4DDN Network Manager's Guide

IRIS-4D Series

SiliconGraphics
Computer Systems

Document number: 007-0821-010

4DDN Network Manager's Guide

Version 1.0

Document Number 007-0821-010

9/90

Technical Publications:

Kevin Walsh
Diane Wilford

Engineering:

Vernon Schryver
Andrew Cherenson

© Copyright 1986, Technology Concepts Inc., Sudbury, MA.
© Copyright 1988, Silicon Graphics, Inc. - All rights reserved

This document contains proprietary and confidential information of Silicon Graphics, Inc., and is protected by Federal copyright law. The contents of this document may not be disclosed to third parties, copied or duplicated in any form, in whole or in part, without the express written permission of Silicon Graphics, Inc.

This manual has been adapted from the *CommUnity-UNIX Network Manager's Guide* by Technology Concepts, Inc. This material may be changed without notice by Technology Concepts and Silicon Graphics, Inc. Technology Concepts Inc. is not responsible for any errors that may appear herein.

U.S. Government Limited Rights

Use, duplication or disclosure of the technical data contained in this document by the Government is subject to restrictions as set forth in subdivision (b) (2) of the Rights in Technical Data and Computer Software clause at 52.227-7013. Contractor/manufacturer is Silicon Graphics Inc., 2011 N. Shoreline Blvd., Mountain View, CA 94039-7311.

4DDN Network Manager's Guide
Version 1.0
Document Number 007-0821-010

Silicon Graphics, Inc.
Mountain View, California

The words IRIS, Geometry Link, Geometry Partners, Geometry Engine and Geometry Accelerator are trademarks of Silicon Graphics, Inc.

UNIX is a trademark of AT&T Bell Laboratories.
IRIX is a trademark of Silicon Graphics, Inc.
DECnet, RSX, ULTRIX, VAX and VMS are trademarks of Digital Equipment Corporation.

Contents

1. **Introduction** 1-1
   - 1.1 Structure of the Guide 1-1
   - 1.2 Conventions 1-2
   - 1.3 Related Documentation 1-3
   - 1.4 Product Support 1-3

2. **General Information** 2-1
   - 2.1 Networking Terminology 2-2
   - 2.2 The Configuration Database 2-6
   - 2.3 Server Processes 2-7
   - 2.4 Default Accounts 2-8
   - 2.5 Proxy Log-in 2-10
   - 2.6 Configuring the 4DDN Software 2-11

3. **Network Management** 3-1
   - 3.1 Invoking NCP 3-2
   - 3.2 NCP Command Syntax 3-3
   - 3.3 Remote Network Management Mode 3-4
   - 3.4 NCP Responses 3-5
     - 3.4.1 Error Messages and Meanings 3-5
   - 3.5 Exiting NCP 3-8

4. **Configuring and Controlling the Network** 4-1
   - 4.1 Node Commands 4-2
     - 4.1.1 Node Parameters 4-2
     - 4.1.2 Executor Node Commands 4-3
     - 4.1.3 Copying Between Databases 4-4
     - 4.1.4 Node Identification Commands 4-6
     - 4.1.5 Setting the Local Node State 4-7
     - 4.1.6 Resetting Node Counters 4-7
   - 4.2 Circuit Commands 4-8
   - 4.3 Line Commands 4-9

5. Monitoring the Network 5-1
5.1 Display Types 5-1
5.2 Monitoring Nodes 5-2
5.3 Monitoring Circuits 5-3
5.4 Monitoring Lines 5-3

6. Testing the Network 6-1
6.1 Loop Parameters 6-1
6.2 Local Loopback Test 6-2
6.3 Remote Loopback Test 6-3

7. NCP Reference Guide 7-1
7.1 CLEAR EXECUTOR NODE 7-2
7.2 DEFINE EXECUTOR ALL 7-2
7.3 DEFINE KNOWN NODES ALL 7-2
7.4 DEFINE NODE 7-3
7.5 EXIT 7-3
7.6 LOOP EXECUTOR 7-4
7.7 LOOP NODE 7-5
7.8 SET EXECUTOR ADDRESS 7-6
7.9 SET EXECUTOR NAME 7-6
7.10 SET EXECUTOR STATE 7-7
7.11 SET EXECUTOR ALL 7-7
7.12 SET EXECUTOR NODE 7-8
7.13 SET KNOWN NODES ALL 7-8
7.14 SET NODE 7-9
7.15 SET NODE ALL 7-9
7.16 SHOW NODE 7-10
7.17 SHOW CIRCUIT 7-25
7.18 SHOW LINE 7-30
7.19 TELL 7-34
7.20 ZERO EXECUTOR COUNTERS 7-34
7.21 ZERO KNOWN CIRCUIT COUNTERS 7-34
7.22 ZERO KNOWN LINE COUNTERS 7-34
7.23 ZERO KNOWN NODES COUNTERS 7-34
7.24 ZERO NODE COUNTERS 7-35

A. Default Values for Parameters A-1
A.1 Node Parameters A-1
A.2 Loop Parameters A-1

1. Introduction

IRIS-4DDN is a software communications product that enables your IRIS-4D workstation to communicate on a DECnet network as an Ethernet end node.

The information in this document is intended for network managers and system managers. This document provides an overview of the IRIS-4DDN network management software, explains how to configure the network and create the network database, explains NCP, describes how to diagnose network problems and test the network, and lists the NCP commands alphabetically. This document assumes that the audience is familiar with some networking concepts. However, concepts and technical terms are defined in chapter 2.

1.1 Structure of the Guide

This guide consists of seven chapters and an appendix.

Chapter 1: This chapter contains the introduction.

Chapter 2: This chapter contains general information about 4DDN, including configuration instructions, and defines some networking concepts.

Chapter 3: This chapter explains NCP, how to invoke or exit NCP and provides the syntax rules of NCP commands.

Chapter 4: This chapter explains how to set parameters and control the DECnet/4DDN network.

Chapter 5: This chapter explains how to monitor the DECnet/4DDN network.

Chapter 6: This chapter explains how to test the DECnet/4DDN network.

Chapter 7: This chapter lists the available NCP commands alphabetically for easy reference.

Appendix A: This appendix lists the default values for parameters.

1.2 Conventions

This document uses the standard IRIX™ convention when referring to entries in the IRIX documentation. The entry name is followed by a section number in parentheses. For example, *cc*(1) refers to the *cc* manual entry in Section 1 of the *IRIS-4D User's Reference Manual, Volume 1*.

In command syntax descriptions and examples, square brackets surrounding an argument indicate that the argument is optional. Variable parameters are in *italics*. You replace these variables with the appropriate string or value.

In text descriptions, filenames and IRIX commands are also in *italics*. Names of variables and error codes used in the VMS environment are in uppercase.

Other conventions used in the DECnet environment are listed below:

CAPITAL	Words that are capitalized and in bold type are commands that must be entered by the user. Press Return after making the entry.
NCP>	Screen prompt
node-name	Multi-word variables are hyphenated.

1.3 Related Documentation

Silicon Graphics, Inc. *4DDN User's Guide*	007-0820-010
Digital Equipment Corporation *DECnet Digital Network Architecture (Phase IV) General Description*	DEC part #: *AA-N149A-TC*
Digital Equipment Corporation *Guide to Networking on VAX/VMS*	DEC part #: AA-Y512A-TE

1.4 Product Support

Silicon Graphics, Inc., provides a comprehensive product support and maintenance program for IRIS products. For further information, contact your service organization.

2. General Information

IRIS-4DDN is a comprehensive, packet-oriented networking software product that allows various computer systems to share information and data. 4DDN is based on the Digital Network Architecture (DNA) structure and protocols, which makes it completely compatible with DECnet Phase-IV. The 4DDN software enables IRIS-4D Series workstations to participate on a DECnet Phase-IV network as ethernet end-nodes.

