already communicated with the local node are displayed. For router networks (as in this example) nodes configured in each node's NETXPORT.NI.*niName*.MAPPING.*mapentry* configuration screens are displayed. All node names (if more than one was configured) are displayed for each node.



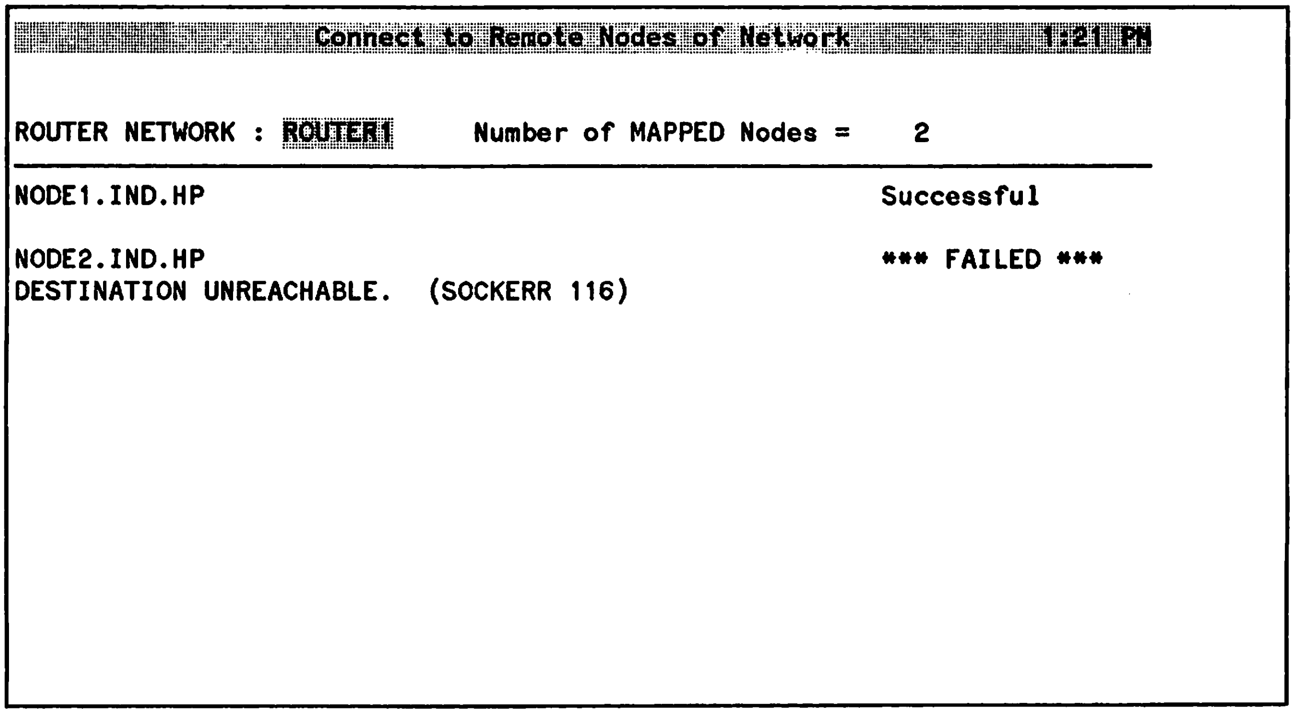
```
                         Network Mapped Nodes                      10:00 AM

ROUTER NETWORK : ROUTER1      Number of MAPPED Nodes =     2
------------------------------------------------------------------------
NODE2.IND.HP
C 192.001.000 003   LAPB56K   LOCAL


NODE1.IND.HP
C 192.001.000 005   LAPB56K   REMOTE
```

Figure 2-16. Network Mapped Nodes Screen

LOCAL corresponds to adjacent and REMOTE corresponds to nonadjacent as displayed in the Router Reachable Nodes Screen in NMMGR. The Number of MAPPED Nodes is the total number of nodes displayed.

Z: Connect To Remote

Enter a valid LAN or ROUTER type network name. This function attempts to connect you to all remote nodes accessible from a given network. For a LAN (IEEE 802.3 network), only remote nodes that have already communicated with the local node are displayed. For router networks (as in this example) nodes configured in NETXPORT.NI.*niName*.MAPPING.*mapentry* screens are displayed.



```
                 Connect to Remote Nodes of Network                1:21 PM

ROUTER NETWORK : ROUTER1      Number of MAPPED Nodes =    2
_____________________________________________________________________
NODE1.IND.HP                                            Successful

NODE2.IND.HP                                            *** FAILED ***
DESTINATION UNREACHABLE.  (SOCKERR 116)
```

Figure 2-17. Connect Remote Nodes of Network Screen

The Number of MAPPED Nodes is the total number of remote nodes mapped to the local node. This example shows one successful connection and one failed connection.

LOG AND TRACE FILES

SECTION

3

DIAGNOSTIC FUNCTIONS

HP AdvanceNet products provide both tracing and logging services for use as diagnostic and debugging aids. User-level tracing provides a record of data communications subsystem intrinsic calls. Internal level tracing records internal state transitions and the sequences of module execution within data communications subsystems. You should only use internal tracing under the recommendation of an HP service representative. Logging records subsystem events including normal events as well as error events. Use logging in problem determination and in monitoring network usage and resources. A diagnostic utility, NMDUMP, is provided to format trace and log files.

The Tracing Facility

Tracing is provided for the Network Services subsystem, Network IPC, the Network Transport subsystem, and the link subsystems.

Tracing for Network IPC applications is enabled with the NetIPC intrinsic IPCCONTROL, which is explained in the *NetIPC3000/V Programmer's Reference Manual.*

Tracing for the Network Services is enabled by the DSLINE command for each user's services. Network Service tracing is used to trace messages generated by your applications. For more information, see the *NS3000/V User/Programmer Reference Manual.*

Tracing for the Network Transport is selectively enabled with the NETCONTROL command as described in Section 1, "Commands," in this volume.

Tracing at the link level can be enabled in the NMMGR configuration for a particular link, as explained in Section 7, Link Configuration in Volume I of this manual set. Link level tracing can also be enabled with the LINKCONTROL command as described in Section 1, Commands of this volume.

Traces from Network Services, Network IPC, and Network Transport can be formatted with the NMDUMP formatting utility, as described later in this section.

Traces from link subsystems can be formatted with the CSDUMP formatting utility as described in the *LAN/3000 Diagnostic and Troubleshooting Reference Manual* and the *Fundamental Data Communications Handbook.*

Trace Files

Trace records are written to disc files and are of file type NTRAC. There are two ways that trace files are named. First, you can explicitly specify a file name (in the configuration file or with the NETCONTROL command). In this case, the contents of the file can be overwritten each time a new trace is started. No warning is issued. Secondly, you can allow the trace file name to default. NMS uses NMTC*nnnn*.PUB.SYS where *nnnn* is a number from 0000 to 9999. In this case, NMS opens a new file each time a new trace is started, and names it by incrementing *nnnn* by one. If this new trace file name is the name of a file that already exists, NMS continues to increment *nnnn* by one until it produces the name of a new (non-existing) file. Trace files are **wraparound** files. If the NMS trace facility reaches an end-of-file mark while recording to a disc file, it wraps subsequent entries around to the beginning of the file, overwriting the previous entries.

The Logging Facility

Node Management Services, NMS, provides logging services for Network Services, NetIPC, Network Transport, Link Manager, PC Link Manager, and the NCMS Subsystems. Logging is performed at three levels: network logging, event logging, and link level logging. Network logging records the usage of the communications network resources. It serves as a tool in resolving network problems. Event logging records the major events of subsystem processing. Link level logging records errors and special events encountered by the Link Manager.

The MPE command NSCONTROL with the LOG= option can be used to enable or disable detailed event logging for the Network Services (see Section 1, "Commands," in this volume for more information).

NMMGR logging configuration is used to direct the logfile output to a file or to the console. For Network Services, this is CLAS0004 of the SUB0006 subsystem ID. See Section 14, Logging Configuration, in Volume I of this manual set for more information.

The three MPE commands available to the network manager to manage log files are SHOWNMLOG, SWITCHNMLOG, and RESUMENMLOG as explained in Section 1 in this volume. The network manager can determine the name and available space in the log file, close the current file and open a new one, and activate logging after a recoverable error.

Logging is enabled for each subsystem during configuration with NMMGR. Each subsystem is numbered; Network Transport is subsystem 0003, NetIPC is subsystem 0005, Network Services is subsystem 0006, Link Manager is subsystem 0008, and the NCMS Subsystems are 0021 and 0029. Within each subsystem, the logging function is organized into numbered classes according to the type of messages logged. The network manager configures the logging classes and the destinations of the logging records for each class. Each subsystem sends logging records to NMS which prefixes a subsystem identifier, log class, and a time stamp. NMS outputs the log records to the configured destination associated with the logging class.

The Network Transport log classes are:

- Class 1 -- Internal software errors, such as critical Buffer Manager errors or the detection of corrupted data structures.

- Class 2 -- Errors requiring operator attention or intervention. Errors of this class typically affect all transport users, such as link errors, configuration errors, or buffer pool depletion.

- Class 3 -- Non-critical errors typically affecting single users, such as inbound checksum errors, outbound path failures, or protocol module retransmissions and error terminations.

- Class 4 -- Nodal Logging, such as NETCONTROL (start or stop).

- Class 5 -- Miscellaneous informative logging, such as transport protocol module initiation and termination.

- Class 6 -- Statistical information, such as statistics per connection.

The Network IPC log classes are:

- Class 0 -- Internal errors.
- Class 1 -- Resource errors.
- Class 2 -- Informative messages.

The Network Services log classes are:

- Class 2 -- Resource errors. These are errors that occur because of insufficient system resources. Occasional resource errors can be expected during peak usage. Frequent resource errors are an indication that some system resource is not properly configured. Examples of resource error include:

  - no extra data segments available
  - no NS pseudo terminals available
  - insufficient stack space for an NS operation

- Class 3 -- Internal errors. An internal error occurs only when the NS software detects an abnormal condition, usually the result of a failure somewhere in the system software. Most internal errors should be reported to HP. Examples of internal errors are:

  - corrupted internal data structures
  - failures of operating system intrinsics
  - invalid internal messages

- Class 4 -- Detailed events. It is possible to log normal events occurring in some of the NS modules (DSDAD, DSSERVER, ENV, and RPM). Logging of these events is usually disabled (because it generates a large volume of logged data), but can be enabled using the LOG function of the NSCONTROL command. The events include:

  - service initiation and termination
  - local and network message traffic
  - definition of environments

- Class 5 -- NETIPC errors. A special class is used to log NwtIPC errors detected by the NS software An IPC error is logged along with the corresponding Protocol Module error for more complete information on network problems. Most IPC errors occur because of transport-related problems. These may be due to:

  - network failures
  - remote nodes which are down
  - transport congestion
  - internal errors in the NS software

Link logging, SUB0008, logs internal errors reported by the link manager. Class 0 is used for messages from the link manager on all link types except ASNP links. Class 1 is used for messages from the PC link manager (for ASNP links only). Many of the messages correspond to NMERR messages, which are documented in the *NS3000/V Error Message and Recovery Manual.* (Note that the log error number printed in the logging record is an internal number and does not correspond to the NMERR message number.) Link Manager works in conjunction with higher level subsystems; therefore, the link level messages should always be interpreted in conjunction with other subsystem messages.

NCMS subsystems are active if you have installed the OpenView NS Monitor Applications, the OpenView Core software, and the Network Control Server (NCS) software (provided with FOS). See the *OpenView NS Monitor Applications Manager's Guide* (part number 32051-90001) for more information about these products. The Network Control Management Server (NCMS) consists of two subsystems: the NCMS - Subsystem Management Server (SUB0021) and the Network Control Management Server (SUB0029). Logging must be configured in NMCONFIG.PUB.SYS. See the Logging Configuration section in the *NS 3000/V Network Manager Reference Manual, Volume I* for more information about logging for NCMS.