2.1 Networking Terminology

Node

A node is a computer using DNA protocols to communicate with other computers across a network. The data path connecting two nodes is determined by a network function called routing.

Designated Router

Routers are nodes that can forward or route messages from one node to another. When there is more than one routing node on an ethernet, one of them is selected as the designated router. The designated router can be either an area router (one capable of routing between different areas) or a local router (one capable of routing only within an area.)

Endnode

When a node has a single circuit connecting it to the rest of the network, it is called an endnode. An endnode can send packets from itself to any other DECnet or 4DDN node and receive packets addressed to itself from other Phase-IV nodes. An endnode is a non-routing node; it cannot route packets.

Local Node

The node at which you are physically located is called the local node.

Remote Node

All other nodes in relation to the local node are considered remote nodes.

Executor Node

This is the node at which NCP (Network Control Program) commands are executed. The executor node is usually the local node. However, through the NCP command SET EXECUTOR any remote node can be designated as the executor node, and thus, any NCP commands issued on the local node are executed on the remote node.

Reachable or Active Node

A node is considered reachable or active when it is available for connection requests. A node is unreachable or inactive when it is powered down or unavailable for connection requests.

Adjacent Node

A node that is connected to the local node by a single physical line is called adjacent. All nodes on a single ethernet line are considered adjacent.

Node State

The operational state of the local node on the network can be controlled. Use this control to restrict the operation of the node or shut it down. When the state is set to OFF, the node is unreachable. When the state is set to ON, the node is active or reachable.

Circuits

Circuits are logical communications data paths between nodes. They operate over physical lines.

Ethernet circuits enable multi-access connection between a number of nodes on the same physical medium. Each node is considered adjacent to every other node on the circuit and equally accessible. Each node is identified by an ethernet address (See the DECnet documentation for a complete definition of an ethernet address.)

Lines

Lines are physical data paths between nodes and are the lowest level communications path. The ethernet line physically connects the different nodes in the local network.

Counters

Counters are performance variables that track various events in a network. Information obtained from counters may be useful in measuring the performance and throughput for a given circuit.

There are counters tracking performance on nodes, circuits, and lines.

Note that node counters are only available after a connection has been attempted between the executor and the specified node. If there has been no logical link established between the executor and the specified node, the SHOW NODE <node-id> COUNTERS command returns with the message, NO INFORMATION AVAILABLE.

Node Identification

The following information identifies a node:

Node Address — A unique number assigned by the network manager; it must conform to the following format:

```
area-number.node-number
```

where

The area number is a group of nodes in the network that can run independently as a subnetwork. Each area has a unique number in a network. The area number must be an integer in the 1-63 range. The area number defaults to the local area if none is provided.

The node-number must be an integer in the 1-1023 range. Node numbers must be unique to the specific network area.

Node Name — One node name is allowed for each node. The node name can have up to 6 characters (letters and/or numbers). The first character must be a letter. The node name is not case-sensitive. Lowercase letters are converted to uppercase.

The address and name for the local and the remote nodes are stored in the configuration database. The configuration database for the local node must contain information about the local node and the remote nodes.

2.2 The Configuration Database

The information each node has about the network is stored as parameters in a configuration database. These parameters identify the local node, the remote nodes, the local ethernet line, the circuit associated with the ethernet line, and the state. The 4DDN configuration database consists of two different databases, one permanent and one volatile.

The permanent database is stored on disk and specifies the initial values for the volatile database. Changes in the volatile database can be copied to the permanent database. The permanent database can be copied to the volatile database and vice-versa.

The volatile database resides in memory and controls the operation of the network without modifying the permanent database. When the system is booted, the volatile database is automatically generated. It is usually a copy of the permanent database. The parameters in the volatile database can be changed while the system is running and these changes are effective until further modification or network shutdown. Commands are available to copy the volatile to the permanent database and vice-versa.

2.3 Server Processes

Servers are processes that provide network services. The standard server processes that are provided with the 4DDN system are the master server (dnserver), the File Access Listener (FAL), the Network Management Listener (NML), and the Virtual Terminal Server (sethostd). These processes (or their equivalents) also reside on remote nodes in the 4DDN/DECnet network. The following is a short description of these servers and the protocols that allow them to operate with other nodes.

The master server (dnserver) is a process that runs on the 4DDN system. It invokes the other servers when requests for connections to other servers are received. Dnserver is a continuously running UNIX process.

FAL, the File Access Listener, is the server that enables remote users to access files on the system. Specifically, FAL provides the local file system with network file access functions such as file copies and directory listings. FAL uses DAP (Data Access Protocol) to communicate with processes on other nodes.

NML, the Network Management Listener, is the server that enables the use of NCP, the Network Control Program. Together with NCP, the NML enables you to determine the status of a network, zero lines and counters, and perform other network management tasks. NML executes NICE (Network Information and Control Exchange) protocol messages that it receives from NCP. The responses to the protocol messages are returned to the appropriate NCP for display.

Sethostd, the virtual terminal server, is the process that enables a user on one node to remotely log in to another node. The server on one node responds to requests from the SET HOST (or sethost) command of another node. SET HOST uses the CTERM protocol to communicate with sethostd.

2.4 Default Accounts

All DECnet and 4DDN logical link connection requests contain access information (a user name and password). This access information is validated by the targeted server at its discretion. On 4DDN nodes, the dnserver verifies the access information before the targeted server is started and the logical link is accepted. Dnserver uses the */etc/passwd* file to perform this verification.

Users do not need to explicitly specify usernames and passwords for all network-related functions. In such cases, defaults are used, and because the use of these defaults permits easy access to data, the defaults should be set up to have no privileges and very few access rights.

On 4DDN nodes, the initiating (client) side does not default these fields (they are left as null), and the dnserver defaults them to "decnet" and no password. For example:

```
% dnls pluto::/usr/bin
```

In this example, the username "decnet" is filled in automatically by dnserver and no password is required.

On VMS systems, the defaults are filled in on the initiating (client) side. These defaults are stored in the NCP databases on each node and can be set with the NCP commands:

```
NCP> SET NODE <nodename> NONPRIV USER <username>
NCP> SET NODE <nodename> NONPRIV PASSWORD <password>
```

DECnet uses one other account, a privileged account for certain network management functions. These are setup on the VAX using the NCP commands:

```
NCP> SET NODE <nodename> PRIV USER <username>     ! default = decnet
NCP> SET NODE <nodename> PRIV PASSWORD <password> ! default = DECNET
```

If this account is not created, then certain network management functions do not work.

With the existence of the default decnet account, the following commands issued on a VAX/VMS node are equivalent:

```
$ DIR pluto::usr:[bin]
$ DIR  pluto"decnet DECNET"::usr:[bin]
```

In the first remote directory listing of a 4DDN node, "decnet DECNET" is the default account and password and does not need to be entered. The VAX/VMS system has filled it in with known default information.

2.5 Proxy Log-in

Effective communication with remote computers on a network requires that users have access and knowledge of password protected accounts. This network requirement is particularly challenging since user lists on individual computer nodes are subject to change, e.g., a user is added or deleted from the list, a new password is assigned, etc. These changes in the user/account/password list must be made known to all nodes wishing to communicate with the computer (i.e. where these changes were made). The "proxy log-in" facility under VMS addresses this situation.

The proxy log-in feature maintains a local database on each node of remote network users that have proxy access to specific local accounts. The database provides a one-to-one mapping between the remote user, identified as NODE::USERNAME, and the local proxy account. An example is a logical link request having no specific access control information from a remote user whose name is in the database. In this case, the process created by NETACP to handle the logical link is run using the authorization context of the proxy account. The Proxy Log-In enables remote users to be added to or deleted from a particular local proxy account without prior knowledge of the account.

To incorporate the concept of proxy access, 4DDN provides the user's identification number (UID) as access control information in all outgoing connects. This three digit number is used by the VAX as proxy information. Since "proxy log-in" is a VMS concept and there is no exact UNIX analogy, incoming connects to 4DDN nodes do not make use of proxy information.

2.6 Configuring the 4DDN Software

Once the system is rebooted using a kernel with the 4DDN software, edit the */usr/etc/dn/nodes* to define the names and addresses of DECnet nodes, including your IRIS-4D, on your network. Then do the following:

```
su
cd  /usr/etc/dn
sh  dninstall.sh
```

The *dninstall.sh* installation script creates the NCP databases using the information in the *nodes* file. Now start 4DDN by typing:

```
sh  dnstart.sh
```

This script loads the node database and starts the *dnserver* daemon. To make sure that your machine is running 4DDN, type:

```
./ncp  show know nodes
```

Also type:

```
ps -e | grep dnserver
```

and look for the *dnserver* process. If you do not find *dnserver*, try the dnstart procedure again. If you do not succeed, contact your service organization.

To have 4DDN started automatically during system initialization, create the 4DDN configuration flag file:

```
/etc/chkconfig 4DDN on
```

The network initialization script, */etc/init.d/network* checks for this file to determine if 4DDN should be started.

To make the default account operational on 4DDN nodes, add the following entry to the */etc/passwd* file:

```
decnet:EHZFcYdsXgq3c:998:998:DECnet guest acct:/usr/tmp:/bin/login
```

The user-ID chosen for the accounts is the same as the guest account. The password is DECNET.

When accessing VMS systems, the default nonprivileged account used by 4DDN is DECNET with the password DECNET. The VMS node should set up to use these values:

```
NCP> SET NODE <4DDN-nodename> NONPRIV USER DECNET
NCP> SET NODE <4DDN-nodename> NONPRIV PASSWORD DECNET
```

The file */usr/etc/dn/servers.priv* contains the default and privileged account names. The default account name is assigned to remote programs that provide a null username in their access information. The privileged account is an IRIX account, such as root, that has super-user privilege (user ID number is 0). Remote programs that provide this name as the username are denied access unless this name is in the *servers.priv* file. The format of the file is:

```
null   default-account
priv   superuser-account
```

Both entries must be defined or the contents are ignored.

3. Network Management

Through the Network Control Program (NCP), IRIS-4DDN provides a set of interactive commands to configure, control, monitor, and test the network and ensure its effective operation.

To use the NCP DEFINE, LOOP, SET, and ZERO commands, super user privileges are required. All users can use the NCP SHOW commands.

NCP enables the following operations:

- Changing parameters (SET and DEFINE commands)
- Zeroing counters (ZERO command)
- Gathering information to monitor the operation of the network (SHOW command)
- Testing the network (LOOP command)

3.1 Invoking NCP

To invoke NCP, type:

```
/usr/etc/dn/ncp
```

and press the <Return> key. NCP displays the "NCP>" prompt on the next line. In this document, examples of interactive commands are shown with the "NCP>" prompt. For example:

```
Prompt  User types

NCP>    SHOW KNOWN CIRCUIT SUMMARY
```

It is also possible to enter a single NCP command without actually entering the NCP program first. To do this, type NCP followed by the command, all on the same line. For example:

```
/usr/etc/dn/ncp show executor counters
```

3.2 NCP Command Syntax

NCP commands are not case-sensitive except when specifying a user ID or password. (Lowercase letters are automatically converted to uppercase internally.) They consist of a command keyword, an entity, and one or several entity-options qualifying the command and supplying additional information. For example:

```
NCP> SHOW KNOWN NODES SUMMARY
```

Where

NCP>	is the screen prompt and is not part of the command.
SHOW	is the command keyword specifying the operation.
KNOWN NODES	is the entity and specifies the component (plural in this case) to which the operation applies.
SUMMARY	is the entity-option. There are four available entity-options – CHARACTERISTICS, SUMMARY, STATUS and COUNTERS. SUMMARY is the default.

Some NCP commands may require more than one line on the screen. To continue a command line on the next line, a hyphen must be used, as shown below:

Example:

```
NCP>SHOW KNOWN NODES -
SUMMARY
```

A space is required before the hyphen if it separates two words.

Command keywords can be abbreviated, e.g., CHAR for CHARACTERISTICS or COUNT for COUNTERS.

For more information on NCP commands and screen output, refer to Chapter 7, NCP Reference Guide.

3.3 Remote Network Management Mode

Remote network management is used to connect to any node in the network, examine its parameters and counters, zero its counters, and perform loop testing.

To switch to remote network management mode, enter the following NCP command:

```
NCP>SET EXECUTOR NODE node-id
```

Where node-id is a node name or node address.

NCP forwards any commands it receives to the node specified in node-id until you type a new SET EXECUTOR NODE or CLEAR EXECUTOR NODE command.

TELL NODE Command

Alternatively, use the TELL prefix to identify the executor node for a particular NCP command. TELL sets the executor for only one command and must prefix the command for which it is intended. For example:

```
NCP>TELL pluto SHOW EXECUTOR CHAR
```

This command sets the executor to node pluto where the SHOW EXECUTOR CHAR executes.

3.4 NCP Responses

Depending on the command entered, NCP responds by displaying one of the following:

- The requested data after a SHOW command,
- The prompt "NCP>" after a SET or DEFINE command, or
- One of the error messages described in the next section after a SET or DEFINE command.

3.4.1 Error Messages and Meanings

NCP generates error messages for a number of system-level and network-level error conditions. Most of these messages include a brief description of the error and a flag specifying the parameter that produced the error. The invalid parameter is enclosed by slashes (/). For example:

```
NCP>SHOW FOO
NCP: Syntax Error
SHOW /FOO/
```

The following is a list of error messages generated by NCP:

Unimplemented Function or Option

The NCP command is not supported by the implementation.

System-specific management function not supported

The NCP command is not supported by the implementation.

Unimplemented Parameter Type

NCP command parameter is not supported by the implementation.

Incompatible Management Version

Tried to connect to an earlier version of Network Management (lower than 4.0.0).

Unrecognized Component

No database entry for the indicated entity.

Component in Wrong State

This message may be generated when a user tries to change the node address while the operational state of the node is set to ON.

Rejected Access Control, Privilege Violation

These messages may be generated when a user attempts to perform an operation to which he has insufficient privilege.

Object Not Found

The Network Management Listener (NML) was not found on the remote node.

Invalid Parameter Value

The parameter value sent to NML was not acceptable.

Unrecognize Node Name

The node name is missing from the NCP database.

Privilege Violation

The user did not have super-user privilege to modify the NCP database.

Link Access Failure

The link to the remote node cannot be established.

Link Access Failure

The link to the remote node cannot be established.

Management Program Error

4DDN software error.

3.5 Exiting NCP

To exit NCP, type ^D (control-D) or type EXIT after the "NCP>" prompt and press the return key. The EXIT command is listed in Chapter 7, NCP Reference Guide.

4. Configuring and Controlling the Network

This chapter explains how to configure and control the DNA network using NCP commands.

These commands are divided into the following groups:

- Node commands
- Circuit commands
- Line commands
- Logical Link commands

For additional information on NCP commands, refer to Chapter 7, NCP Reference Guide.

4.1 Node Commands

To establish a system running IRIS-4DDN as a node in the DECnet network, a node database must be built on the IRIS. This section explains how to identify the executor node and the remote nodes and supply the node parameters that are required to build the operational network node database.

4.1.1 Node Parameters

The following command parameters apply to local and remote nodes: (See the "Node Identification" section in Chapter 2 for the definition of node address and name formats.)

- NODE ADDRESS – Specifies the node by its DECnet address.
- NODE NAME – Specifies the node by its DECnet host name.

The following parameters apply to the local node only:

- EXECUTOR STATE ON – Allows new logical links to be created. ON is the normal operational state of a node on the network.
- EXECUTOR STATE OFF – Prevents the creation of new logical links and terminates the existing links.