The NCMS - Subsystem Management Server (SUB0021) log classes are:

- Class 1 -- Serious internal error.
- Class 4 -- Start/stop messages.
- Class 5 -- Error responses from NCS.
- Class 7 -- Informative messages.

The Network Control Management Server (SUB0029) log classes are:

- Class 1 -- Serious internal errors.
- Class 2 -- Non-recoverable internal errors.
- Class 3 -- Recoverable internal errors.
- Class 4 -- NCM/NCS start/stop messages.
- Class 5 -- Error responses from NCS.
- Class 6 -- NCM/NCS connection establishment/termination and negative event acknowledgement.
- Class 7 -- Informative messages.

INTRODUCTION OF NMDUMP

Because logging records and trace messages are created in coded form, NMS provides a utility to format log and trace files. This utility is called NMDUMP (NMS Trace/Log File Analyzer).

The remainder of this section describes the NMDUMP utility and contains the following subsections:

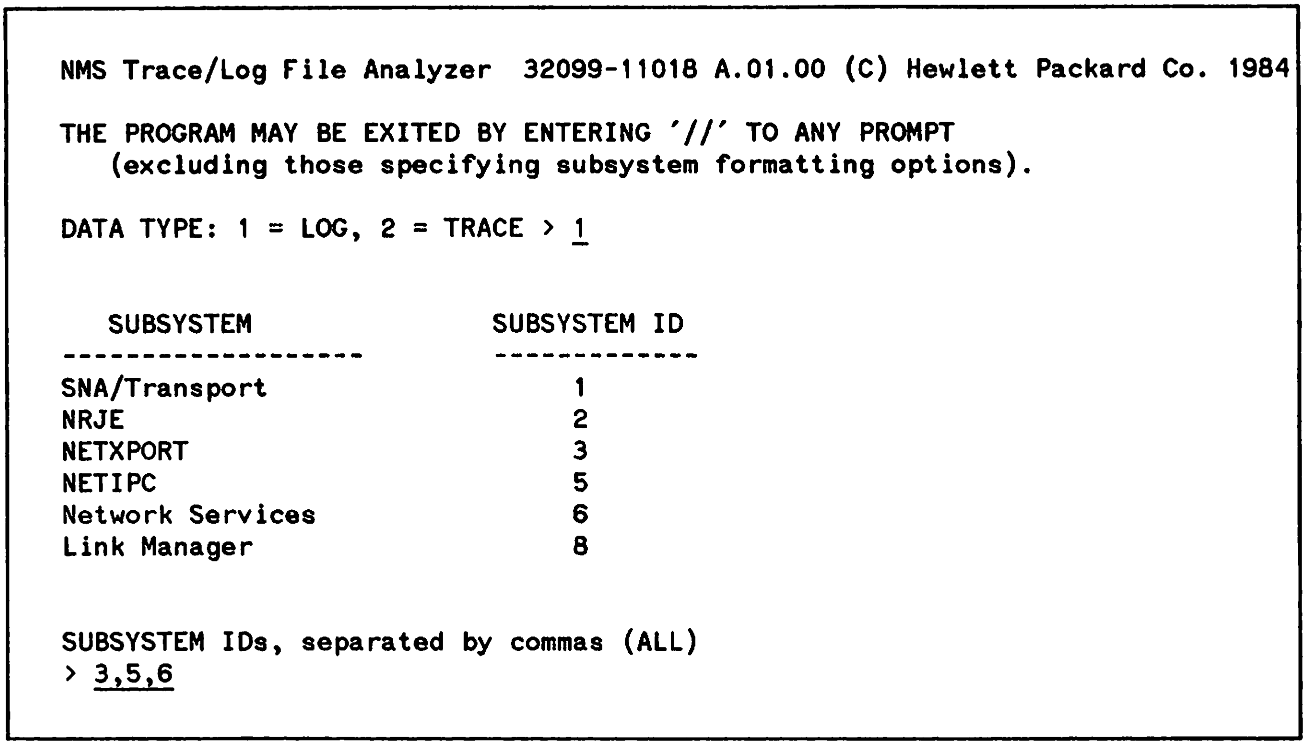
- The NMS Trace/Log Formatter (NMDUMP)
  - Specifying a File Name
  - Selecting a Time Range
- NETXPORT, NetIPC, and Network Services Formatting
- NETXPORT, NetIPC, and Network Services Menu Options
  - Option ?, Redisplay Options
  - Option 0, Set Defaults
  - Option 1, ASCII On or Off
  - Option 2, Octal or Hex
  - Option 3, Max # of Bytes
  - Option 4, Verbosity High or Low
  - Log Option 5, Class
  - Log Option 6, Entity or Module
  - Log Option 7, Event or PIN
  - Trace Option 5, Type or Descriptor
  - Trace Option 6, Entity or Service
  - Trace Option 7, Event or PIN
  - Trace Option 8, NS Messages

THE NMS TRACE/LOG FORMATTER (NMDUMP)

The NMDUMP utility is accessed by running the program:

```
:RUN NMDUMP.PUB.SYS
```

The following menu is displayed:



```
NMS Trace/Log File Analyzer  32099-11018 A.01.00 (C) Hewlett Packard Co. 1984

THE PROGRAM MAY BE EXITED BY ENTERING '//' TO ANY PROMPT
   (excluding those specifying subsystem formatting options).

DATA TYPE: 1 = LOG, 2 = TRACE > 1


   SUBSYSTEM                SUBSYSTEM ID
-------------------         ------------
SNA/Transport                    1
NRJE                             2
NETXPORT                         3
NETIPC                           5
Network Services                 6
Link Manager                     8


SUBSYSTEM IDs, separated by commas (ALL)
> 3,5,6
```

Figure 3-1. Initiating the NMDUMP Formatting Program.

You can select either logging or tracing by entering 1 or 2, respectively. In the example shown in Figure 3-1, the user selects logging. Next, NMDUMP displays the name and the ID number of each subsystem that uses NMS logging or tracing. At the prompt, enter the subsystem ID numbers, separated by commas, for each subsystem you want formatted. NMDUMP indicates the default for a prompt in parentheses. Press RETURN if you want to select the default. The user in Figure 3-1 selects the Network Transport (3), NetIPC (5), and Network Services (6) subsystems.

For most of the subsystems, NMDUMP calls an associated formatting menu that allows you to specify which log records or trace messages are to be formatted and how to display them. If you specify that you want to format several subsystem IDs, NMDUMP displays the menu for the first subsystem specified and prompts you for information. When you are finished using the first menu, NMDUMP displays the next menu, and so on. These menus are described in more detail in "NETXPORT, NetIPC, and Network Services Formatting" later in this section.

NOTE

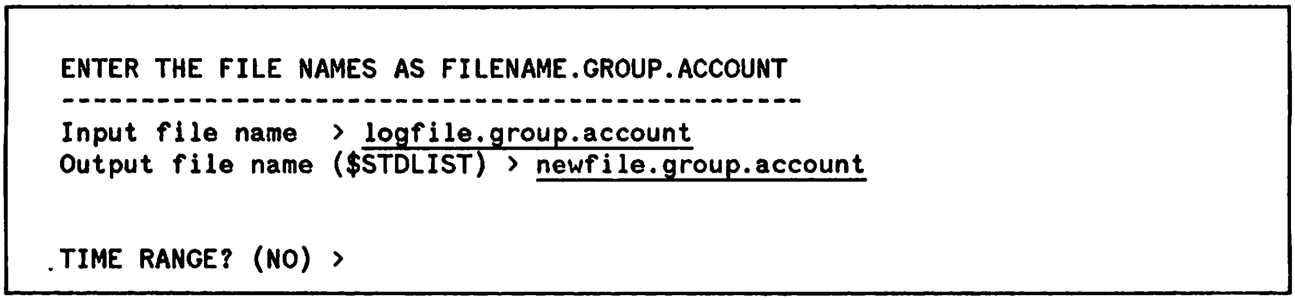
Menus for the SNA/Transport and NRJE subsystems are displayed with the ALL option, even if these products are not installed on your system. With NMDUMP, you can format log or trace files from any system, regardless of what products are installed.

After the menus for the specified subsystems have been displayed, NMDUMP prompts you for the name of the log or trace file. You can either specify a file name for the formatted output or use a file equation to redirect the output. This procedure is described in more detail in "Specifying a File Name" later in this section. You may also specify a time range within the file to be formatted. This is described in more detail in "Selecting a Time Range" later in this section.

You may type // at any of the main NMDUMP prompts to exit the program. If NMDUMP is displaying a subsystem menu, you must press RETURN to exit the menu and return to the main NMDUMP prompt, where typing // exits the program.

Specifying a File Name

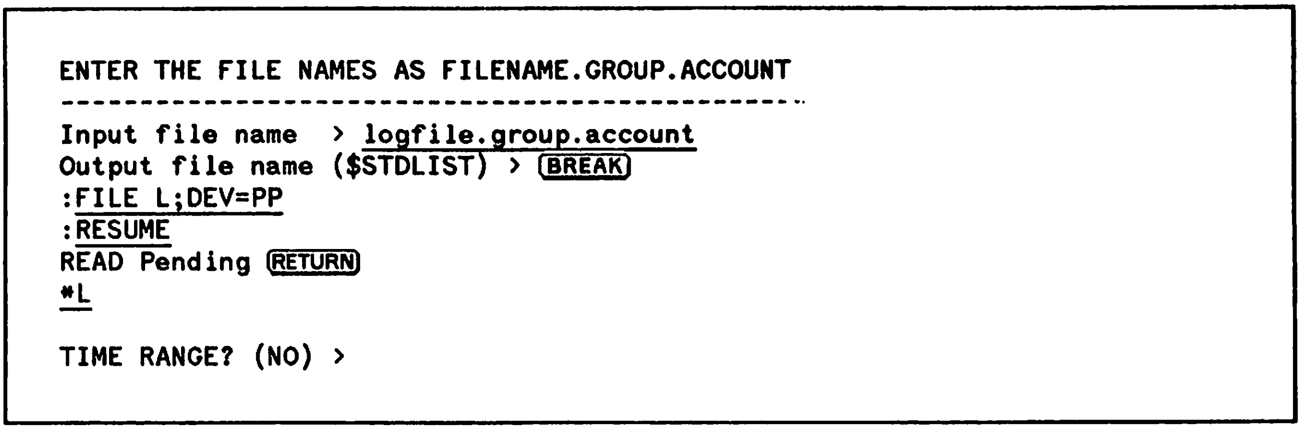
After you set the formatting options, NMDUMP prompts you for the file name of the log or trace file, as shown in the following three figures.



```
ENTER THE FILE NAMES AS FILENAME.GROUP.ACCOUNT
----------------------------------------------
Input file name  > logfile.group.account
Output file name ($STDLIST) > newfile.group.account


.TIME RANGE? (NO) >
```