Rules For Supplying Node Parameters

- For the local node, use the SET EXECUTOR command to specify node parameters. For a remote node, use the SET NODE command, which is described in Chapter 7.
- The operational state of the local node must be OFF to change the NODE ADDRESS parameter for the local node.
- Do not mix parameters applying to two different nodes in the same command.

4.1.2 Executor Node Commands

There are two major parts to network management: NCP, which serves as the requester of information and services; and NML (Network Management Listener), which services these requests. The NCP and NML communicate via DECnet and, therefore, do not have to be located on the same node. The node on which NML is resident is called the executor node.

For example, to select a node named PLUTO as the executor node for the commands to be issued, type at the local 4DDN node:

```
NCP> SET EXECUTOR NODE pluto
```

NCP commands entered at the local node are executed at the executor node, PLUTO. Each command is interpreted as if issued at the executor node. Any display of information resulting from the execution of a command appears at the local node.

Note: In remote network management, setting the executor state to OFF breaks the link. Usually this is an undesirable condition.

To reset the executor to the local node, use the following NCP command:

```
NCP> CLEAR EXECUTOR NODE
```

The local node is automatically the executor when NCP is invoked.

4.1.3 Copying Between Databases

Several commands allow users to copy the content of one database to the other database:

Copying from the Volatile to the Permanent Database

- For a single node:

  ```
  NCP> DEFINE NODE saturn ALL
  ```

  This command copies the executor's volatile data about node SATURN to its permanent database.

- For all known nodes (known to that particular node as defined in the database):

  ```
  NCP> DEFINE KNOWN NODES ALL
  ```

  This command copies, for each of the known nodes, the content of each volatile database record to the permanent database.

Copying from the Permanent to the Volatile Database

- For a single node:

```
NCP> SET NODE saturn ALL
```

  This command copies the executor's permanent database about node SATURN to its volatile database.

- For all known nodes (known to that particular node as defined in the database.)

```
NCP> SET KNOWN NODES ALL
```

  This command copies, for each of the known nodes, the content of each permanent database to the volatile database.

NOTE: When the dnstart procedure is executed, parameter values are automatically copied from the permanent database to the volatile database.

4.1.4 Node Identification Commands

When building the executor's configuration database, identify the local node and all the remote nodes. Nodes are identified by a node address and a node name. (See the section titled "Node Identification" for the definitions and rules concerning node address and node name.)

To establish the local node in the volatile database, use the following commands:

```
NCP> SET EXECUTOR ADDRESS AA.NNNN
```

where AA is the area number and NNNN is the node number.

This command specifies an address for the executor node. It should be the first command issued when initially setting the executor node parameters.

It is not necessary to specify an ethernet address for a node on an ethernet line. The software at the ethernet node sets its own ethernet address. Users do not need to know the node's ethernet address for normal network operations.

Note: When an executor's address has been defined in the volatile database, it can only be changed by setting the nodes state to an off condition and restarting the node.

The next command specifies a name for the executor node. A node name is optional, if the node address is already defined.

```
NCP> SET EXECUTOR NAME saturn
```

To insert a remote node's name into the volatile database, use the SET NODE command to specify node name and node address:

```
NCP> SET NODE 63.1 NAME sam
```

It is possible to change the local node's remote node database as follows:

```
NCP> SET NODE sam ADDRESS 63.2
```

Multiple parameters can be specified in one command. For example, for the executor node, type:

```
NCP> SET EXECUTOR NAME bill ADDRESS 63.1
```

4.1.5 Setting the Local Node State

The operational state of the local node is set by using the SET EXECUTOR command with the STATE parameter.

Possible values are OFF and ON. (See the section titled Node Parameters for details on these two different states.) For example:

```
NCP> SET EXECUTOR STATE ON
```

This command sets the executor node's operational state on the network to ON. Thus, the executor node becomes accessible from other nodes in the network.

4.1.6 Resetting Node Counters

4DDN collects statistical information about network traffic between the local node and other nodes. This information includes the number of bytes or messages sent or received. See Chapter 7, NCP Reference Guide, for a description of the SHOW command.

The counters can be reset for all known nodes or for a given node by using the ZERO command and the appropriate parameters as follows:

```
NCP> ZERO KNOWN NODES COUNTERS
NCP> ZERO NODE pluto COUNTERS
NCP> ZERO EXECUTOR COUNTERS
```

4.2 Circuit Commands

4DDN currently supports only a single ethernet circuit. The circuit name is based on the controller type. The output of the SHOW commands identifies circuits using the following format:

```
dev-c
```

where

dev is a device name, e.g., UNA on a VAX, ENP on some IRIS-4Ds.

c is an empty string or a decimal number (0 or a positive integer) identifying the controller for the device.

Resetting Circuit Counters

4DDN maintains statistical information about on the circuit performance. This information includes: the number of data packets sent; the number of data packets received, or lost over the circuit; timeouts; and the time elapsed since counters were last zeroed.

When the network is running, reset counters can be set to zero for the known circuit as follows:

```
NCP> ZERO KNOWN CIRCUIT COUNTERS
```

See Chapter 7, "SHOW KNOWN CIRCUIT COUNTERS," for a complete list and description of circuit counters.

4.3 Line Commands

4DDN currently supports only a single ethernet line. The line name is based on the controller type. The output of the SHOW command identifies lines using the circuit format described above.

Resetting Line Counters

4DDN maintains statistical information about the line's performance. This information includes: the number of bytes, data blocks sent or received; local and remote process errors; and the time elapsed since the counters were last zeroed.

When the network is running, counters can be reset to zero for the known line. For example:

```
NCP> ZERO KNOWN LINE COUNTERS
```

See Chapter 7 for the SHOW KNOWN LINE COUNTERS command, where there is a complete list and description of line counters.

5. Monitoring the Network

You can monitor the activity of the operational network by using the NCP SHOW command. The SHOW command displays static, dynamic, or counter information about nodes, circuits, or lines, as stored in the volatile database.

The selected component can be singular or plural, e.g., NODE is singular, KNOWN NODES is plural.

See Chapter 7, NCP Reference Guide, for a complete description of screen output.

5.1 Display Types

The following display types are available with the SHOW command: Characteristics, Status, Summary, and Counters.

Characteristics

Characteristics are static parameters that are specified in the volatile configuration database, e.g., software version number for each layer, identification, maximum number of links, delay information, node type, etc.

Status

Status is dynamic or changing network parameters reflecting network operations for the operational network. Status information may include: the local node and its operational state; reachable and unreachable nodes and their operational state; and circuits with their operational state. Status also may include identification of the line and its operational state.

Summary

A summary is the most useful network information compiled from both static and dynamic sources. A summary is an abbreviation of the status display type. A summary is the default option if no display type is selected.

Counters

Counters provide network information for circuits, lines, and nodes, including the local node. (See Chapter 7, NCP Reference Guide, for more details on counters.)