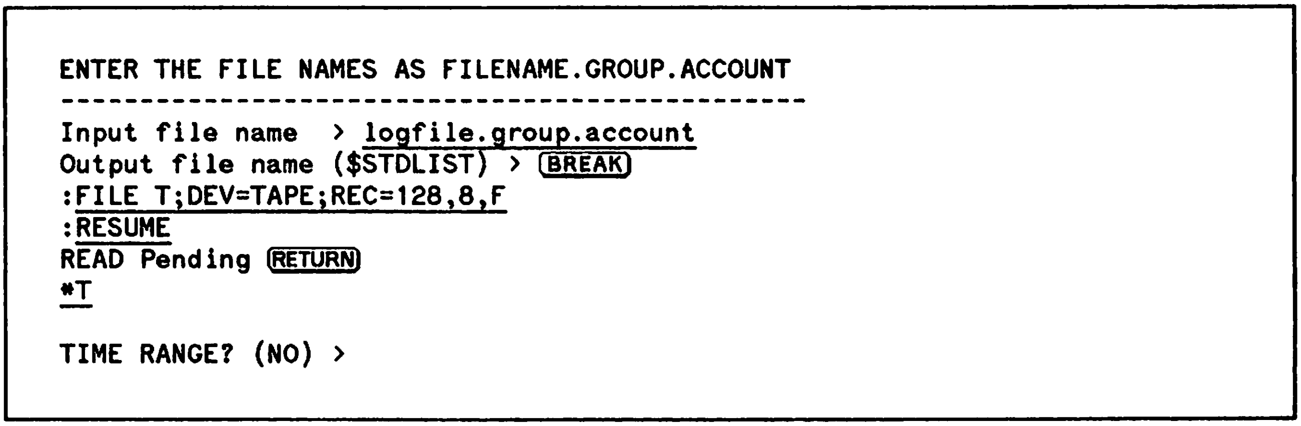
Figure 3-2. Specifying a File Name

The input file name must be the name of an existing log file. The output file name must not exist on the system. If the group and account are not given, the logon group and account are used. The default output file for RETURN is $STDLIST. You can use a file equation to back-reference the name of an input or output file. A file equation can be entered in break mode with NMDUMP, as shown in Figures 3-3 and 3-4. You can also enter a file equation before running NMDUMP.



```
ENTER THE FILE NAMES AS FILENAME.GROUP.ACCOUNT
----------------------------------------------
Input file name  > logfile.group.account
Output file name ($STDLIST) > BREAK
:FILE L;DEV=PP
:RESUME
READ Pending RETURN
*L

TIME RANGE? (NO) >
```

Figure 3-3. Specifying a File Equation for a Printer

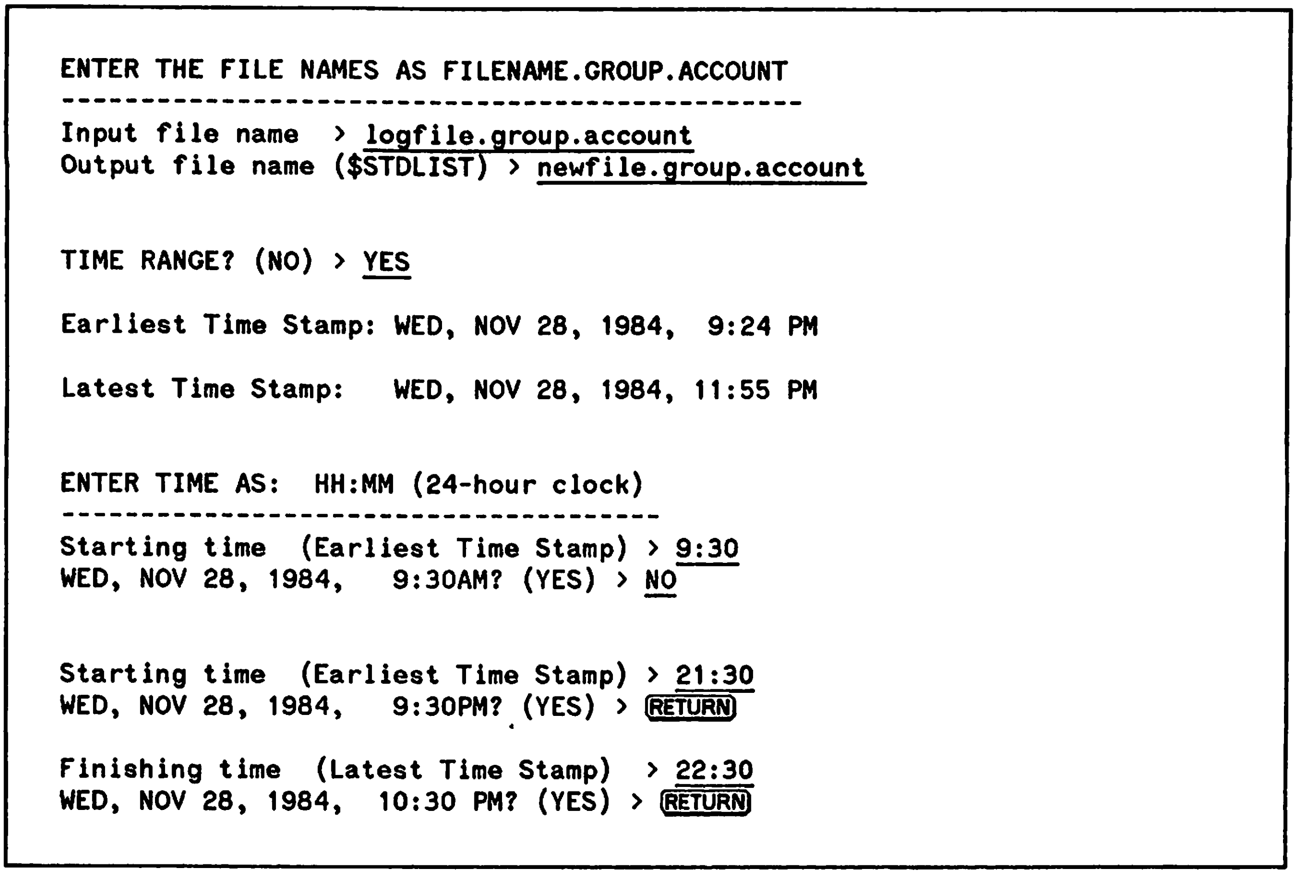


```
ENTER THE FILE NAMES AS FILENAME.GROUP.ACCOUNT
----------------------------------------------
Input file name  > logfile.group.account
Output file name ($STDLIST) > BREAK
:FILE T;DEV=TAPE;REC=128,8,F
:RESUME
READ Pending RETURN
*T

TIME RANGE? (NO) >
```

Figure 3-4. Specifying a File Equation for Tape

Selecting a Time Range

After you enter the file names, NMDUMP asks if you want to specify a time range. If you do not choose to specify a time range, NMDUMP formats the entire log file without further prompts. However, if you choose to specify a time range, NMDUMP supplies you with the actual time range of the entire log file, then prompts you for a range within the time boundaries of the file. After you specify a time range, NMDUMP formats only the records with time stamps within the range.



```
ENTER THE FILE NAMES AS FILENAME.GROUP.ACCOUNT
----------------------------------------------
Input file name  > logfile.group.account
Output file name ($STDLIST) > newfile.group.account


TIME RANGE? (NO) > YES

Earliest Time Stamp: WED, NOV 28, 1984,  9:24 PM

Latest Time Stamp:   WED, NOV 28, 1984, 11:55 PM


ENTER TIME AS:  HH:MM (24-hour clock)
-------------------------------------
Starting time  (Earliest Time Stamp) > 9:30
WED, NOV 28, 1984,    9:30AM? (YES) > NO


Starting time  (Earliest Time Stamp) > 21:30
WED, NOV 28, 1984,    9:30PM? (YES) > RETURN

Finishing time  (Latest Time Stamp)  > 22:30
WED, NOV 28, 1984,  10:30 PM? (YES) > RETURN
```

Figure 3-5. Selecting a Time Range

In the example in Figure 3-5, the user selects to specify a time range. NMDUMP searches the entire log file, then displays the earliest and latest time stamps. Next, NMDUMP prompts the user for the starting and finishing times. If the dates for the two time stamps are not the same, NMDUMP issues prompts for these dates. In Figure 3-5, the user chooses to format a one-hour time span. Notice that NMDUMP translates the time entered from 24-hour time to standard time, then asks the user to verify it. In the example, the user first enters the time incorrectly. Seeing that the verification prompt shows AM instead of PM, the user enters NO. NMDUMP repeats the starting-time prompt. The user now enters the correct 24-hour time, 21:30, to represent 9:30 p.m.

After NMDUMP formats and outputs all records within the specified time range, it prompts you for another time range:

```
Another TIME RANGE? (NO)? >
```

If you enter NO, which is the default, the NMDUMP program finishes running. If you enter YES, NMDUMP again prints the time range of the file, prompts you for starting and finishing times, then processes the records for the specified time range. This process continues until you specify that there are no more time ranges.

NETXPORT, NetIPC, and Network Services Formatting

The example in Figure 3-6 shows a typical session using the NMDUMP formatting menus. The user is formatting a log file and has chosen the Network Services, subsystem ID 6. NMDUMP displays the formatting menu for the Network Services Log Format, which provides nine options. The current value of each option is indicated in parentheses. When the menu is first displayed, as shown in Figure 3-6, these values are set to their default values. Pressing RETURN means that you are selecting the values displayed in the parentheses. However, you can change anv of the values by entering the number of the option and pressing RETURN. In Figure 3-6, the user selects option 4, which toggles the setting for verbosity from HIGH to LOW. Notice that selecting low verbosity also sets the maximum number of bytes per record to zero. When values of options are changed, the menu is redisplayed with the new values shown.

```
SUBSYSTEM IDs, separated by commas (ALL)
>6 RETURN

NETWORK SERVICES LOG FORMATTER  (A0000000)

0 - Set all options back to their default values
1 - Additional ASCII representation (ON)
2 - Output format                   (OCTAL)
3 - Maximum # of bytes/rec to dump  (128)
4 - Label verbosity level           (HIGH)
5 - Class selection                 (ALL)
6 - Module selection                (ALL)
7 - PIN selection                   (ALL)
? - Redisplay current options

Enter number of option or <CR> to select current options: 4 RETURN

Max number of bytes/record to raw dump set to 0

NETWORK SERVICES LOG FORMATTER  (A0000000)

0 - Set all options back to their default values
1 - Additional ASCII representation (ON)
2 - Output format                   (OCTAL)
3 - Maximum # of bytes/rec to dump  (0)
4 - Label verbosity level           (LOW)
5 - Class selection                 (ALL)
6 - Module selection                (ALL)
7 - PIN selection                   (ALL)
? - Redisplay current options

Enter number of option or <CR> to select current options : RETURN
```

Figure 3-6. A Typical Session with NMDUMP

Four of the options (1-4) reflect the common structure of log and trace files. They are provided on both log and trace file formatting menus for all three subsystems--Network Transport, NetIPC, and Network Services. Two options, ? and 0, are used for menu display and are also provided on all menus. The other options vary, depending on the subsystem and the type of file. Table 3-1 shows the options that vary by subsystem for log files, and Table 3-2 shows the options that vary by subsystem for trace files.

TABLE 3-1. SUBSYSTEM OPTIONS FOR LOG FILES

Option	Network Transport	NetIPC	Network Services
5	Class	Class	Class
6	Entity		Module
7	Event	PIN	PIN

TABLE 3-2. SUBSYSTEM OPTIONS FOR TRACE FILES

Option	Network Transport	NetIPC	Network Services
5	Trace Type	Descriptor	Descriptor
6	Entity		Service
7	Event	PIN	PIN
8			NS Message only

You cannot format information that was not entered in the trace or log file, regardless of which option you select on the formatting menu. The formatting menu options are changed in one of three ways:

- Most of the options toggle between two possible values, such as YES or NO, and ON or OFF. If you wish to change the displayed value of this type of option, type in the option number and press RETURN. The menu is redisplayed with the new value shown in parentheses. An example of this type of option (4) is shown in Figure 3-2.

- Some options allow more choices. For these options, you enter the option number and press RETURN to display an additional menu and prompt:

  ```
  NETWORK SERVICES LOG FORMATTER (A0000000)

  5 - Class selection                (ALL)

  Enter number of option or <CR> to select current options: 5
   2   - Resource errors         4   - Detailed events
   3   - Internal errors         5   - IPC errors
  Enter class numbers separated by commas:
  >
  ```

  The items on the additional menu are numbered. Select the items that you want formatted by entering their numbers, separated by commas. When you press RETURN, this menu is redisplayed with the selected numbers shown in parentheses.

- Several options prompt you to enter the numbers ot certain items:

  ```
  NETWORK SERVICES LOG FORMATTER (A0000000)

  7 - PIN Selection             (ALL)

  Enter number of option or <CR> to select current options: 7
  Enter PIN numbers separated by commas:
  >
  ```

  Multiple numbers, separated by commas, are allowed. Enter the number(s) requested and press RETURN. The formatting menu is redisplayed with the number(s) shown in parentheses.

Details on each option are described below.

NETXPORT, NetIPC and Network Services Menu Options

Option ?, Redisplay Options.

```
? - Redisplay current options
```

The system redisplays the menu with the current settings for the options shown.

Option 0, Set Defaults.

```
0 - Set all options back to their default values
```

This option is self-explanatory.

Option 1, ASCII On or Off.

```
1 - Additional ASCII representation (ON)
```

This. option has one of two possible values, ON or OFF. Enabling this option gives you an ASCII representation of the data, which is given in octal (octal is the default; data will be given in hexadecimal if Option 2 is enabled). The default is ON. You may wish to disable this option if no data is being produced, or if you wish to save space. Entering the option number 1 automatically turns ASCII representation OFF.

Option 2, Output Format, Octal or Hex.

```
2 - Output format                  (OCTAL)
```

This option has one of two possible values, OCTAL or HEX. Enabling this option gives you the unformatted data in hexadecimal rather than octal form. Entering the option number 2 automatically changes the data representation to HEX.

Option 3, Maximum Number of Bytes.

```
3 - Maximum # of bytes/rec to dump  (128)

Enter number of option or <CR> to select current options: 3
Enter maximum number of bytes/record to raw dump. (0..8192)
>
```

This option allows you to limit the amount of data printed in the information and data sections. You may want to reduce the number of bytes per record if the information you need is located in the first few bytes of the information section.

Option 4, Verbosity High or Low.

```
4 - Label verbosity level          (HIGH)
```

This option has two possible settings, HIGH or LOW. Use LOW if you want to produce a summary of the log records to quickly isolate the time range for the event or error you are interested in. When LOW verbosity is selected, the maximum number of bytes per record (Option 3) is set to zero. Selecting LOW verbosity for trace files gives you the formatted header and raw messages. Once you have found the time range you are interested in, you can use HIGH verbosity to produce a detailed report for this time range.

Log Option 5, Class Selection. If you select this option on the log formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:

```
NETXPORT LOG FORMATTER (A0000000)

5 - Class selection                 (ALL)

Enter number of option or <CR> to select current options: 5
 1   - Int. software errors      4  - Nodal logging
 2   - Op. attention required    5  - Informative
 3   - Non-critical errors       6  - Statistics
Enter class numbers separated by commas:
 >

NETIPC LOG FORMATTER (A0000000)

5 - Class selection                 (ALL)

Enter number of option or <CR> to select current options: 5
 0   - Internal software errors  2  - Informative logging
 1   - Socket resource errors
Enter class numbers separated by commas:
 >

NETWORK SERVICES LOG FORMATTER (A0000000)

5 - Class selection                 (ALL)

Enter number of option or <CR> to select current options: 5
 2   - Resource errors          4   - Detailed events
 3   - Internal errors          5   - IPC errors
Enter class numbers separated by commas:
 >
```

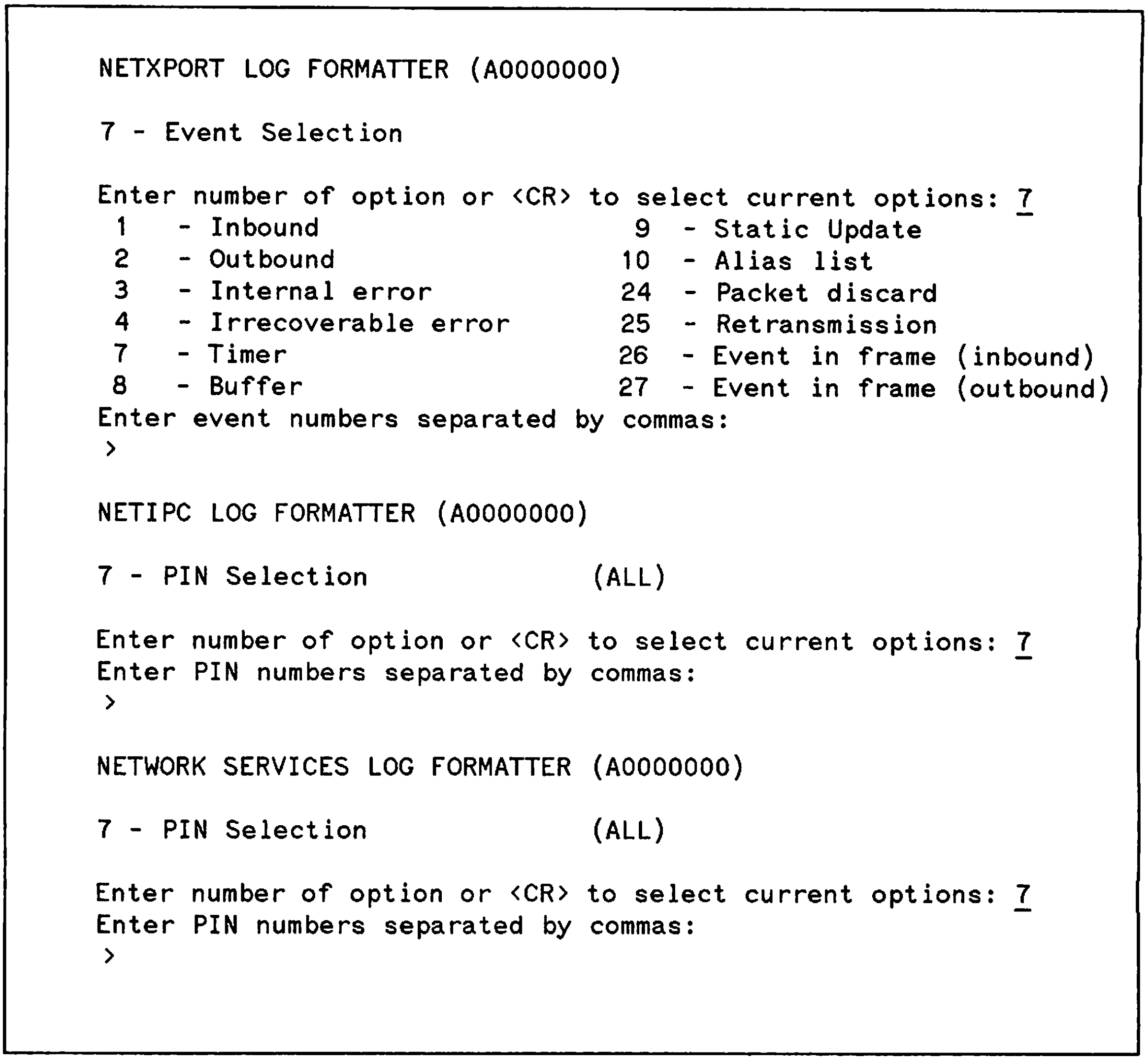
The options refer to the log classes configured using NMMGR, as described in Section 14 of Volume I. Log records will exist in the log file for each of the log classes configured to disc. Refer to Table 14-1 of Volume I for information on configuring logging to a disc file. If you want to format the records of specific log classes, enter the class numbers, separated by commas, at the prompt. When you press RETURN, the menu is redisplayed with the new settings.

Log Option 6, Entity or Module Select. If you select this option on the log formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:

```
NETXPORT LOG FORMATTER (A0000000)

6 - Entity selection              (ALL)

Enter number of option or <CR> to select current options: 5
 0   - LAN NI                     105  - PXP
 1   - Gateway Half NI            106  - PXP SIP
 3   - Router NI                  109  - IP
 6   - X.25 NI                    110  - IP Update
 9   - Loopback NI                111  - Probe
 101 - User Interface             112  - Dial
 102 - Control Process            113  - Path
 103 - TCP                        118  - X.25
 104 - TCP SIP
Enter entity numbers separated by commas:
 >

NETWORK SERVICES LOG FORMATTER (A0000000)

6 - Module selection              (ALL)

Enter number of option or <CR> to select current options: 6
 10  - DSDAD                       20  - RFA
 11  - DSSERVER                    30  - VT
 12  - CX                          40  - NFT
 13  - BFM                         50  - PTOP
 14  - ENV                         51  - RPM
 15  - DSUTIL
Enter module numbers separated by commas:
 >
```

You may have encountered an error message which defines the entity or module that caused the problem. With this option, you can select specific entities or modules to format.

Log Option 7, Event or Pin Selection. If you select this option on the log formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:



```
NETXPORT LOG FORMATTER (A0000000)

7 - Event Selection

Enter number of option or <CR> to select current options: 7
 1    - Inbound                      9   - Static Update
 2    - Outbound                    10   - Alias list
 3    - Internal error              24   - Packet discard
 4    - Irrecoverable error         25   - Retransmission
 7    - Timer                       26   - Event in frame (inbound)
 8    - Buffer                      27   - Event in frame (outbound)
Enter event numbers separated by commas:
>

NETIPC LOG FORMATTER (A0000000)

7 - PIN Selection               (ALL)

Enter number of option or <CR> to select current options: 7
Enter PIN numbers separated by commas:
>

NETWORK SERVICES LOG FORMATTER (A0000000)

7 - PIN Selection               (ALL)

Enter number of option or <CR> to select current options: 7
Enter PIN numbers separated by commas:
>
```

For Network Transport event selection, the menu shown at the top of the above box is provided.

For Network Services and NetIPC subsystems, this option allows you to narrow your focus to a certain PIN. Each NS log message includes the Process Id Number (PIN) of the server or user process in which the error or event occurred. If information about a particular process is desired, and if the PIN is known, the log messages for that process can be selected. If you do not know the PIN, you can use the NSCONTROL STATUS=SERVERS display to determine the PIN. Alternatively, you can set formatting option 4 (verbosity) to LOW and default the output to $STDLIST. NMDUMP displays abbreviated output to your screen, and you can find the PIN in this display.