5.2 Monitoring Nodes

NCP displays executor node information through the following commands:

```
NCP> SHOW EXECUTOR CHARACTERISTICS
NCP> SHOW EXECUTOR COUNTERS
NCP> SHOW EXECUTOR STATUS
NCP> SHOW EXECUTOR SUMMARY
```

NCP monitors a single node through the following commands:

```
NCP> SHOW NODE pluto CHARACTERISTICS
NCP> SHOW NODE pluto COUNTERS
NCP> SHOW NODE pluto STATUS
NCP> SHOW NODE pluto SUMMARY
```

NCP monitors all active nodes. A node is considered active when it is reachable and adjacent. This is accomplished through the following commands:

```
NCP> SHOW ACTIVE NODES CHARACTERISTICS
NCP> SHOW ACTIVE NODES COUNTERS
NCP> SHOW ACTIVE NODES STATUS
NCP> SHOW ACTIVE NODES SUMMARY
```

NCP monitors known nodes (nodes known to the local node) through the following command:

```
NCP> SHOW KNOWN NODES CHARACTERISTICS
NCP> SHOW KNOWN NODES COUNTERS
NCP> SHOW KNOWN NODES STATUS
NCP> SHOW KNOWN NODES SUMMARY
```

5.3 Monitoring Circuits

NCP displays information on the known circuits (the circuits available to the local node) through the following commands:

```
NCP> SHOW KNOWN CIRCUITS CHARACTERISTICS
NCP> SHOW KNOWN CIRCUITS COUNTERS
NCP> SHOW KNOWN CIRCUITS STATUS
NCP> SHOW KNOWN CIRCUITS SUMMARY
```

5.4 Monitoring Lines

NCP displays information on the known lines (the lines available to the local node) through the following command:

```
NCP> SHOW KNOWN LINES CHARACTERISTICS
NCP> SHOW KNOWN LINES COUNTERS
NCP> SHOW KNOWN LINES STATUS
NCP> SHOW KNOWN LINES SUMMARY
```

6. Testing the Network

NCP provides the LOOP command to help you determine whether the network is operating properly.

Node-level loopback tests enable you to evaluate the operation of logical links and network software by exchanging test data between processes on two different nodes or between processes on the same node but in different layers.

If a test is successfully completed, the data messages will loop back to the source without any change. If a test fails, the messages do not return to the source or they return in a corrupted state. In this case, NCP displays an error message indicating a test failure, specifying the reason for the failure, and providing a count of the data messages that were not returned.

The loopback mirror resides above the Session layer. It allows checking of the following layers: Session, End Communication, and Routing for a local test; and Data Link if it is a local-to-remote test.

6.1 Loop Parameters

The LOOP command can be executed with various parameters:

COUNT Specifies the number of blocks sent during loopback testing over the line or node. This parameter must be an integer in the 1-65535 range. The default value is 1.

LENGTH Specifies the length in bytes of the blocks to be sent during loopback testing. This parameter must be an integer in the 1-65535 range. The default value is 40.

WITH Specifies the type of binary information to be sent during testing. The options are: MIXED, ONES, ZEROS. MIXED is the default value.

USER Specifies the user ID for access control information.

PASSWORD Specifies the password that corresponds to the user's account.

6.2 Local Loopback Test

The local loopback test evaluates the local 4DDN software using an internal logical link path.

Issue the following command:

```
NCP>LOOP EXECUTOR COUNT 5 LENGTH 40 WITH zeros
```

Figure 6-1 illustrates a local loopback test.

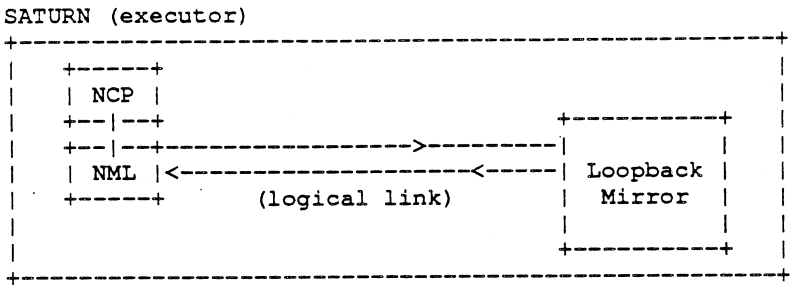

Figure 6-1. Local Loopback Testing

6.3 Remote Loopback Test

The LOOP NODE command is used to test the logical link connection between two nodes. The remote node must have been identified using the SET NODE command.

For example:

```
NCP> LOOP NODE pluto COUNT 5 LEN 40 WITH zeros -
     USER james PASSWORD bond
```

Figure 6-2 illustrates a local to remote loopback test and Figure 6-3 shows the various protocol layers involved in the test.

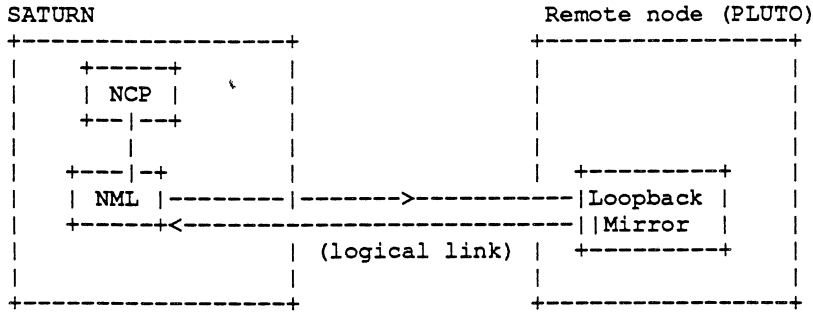

Figure 6-2. Local to Remote Loopback Testing

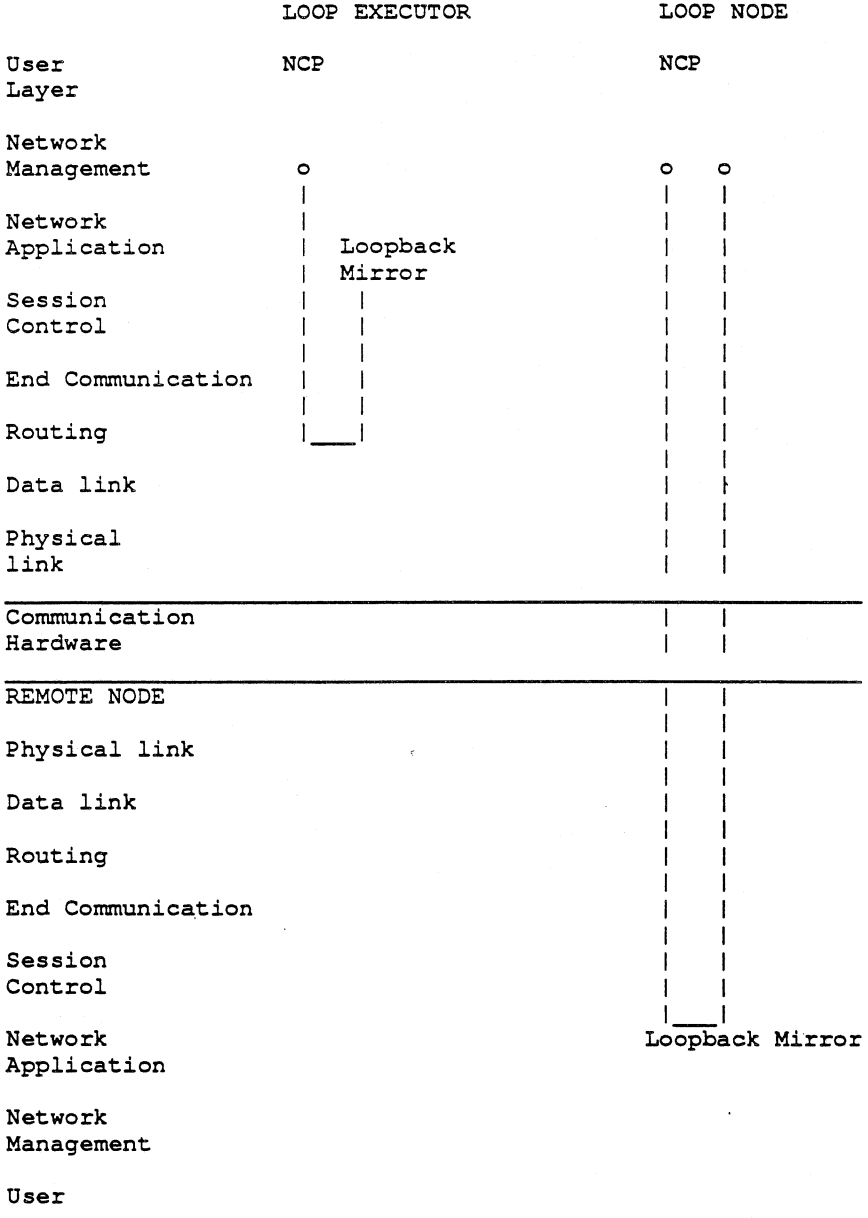

Figure 6-3. Protocol Layering in Local to Remote Loopback Testing

7. NCP Reference Guide

This chapter contains all the NCP commands currently supported by IRIS-4DDN. They are alphabetically listed for quick reference. For each command, the function, the format, and, if necessary, the output are described.