Trace Option 5, Type or Descriptor Selection. If you select this option on the trace formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:

```
NETXPORT TRACE FORMATTER (A0000000)

5 - Type selection                (ALL)

Enter number of option or <CR> to select current options: 5
 0   - Internal trace            4  - Data trace
 1   - State trace               5  - Nodal management trace
 2   - Port message trace        6  - Buffer trace
 3   - Protocol header trace     7  - Timer trace

NETIPC TRACE FORMATTER (A0000000)

5 - Descriptor selection          (ALL)

Enter number of option or <CR> to select current options: 5
Enter descriptor numbers separated by commas:

NETWORK SERVICES TRACE FORMATTER (A0000000)

5 - Descriptor selection          (ALL)

Enter number of option or <CR> to select current options: 5
Enter descriptor numbers separated by commas:
```

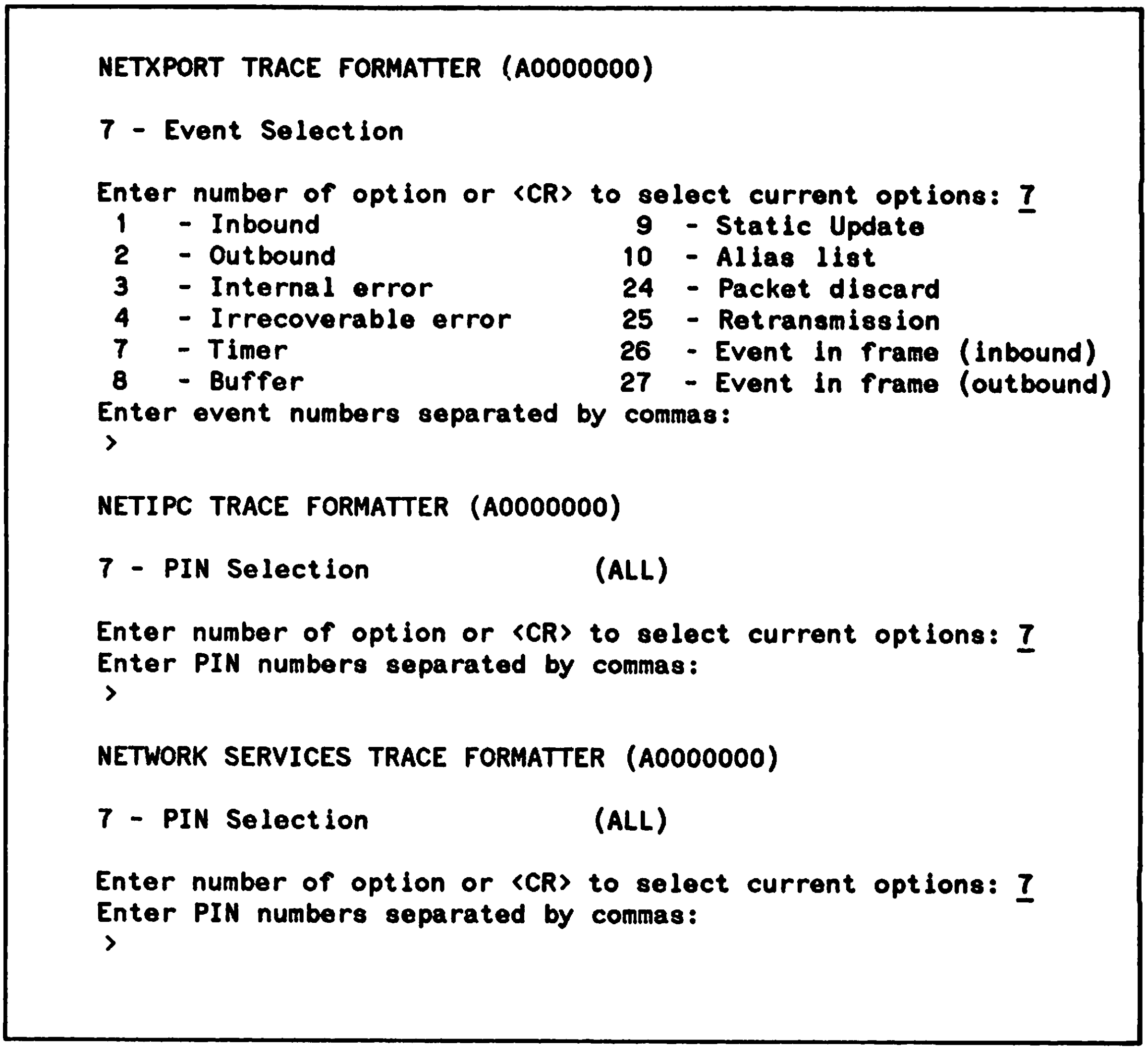
A socket descriptor is created every time a socket or connection is created in a process. This option allows you to format only those events associated with a particular descriptor number. This is useful when reading a trace which pinpoints a particular socket or connection as causing the problem, and you want to see only the errors concerning that socket or connection. If you do not know the descriptor, you can set formatting option 4 (verbosity) to LOW and default the output to $STDLIST. NMDUMP displays abbreviated output on your screen. You can find the descriptor in that display.

Trace Option 6, Entity or Service Selection. If you select this option on the trace formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:

```
NETXPORT TRACE FORMATTER (A0000000)

6 - Entity selection                (ALL)

Enter number of option or <CR> to select current options: 6
 0   - LAN NI                    105  - PXP
 1   - Gateway Half NI           106  - PXP SIP
 3   - Router NI                 109  - IP
 6   - X.25 NI                      110  - IP Update
 9   - Loopback NI               111  - Probe
 101 - User Interface            112  - Dial
 102 - Control Process           113  - Path
 103 - TCP                       118  - X.25
 104 - TCP SIP
Enter entity numbers separated by commas:
 >

NETWORK SERVICES TRACE FORMATTER (A0000000)

6 - Service selection                (ALL)

Enter number of option or <CR> to select current options: 6
 1   - VT                          5 - RPM
 2   - NFT                         6 - PTOP
 3   - RFA / RDBA
Enter service numbers separated by commas:
 >
```

This option allows you to narrow your focus to a specific entity or service. You may have an error message which defines the entity or module that caused the problem, and you may want to look at only these records. The menu shown at the top of the above box allows you to select certain entities or modules to format.

Trace Option 7, Event or PIN Selection. If you select this option on the trace formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:



```
NETXPORT TRACE FORMATTER (A0000000)

7 - Event Selection

Enter number of option or <CR> to select current options: 7
 1   - Inbound                   9  - Static Update
 2   - Outbound                 10  - Alias list
 3   - Internal error           24  - Packet discard
 4   - Irrecoverable error      25  - Retransmission
 7   - Timer                    26  - Event in frame (inbound)
 8   - Buffer                   27  - Event in frame (outbound)
Enter event numbers separated by commas:
 >

NETIPC TRACE FORMATTER (A0000000)

7 - PIN Selection              (ALL)

Enter number of option or <CR> to select current options: 7
Enter PIN numbers separated by commas:
 >

NETWORK SERVICES TRACE FORMATTER (A0000000)

7 - PIN Selection              (ALL)

Enter number of option or <CR> to select current options: 7
Enter PIN numbers separated by commas:
 >
```

Selecting this option allows you to narrow your focus to a specific event or PIN.

For Network Transport event selection, the menu shown at the top of the above box is provided.

For Network Services and NetIPC subsystems, this option allows you to narrow your focus to a certain PIN. Each NS trace message includes the Process Id Number (PIN) of the server or user process in which the error or event occurred. If information about a particular process is desired, and if the PIN is known, the trace messages for that process can be selected. If you do not know the PIN, you can use the NSCONTROL STATUS=SERVERS display to determine the PIN. Alternatively, you can set formatting option 4 (verbosity) to LOW and default the output to $STDLIST. NMDUMP displays abbreviated output to your screen, and you can find the PIN in this display.

Trace Option 8, NS Messages. If you select this option on the trace formatter menu, NMDUMP displays the following menu of options, depending on the subsystem:

```
NETWORK SERVICES TRACE FORMATTER (A0000000)

8 - Format NS Messages only     (OFF)

Enter number of option or <CR> to select current options: 8
>
```

If you enter 8 and press RETURN, the value in parentheses is toggled from OFF to ON, or vice versa, depending on the original value. If this value is OFF, you see the actual NetIPC intrinsic calls used for the Network Services.

LINK SUBSYSTEM FORMATTING

The Link Manager subsystem does not call any menus. Instead, the link-level logging records are formatted as shown in Figure 3-7. For the Link Manager subsystem, NMDUMP formats logging messages only. The first line labels the error as a Link Manager error, and shows whether it is fatal or non-fatal. The next line contains the time and date of the event. This line is followed by the log message, the Requestor ID, the Link name, and the LDEV of the link. The Requestor ID is for internal use only. Any underlying errors are reported below the LDEV field. Many of these messages correspond to NMERR messages, which are documented in the *NS3000/V Error Message and Recovery Manual.* The example below includes an underlying CS error (CS error #86).

```
*** LINK MANAGER ERROR   (NON FATAL)  ***
-
-   DATE:    10/18/83    TIME:    15:23:34
-
-   CALL to CCONTROL intrinsic failed
-   Requestor ID:  1
-   Link name: SANJOSE
-   LDEV #45
-
-   CS error #86
-
```

Figure 3-7. Link Level Logging Format

CHANGING A NETWORK

SECTION 4

CONFIGURATION SCENARIOS

In this section, we will describe four common scenarios that you may encounter after you have completed your original network configurations. We will look at each case from the point of view of one node, but will also consider how each remote node will be affected. The scenarios consist of adding a node to a router network, adding a node to a LAN, configuring an existing node to also serve as a gateway half, and adding a node to an X.25 network. Most of the steps listed will also apply to deletions for the given scenario; in the steps where there are differences, however, these differences will be listed.

Adding A Router Node

Suppose you wish to add a node to a router network. Here are the basic steps you would take:

1) **Update worksheets.** (Refer to Section 3 and Appendix A of Volume I.) For the catenet worksheets, update the catenet map only if the node being added is a gateway node. For the network worksheets, update the network map to show the node to be added. Also, show any links to be added. For example, in Figure 4-1, we have used a dotted line to indicate that Link1 will be a new link, and we have labelled Node A because it will be the new node. The Network Table must also be updated to incorporate information regarding the new node. The Router Internet Routing Table needs to be updated only if the new node will be a gateway node.

   As for the node worksheets, you need to complete a Router Node Intranet Routing Table for the new node. In Figure 4-1, for example, Node A must know how to route messages to the other three nodes on the network. You also need to complete a Router Node Internet Routing Table for the new node. Next, you can fill out the information for "Subsequent Router Nodes" in Appendix A.

   All other nodes on the router network will be affected by the addition of a node. For example, each node must update its Router Node Intranet Routing Table to reflect the new node and each new link. Depending on where you add a link, all intranet routes may be affected. Regardless of how existing routes will be affected, you need to enter intranet routing information about how to communicate with the new node.

   If the new node will also be a gateway node, then all existing nodes on the router network will need to update their Router Node Internet Routing tables so that they can use the new node as a gateway to other networks. If the new node is not going to be a gateway, the other nodes will not need to change their internet routing entries.

   For the existing nodes on the network, refer next to the information on the node worksheets where you filled in values for fields configured during guided configuration. You should update this information as necessary, referring to new and updated maps and tables, because link, mapping and internet routing information may need to be changed.

2) **Shut down the network transport.** Make sure all users are logged off, then issue a NETCONTROL STOP command from the console.

3) **Perform a system backup.** Using the SYSDUMP or FULLBACKUP commands, create a complete coldload backup tape for safekeeping.

4) **Make system hardware changes.** Use the SHUTDOWN command to shut down the system, and install new hardware (INP/ATP). Then, using SYSDUMP, make the appropriate system configuration changes required to support the new INP or ATP. These changes will cause a coldload/update tape to be created. LOAD the system from the coldload tape.

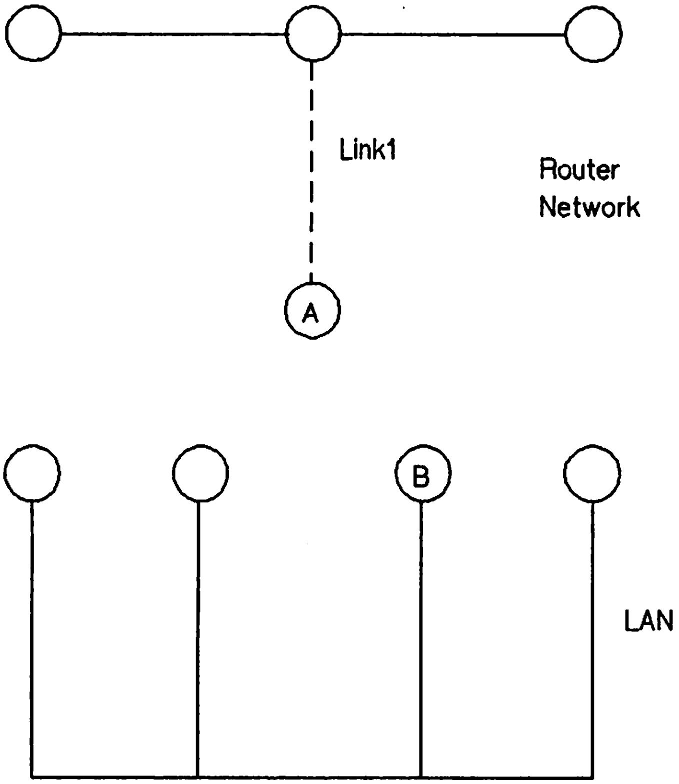


Figure 4-1. Adding Nodes

5) **Make software configuration changes.**

   a) Perform guided configuration, which will take you through a set of screens where you will enter information from your worksheets. See Section 6 of Volume I for details regarding the screens and fields you will visit during guided configuration.

   b) The last step in the guided configuration process is validating the configuration. Guided configuration will take you to the validation screen. All configuration data is cross-checked for accuracy and completeness. If there are any errors with the configuration, you will be shown the incorrect data and the associated screens. A summary of the validation is sent to the line printer for your reference in making corrections. Using this summary listing, revisit the screens with invalid data, and correct the values. To revisit these screens, use the direct-path branching feature allowed only in manual (not guided) configuration. The direct-branching paths you will need are displayed during validation. Repeat this step--validating and correcting--until there are no errors in the configuration.

6) **Start up the network transport.** Issue the NETCONTROL START command from the system console. This will cause your new configuration to be read in and activated.

7) **Make changes on remote nodes.** Repeat steps 2 through 6 on the adjacent nodes to which the router link is connected. For the remaining nodes on the router network, you can use the Go To Update function key of guided configuration to update mapping (intranet routing) and internet routing as necessary, resulting from the addition or deletion of a node in the network.

8) **Copy network directory.** (Refer to Section 15 of Volume I.) Copy NSDIR.NET.SYS and NSDIRK.NET.SYS from another node on the router network to the new node by using the STORE command to store the remote node's directory files on tape. RESTORE the files onto the new node.

9) **Update remote system directories.** This step typically will be performed by the network manager using the MAKESTREAM process described in Section 16 of Volume I. In short, the new node will send its directory entry to the central administrative node on the network. It will be up to the central administrative node to enter this information, then stream a job to update the directories on the other nodes. (If deleting a node, its entry should be removed from all network directories.)

Adding A LAN Node

Suppose you wish to add a node to a LAN. Here are the basic steps you would take:

1) **Update worksheets.** (Refer to Section 3 and Appendix A of Volume I.) For the catenet worksheets, update the catenet map only if the node being added is a gateway node. For the network worksheets, update the network map to show the node to be added. In Figure 4-1, we have labelled Node B to indicate that we are adding this node to the network. Also, the Network Table must be updated to incorporate information about the new node. The LAN Internet Routing Table needs to be updated only if the new node will be a gateway node.

   As for the node worksheets, you need to complete a LAN Node Internet Routing Table for the new node. Next, fill out the information for "Subsequent LAN Nodes" in Volume I, Appendix A. If the new node will also be a gateway node, then all existing nodes on the LAN will need to update their LAN Node Internet Routing tables so that they can use the new node as a gateway to other networks. Also, for these existing nodes, refer to the internet routing information on the node worksheets and add the appropriate information. If the new node is not going to be a gateway, the other nodes will not need to change their internet routing entries.

2) **Shut down the network transport.** Make sure all users are logged off, then issue a NETCONTROL STOP command from the console.

3) **Perform a system backup.** Using the SYSDUMP or FULLBACKUP commands, create a complete coldload backup tape for safe keeping.

4) **Make system hardware changes.** Use the SHUTDOWN command to shut down the system, and install new hardware (LANIC). Then, using SYSDUMP, make the appropriate system configuration changes required to support the new LANIC. These changes will cause a coldload/update tape to be created. LOAD the system from the coldload tape.

5) **Make software configuration changes.**

a) Perform guided configuration, which will take you through a set of screens where you will enter information from your worksheets. See Section 6 of Volume I for details regarding the screens and fields you will visit during guided configuration.

b) The last step in the guided configuration process is validating the configuration. Guided configuration will take you to the validation screen. All configuration data is cross-checked for accuracy and completeness. If there are any errors with the configuration, you will be shown the incorrect data and the associated screens. A summary of the validation is sent to the line printer for your reference in making corrections. Using this summary listing, revisit the screens with invalid data, and correct the values. To revisit these screens, use the direct-path branching feature allowed only in manual (not guided) configuration. The direct-branching paths you will need are displayed during validation. Repeat this step--validating and correcting--until there are no errors in the configuration.

6) **Start up the network transport.** Issue the NETCONTROL START command from the system console. This will cause your new configuration to be read in and activated.

7) **Make any necessary changes on remote nodes.** Changes to existing nodes on the LAN will be necessary only if the new node is going to be a gateway node. In this case, you can use the Go To Update function key of guided configuration to update internet routing information for the other LAN nodes.

8) **Copy network directory (optional).** (Refer to Section 15 of Volume I.) If the new node is to be a proxy server, then copy NSDIR.NET.SYS from another proxy server (must be an HP node) on the LAN to the new node by using the DSCOPY command. Because the network directory uses a KSAM file pair, be sure to copy both the data file and the key file.

9) **Update remote system directories.** This step typically will be performed by the network manager using the MAKESTREAM process described in Section 16 of Volume I. In short, the new node will send its directory entry to the central administrative node on the network. It will be up to the central administrative node to enter this information, then stream a job to update the directories on the other nodes. (If deleting a node, its entry should be removed from all network directories.)

Adding A Gateway Half NI

In Figure 4-2, we wish to add a gateway-half link to connect nodes A and B so that the LAN and the router network can communicate with each other using these nodes as gateway halves. This means that we must configure nodes A and B to include gateway half NIs in addition to their respective router and LAN NIs.

The steps below discuss how you would configure a node as a gateway half after it has been configured as part of a LAN. Read the "Adding A LAN Node" subsection for information about how adding a gateway affects other nodes on that network.

1) **Update Worksheets.** (Refer to Section 3 and Appendix A of Volume I.) For the catenet worksheets, update the catenet map to show the new gateway node. For the network worksheets, draw a gateway-half map that shows the node and the gateway half it will be connected to. Also, show the gateway-half link and indicate the networks that will be directly connected by this link. Then, complete a Gateway-Half Network Interface Table.

As for the node worksheets, complete a Gateway-Half Node Internet Routing Table. Next, fill out the information for "Subsequent Gateway-Half Nodes" in Volume I, Appendix A.

2) **Shut down the network transport.** Make sure all users are logged off, then issue a NETCONTROL STOP command from the console.

3) **Perform a system backup.** Using the SYSDUMP or FULLBACKUP commands, create a complete coldload backup tape for safe keeping.

4) **Make system hardware changes.** Use the SHUTDOWN command to shut down the system, and install new hardware (INP/ATP). Then, using SYSDUMP, make the appropriate system configuration changes required to support the new INP or ATP. These changes will cause a coldload/update tape to be created. LOAD the system from the coldload tape.

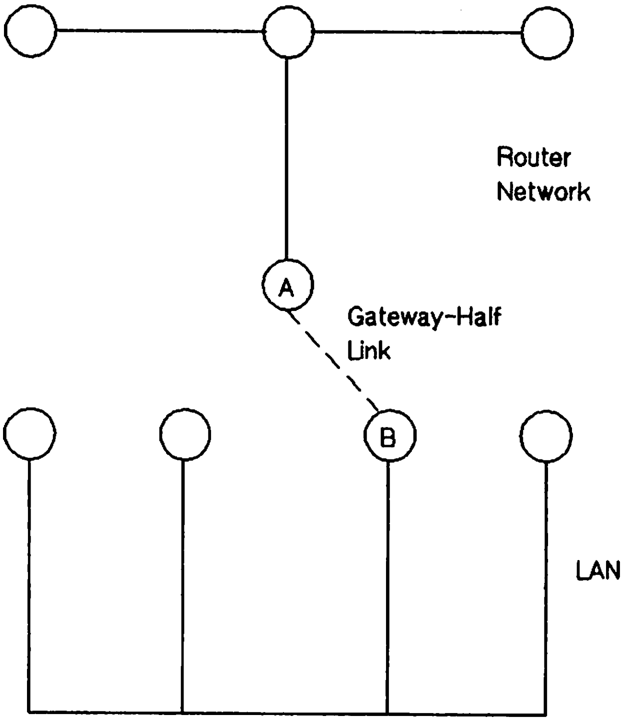


Figure 4-2. Adding a Gateway Half NI

5) **Make software configuration changes.**

a) Perform guided configuration, which will take you through a set of screens where you will enter information from your worksheets. See Section 6 of Volume I for details regarding the screens and fields you will visit during guided configuration.

b) The last step in the guided configuration process is validating the configuration. Guided configuration will take you to the validation screen. All configuration data is cross-checked for accuracy and completeness. If there are any errors with the configuration, you will be shown the incorrect data and the associated screens. A summary of the validation is sent to the line printer for your reference in making corrections. Using this summary listing, revisit the screens with invalid data and correct the values. To revisit these screens, use the direct-path branching feature allowed only in manual (not guided) configuration. The direct-branching paths you will need are displayed during validation. Repeat this step--validating and correcting--until there are no errors in the configuration.

6) **Start up the network transport.** Issue the NETCONTROL START command from the system console. This will cause your new configuration to be read in and activated.

Adding An X.25 Node

Suppose you wish to add a node to a X.25 network. Here are the basic steps you would take:

1) **Update worksheets.** (Refer to Section 3 and Appendix A of Volume I.) For the catenet worksheets, update the catenet map only if the node being added is a gateway node. For the network worksheets, update the network map to show the node to be added. Also, show any links to be added.

   As for the node worksheets, you need to complete an Intranet X.25 Table for the new node. You also need to complete an Internet X.25 Table for the new node. Next, you can fill out the information for "Subsequent X.25 Nodes" in Appendix A.

   All other nodes on the X.25 network will be affected by the addition of a node. For example, each node must update its Intranet X.25 Table to reflect the new node.

   If the new node will also be a gateway node, then all existing nodes on the X.25 network will need to update their Internet X.25 tables so that they can use the new node as a gateway to other networks. If the new node is not going to be a gateway, the other nodes will not need to change their internet X.25 entries.

   For the existing nodes on the network, refer next to the information on the node worksheets where you filled in values for fields configured during guided configuration. You should update this information as necessary, referring to new and updated maps and tables, because link, mapping and internet routing information may need to be changed.

2) **Shut down the network transport.** Make sure all users are logged off, then issue a NETCONTROL STOP command from the console.

3) **Perform a system backup.** Using the SYSDUMP or FULLBACKUP commands, create a complete coldload backup tape for safekeeping.

4) **Make system hardware changes.** Use the SHUTDOWN command to shut down the system, and install new hardware (INP). Then, using SYSDUMP, make the appropriate system configuration changes required to support the new INP link. These changes will cause a coldload/update tape to be created. LOAD the system from the coldload tape.

5) **Make software configuration changes.**

   a) Perform guided configuration, which will take you through a set of screens where you will enter information from your worksheets. See Section 6 of Volume I for details regarding the screens and fields you will visit during guided configuration.

   b) The last step in the guided configuration process is validating the configuration. Guided configuration will take you to the validation screen. All configuration data is cross-checked for accuracy and completeness. If there are any errors with the configuration, you will be shown the incorrect data and the associated screens. A summary of the validation is sent to the line printer for your reference in making corrections. Using this summary listing, revisit the screens with invalid data, and correct the values. To revisit these screens, use the direct-path branching feature allowed only in manual (not guided) configuration. The direct-branching paths you will need are displayed during validation. Repeat this step--validating and correcting--until there are no errors in the configuration.