Note: The commands listed in this chapter describe the NCP commands supported by 4DDN. If the SET EXECUTOR NODE command is used to run NCP remotely on another DECnet system, the output of the NCP commands may differ from the output described here. The remote DECnet system may also contain additional NCP commands that are not described here. Refer to the appropriate DECnet documentation for more information.

Please refer to Chapter 3 for a description of NCP command syntax.

7.1 CLEAR EXECUTOR NODE

This command resets the executor to the local node. It clears the default executor node designation previously specified through the SET EXECUTOR NODE command.

7.2 DEFINE EXECUTOR ALL

This command copies the contents of the volatile database to the permanent database.

7.3 DEFINE KNOWN NODES ALL

This command copies, for each of the known nodes, the content of each volatile database record to the permanent database.

7.4 DEFINE NODE

This command copies, for the specified node, the content of the volatile database to the permanent database.

Format:

```
DEFINE NODE node-id ALL
```

7.5 EXIT

This command exits NCP. Typing ^D (control-D) also exits NCP.

7.6 LOOP EXECUTOR

This command tests the logical links within a single node. This test is performed by looping messages to the loopback mirror on the local node. Note that the count, length, and block-type are optional, but when used, must be used in the specified order, i.e., count, followed by length, followed by block-type.

Format:

```
LOOP EXECUTOR [COUNT count] [LENGTH length]
 [WITH block-type] [USER user-id PASSWORD password]
```

where

count	specifies the number of times the command is be repeated. The default is 1 and the maximum is 65,535.
length	specifies the number of bytes in the loop message. The default value is 40 and the maximum length is the maximum segment buffer size, which can be displayed using the SHOW EXEC CHAR command.
block-type	specifies the test data. Three options are available: ZEROS ONES MIXED (default value)
user-id	specifies the user name to be used for access control information in connecting to the remote mirror.
password	specifies the password that corresponds to the specified user-id.

7.7 LOOP NODE

This command tests the logical links to a remote node. Note that the count, length, and block-type are optional, but when used, must be used in the specified order, i.e., count, followed by length, followed by block-type.

Format:

```
LOOP NODE node-id [COUNT count] [LENGTH length]
 [WITH block type] [USER user-id PASSWORD password]
```

where

node-id	identifies the node (name or address)
count	specifies the number of times the command is repeated. The default is 1 and the maximum is 65,535.
length	specifies the number of bytes in the loop message. The default value is 40 and the maximum length is the maximum segment buffer size, which can be displayed using the SHOW EXEC CHAR command.
block-type	specifies the test data. Three options are available: ZEROS ONES MIXED (default value)
user-id	specifies the user name to be used for access control information in connecting to the remote mirror.
password	specifies the password that corresponds to the specified user-id.

7.8 SET EXECUTOR ADDRESS

This command specifies a new node address for the executor node. This command can only be issued if the state of the executor is OFF. It is not possible to issue NCP commands to a remote executor node when the state of that node is OFF. Therefore, it is essential, when issuing this command, that the executor node be the local node.

Format:

```
SET EXECUTOR ADDRESS node-address
```

where

node-address specifies the node address for the local (executor) node

7.9 SET EXECUTOR NAME

This command specifies a new node name for the executor node. (The restrictions for SET EXEC ADDRESS apply to this command, too).

Format:

```
SET EXECUTOR NAME node-name
```

where

node-name specifies a node name for the executor node

Alternatively, use the TELL prefix to identify the executor node for a particular NCP command.

7.7 LOOP NODE

This command tests the logical links to a remote node. Note that the count, length, and block-type are optional, but when used, must be used in the specified order, i.e., count, followed by length, followed by block-type.

Format:

```
LOOP NODE node-id [COUNT count] [LENGTH length]
 [WITH block type] [USER user-id PASSWORD password]
```

where

node-id	identifies the node (name or address)
count	specifies the number of times the command is repeated. The default is 1 and the maximum is 65,535.
length	specifies the number of bytes in the loop message. The default value is 40 and the maximum length is the maximum segment buffer size, which can be displayed using the SHOW EXEC CHAR command.
block-type	specifies the test data. Three options are available: ZEROS ONES MIXED (default value)
user-id	specifies the user name to be used for access control information in connecting to the remote mirror.
password	specifies the password that corresponds to the specified user-id.

7.8 SET EXECUTOR ADDRESS

This command specifies a new node address for the executor node. This command can only be issued if the state of the executor is OFF. It is not possible to issue NCP commands to a remote executor node when the state of that node is OFF. Therefore, it is essential, when issuing this command, that the executor node be the local node.

Format:

```
SET EXECUTOR ADDRESS node-address
```

where

node-address specifies the node address for the local (executor) node

7.9 SET EXECUTOR NAME

This command specifies a new node name for the executor node. (The restrictions for SET EXEC ADDRESS apply to this command, too).

Format:

```
SET EXECUTOR NAME node-name
```

where

node-name specifies a node name for the executor node

Alternatively, use the TELL prefix to identify the executor node for a particular NCP command.

7.14 SET NODE

This command specifies node names and addresses when building the executor's volatile configuration database.

Format:

```
SET NODE node-id ADDRESS node-address NAME node-name
```

where

node-id	identifies the local node and can be a name or an address.
node-address	specifies the address of the node you want to include in the configuration database.
node-name	specifies the name of the node you want to include in the configuration database. Only one name can be assigned to a node address. Duplicate node names are not permitted.

For example,

```
SET NODE 1.7 NAME SATURN
```

creates an entry for the node SATURN at address 1.7.

7.15 SET NODE ALL

This command is used to copy the content of the permanent database to the volatile database for a single node.

Format:

```
SET NODE node-id ALL
```

where

node-id	specifies the local or remote node.

7.16 SHOW NODE

The SHOW NODE is a collection of subcommands that fall into four categories, or formats. This command displays static node information for the executor, a specified node, all active nodes, or all known nodes.

Format 1: Displaying Node Characteristics Command Set

```
SHOW EXECUTOR CHARACTERISTICS
SHOW NODE node-id CHARACTERISTICS
SHOW ACTIVE NODES CHARACTERISTICS
SHOW KNOWN NODES CHARACTERISTICS
```

where

node-id identifies the local or remote node (name or address).

Description of the Screen Display for the Executor

- Identification – 4DDN version string.
- Management Version – Version number of the Network Management layer.
- NSP Version – Version number of the End Communication layer.
- Maximum Links – Maximum number of logical links for the node.
- Delay Factor – The value of the DELAY FACTOR parameter is multiplied by one sixteenth of the estimated round trip delay time to determine the appropriate value for the time to retransmit certain NSP messages.

When specifying values for the DELAY FACTOR parameter, use a value in the range 0 to 255. For example:

```
NCP> SET EXEC DELAY FACTOR 10
```

- Delay Weight – The NSP layer estimates the current delay in the round trip transmission to a node with which it is communicating. The value of the DELAY WEIGHT parameter is used to calculate a new value of the estimated round trip delay. The value is in the 0-255 range. For example:

```
NCP> SET EXEC DELAY WEIGHT 15
```

- Inactivity timer – A logical link is inactive when no data is transmitted in either direction for a given interval of time. The INACTIVITY TIMER regulates the frequency with which 4DDN tests the viability of an inactive link. The INACTIVITY TIMER parameter is used to specify the maximum duration of inactivity before the local 4DDN node tests the viability of the link. For example, this command sets the inactivity interval to 30 seconds:

```
NCP> SET EXECUTOR INACTIVITY TIMER 30
```

  When this timer expires, 4DDN generates artificial traffic to test the link.

- Retransmit factor – This specifies the number of times a packet may be retransmitted before a link is declared broken. The value of the RETRANSMIT FACTOR parameter regulates the number of times the Network Servers Protocol (NSP) Layer reattempts a transmission when its retransmission timer expires for a logical link. Use a number in the range of 0 to 65,535 for this value. For example, the following example specifies that NSP reattempts a transmission no more than 10 times:

```
NCP> SET EXECUTOR RETRANSMIT FACTOR 10
```

  If NSP tries to retransmit the eleventh time, the logical link is disconnected.

- Routing Version – Version number of the Routing layer.
- Type – Type of the specified node. The values are:
  - ROUTING III
  - ROUTING IV
  - AREA
  - NONROUTING III
  - NONROUTING IV
- Maximum Address – Highest address that the local node will recognize.
- Max Broadcast Nonrouters – Maximum number of endnodes the executor node can have on its ethernet circuits. Valid values are in the 0-1023 range.
- Segment buffer size – Maximum size of transmit buffers. This value is system dependent buffers. This value must be in the 512-1460 range.

Output Format

```
NCP> SHOW EXEC CHAR
```

Node Volatile Characteristics as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

Identification	= NML 1.0 SGI 4DDN Release 1.0
Management Version	= V4.0.0
NSP Version	= V4.0.0
Maximum Links	= 32
Delay Factor	= 80
Delay Weight	= 5
Inactivity Timer	= 10
Retransmit Factor	= 10
Routing Version	= V2.0.0
Type	= Nonrouting IV
Maximum Address	= 1023
Max Broadcast Nonrouters	= 1023
Segment Buffer Size	= 1460

```
NCP> SHOW NODE topcat CHAR
```

Node Volatile Characteristics as of Mon Jul 4 12:00:00 1988

Remote Node = 1.2 (TOPCAT)

NO INFORMATION AVAILABLE

Format 2: Displaying Node Counters

```
SHOW EXECUTOR COUNTERS
SHOW NODE 'node-id COUNTERS
SHOW ACTIVE NODES COUNTERS
SHOW KNOWN NODES COUNTERS
```

where

node-id identifies the local or remote node (name or address).

Description of the Display

The node counters provide the following information about user traffic between the executor and the specified node.

- Seconds since last zeroed – Number of seconds that elapsed since the node counters were zeroed.
- User data bytes received – Number of user data bytes received by your node. This does not include protocol headers and checksums.
- User data bytes sent – Number of bytes of data sent by your node.
- User data messages received – Number of user data messages received by your node.
- User data messages sent – Number of user data messages sent by your node.
- Connects received – Number of logical link connection requests received by your node.
- Connects sent – Number of logical link connection requests sent by your node.
- Response timeouts – Number of times there was no response to an NSP segment within the allotted timeout period. Usually indicates overloading, messages being discarded, and necessary retransmission.
- Received connect resource errors – Number of incoming connection requests for which the local node did not have enough resources.

- Packet format error – Number of packet format error that occur because of invalid packet control information. This counter is kept for the executor node only.

NOTE: If no links have been established on the executor, SHOW EXEC COUNT will only display the "Packet format error" count.

Output Format

```
NCP> SHOW EXEC COUNT
```

Node Counters as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

>65534	Seconds since last zeroed
107	User data bytes received
54	User data bytes sent
144	User data messages received
144	User data messages sent
56	Connects received
60	Connects sent
0	Response timeouts
0	Received connect resource errors
0	Packet format error

```
NCP> SHOW NODE saturn COUNT
```

Node Counters as of Mon Jul 4 12:00:00 1988

Remote Node = 1.7 (SATURN)

>65534	Seconds since last zeroed
2	User data bytes received
0	User data bytes sent
0	User data messages received
20	User data messages sent
2	Connects received
0	Connects sent
20	Response timeouts
0	Received connect resource errors

```
NCP> SHOW NODE 2 COUNT
```

Node Counters as of Mon Jul 4 12:00:00 1988

Remote Node = 1.2 (TOPCAT)

No information available

```
NCP> SHOW ACT NODE COUNT
```

Active Node volatile Counters as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

>65534	Seconds since last zeroed
30225	User data bytes received
30277	User data bytes sent
1189	User data messages received
1192	User data messages sent
3	Connects received
3	Connects sent
0	Response timeouts
0	Received connect resource errors
0	Packet format error

Remote Node = 1.5 (ROMULA)

>65534	Seconds since last zeroed
131773	User data bytes received
37128	User data bytes sent
1291	User data messages received
1275	User data messages sent
0	Connects received
20	Connects sent
0	Response timeouts
0	Received connect resource errors

Remote Node = 1.7 (SATURN)

>65534	Seconds since last zeroed
47270	User data bytes received
18196	User data bytes sent
1849	User data messages received
1851	User data messages sent
2	Connects received
12	Connects sent
10	Response timeouts
0	Received connect resource errors

```
NCP> SHOW KNOWN NODE COUNT
```

Known Node Volatile Counters as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

>62545	Seconds since last zeroed
2	User data bytes received
0	User data bytes sent
0	User data messages received
20	User data messages sent
2	Connects received
0	Connects sent
20	Response timeouts
0	Received connect resource errors

Remote Node = 1.2 (TOPCAT)

No information available

Format 3: Displaying Node Status

```
SHOW EXECUTOR STATUS
SHOW NODE node-id STATUS
SHOW ACTIVE NODES STATUS
SHOW KNOWN NODES STATUS
```

where

node-id identifies the local or remote node (name or address).

Description of the Display for the Executor Node

- Node address and name.
- State – On or Off.
- Ethernet Physical Address.

Description of the Display for a Remote Node

- Node address and name.
- Routing state – reachable or unreachable.
- Number of active logical links associated with the node.
- Delay time to set the retransmission.
- Node type – ROUTING IV, NONROUTING IV, or AREA.

Output Format

```
NCP> SHOW EXEC STAT
```

Node Volatile Status as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

State = On
Physical Address = aa-00-04-00-04-04

```
NCP> SHOW NODE 1 STAT
```

If node 1 is the executor, see SHOW EXEC STAT command.

```
NCP> SHOW NODE 2 STAT
```

Node Volatile Status as of Mon Jul 4 12:00:00 1988

Node	State	Active Links	Delay	Type	Cost	Hops	Circuit
1.2 (TOPCAT)	reachable			routing IV			ENP

```
NCP> SHOW ACTIVE NODE STAT
```

Node Volatile Status as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

State = On
Physical Address = aa-00-04-00-04-04

Node	State	Active Links	Delay	Type	Cost	Hops	Circuit
1.2 (TOPCAT)	reachable			routing IV			ENP

```
NCP> SHOW KNOWN NODE STAT
```

Known Node Volatile Status as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

State = On
Physical Address = aa-00-04-00-04-04

Node	State	Active Links	Delay	Type	Cost	Hops	Circuit
1.2 (TOPCAT)	reachable			routing IV			ENP

Format 4: Displaying a Node Summary

```
SHOW EXECUTOR SUMMARY
SHOW NODE node-id SUMMARY
SHOW ACTIVE NODES SUMMARY
SHOW KNOWN NODES SUMMARY
```

where

node-id identifies the local or remote node (name or address).

Description of the Display

- Node address and name.
- Routing state – reachable or unreachable.
- Number of active logical links associated with the node.
- Retransmission delay time in seconds.