6) **Start up the network transport.** Issue the NETCONTROL START command from the system console. This will cause your new configuration to be read in and activated.

7) **Make changes on remote nodes.** On the other nodes on the network who wish to communicate with the new node, you can use the Go To Update function key of guided configuration followed by the Go To X.25 SVC function key to add the new nodes, and the X.25 address to the SVC path table. The new node's node name and the IP addresses must also be added to the network directory.

8) **Copy network directory.** (Refer to Section 15 of Volume I.) Copy NSDIR.NET.SYS and NSDIRK.NET.SYS from another node on the X.25 network to the new node by using the STORE command to store the remote node's directory files on tape. RESTORE the files onto the new node.

9) **Update remote system directories.** This step typically will be performed by the network manager using the MAKESTREAM process described in Section 16 of Volume I. In short, the new node will send its directory entry to the central administrative node on the network. It will be up to the central administrative node to enter this information, then stream a job to update the directories on the other nodes. (If deleting a node, its entry should be removed from all network directories.)

Network Management with OpenView

APPENDIX

A

Network Management

Network Management is provided by the HP OpenView NS Monitor Applications (product numbers 32051A and 32053A). From a management node, these applications provide you with network status, link usage information, and diagnostic tests for managed nodes in the network. In order to use the monitor applications, you need the OpenView Windows product and the software for the HP 3000 as described below. The software on the HP 3000 that provides the kernel level services to the applications consists of two parts:

OpenView Core Software

The OpenView Core software is the separately orderable product (#32052) that works with the Network Control Server (NCS) software on the HP 3000. OpenView Core software contains the Network Control Manager (NCM). NCM is the coordinator of all of the network management facilities. NCM resides on one of the HP 3000 nodes, which is designated as the management node.

FOS Containing NCS

The HP 3000 MPE V Fundamental Operating System (FOS) contains the Network Control Server (NCS) software. NCS resides on each managed HP 3000 node in the network. NCS initiates network management functions on the node on which it resides. NCS interfaces with, and receives information from NS3000/V links. NCS is installed with NS3000/V on each node in the network.

The communication functionality between the OpenView Core software and NCS is called the Network Control Management Server (NCMS). During the installation of NCS or while running the OpenView NS Monitor Applications, NCMS messages are displayed on the HP 3000 console. The *NS3000/V Error Message and Recovery Manual* contains NCMS console messages (with meaning/cause and action). For more information about network management functionality provided by these products see the *OpenView NS Monitor Applications Manager's Guide* (part number 32051-90002).

NCSCONTROL Command

The NCSCONTROL command controls whether or not a node is monitored using the OpenView NS Monitor applications. For more information about NCSCONTROL, see the *OpenView NS Monitor Applications Manager's Guide* (part number 32051-90002).

NCSCONTROL command parameters and functions are summarized below:

Parameters

STATUS	Gives the status of NCS.
ADD	Changes the NCS configuration file such that NCS may now be started up (manually or when the system starts up), but does not startup NCS. In other words, NCS is "added" to the domain of nodes that can be managed.
DELETE	Changes the NCS configuration file such that NCS can no longer be started up, but does not shutdown the currently running NCS. (NCS is "deleted" from the domain of nodes that can be managed.)
STOP	Shutsdown the currently running NCS.
START	Starts NCS up (unless NCSCONTROL DELETE has been done to keep NCS from starting up).

NCSCONTROL Messages

Messages may appear on the console of the management or managed node after you have issued the NCSCONTROL command. The *NS3000/V Error Message and Recovery Manual* contains NCSCONTROL messages (with meaning/cause and action).

In addition, NCSCONTROL STATUS can display the following messages, depending on the functions that have been performed:

```
NCS UP AND NCSCONTROL ADD HAS BEEN DONE
```

NCS is running and will be started up whenever the system is brought up.

```
NCS UP, MUST DO NCSCONTROL ADD BEFORE STARTING NCS
```

NCS is running, but if the system is shutdown, and then brought up again, NCS will not startup.

```
NCS DOWN, NCSCONTROL ADD HAS BEEN DONE
```

NCS is not currently running, but will be started the next time the system is brought up.

```
NCS DOWN, MUST DO NCSCONTROL ADD BEFORE STARTING NCS
```

NCS is not currently running, and will not be started up the next time the system is brought up.

GLOSSARY

A

address - in networking, a numerical identifier defined and used by a particular protocol and associated software to distinguish one node from another.

address resolution - in NS networks, the mapping of node names to IP addresses and the mapping of IP addresses to subnet addresses.

Address Resolution Protocol (ARP) - used by the IEEE 802.3 links for IP to LAN station address resolution for nodes on the LAN that support Ethernet. With the concurrent configuration of IEEE 802.3 and Ethernet, both the Probe protocol and Address Resolution Protocol broadcast requests to all nodes on the LAN to resolve the 48-bit LAN station address of a given remote node.

adjacent - describes a node on a router network that is connected to another node by a single link, with no intervening nodes.

Advanced Terminal Processor (ATP) - a hardware card that fits into the backplane of the HP 3000 and that provides a physical layer interface for Asynchronous SERIAL Network Links.

ARP - see **Address Resolution Protocol**

Asynchronous Serial Network Protocol (ASNP) - a Data Link Layer protocol used for Asynchronous SERIAL Network Links.

ASNP - see **Asynchronous Serial Network Protocol**

ATP - see **Advanced Terminal Processor**

autodial - describes a dial link in which the remote node's telephone number is automatically dialed by modem or other device with this capability.

AUI cable - Attachment Unit Interface cable, a cable joining the LANIC to the MAU (Media Attachment Unit) for coaxial cable IEEE 802.3 local area network links.

B

Bisynchronous Communication protocol - see **BSC protocol.**

boundary - see **network boundary.**

broadcast - a method of communcation in which all nodes on the network share the same communcications channel (referred to as the communications bus). Messages are transmitted to all nodes on the same bus at the same time. IEEE 802.3 networks are broadcast networks.

brother branching - the process of proceeding to another screen in NMMGR by using the NEXT command. The screen displayed after entering NEXT will be one on the same hierarchal level as the current screen in the configuration tree structure.

BSC protocol - a data link layer protocol that can be used over NS Point-to-Point 3000/V Links that use dial links (switched lines). BSC is an acronym for the Bisynchronous Communication protocol.

buffer - a logical grouping of a system's memory resources used by NS3000/V.

C

catenet - a group of computer networks that are connected to one another.

Catenet Administrator - the person having responsibility for coordinating network management tasks among all networks in the catenet.

catenet lifecycle - see **lifecycle, network/catenet**

central administrative node - a node designated as the node in the catenet to be the first one updated with new or changed internet routing and network directory information about any other nodes in the network.

Communications Services (CS/3000) - software included with NS3000/V links that provides some diagnostic and link management software.

computer name - term used in some HP networking documentation to refer to node name. (See **node name.**)

computer network - a group of computer systems connected in such a way that they can exchange information and share resources.

configuration - the process of defining the characteristics of the network in software. Two kinds of configuration must be performed for each node in an NS3000/V network: system configuration, accomplished with the SYSDUMP utility, and network configuration, accomplished with the NMMGR utility.

CSLIST - a utility that lists version numbers for the software modules of the CS/3000 subsystem, and provides information on the INP and LANIC download files.

CSDUMP - A utility that formats files created by the CSTRACE utility, which traces link activities.

D

DADCONF.PUB.SYS configuration file - a file that must be installed to allow initialization of purchased network services. This file must be created after system configuration has occurred, and the SYSDUMP tapes are loaded, and before network configuration with NMMGR.

data screen - a type of screen displayed by NMMGR that allows you to configure data.

Dial ID protocol - a proprietary Hewlett-Packard protocol that provides security checking and address exchange for dial links.

dial link - a connection between network nodes made through pubic telephone lines.

direct path branching - the process of proceeding from screen to screen in NMMGR by entering path names in the `Path:` field.

directly connected - describes nodes that are members of the same network.

distributed network - a computer network in which connected systems are independent and equally in control of the network's operation.

DSLIST - a utility that lists the version numbers of software modules that are part of the DS subsystem of NS3000/V.

DSDUMP - a utility that can format link trace files created with CSTRACE. Formatting with DSDUMP rather than CSDUMP provides more information on the data link and network level activities when the BSC or DS X.25 protocols are in use.

DSM - the INP Diagnostic Support Monitor, a utility used to test INP cards.

driver - software that controls I/O devices, including NS3000/V links.

E

entry, network directory - the data in a network directory that consists of a node's name and its path report list.

entry priority - the ranking used to identify the most desirable to the least desirable routes used to reach a given remote node from a given local node in a router network.

environment - a session established on a remote node.

F

full gateway - a node that is a member of more than one network. Because it is a member of more than one network, the node can pass messages from one network to another.

G

gateway - a node that can provide communication between networks. A gateway can be either a full gateway or a gateway half.

gateway half - a node that, in conjunction with another node on another network, performs the function of a gateway. Together, two gateway halves can link two networks in the same catenet.

gateway half link - A link that joins two nodes that form a gateway half pair. The link must be configured as the link used for the gateway half network interface at each of the two nodes. The NS Point-to-Point 3000/V Link and the Asynchronous SERIAL Network Link can be used as gateway half links.

gateway half pair - a set of two nodes that are joined by a gateway half link. Each node in the pair must have a gateway half network interface configured, using that link.

gateway node - a node used to connect networks in the same catenet. To be a gateway node on an NS3000/V network, a node must be configured as part of more than one network, or must be configured as a gateway half in conjunction with another gateway half node on another network.

general protocols - protocols used by an NS3000/V node regardless of link type: these are the IPU (IP Update), TCP, and PXP protocols.

global entry - a network directory entry that can be merged into other directories, and that can therefore be used by other nodes in the network.

Guided Configuration - a method of configuring a node in which a subset of the complete NMMGR interface is presented, and in which defaults of configurable values are used automatically.

H

hop count - see **internet hop count** and **intranet hop count.**

I

IEEE 802.3 - a standard for a broadcast local area network published by the Institute for Electrical and Electronics Engineers (IEEE).

IEEE 802.3 networks - networks whose operation is based on the IEEE 802.3 standard for local area networks. ThinLAN/3000, LAN3000/V, and StarLAN/3000 are NS3000/V links that can be used to create IEEE 802.3 networks.

inbound - pertaining to data being received at a given node.

INPDPAN - INP Dump Analysis, a utility that generates a formatted dump of an INP log file. An INP log file is generated when an error occurs; it contains the contents of the INP memory.

Intelligent Network Processor (INP) - a hardware card that fits into the backplane of the HP 3000 and provides a physical layer interface for NS Point-to-Point and NS X.25 3000/V Links. The INP is also used for DS links; however, a single INP cannot provide an interface for an NS link and a DS link concurrently, unless the Software Switch is used.

internet communication - communication that occurs between networks.

Internet Protocol (IP) - a protocol used to provide routing among different networks in the same catenet, as well as among nodes in the same router network. The Internet Protocol corresponds to layer 3, the Network Layer, of the OSI model.

internet hop count - the number of gateways that are used to route a message to its destination network.

intranet communication - communication that occurs between nodes in a single network.

intranet hop count - the number of intermediate nodes that lie between a source and destination node on the same router network.

IP address - an address used by the Internet Protocol to perform internet routing, and used to provide intranet addresses in NS3000/V router networks.