Output Format

```
NCP> SHOW EXEC SUMM
```

Node Volatile Summary as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

State = On
Identification = NML 1.0 SGI 4DDN Release 1.0

if node is the executor, see the SHOW EXEC SUMM command.

```
NCP> SHOW NODE 2 SUMM
```

Node Volatile Summary as of Mon Jul 4 12:00:00 1988

Node	State	Active Links	Delay	Circuit	Next Node
1.2 (TOPCAT)	reachable			ENP	

```
NCP> SHOW ACT NODE SUMM
```

Node Volatile Summary as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

State = On
Active Links = 4

Node	State	Active Links	Delay	Circuit	Next Node
1.2 (TOPCAT)	reachable			ENP	

```
NCP> SHOW KNOWN NODE SUMM
```

Known Node Volatile Summary as of Mon Jul 4 12:00:00 1988

Executor Node = 1.4 (PLUTO)

State = on
Active Links = 2

Node	State	Active Links	Delay	Circuit	Next node
1.2 (TOPCAT)	reachable			ENP	
1.5 (ROMULA)	reachable	1	4	ENP	
1.6 (VULCAN)	unreachable				
1.7 (SATURN)	unreachable				

7.17 SHOW CIRCUIT

This command displays static line information.

Format 1: Displaying Circuit Characteristics

```
SHOW KNOWN CIRCUIT CHARACTERISTICS
```

Description of Characteristics

- State – Operational state of a circuit. There are 2 circuit states: OFF and ON.
- Designated router – Node that is used for routing to non-adjacent nodes.
- Hello timer – Frequency in seconds of routing layer hello messages sent to the adjacent node(s) on the circuit.
- Type – Circuit type: ethernet.
- Adjacent node – Adjacent node on the circuit. On an ethernet circuit, there can be several adjacent nodes.
- Listen timer – Maximum time allowed before a message is received from an adjacent node on the circuit. If this time is exceeded the node is declared unreachable.

Output Format

```
NCP> SHOW KNOWN CIRC CHAR
```

Known Circuit Volatile Characteristics as of Mon Jul 4 12:00:00 1988

Circuit = ENP

State	= on	
Designated router	= 1.1 (VENUS)	
Hello timer	= 15	(Display for 1st Node)
Type	= Ethernet	
Adjacent node	= 1.1 (VENUS)	
Listen timer	= 15	

Circuit = ENP

Adjacent node	= 1.2 (TOPCAT)	(Display for all Other Nodes)
Listen timer	= 15	

Format 2: Displaying Circuit Counters

```
SHOW KNOWN CIRCUIT COUNTERS
```

Description of Counters

- Seconds since last zeroed – Time elapsed since counters were zeroed for the last time.
- Bytes sent – Number of bytes of data sent by the local node over the circuit.
- Bytes received – Number of bytes of data received by the local node over the circuit.
- Data blocks sent – Number of data blocks sent by the local node.
- Data blocks received – Number of data blocks received by the local node.

Output Format

```
NCP> SHOW KNOW CIRC COUNT
```

Known Circuit Volatile Counters as of Mon Jul 4 12:00:00 1988

Circuit = ENP

>65534	Seconds since last zeroed
21499	Data blocks sent
20771	Data blocks received
497539	Bytes sent
263192	Bytes received

Format 3: Displaying Circuit Status

```
SHOW KNOWN CIRCUIT STATUS
```

Description of the Display

- Circuit ID – Identifies a particular circuit for which the information is displayed.
- Circuit current state – ON or OFF.
- Address and name of adjacent nodes on that circuit.
- Adjacent node ID.
- Block size.

Output

```
NCP> SHOW KNOWN CIRCUIT STATUS
```

Known Circuit Volatile Status as of Mon Jul 4 12:00:00 1988

Circuit	State	Loopback Name	Adjacent Node	Block Size
ENP	on		1.1 (VENUS)	
ENP			1.2 (TOPCAT)	

Format 4: Displaying a Circuit Summary

```
SHOW KNOWN CIRCUIT SUMMARY
```

Description of the Display

- Circuit ID – Identifies a particular circuit for which the information is displayed.
- Circuit current state – ON or OFF.
- Address and name of adjacent nodes on that circuit.

Output

```
NCP> SHOW KNOWN CIRCUIT SUMMARY
```

Known Circuit Volatile Summary as of Mon Jul 4 12:00:00 1988

Circuit	State	Loopback Name	Adjacent Node
ENP	on		1.1 (VENUS)
ENP			1.2 (TOPCAT)

7.18 SHOW LINE

This command displays the line information.

Format 1: Displaying Line Characteristics

```
SHOW KNOWN LINE CHARACTERISTICS
```

Description of the display

- Protocol
- Hardware address

Output

```
NCP> SHOW KNOWN LINE CHAR
```

Known Line volatile characteristics as of Mon Jul 4 12:00:00 1988

Line = ENP

Protocol	= Ethernet
Hardware address	= aa-00-04-00-04-04

Format 2: Displaying Line Counters

```
SHOW KNOWN LINE COUNTERS
```

Description of Line Counters

- Seconds since last zeroed – Number of seconds elapsed since the line counters were zeroed.
- Data blocks received – Number of data blocks received over the line.
- Data blocks sent – Number of data blocks sent over the line.
- Blocks sent, multiple collisions – Number of times that a frame was successfully transmitted on the third or later attempt after normal collisions on previous attempts.
- Collision detect check failure – Number of collision detect failures not sensed after a transmission.
- System buffer unavailable – Number of times no data link buffer was available for an incoming frame.

Output

```
NCP> SHOW KNOW LINE COUNT
```

Known Line volatile Counters as of Mon Jul 4 12:00:00 1988

Line = ENP

>65534	Seconds since last zeroed
76622	Data blocks sent
36046	Data blocks received
397468	Bytes sent
271671	Bytes received
0	Blocks sent, multiple collisions
0	Collision detect check failure
0	Unrecognized frame destination
0	System buffer unavailable

Format 3: Displaying Line Status

```
SHOW KNOWN LINES STATUS
```

Description of the Display

- Line name (e.g., ENP).
- State – On or Off.

Output

```
NCP> SHOW KNOWN LINE STAT
```

Known Line volatile Status as of Mon Jul 4 12:00:00 1988

Line	State
ENP	On

Format 4: Displaying Line Summaries

```
SHOW KNOWN LINES SUMMARY
```

Description of the Display

- Line name (e.g., ENP).
- State – On or Off.

Output

```
NCP> SHOW KNOWN LINE SUMM
```

Known Line Volatile Summary as of Mon Jul 4 12:00:00 1988

Line	State
ENP	On

7.19 TELL

Use the TELL prefix to identify the executor node for a particular NCP command. TELL sets the executor for only one command and must prefix the command for which it is intended. For example:

```
NCP> TELL pluto SHOW EXECUTOR CHAR
```

This command sets the executor to node pluto where the SHOW EXECUTOR CHAR executes.

7.20 ZERO EXECUTOR COUNTERS

This command resets all counters to zero on the executor node.

7.21 ZERO KNOWN CIRCUIT COUNTERS

This command resets circuit counters to zero for all known circuits.

7.22 ZERO KNOWN LINE COUNTERS

This command resets line counters to zero for all known lines.

7.23 ZERO KNOWN NODES COUNTERS

This command resets node counters to zero for all known nodes.

7.24 ZERO NODE COUNTERS

This command resets node counters to zero for a specified node, remote or local.

```
ZERO NODE node-id counters
```

where

node-id	identifies the local or remote node (name or address)

Appendix A: Default Values for Parameters

A.1 Node Parameters

Delay Factor	80
Delay Weight	5
Inactivity Timer	10
Retransmit Factor	10
Maximum Address	1023
Maximum Broadcast Nonrouters	1023
Segment Buffer Size	1460
Address	1
Type	Nonrouting IV

A.2 Loop Parameters

Count	1
Length	40
With	Mixed