IP - see **Internet Protocol.**

IPC line test - a software program that tests whether the Network Transport is operating correctly.

L

LANDPAN - LANIC Dump Analysis, a utility that generates a formatted dump of a LANIC log file. The LANIC log file is produced when an error occurs; it contains the contents of the LANIC memory.

LANIC - see **Local Area Network Interface Controller**

LANIC Self-Test - a ROM-based program on the LANIC card that tests and reports the status of LANIC hardware.

LANDIAG - LAN Node Diagnostic, an interactive utility designed to help identify malfunctioning hardware components of the LAN3000/V Link.

LAP-B (Link Access Procedure, Balanced) protocol - a data link layer protocol that can be used by NS Point-to-Point and NS X.25 3000/V Links. LAP-B must be used over direct-connect NS Point-to-Point or NS X.25 3000/V Links.

leased line - data-grade telephone lines leased directly to a subscriber and allocated specifically for the subscriber's needs.

lifecycle, network/catenet management - the stages of development of a network or catenet. The four stages consist of design, implementation, operation, and tuning and growth.

line speed - a measure of the rate at which data passes through a physical link (usually measured in bits or kilobits per second).

link product - one of the NS3000/V Links: the NS Point-to-Point 3000/V Link, the NS X.25 3000/V Link, the ThinLAN/3000 Link (including the LAN3000/V thick cable option), and the Asynchronous SERIAL Network Link. Each link product consists of software and hardware that together perform the functions of layers 1 through 4 of the OSI 7-layer networking model.

Link Support Services - an NS3000/V software subsystem that opens, closes, and otherwise controls physical links.

Local Area Network Interface Controller (LANIC) - a hardware card that fits into the backplane of the HP 3000 and provides a physical layer interface for IEEE 802.3 local area networks.

local entry - a network directory entry that cannot be distributed to other nodes in the network.

local node - the node (computer that is part of a network) that you are currently using or referring to.

logging - the process of recording the usage of network resources. NS3000/V logging is performed at three levels: network, event, and link levels.

log class - a designation indicating the subset of information that will be logged.

loopback - the routing of messages originating from a node to that node itself.

M

manual dial - describes a dial connection in which the remote node's telephone number must be physically entered by someone such as the system operator.

map, catenet or network - a drawing of a network or catenet that shows its topology, node and network names, addresses, network boundaries (for a catenet map), and link types.

mapping - A set of characteristics that describe a route taken by messages to reach a destination node. This set of characteristics is configured with NMMGR at every node on a router network. One mapping is configured at each node for every other node on the network to which messages will be sent.

MAU - Media Attachment Unit, a device attached to a coaxial cable for a LAN3000/V Link. The MAU provides physical and electrical connection from the AUI cable to the coaxial cable.

menu screen - a type of screen displayed by NMMGR that allows you to select an NMMGR function, such as proceeding to a certain configuration branch, refreshing the screen display, or returning to the previously displayed screen.

MPE-V - Multiprogramming Executive V, the operating system of Series 37 through 70 HP 3000s and MICRO 3000s. NS3000/V operates in conjunction with the MPE-V operating system.

N

name - In the context of NS3000/V networks, a name is a character string used to identify some portion or component of a network or catenet.

neighbor gateway - a gateway that is in the same network as a given node.

neighbor node - a node that is in the same network as a given node.

NetIPC - Network Interprocess Communication, software that enables programs to access network transport protocols.

network - see **computer network**.

network address - the network portion of the IP address. The IP address consists of a network portion and a node portion.

network architecture - the plan that defines the characteristics and interactions of the hardware and software used to create a network.

network boundary - the logical division between networks in a catenet.

network directory - a file containing information required for one node to communicate with another node in the catenet. The active network directory on a node must be named NSDIR.NET.SYS.

network interface - the collective software that provides an interface between a system and a network. A node will possess a network interface for each of the networks to which it belongs, and for each of the gateway half links of which it is a part.

network management - the collective tasks required to design, install, configure, maintain, and if necessary, change a network.

network management lifecycle - see **lifecycle, network/catenet**

network manager - the person responsible for performing and coordinating network management tasks for an entire network.

Network Management (NM) Protocol - the network management procedure which is responsible for IP and link level echo and protocol echo testings. (Used by the OpenView NS Monitor Applications products.)

Network Transport - software that corresponds to layers 3 and 4 of the OSI network architecture model. The function of this software is to send data out over the appropriate communications link, and to receive incoming data, and to route the incoming or outgoing data to the appropriate destination node.

NFT - Network File Transfer, the Network Service that transfers disc files between nodes on the network.

NMCONFIG.PUB.SYS configuration file - The configuration file containing information needed for link level and NetIPC logging.

NMDUMP - a utility used to format log and trace files.

NMMAINT - a utility that lists the software module version numbers for all HP AdvanceNet products, including NS3000/V.

NMMGR - see **Node Management Configurator**

NM protocol - see **Network Management (NM) protocol**

node - a computer that is part of a network.

node address - the node portion of the IP address, which consists of a node portion and a network portion.

Node Management Configurator (NMMGR) - an NS3000/V software subsystem that enables you to configure each node on a network.

Node Management Services - an NS3000/V software subsystem that provides configuration file version checking and logging.

Node Manager - the person responsible for performing network management tasks for a node on a network.

node name - a character string used to identify each system that is a node in a network or catenet. Each node name in a network or catenet must be unique; however, a single node can be identified by more than one node name.

non-adjacent - describes a node on a router network that is separated from a given node by intervening, or intermediate nodes.

NSCONF.NET.SYS - Default name, and name recommended by HP for the network transport configuration file residing on each node. This name is used throughout the *NS3000/V Network Manager Reference Manual* to refer to this configuration file.

NSDIR.NET.SYS - Name of the active network directory file. For convenience, this name is used throughout the *NS3000/V Network Manager Reference Manual* to refer to the network directory file.

NS3000/V - a Hewlett-Packard data communications product that provides networking capabilities for HP 3000 minicomputers. NS3000/V also provides communication between HP 3000s and other types of computers.

NS3000/V Network Services - software applications that can be used to access data, initiate processes, and exchange information among nodes in the network. The services are: RPM, VT, RFA, RDBA, NFT, and PTOP.

NS3000/V Link - software and hardware that provides the connection between nodes on a network. Four NS3000/V links are available: the NS Point-to-Point 3000/V Link, the NS X.25 3000/V Link, the ThinLAN/3000 Link (including the ThickLAN thick cable option), StarLAN/3000 Link, and the Asynchronous SERIAL Network Link.

O

outbound - pertaining to data being sent from a given node.

OpenView NS Monitor Applications - HP products that provide network management functionality when used in conjunction with the OpenView Core software and the Network Control Server (NCS) software (provided with FOS).

OSI (Open Systems Interconnection) model - a model of network architecture devised by the International Standardards Organization (ISO). The OSI model defines seven layers of a network architecture, each layer performing specified functions.

P

Packet Exchange Protocol (PXP) - a Transport Layer protocol used in NS3000/V links to initially establish communication between nodes when the NetIPC socket registry is used.

partner gateway half - two gateway halves connected by a link and configured to provide communication between two networks are partner gateway halves.

path name - in configuration with NMMGR, a string that can be typed in the "Path:" field of NMMGR display screens that causes another screen to appear. Each screen has a unique path name that corresponds to its location in the hierarachy of configuration screens presented by NMMGR.

path report - a data structure containing name-to-address mapping information for a node, as well as the networking protocols used by the node for its interface to a given network.

path report list - the set of all path reports for a node, which includes path reports for all the networks the node belongs to.

point-to-point network - networks in which messages are transmitted from node to node in the network over a defined route until reaching their destination.

pools - virtual terminals are configured in groups called **pools**. A pool of virtual terminals is shared by all NS communications devices on a system.

Probe protocol - an HP protocol used by nodes on NS3000/V IEEE 802.3 networks to obtain information about other nodes on the network.

Probe proxy server - a node on an IEEE 802.3 network that possesses a network directory and is therefore used to provide information about nodes on other networks in the catenet to nodes on the IEEE 802.3 network.

protocols - the rules and conventions that define the functions to be performed and the format of messages exchanged by each layer of network architecture.

PTOP - Program-to-Program Communication, the Network Service that allows programs residing on different nodes to exchange information with one another in a master/slave relationship.

PXP - see **Packet Exchange Protocol.**

Q

QuickVal - a software program that tests whether Network Services are operating correctly between nodes.

R

RDBA - Remote Data Base Access, the Network Service that allows users to access data bases on remote nodes.

reachable network - a network that can be accessed (with additional internet hops possibly required) by a particular gateway.

remote node - a node on the network other than the node you are currently using or referring to.

resolution, of names and addresses - see **address resolution.**

RFA - Remote File Access, the Network Service that allows users to access files and devices on remote nodes.

RPM - Remote Process Management, the Network Service that allows a process to programmatically initiate and terminate other processes throughout a network from any node on the network.

router network - one of the types of networks that can be created with NS3000/V link products. Router networks are point- to-point networks. The NS Point-to-Point 3000/V Link and the Asynchronous SERIAL Network Link can be used to create router networks.

routing - the process used to determine the path that packets, or pieces of a message, take through a network or catenet to reach a destination node.

S

security string - an alphanumeric character string that functions as a password for dial links. The security string is used by the Dial ID protocol.

select screen - a type of screen displayed by NMMGR that allows you to select identifiers (such as names) to add, delete, rename, or update.

shared dial - describes a dial link that provides connection to more than one remote system, although to only one at a time.

station address - a 48-bit link-level address used by the IEEE 802.3 and Ethernet protocols that is assigned to every node on an IEEE 802.3 network.

store-and-forward - a technique in which messages are passed from one node to another in a network to reach their destination. Point-to-point networks use the store-and-forward technique to transmit messages.

subnet - another name for a network, especially if the network is part of a catenet. The word **subnet** is also a synonym for **intranet**.

synchronization - the process of creating and modifying network directories so that all directories in the network or catenet are identical or at least contain the information required for the network to operate as planned.

SYSDUMP - the software program that allows you to perform system configuration on HP 3000s.

system configuration - the means of defining to MPE-V the peripheral devices attached to the HP 3000 for the input and output of data, and the parameters required for system operation.

T

TCP - see **Transmission Control Protocol.**

TERMDSM - a utility used to diagnose problems with the ATP. The ATP is a card installed in the HP 3000 for Asynchronous SERIAL Network links.

topology - the physical arrangement of nodes in a network. Some common topologies are bus, star, and ring.

Transmission Control Protocol (TCP) - a network protocol that establishes and maintains connections between nodes. TCP regulates the flow of data, breaks messages into smaller fragments if necessary (and reassembles the fragments at the destination), detects errors, and retransmits messages if errors have been detected.

U

utility, NMMGR - NMMGR utilities, which are accessed from the UTILITY menu screen, consist of Output Configuration File, Compress, Validate, and Configuration Subtree Copy utilities.

V

validation - the process of ascertaining whether the transport configuration file has been correctly configured. This is accomplished by using the NMMGR Validate Configuration File screen.

virtual terminal - software that simulates the function of a terminal to MPE. Devices that provide incoming data to a system require the configuration of virtual terminals.

VPLUS - software used to generate screens such as those displayed by NMMGR.

X

XPT line test - a software program that tests whether the Network Transport is operating correctly.

X.25 Address - is the X.25 (subnet) address provided by the network administration if you are connectedto a Public Data Network.

X.25 Address Key - a label identifying a node and its associated X.25 parameters.

INDEX

Part No. 32344-90012
Printed in U.S.A. 05/87
U0788

HEWLETT
PACKARD

Part No. 32344-90012
Printed in U.S.A. 05/87
E0587

HEWLETT
PACKARD